

# Sample-based SMPC for tracking control of fixed-wing UAV

Martina Mammarella<sup>1</sup>, Elisa Capello<sup>2</sup>, Fabrizio Dabbene<sup>3</sup> and Giorgio Guglieri<sup>1</sup>

**Abstract**—In this paper, a guidance and tracking control strategy for fixed-wing unmanned aerial vehicle autopilots is presented. The proposed control exploits recent results on sample-based stochastic model predictive control, which allows coping in a computationally efficient way with both parametric uncertainty and additive random noise. Different application scenarios are discussed, and the implementability of the proposed approach are demonstrated through simulations. The capability of guaranteeing probabilistic robust satisfaction of the constraint specifications represents a key-feature of the proposed scheme, allowing real-time tracking of the designed trajectory with guarantees in terms of maximal deviation with respect to the planned one. The presented simulations show the effectiveness of the proposed control scheme.

## I. INTRODUCTION

Remotely piloted aircraft systems (RPAS) or unmanned aerial vehicles (UAVs), commonly known as drones, are being widely studied and developed due to their mission flexibility, reconfigurable architecture and cost effectiveness. The list of practical and potential applications of UAVs is wide, well beyond the boundaries of conventional remote sensing scenarios, see e.g. [1], [2] for a discussion.

The majority of studies available in the literature are related to the use of *multi-rotor* UAVs, which are characterized by a high degree of maneuverability and flexibility in configuration. On the other hand, these advantages are paired with some rather serious drawbacks in terms of fuel consumption / mission duration, and relative low speed. These considerations have recently revived the interest in employing fixed-wing UAVs (FW-UAVs), which are in general able to perform longer missions, cover larger areas, and have lower construction costs. Some typical fields of application are: (i) rescue and civil protection, (ii) precision farming, and (iii) urban monitoring. The interested reader is referred to [3] for a detailed discussion of these three scenarios. In particular, some common features can be highlighted: (i) high uncertainty and variability, such as the need of producing low-cost FW-UAVs induces uncertainty in their characteristics, (ii) safety issues (i.e., the necessity to guarantee mission compliance with guaranteed deviation from the planned trajectory).

\*This work was not supported by any organization

<sup>1</sup>M. Mammarella and G. Guglieri are with the Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, [martina.mammarella@polito.it](mailto:martina.mammarella@polito.it)

<sup>2</sup>E. Capello is with the Department of Mechanical and Aerospace Engineering, Politecnico di Torino and with the CNR-IEIIT, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, [elisa.capello@polito.it](mailto:elisa.capello@polito.it)

<sup>3</sup>F. Dabbene is with the CNR-IEIIT, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, [f.dabbene@ieiit.cnr.it](mailto:f.dabbene@ieiit.cnr.it)

The literature on control design for UAVs is large, and ranges from proportional derivative integrative (PID) control [4], [5] to adaptive control [6], [7]. Even if these strategies can deal with bounded disturbances, they do not take into account system constraints. From an implementation viewpoint, the class of mini/micro UAVs shares limitations with most computer embedded systems: limited space, limited power resources, increasing computation requirements, complexity of the applications, see e.g. [8]. Moreover, as explained in [9], most modern autopilots incorporate control algorithms to meet the increasingly stringent requirements for the mission accomplishment. The computational effort required by the control algorithms justifies the trend of most commercial autopilots to rely on PID controller, which require low computational power.

The combination of guidance and classical model predictive control (MPC) schemes for UAV applications, in particular for rotorcraft and multi-rotor systems, can be found in literature, see e.g. [10], [11], [12]. For example, in [13] a combination of a linear quadratic regulator and nonlinear MPC (NMPC) is proposed for high-level dynamics. A fixed-wing application is proposed in [14], in which high-level dynamics is controlled by a NMPC. Motivated by the considerations above, this paper develops a framework for FW-UAVs guidance and control, guaranteeing that the trajectories remain in a pre-defined neighborhood on the nominal one with high probability, thus resulting in a reduction of the time of flight. The aim of this paper is twofold: (i) real-time implementation of a sample-based stochastic MPC (SMPC), suitably designed for a customized or open source mini-UAV autopilot, in which the code can be written and implemented by the user, and (ii) a combination of guidance and control algorithms applied to an urban scenario, in which no-aggressive maneuvers are analyzed.

