Assessment of Quadrotor PID Control Algorithms using six-Degrees of Freedom CFD simulations

Assessment of Quadrotor PID Control Algorithms using six-Degrees of Freedom CFD simulations

Manuel Carreño Ruiz[1], Nicoletta Bloise[1], Elisa Capello[2], Domenic D'Ambrosio[1] and Giorgio Guglieri[2]

***Abstract*— The evolution of technology has made increasingly advantageous the introduction of Unmanned Aerial Systems (UASs) in various applications, especially by exploiting their ability for autonomous flight. This paper presents an innovative approach to simulating UAS maneuvers that integrates a Computational Fluid Dynamics (CFD) model and a closed-loop control algorithm for both position and attitude dynamics. We chose the Proportional-Integrative-Derivative (PID) controller for this preliminary research activity because of its simple implementation and widespread employment in commercial autopilot systems. The numerical simulation of the UAS aerodynamics allows for performing an accurate analysis in critical situations. These include, for example, ground effect or wind gusts scenarios, which require an enhanced propulsive model to capture the interaction between vehicle dynamics, aerodynamics, and environmental conditions. The coupled CFD/PID framework can be a virtual testing environment for UAS platforms. Here we report on its validation. The paper compares such an innovative in-the-loop CFD approach and a classical simplified propulsive model that adopts constant thrust and torque coefficients.**

Index Terms — Unmanned Aerial Systems, PID Controller, Computational Fluid Dynamics

I. Introduction

In the last decade, the interest in Unmanned Aerial Systems (UASs), commonly known as drones, has rapidly grown in many different sectors, and challenges are arising in the field of robotics, communication, and Big Data [1]. In the next future, unpiloted aircraft will be involved more and more in urban air mobility (for example in smart-cities [2]), as well as in the farming area (for example in precision agriculture applications [3]). For all these operations, it is mandatory to guarantee the maximum level of automation and to compute safe flight missions [4].

The field is full of opportunities and challenges that require interdisciplinary studies. Previous research highlights the need to analyze disturbances induced by aerodynamic effects, for example, using an Active Disturbance Rejection Control (ADRC) to maintain stability even with external disturbances [5]. Especially considering multicopters with small inertia, their performances are significantly affected by disturbances. Some authors [6], [7] present a disturbance rejection mechanism for Micro Air Vehicles (MAV) based on two control algorithms that switch on according to the process. Likewise, other authors [8] investigate the position and attitude control in environments with extreme external disturbances such as wind gusts. They present a robust adaptive controller with an aerodynamics, wind gust, and control model integrated into a six-degree-of-freedom UAS dynamics solver. In both the previous works, a Proportional–Integral–Derivative (PID) controller is tuned by adding another adaptive control term. In Ref. [9], an optimized PID is combined to improve disturbance rejection.

The novelty of our research is using a Computational Fluid Dynamics (CFD) solver to evaluate the state of the UAS using a closed-loop feedback control that can reduce the error value over time. We have no record of publications reporting this kind of approach in the context of UAS flight. However, there are some examples of PID-controlled CFD simulations [10], where the authors propose a new method of tuning the PID controller parameters using CFD simulation for a greenhouse climate control system, avoiding the implementation of the sensors model. A similar CFD-based ventilation test method is presented in Ref. [11] for a conditioned room model to evaluate the ventilation system and control performance. On the other hand, several authors simulated quadcopters with some degrees of freedom but without a control strategy or the generality of a 6-DOF simulation. For example, some authors [12] present a somehow similar approach in the context of urban air mobility using loosely-coupled high-fidelity CFD simulations with the reduced-order model CAMRAD-II to achieve a trimmed cruise condition. Also, Ref. [13] focuses on a vehicle that moves near obstacles to studying wall and ground effects. However, since the motion is predefined, the simulation cannot capture the quadcopter's dynamics, evidencing the great advantage embedded in our methodology.