The proposed approach successfully applies in a real-time framework recent theoretical results obtained by the research team involved in the project: a guidance scheme developed in [15], and an offline sample based SMPC approach developed in [16], [17]. The use of this latter is important for several reasons: (i) the scheme guarantees probabilistic robust satisfaction of the imposed constraints, at a lower computational cost, and (ii) it is shown that such approach is feasible for medium-high sampling frequencies involved in the application. The effectiveness of a similar approach for real-time implementation was validated for a space application on an experimental testbed in [17]. In the application considered here, the goal is more challenging due to the much faster dynamics characterizing the UAV with respect to the spacecraft dynamics, and to the inclusion of atmospheric disturbances

(i.e. additive noise), model uncertainties (variations on speed  $V$  and mass  $m$ ) and platform inaccuracies (variations on the moments of inertia  $J$ ). The main theoretical results are presented in Section III, while the framework is tested by means of simulations in Section IV.

## II. FIXED-WING-UAV AND ON-BOARD SYSTEMS

The MH850 fixed-wing UAV in Fig. 1(a) is one of the platforms of the *MicroHawk family* of UAVs [18]. It was developed for low cost alpine surveillance missions, in which high altitude, low temperature and strong winds represent critical conditions for the mission. This platform has enough specific excess power to climb with non-marginal rates at altitude [18]. Usually, in piloted-mode, only 45% of the maximum power is required for level flight (at sea level). This FW-UAV has an expanded poly-propylene (EPP) wing and a sintered nylon fuselage, with trailing edge elevon (symmetric deflection for elevator  $\delta_e$  and antisymmetric for aileron  $\delta_a$ ), and is equipped by an electric brushless motor and a Lithium polymer battery of 5000 mAh with two cells 2S (7.4 V) with a discharge efficiency of 0.875. With this battery and a small camera as a payload, the endurance is about 45–50 min at an airspeed of 13.5 m/s and at an altitude of 100 m. A database including all the aerodynamic derivatives was employed to design the linear and nonlinear aircraft models [18].

The complete nonlinear model, as defined in [19], [20], is reported in [3]. It is composed by a set of nine equations, referred to a body reference frame, describing the forces  $[F_X F_Y F_Z]^T$ , moments  $[L M N]^T$ , linear velocity  $V = [u v w]$  (and, as a consequence, the angle of attack  $\alpha \simeq \frac{w}{V}$ ), angles  $[\phi \theta \psi]$  and angular speeds  $[p q r]$ , which characterize the flight condition of the aircraft. The aircraft control input is based on trailing edge elevon (symmetric deflection for elevator  $\delta_e$  and antisymmetric for aileron  $\delta_a$ ).

A custom-made autopilot is installed on-board, designed and produced at DIMEAS, Politecnico di Torino [21] (see Fig. 1(b)). Main characteristics comprehend an open architecture, the possibility to be reprogrammed in flight, and real-time telemetry. Sensors include GPS, barometric sensor, differential pressure sensor, and three-axis gyros and accelerometers. The CPU is the ATXMEGA256A3U-3U model with 256Kb flash memory and 16Kb of RAM. A Radiomodem Xbee Pro S1 is used for the communication link between the Ground Control Station (GCS) and the autopilot.

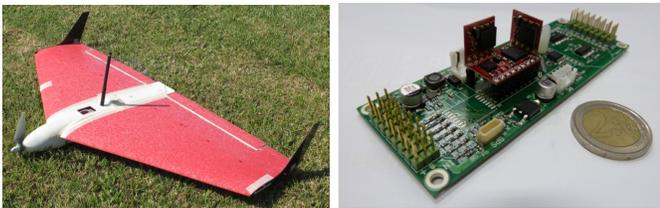


Fig. 1. The MH850 mini-UAV (left) and the custom-made autopilot board designed and produced in the Politecnico di Torino DIMEAS (right).

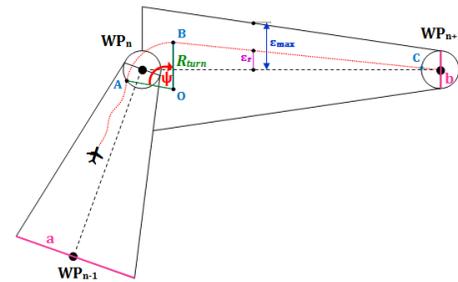
The system can be equipped with different on-board sensors, depending on the defined monitoring mission. For example,

for urban monitoring and risk mapping, a camera and video telemetry can be installed on-board.

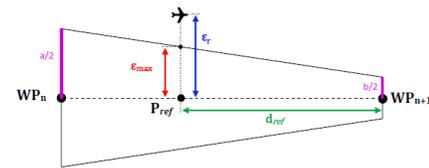
## III. GUIDANCE AND CONTROL STRATEGY

### A. Guidance Algorithm

The guidance algorithm adopted in this study is described in details in [15], in which some simplifying hypotheses are made to take into account the flash memory limitation of the autopilot micro controller. A given set of waypoints is considered, with assigned North, East and altitude coordinates. This set of waypoints includes the starting point, which is the point where the FW-UAV finishes its climb and the autonomous flight starts. The starting point and all waypoints are assumed to be at the same altitude; thus, a 2D path is considered. A trajectory smoother, that renders the assigned trajectory kinematically feasible in terms of speed and turn rate constraints, is implemented. For the evaluation of the performance of the guidance algorithm for aerial mapping, the cross-track error (CTE)  $\epsilon_r$  is calculated, to monitor the FW-UAV position with respect to the reference path [15]. A look-ahead or proximity distance is included to define the minimum distance of the UAV from the next waypoint. When the distance between the aircraft and the next waypoint is less than this pre-defined value, the waypoint is reached and the aircraft can move to the next waypoint.



(a) Guidance phases



(b) CTE and reference distances

Fig. 2. Guidance algorithm scheme.

Starting from these hypotheses, the guidance sequence can be divided into three phases, as described in [15]. The first phase is the waypoint approach. Assuming that the aircraft is flying with a fixed speed at a defined altitude, this phase is identified by the red dotted line before point A in Fig. 2(a). It is assumed that the waypoint has been reached when the FW-UAV flies into the imaginary circle centered in the waypoint WP<sub>n</sub> with radius equal to the defined proximity distance. In our case the proximity radius is 20 m, according to the MH850 dynamic constraints (in terms of minimum turn radius). In the second phase, the aircraft turns around the waypoint, according with the turn rate constraints. In Fig. 2(a), this segment is

identified by the red dotted curve between points A and B. Finally, the last phase is related to the straight flight that starts at the end of the last turn and ends at the beginning of the next turn, represented by the red spotted line between the points B and C in Fig. 2(a). In this phase, the CTE is applied [15]. Moreover, a no-correction zone is included, in which the corrections on the heading angle are imposed only when the FW-UAV CTE is larger than an assigned value (i.e. maximum acceptable CTE).

### B. Tracking-SMPC Algorithm Based on Offline Sampling

To control the FW-UAV dynamics, we develop upon the approach proposed in [16]. Our goal is to modify this initial framework in order to obtain a flexible reference tracking algorithm that still shares the same properties. To this end, let us consider a typical servo-problem, where the main objective is to track a reference signal  $r_k$ . Hence, the desired output, at steady state, shall be  $y_k = r_k$ . Here, we focus on state-tracking, hence we consider the full-information case, with  $C = I$ . This setup well suits the application case under consideration, in which we have access to all states of the FW-UAV dynamics.

Formally, we start from the following discrete-time system subject to both random noise and stochastic uncertainty

$$x_{k+1} = A(q_k)x_k + B(q_k)u_k + w_k, \quad (1)$$

with state  $x_k \in \mathbb{R}^n$ , control input  $u_k \in \mathbb{R}^m$ , output  $y_k \in \mathbb{R}^n$ , additive disturbance  $w_k \in \mathbb{R}^{m_w}$ , and parametric uncertainty  $q_k \in \mathbb{R}^{n_q}$ . We assume that  $A(q_k)$  and  $B(q_k)$ , of appropriate dimensions, are functions of the uncertainty  $q_k$ , which is bounded within the convex polytope  $\mathbb{Q}$ . These uncertainties are realizations at time  $k$  of independent and identically distributed (iid) multivariate real valued random variables  $Q_k$ . Moreover, the disturbance sequence  $(w_k)_{k \in \mathbb{N}_{\geq 0}}$  is assumed to be a realization of iid, zero-mean random variables  $(W_k)_{k \in \mathbb{N}_{\geq 0}}$ , with bounded and convex support  $\mathbb{W}$ .