Despite the high computational cost of CFD simulations, the evidence demonstrates an increasing interest in this direction. For this reason, for these preliminary results, a traditional PID control strategy is implemented to validate the approach. However, the framework is general and extensible to other control algorithms of increasing complexity. In any case, as reported in Refs. [14] and [15], PID control is the most implemented in the industrial process, as it is a good compromise between robustness and performance for UAS autopilot design. For different approaches to PID UAV control, the reader may refer, for example, to Ref. [16]. In

*This work was not supported by any organization

[1] Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy manuel.carreno@polito.it, nicoletta.bloise@polito.it, domenic.dambrosio@polito.it

[2]Department of Mechanical and Aerospace Engineering, Politecnico di Torino and CNR-IEIIT, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy elisa.capello@polito.it, giorgio.guglieri@polito.it

this work, we study a cascade PID controller for a simple horizontal translation for a limited time due to the extensive computational requirements of CFD simulations.

The main goal of this paper is to present an innovative approach to performing precise simulations of UAS maneuvers. The proposed methodology includes a PID control system embedded in the commercial Computational Fluid Dynamics software STAR-CCM+ that will help the analysis of critical scenarios such as ground/wall effect near obstacles or wind gusts disturbances. Thus, the research contributes to increasing the safety and reliability of UAS missions and improving the tuning of PID controller gains.

The paper layout is the following. In Section II, the methodology and tools are proposed, including a detailed UAS dynamic model, the PID control law, and the CFD modeling. Section III contains the simulations results, with a comparison between simplified and realistic dynamics and environment. Finally, we lay out our conclusions and future activities in Section IV.

II. Methodology and tools

We illustrate in this section our proposed methodology, the mathematical formulation, and the underlying numerical tools. As a preliminary case study and for validation purposes, we chose a simple maneuver, namely a horizontal translation at a constant altitude. Future research will deal with more critical maneuvers, both as interest and computational time are concerned.

A. Quadrotor dynamics

We implemented a six-degree-of-freedom (DOF) UAS rigid body model in a MATLAB/Simulink environment to simulate the desired maneuvers and tune the controller parameters. In particular, we considered a quadrotor model, which represents an under-actuated system with four inputs (the speed of four rotors) that are controllable in position and attitude dynamics through thrust and torque generation [17]. Two reference frames define the flight dynamics: i) the North-East-Down (NED) reference frame, ergo the inertial frame, and ii) the Body reference frame centered in the Center of Gravity (CoG) of the UAS. Indeed, to describe the quadrotors' attitude, Euler angles (ϕ, θ, ψ) identify a sequence of three rotations from the body frame to the inertial one.

In this work, we considered the "+" (cross) quadrotor orientation, where the drone has two rotors parallel to the body x-axis (rotors 1 and 3 with counterclockwise rotation) and two rotors parallel to the body y-axis (rotors 2 and 4 with clockwise rotation), as shown in Fig. 1. The main quadrotor parameters taken as a reference are in Table I. As mentioned before, four control inputs influence the quadrotor dynamics. Parameter u_1 is the sum of all rotor forces, u_2 is the rolling moment (the torque generated along y_b), u_3 is the pitching moment (the torque generated along x_b) and u_4 is the yawing moment around z_b. A propulsive model relating the thrust and torque of each rotor ($i = 1,...,4$) as a function of the rotor speed (ω_i) has to be defined to obtain these forces and moments. For the simplified Simulink model, a simple constant coefficient model (k_T and k_D, thrust and torque coefficient respectively) is used as shown in Eqs.1 and 2.

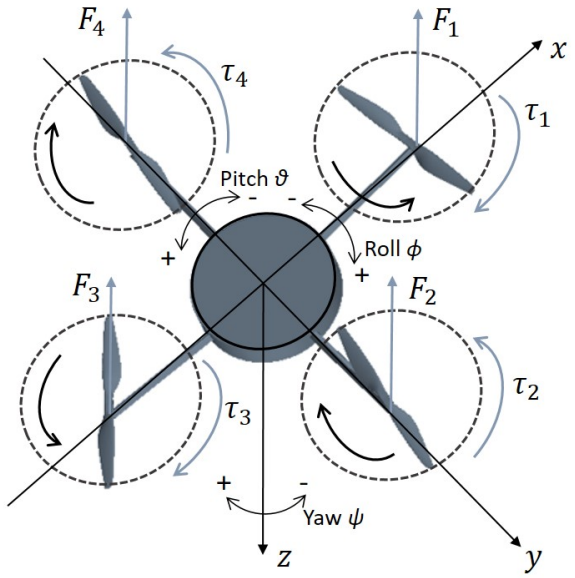

Fig. 1. Forces and torques on the quadrotor in the body frame

UAS model parameters			
Mass	2.45 kg	ω_{max}	523.6 rad/s
Ix	0.02 Kgm^2	ω_{min}	104.72 rad/s
Iy	0.02 Kgm^2	**r**	0.19 m
Iz	0.04 Kgm^2	k_T	4.51 10^{-5}
l	0.4 m	k_D	8.57 10^{-7}

TABLE I
UAS parameters used in the simulation

$$F_i = k_T\ \omega_i^2 \tag{1}$$

$$\tau_i = (-1)^{i+1} k_D\ \omega_i^2 \tag{2}$$

Finally, summarizing the UAS model described in detail in Ref. [17], it is possible to express the quadrotor position dynamics in the inertial frame as

$$\begin{bmatrix} \ddot{x}^{NED} \\ \ddot{y}^{NED} \\ \ddot{z}^{NED} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} (-[c(\phi)c(\psi)s(\theta)+s(\phi)s(\psi)]u_1 - k_{Dx}\dot{X}^{NED}) \\ (-[c(\phi)s(\psi)s(\theta)-c(\psi)s(\phi)]u_1 - k_{Dy}\dot{Y}^{NED}) \\ (-[c(\phi)c(\theta)]u_1 - k_{Dz}\dot{Z}^{NED}) + g \end{bmatrix} \tag{3}$$

and the attitude dynamics as

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 1/I_x[(I_y - I_z)qr - I_r q(\omega_1 - \omega_2 + \omega_3 - \omega_4) + u_2] \\ 1/I_y[(I_z - I_x)pr + I_r p(\omega_1 - \omega_2 + \omega_3 - \omega_4) + u_3] \\ 1/I_z[(I_x - I_y)pq + u_4] \end{bmatrix} \tag{4}$$

where I_r is the rotational inertia of a single rotor and I_x, I_y, I_z are the principal moment of inertia measured from the CAD model.

To control the angular rates (p, q, r) in the body frame of reference, the derivatives of the Euler angles ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) must be transformed in body frame through the following rotation matrix:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & s(\phi)c(\theta) \\ 0 & -s(\phi) & c(\phi)c(\theta)] \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (5)$$

Combining the rotor forces and moments as defined in Fig. 1, the control algorithm (discussed in II-B) provides four control input depending on the rotor speeds:

$$\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \begin{bmatrix} k_T & k_T & k_T & k_T \\ 0 & -lk_T & 0 & lk_T \\ lk_T & 0 & -lk_T & 0 \\ k_D & -k_D & k_D & -k_D \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \quad (6)$$

Given the control inputs (u_1, u_2, u_3, u_4), one can calculate the rotor speeds required for each motor using Eq. 6 . Such output is the input (and control action) for the CFD model.

B. PID control design

We propose a Proportional-Integral-Derivative (PID) controller to test its effectiveness in combination with CFD simulations. As a feedback control, the system acquires the measurement of the controlled variable to stabilize the system and to reduce the error $e(t) = r(t) - y(t)$ between the reference and the measured variable. PID controllers are popular because of their simplicity, robustness, and effectiveness in forcing the controlled variable $y(t)$ to follow as closely as possible a reference variable $r(t)$ defined by the guiding law [18]. Moreover, adjusting the control parameters to reach the reference in a finite time is relatively easy. The control signal $u(t)$ becomes

$$u(t) = K_P e(t) + K_I \int e(\tau) d\tau + K_D \frac{d}{dt} e(t) \quad (7)$$

where K_P is the proportional gain, K_I is the integral gain, and K_D is the derivative gain. These are tuned with a trial and error approach to accomplish required performance.

To track position and attitude references (x_{ref}, y_{ref}, z_{ref}, ψ_{ref}), a robust cascade PID control algorithm has been developed, as shown in Fig. 2. Inputs to the control logic are (x, y, z), (ϕ, θ, ψ) and (p, q, r) in body frame. Sensors and noises are not considered in this work. The outermost loops provide input to the inner ones. Overall, we have:

- the outermost loop, based on the error between the desired position and elevation along the (x,y) and z axes, respectively, and the actual state of the system.
- The middle loop, based on the error between the desired angles, which derive from the previous loop, and the actual angles.
- The control loop on the angular velocities, which returns the control pairs u_2, u_3, and u_4.