To achieve tracking, at steady state we shall have  $x_{k+1} = x_k = r_k$ . Hence, the system dynamics (1) can be rewritten in terms of the state deviation  $\delta x_k$  with respect to the reference  $r_k$ , i.e.  $\delta x_k = x_k - r_k$ , as follows

$$\delta x_{k+1} = A(q_k)\delta x_k + B(q_k)u_k + w_k. \quad (2)$$

The system is subject to  $p_x$  individual chance-constraints on each  $\alpha$ -th state and hard constraints on the  $m$  inputs

$$\mathbb{P} \{ [H_x]_\alpha \delta x_{\ell|k} \leq [h_x]_\alpha \} \geq 1 - \epsilon_\alpha, \forall \ell \in \mathbb{N}_{\geq 0}, \alpha \in \mathbb{N}_1^{p_x} \quad (3a)$$

$$H_u u_{\ell|k} \leq h_u, \forall \ell \in \mathbb{N}_{> 0}, \quad (3b)$$

for each predicted  $\ell$  step ahead at time  $k$  and with  $\epsilon_\alpha \in (0, 1)$ . Each chance-constraint represents the maximum admissible deviation of the corresponding state, defined in compliance with the mission under analysis. Following one of the typical approaches in stabilizing MPC, we consider that a suitable terminal set and an asymptotically stabilizing control gain for (2) exists. Formally, we assume that there exists a terminal set  $\mathbb{X}_T = \{ \delta x_k | H_T \delta x_k \leq h_T \}$ , which is robustly forward invariant for (2) under the (given) control law  $u_k = K \delta x_k$ .

That is, given any  $\delta x_k \in \mathbb{X}_T$ , the state and input constraints (3) are satisfied and there exists  $P \in \mathbb{R}^{n \times n}$  such that

$$Q + K^T R K + \mathbb{E}[A_{cl}(q_k)^T P A_{cl}(q_k)] - P \preceq 0 \quad (4)$$

for all  $q_k \in \mathbb{Q}$ , with  $A_{cl}(q_k) \doteq A(q_k) + B(q_k)K$ , and with  $Q \in \mathbb{R}^{n \times n}$ ,  $Q \succ 0$ ,  $R \in \mathbb{R}^{m \times m}$ ,  $R \succ 0$ .

We then follow a dual-mode prediction scheme typical of robust MPC [22], and consider the design of a parametrized feedback policy of the form

$$u_{\ell|k} = K \delta x_{\ell|k} + v_{\ell|k}, \quad (5)$$

where for a given  $\delta x_{0|k} = \delta x_k$ , the correction terms  $\mathbf{v}_k \doteq \{v_{\ell|k}\}_{\ell \in \mathbb{N}_0^{T-1}}$  are determined by the SMPC algorithm as the minimizer of the expected finite-horizon cost

$$J_T(\delta x_k, \mathbf{v}_k) \doteq \mathbb{E} \left\{ \sum_{\ell=0}^{T-1} (\delta x_{\ell|k}^T Q \delta x_{\ell|k} + u_{\ell|k}^T R u_{\ell|k}) + \delta x_{T|k}^T P \delta x_{T|k} \right\}, \quad (6)$$

subject to constraints (3).

### C. Offline Uncertainty Sampling for SMPC

In this section, following the approach proposed in [16], [17], we derive *offline* a sample-based inner approximation for the chance constraint (3), obtaining linear constraints on  $\delta x_k, \mathbf{v}_k$  to be adopted for the optimization control problem. In particular, the considered SMPC scheme develops as follows: first, equation (1) is explicitly solved with prestabilizing input (5) for the predicted states  $\delta x_{1|k}, \dots, \delta x_{T|k}$ , and predicted inputs  $u_{0|k}, \dots, u_{T-1|k}$ . Simple algebraic manipulations show that it is possible to derive suitable transfer matrices

$$\Phi_{\ell|k}^0(q_k), \quad \Phi_{\ell|k}^v(q_k), \quad \Phi_{\ell|k}^w(q_k) \text{ and } \Gamma_\ell,$$