We set the gains of the PIDs to meet some performance requirements in the time domain, particularly in terms of stability, steady-state error, and convergence time.

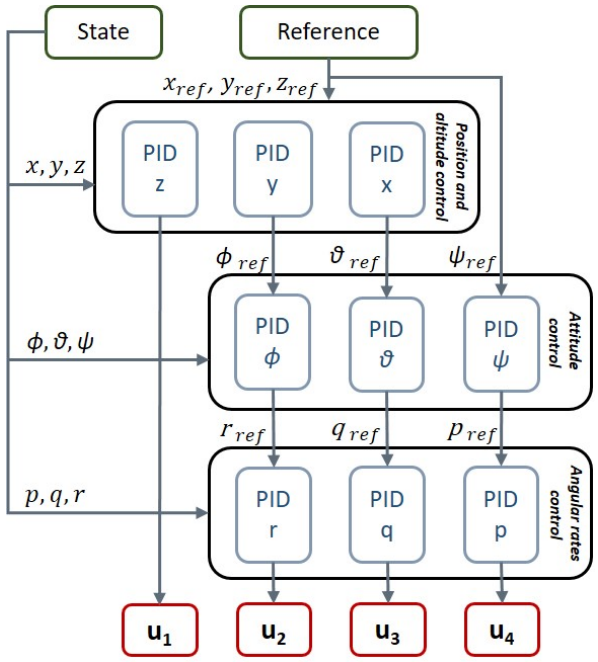

Fig. 2. Robust cascade PID position and attitude control

C. CFD analysis

The commercial CFD software STARCCM+ [19] was used to create a 6 Degrees-of-Freedom numerical model of a quadrotor controlled with the PID approach described in Section II-B. We adopted an overset grid approach to let the UAS move within a background grid. Simulations are performed in a closed cube environment with a height, width, and depth of 20 meters, as shown in Fig. 3. Using such a large domain is possible without the need for too many cells since the background mesh is very coarse. An automatic Adaptive Mesh Refinement (AMR) algorithm refines the background mesh using as a trigger function the interface with the body grid, which moves in solidarity with

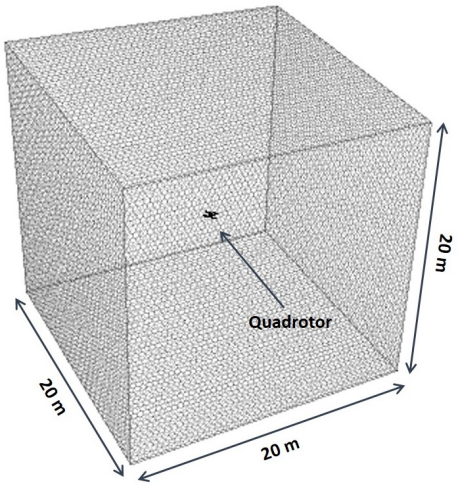

Fig. 3. CFD simulation domain

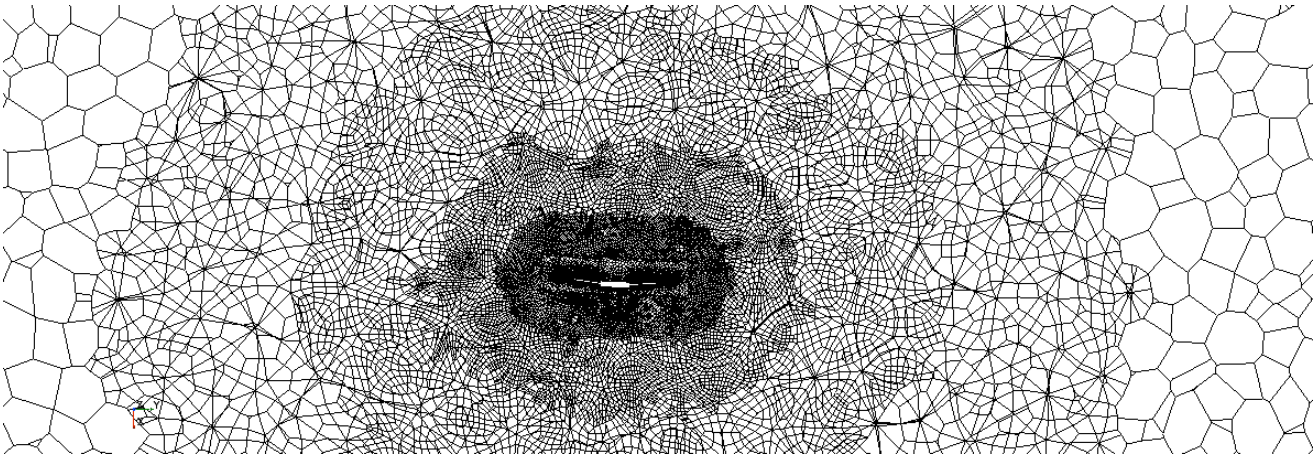

Fig. 4. Computational grid showing adaptive mesh refinement tracking the near-body grid

the quadcopter, as shown in Figure 4. The adaptive mesh refinement occurs every ten integration steps in time to limit the computational cost. A sliding grid approach models the rotors' motion [20] inside the body grid. The time step is defined to allow a maximum rotation of 3^o at the highest angular velocity defined in Table I. The resulting value has an order of magnitude is $10^{-4}s$, so that the physical time that it is possible to simulate is limited depending on the computer system. The adopted time-integration scheme is implicit and second-order accurate. The spatial accuracy is second order also. As a turbulence model, we use the URANS one-equation Spallart-Allmaras model [21]. The grid adopted in this work is under-resolved (4 million points) to accurately simulate aerodynamic interactions such as rotor-rotor interactions. However, its refinement is more than sufficient to describe the entanglement between rotor aerodynamics and quadrotor dynamics. Such a capability confers our simulations a noticeable advantage compared to the propulsive model usually employed in dynamic simulations, which relies on constant thrust and torque coefficients, as shown in Eq. 6. We verified the scalability of such an approach to larger grids by running the simulations on a 4 million and an 8 million cells grid. The results showed an under-linear increase in CPU time due to the reduced importance of the overhead that the 6-DOF solver adds to the simulation. That is an aspect that improves the performance of parallel computing. The link between the controller and the CFD model is performed by adjusting the rotation rates at each time step, and the framework includes an input file that allows the definition of different waypoints. The position, orientation, linear velocities and angular velocities in the CFD simulations are provided by the 6-DOF solver that integrates the resultant accelerations on the vehicle.

Figure 5 illustrates our virtual control test strategy. In particular, the process controller, being a feedback system, computes the error between a reference signal and an actual state vector obtained from the CFD model. Its output, namely forces and torques, is converted in rotor speed according to Eq. 6 and enters directly into the virtual simulator.

We carried out the simulations using the 32 cores of an Intel Xeon Scalable Processors Gold 6130 2.10 GHz. The computational cost of the maneuver presented in this paper is 4000 CPU hours.

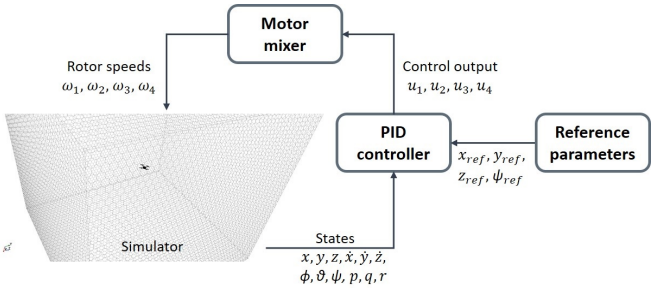

Fig. 5. Configuration of the virtual UAS control test system

III. Simulation Results

Here we summarize the results of a simple preliminary test case. We focus on a horizontal translation of 20 cm on the x-axis at constant altitude. The active control must move the UAS to the desired position while contrasting the gravity force. We compare the simplified Simulink model operating at frequency 500 Hz and the CFD simulation to evaluate seldom-captured non-linear aerodynamic and propulsion effects. Therefore, in order to show the translation along x-axis NED, we present the relevant states x and θ (the pitch angle) in Figs. 6 and 7 . The two states are very similar in both models, and the small discrepancies are due to the transient of the rotors and the fact that the quadcopter has a linear velocity in the x-direction.