whose definition can be found in [16, App. A], such that

$$\delta x_{\ell|k}(q_k, \mathbf{w}_k) = \Phi_{\ell|k}^0(q_k) \delta x_k + \Phi_{\ell|k}^v(q_k) \mathbf{v}_k + \Phi_{\ell|k}^w(q_k) \mathbf{w}_k \quad (7a)$$

$$u_{\ell|k}(q_k, \mathbf{w}_k) = K \Phi_{\ell|k}^0(q_k) \delta x_k + (K \Phi_{\ell|k}^v(q_k) + \Gamma_\ell) \mathbf{v}_k + K \Phi_{\ell|k}^w(q_k) \mathbf{w}_k, \quad (7b)$$

where  $\mathbf{w}_k \doteq \{w_{\ell|k}\}_{\ell \in \mathbb{N}_0^{T-1}}$ . In these equations, both predicted states and inputs are function of the uncertainty  $q_k$  and the noise sequence  $\mathbf{w}_k$ . Then, we note that, from (7), the expected value in (6) can be computed offline, leading to a deterministic quadratic cost function of the form  $J_T(\mathbf{z}_k^T) = \mathbf{z}_k^T \tilde{S} \mathbf{z}_k$  in the variable  $\mathbf{z}_k \doteq [\delta x_k^T, \mathbf{v}_k^T]^T$  (see [16, Appendix A] for details and definition of  $\tilde{S}$ ).

Second, as described in [16], an estimate of the chance constraint sets corresponding to linear constraints (3) may be constructed extracting a number of iid samples  $q^{(i)}$  from  $Q_k$ , which is defined exploiting results from statistical learning theory [23]. This is obtained by extracting, for  $\ell \in \mathbb{N}_0^{T-1}$ ,  $N_\ell^x$  iid samples of the uncertainty and of the noise, according to their respective distribution, obtaining the following *tightened* linear constraints

$$H_x \delta x_{\ell|k}(q^{(i_x)}, \mathbf{w}^{(i_x)}) \leq h_x, \quad (8)$$

for  $i_x \in \mathbb{N}_1^{N_\ell^x}$  and  $\ell \in \mathbb{N}_0^{T-1}$ . In a similar way, the hard input constraints and the terminal constraints can be approximated

by drawing offline respectively  $N_\ell^u$  and  $N_T$  samples, obtaining linear constrain

$$H_u u_{\ell|k}(q^{(i_u)}, \mathbf{w}^{(i_u)}) \leq h_u, \quad (9)$$

$$H_T \delta x_{T|k}(q^{(i_T)}, \mathbf{w}^{(i_T)}) \leq h_T, \quad (10)$$

In [17] the reader can find details on the construction of the matrices involved.

The above linear constraints constitute a probabilistically guaranteed approximation of the actual constraint observed online. In particular, it was shown in [24] that if, for given probabilistic parameters  $\epsilon_\alpha, \epsilon_\beta, \epsilon_\gamma \in (0, 0.14)$ , and  $\delta \in (0, 1)$ , the samples are drawn such that  $N_\ell^x \geq \tilde{N}(n + \ell m, \epsilon, \delta)$ ,  $N_\ell^u \geq \tilde{N}(n + \ell m, \epsilon_\beta, \delta)$ ,  $N_T \geq \tilde{N}(n + T m, \epsilon_\gamma, \delta)$ , with  $\tilde{N}(\cdot, \cdot, \cdot)$  given in [17]. The linear constraints (8), (9), (10), possibly after constraint reduction, can be summarized in the following linear constraint set

$$\mathbb{D} = \left\{ \mathbf{z}_k \mid \tilde{H} \mathbf{z}_k \leq \tilde{h} \right\}. \quad (11)$$

Moreover, as in [16], a first step constraint set is added to (11), to ensure robust recursive feasibility, given by

$$\mathbb{D}_R = \left\{ \delta x_k, \mathbf{v}_k \mid H_\infty A_{cl}(q_k) \delta x_k + H_\infty B(q_k) v_{0|k} \leq h_\infty - H_\infty w_{0|k}, j \in \mathbb{N}_1^{N_c} \right\} \quad (12)$$

with  $A(q_k), B(q_k)$  from Assumption 1 and  $A_{cl}(q_k) = A(q_k) + B(q_k)K$ . The final set of linear constraints to be employed in online implementation is thus given by the intersection of the sets  $\mathbb{D}$  and  $\mathbb{D}_R$ , defined in (11) and (12) respectively, which could still benefit from further offline constraint reduction.