Figure 8 displays the output of the controller and consequently from Eq. 6, each angular velocities of the rotors are evaluated. The force output control (u_1) seems to converge to a value around 15% higher in the CFD model. That is because the value of K_T used in Eq. 6 arises from the simulations described in Ref. [20], where the computational mesh was well-resolved, while the present under-resolved grid predicts slightly lower thrust values for the same rotation

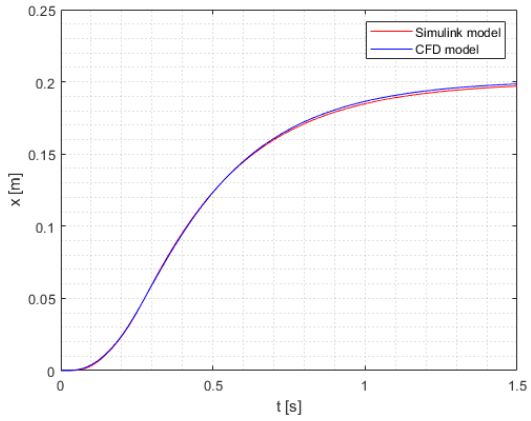

Fig. 6. Comparison of the x-position for the two models

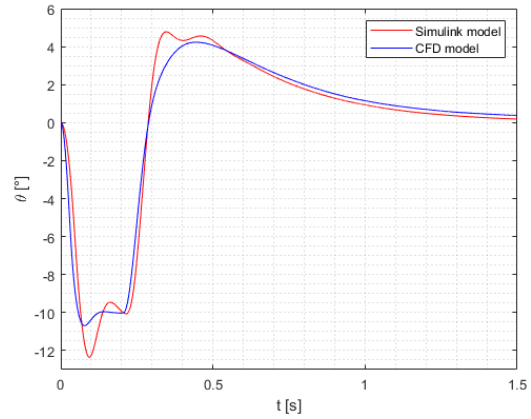

Fig. 7. Comparison of the θ angle for the two models

speed. The same effect is appreciated for the moment output control u_4 due to the differences in the torque coefficient.

The other two outputs respect the trend, except during the transient, which looks damped in the CFD model. That is possibly due to an actual physical phenomenon, namely the time delay between the change in rotation rates and the subsequent update of the aerodynamic forces, which is not instantaneous.

Finally, we show some images from the CFD simulation in Fig. 9, where one can see the simulated advancing maneuver of the drone. It is possible to see the downwash velocities and the drone attitude. Initially, the quadcopter has a negative pitch angle that is needed to accelerate the vehicle to the desired position. Then, a higher thrust, revealed by the increased downwash speeds in rotor 1, helps increment the pitch angle to positive values, as shown in Fig. 7, in an attempt to compensate for the positive velocity that the vehicle has acquired. The last images show that the UAS asymptotically regains a hovering position at an x coordinate of 20 cm.

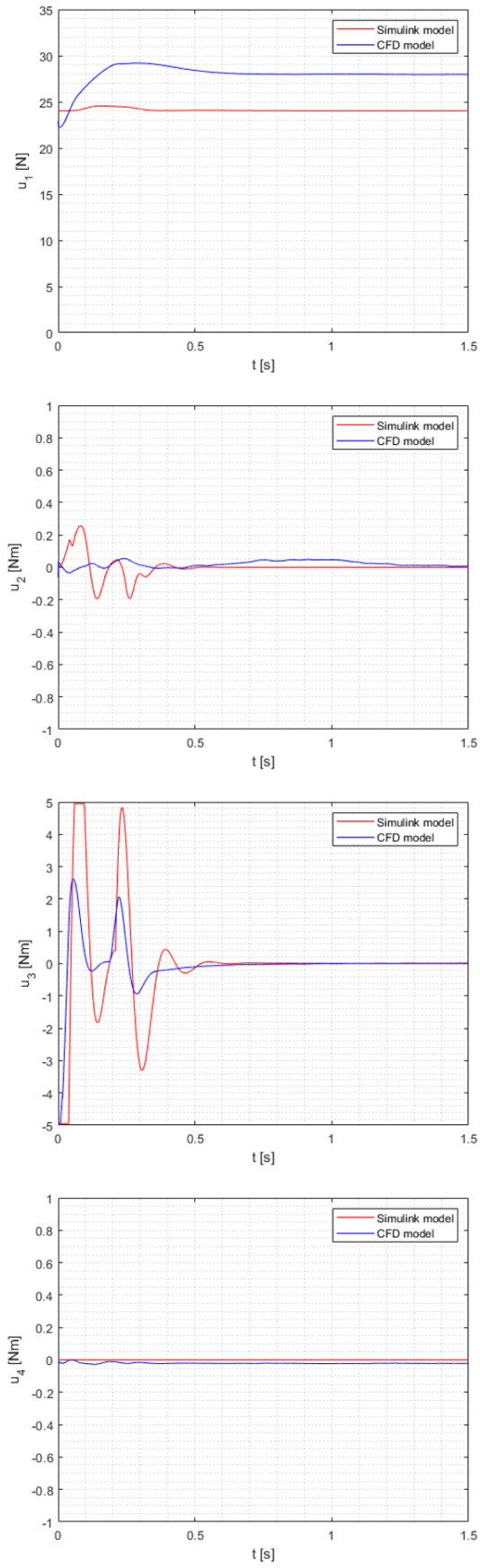

Fig. 8. Comparison of the output control u_1, u_2, u_3, u_4 for the two models

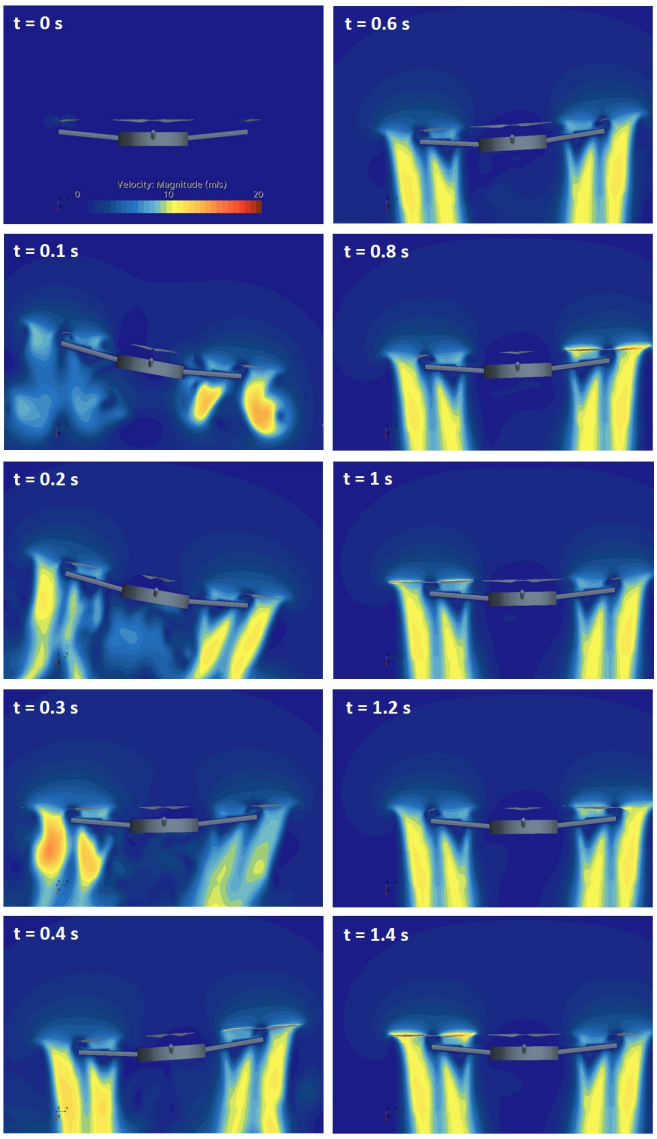

Fig. 9. Visualization of the velocity magnitude field obtained with CFD during the quadrotor maneuver

IV. Conclusions and Future Works

In this paper, we presented a standalone numerical model to simulate the flight of a quadcopter that implements a PID controller in STAR-CCM+ commercial CFD software to reproduce UAS maneuvers accurately. This innovative strategy has produced coherent results compared with simplified propulsion and dynamics. The objective was to validate this simulation framework in a simple case such as a horizontal translation. The enhanced propulsive model and aerodynamics are likely to be important issues for future research simulating more critical maneuvers with ground/wall effect, wind gusts, and contact inspection applications where nonlinear behavior is fundamental to achieving accurate control.

As further work, we will perform experimental testing to validate the model. We aim to assess the model fidelity by comparing CFD virtual flight testing and actual flight logs.