#### IV. SIMULATION RESULTS

The SMPC algorithm described in the previous section is used to control a FW-UAV, whose systems-equation derivations are reported in an extended version of this work [3, Sec. II]. In particular, we consider a linear case, in which the longitudinal and lateral-directional motions result to be decoupled, and each of them can be modeled in the standard discrete time-invariant state-space formulation as (2), where  $A, B$  represent the discrete-time state and input matrices respectively, obtained discretizing the corresponding continuous ones derived starting from the equations in [25]. The SMPC controller is adopted to control both the longitudinal as well as the lateral-directional dynamics. The states are fully measured and the angle of attack  $\alpha$  is evaluated from the Pitot tube sensor (i.e. differential pressure sensor) and from the accelerometers.

The controller provides the control actions in terms of throttle  $\Delta T$ , and elevator and aileron deflections,  $\delta_e$  and  $\delta_a$  respectively. Finally, a PID controller is adopted for the heading angle  $\psi$ , obtaining the reference roll angle  $\phi_{ref}$ . Moreover, the other reference signals,  $u_{ref}$ ,  $h_{ref}$  and  $\psi_{ref}$ , are provided by the guidance algorithm as a function of the identified waypoints, as described in Section III-A.

As anticipated before, a neighborhood in the Downtown Miami's Brickell Financial District, Miami, FL, USA

( $25^\circ 45' 31.3''\text{N}$ ,  $80^\circ 11' 28.3''\text{W}$ ), around the Four Season Hotel and Tower, has been chosen for the urban scenario. Indeed, the Miami police enforcement is already evaluating the possibility to exploit FW-UAVs for drug interdiction, patrol missions, criminal surveillance as well as search-and-rescue operations, but only in life-threatening situations in imminent danger to life or properties.

For the analyzed scenario, the initial conditions have been set as follows: (i) altitude  $h_0 = 100$  m, (ii) airspeed  $V_0 = 13.5$  m/s, (iii) angle of attack  $\alpha_0 = 5.18$  deg, and (iv) ramp angle  $\gamma_0 = 0$  deg. The multi-rate simulator adopted to validate the guidance and control strategy is characterized by two different sample times, as follows: (i) system sample time 0.01 s; (ii) SMPC sample time = 0.1 s. For what concern the main SMPC parameters, they are reported in Table I, in addition to the following settings: (i) prediction horizon  $T = 15$ ; (ii) number of samples  $N_{tot} = 95,472$ ; (iii)  $\epsilon = 0.95$ ; and (iv)  $\delta = 10^{-3}$ . Moreover, the robustly stabilizing feedback gain matrix  $K$  has been evaluated offline exploiting typical robust tools. The results presented in the following have been obtained exploiting a simulator developed with MATLAB/Simulink 2016b run in quasi real-time (i.e. 1 s of simulation in about 1 s) over a Intel Core i7 – 7500U CPU @2.70 GHz with 16 GB of RAM and 512 GB solid-state drive.

TABLE I  
SMPC WEIGHT MATRICES.

Parameter	Value
$\text{diag}(Q_{long})$	$[10^6, 4 \times 10^1, 4 \times 10^1, 10^5, 10^5]$
$\text{diag}(R_{long})$	$[4 \times 10^2, 3 \times 10^{-3}]$
$\text{diag}(Q_{lat})$	$[10^1, 10^1, 10^1, 10^6]$
$R_{lat}$	$10^6$

In terms of uncertainties included in the model, a  $\pm 30\%$  variation of the following parameters has been considered, in addition to those parametric uncertainties ascribable to neglected nonlinearities: (i) cruise speed, with respect to the guidance and control flexibility to different flight conditions; (ii) FW-UAV mass, evaluating the possibility to adopt the same guidance and control strategy on slightly different vehicles; (iii) inertia, due to the manufacturing process. Moreover, we consider the presence of fixed-direction wind turbulence, representing a bounded persistent disturbance affecting the system and modeled as random bounded variable with uniform distribution and maximum intensity of  $\pm 1$  m/s. The considered wind direction is reported in Fig. 3.