References

[1] M. Idrissi, M. Salami, and F. Annaz, "A review of quadrotor unmanned aerial vehicles: Applications, architectural design and control algorithms," *Journal of Intelligent & Robotic Systems*, vol. 104, no. 2, pp. 1–33, 2022.

[2] N. Mohamed, J. Al-Jaroodi, I. Jawhar, A. Idries, and F. Mohammed, "Unmanned aerial vehicles applications in future smart cities," *Technological Forecasting and Social Change*, vol. 153, p. 119293, 2020.

[3] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, "A compilation of uav applications for precision agriculture," *Computer Networks*, vol. 172, p. 107148, 2020.

[4] N. Bloise, S. Primatesta, R. Antonini, G. P. Fici, M. Gaspardone, G. Guglieri, and A. Rizzo, "A survey of unmanned aircraft system technologies to enable safe operations in urban areas," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2019, pp. 433–442.

[5] Y. Zhang, Z. Chen, X. Zhang, Q. Sun, and M. Sun, "A novel control scheme for quadrotor uav based upon active disturbance rejection control," *Aerospace Science and Technology*, vol. 79, pp. 601–609, 2018.

[6] A. Matus-Vargas, G. Rodríguez-Gómez, and J. Martínez-Carranza, "Aerodynamic disturbance rejection acting on a quadcopter near ground," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2019, pp. 1516–1521.

[7] A. Matus-Vargas, G. Rodriguez-Gomez, and J. Martinez-Carranza, "Ground effect on rotorcraft unmanned aerial vehicles: a review," *Intelligent Service Robotics*, vol. 14, no. 1, pp. 99–118, 2021.

[8] M. Kazim, A. T. Azar, A. Koubaa, and A. Zaidi, "Disturbance-rejection-based optimized robust adaptive controllers for uavs," *IEEE Systems Journal*, vol. 15, no. 2, pp. 3097–3108, 2021.

[9] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, and S. M. Smailzadeh, "Attitude control of a quadrotor with optimized pid controller," 2013.

[10] H. Xiao, L. Feng, and Y. Zhi, "Tuning the pid parameters for greenhouse control based on cfd simulation," in *2013 Second International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*. IEEE, 2013, pp. 485–489.

[11] Z. Sun and S. Wang, "A cfd-based test method for control of indoor environment and space ventilation," *Building and Environment*, vol. 45, no. 6, pp. 1441–1447, 2010.

[12] P. Ventura Diaz and S. Yoon, "High-fidelity simulations of a quadrotor vehicle for urban air mobility," in *AIAA SCITECH 2022 Forum*, 2022, p. 0152.

[13] C. Paz, E. Suárez, C. Gil, and C. Baker, "Cfd analysis of the aerodynamic effects on the stability of the flight of a quadcopter uav in the proximity of walls and ground," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 206, p. 104378, 2020.

[14] A. L. Salih, M. Moghavvemi, H. A. Mohamed, and K. S. Gaeid, "Flight pid controller design for a uav quadrotor," *Scientific research and essays*, vol. 5, no. 23, pp. 3660–3667, 2010.

[15] P. Wang, Z. Man, Z. Cao, J. Zheng, and Y. Zhao, "Dynamics modelling and linear control of quadcopter," in *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*. IEEE, 2016, pp. 498–503.

[16] G. Szafranski and R. Czyba, "Different approaches of pid control uav type quadrotor," 2011.

[17] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of Unmanned Aerial Vehicles-5 Volume Set*. New York, NY, USA: Springer, 2014.

[18] L. Becce, N. Bloise, and G. Guglieri, "Optimal path planning for autonomous spraying uas framework in precision agriculture," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2021, pp. 698–707.

[19] P. Siemens, "Star-ccm+ user guide version 14.06," *Siemens PLM software Inc*, 2019.

[20] M. Carreño Ruiz, M. Scanavino, D. D'Ambrosio, G. Guglieri, and A. Vilardi, "Experimental and numerical analysis of multicopter rotor aerodynamics," in *AIAA Aviation 2021 Forum*, 2021, p. 2539.

[21] P. Spalart and S. Allmaras, "A one-equation turbulence model for aerodynamic flows," in *30th aerospace sciences meeting and exhibit*, 1992, p. 439.