TABLE II  
ADDITIVE DISTURBANCES CONSIDERED.

LONG. disturbance	Disturbance value	LAT.-DIR. disturbance	Disturbance value
$d_u$ [m/s]	1.5	$d_v$ [m/s]	1.5
$d_\alpha$ [rad]	$10^{-2}$	$d_p$ [rad/s]	$10^{-2}$
$d_\theta$ [rad]	$10^{-2}$	$d_r$ [rad/s]	$10^{-2}$
$d_q$ [rad/s]	$10^{-2}$	$d_\phi$ [rad]	$10^{-2}$
$d_h$ [m]	$10^{-1}$		

Additional external disturbances have been considered affecting the other states, with an analogous uniform distribution

even if with a minor impact. The reader is referred to Table II for further details.

TABLE III  
STATE CONSTRAINT POLYTOPE VERTEXES.

LONG. variable	Constraint boundaries	LAT.-DIR. variable	Constraint boundaries
$\delta_u$ [m/s]	$\pm 2$	$\delta_v$ [m/s]	$\pm 2$
$\delta_\alpha$ [rad]	$\pm 10^{-1}$	$\delta_p$ [rad/s]	$\pm 10^{-1}$
$\delta_q$ [rad/s]	$\pm 10^{-1}$	$\delta_r$ [rad/s]	$\pm 10^{-1}$
$\delta_\theta$ [rad]	$\pm 10^{-1}$	$\delta_\phi$ [rad]	$\pm 10^{-1}$
$\delta_h$ [m]	$\pm 2$		

Considering a time-varying reference  $r_k$  provided by the guidance algorithm with respect to the waypoints setup for each scenario, the state deviations  $\delta x_k$  have been constrained to be bounded within a polytope, which vertexes are reported in Table III. On the other hand, hard constraints have been imposed on the control variables in compliance with the vehicle mechanical limitations. Hence, the throttle  $\Delta T$  can vary between 0% and 100%, whereas the elevator and aileron deflection saturation was fixed to  $\pm 20\%$ . The waypoints identification is strictly bounded to the mission scenario as well as the payload which the FW-UAV is equipped with. For the Miami Downtown's Brickell Financial District, the path identified by the waypoints has been simulated 20 times and the results obtained are represented in Fig. 3.

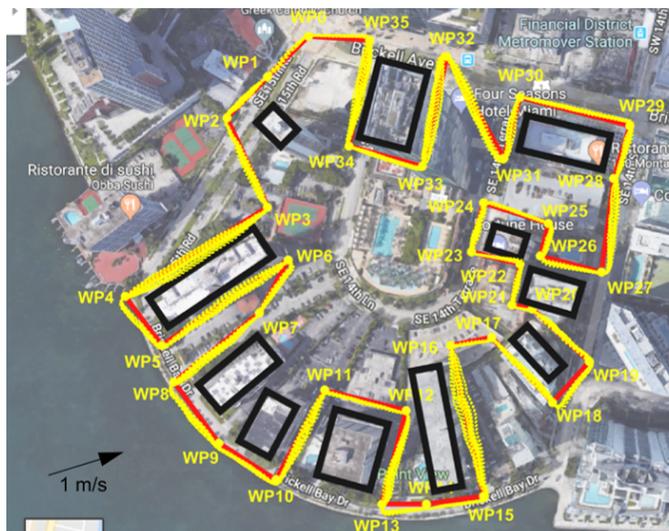


Fig. 3. 20 different piecewise-linear trajectories over Downtown Miami's Brickell Financial District, around the Four Season Hotel and Tower - 2D trajectories.

In this case, the selected waypoints have been identified with respect to the buildings-to-be-monitored location and the safety-area around them, set to be compliant with (possible) risk-mitigation rules. Considering a polygonal chain, which vertexes are identified by 36 waypoints, the ideal path (red line) as well as the simulated ones (yellow lines) are represented in Fig. 3.

In general, we can notice a "tangible" effectiveness of the proposed control strategy since the FW-UAV shows remarkable tracking capabilities with respect to the trajectory

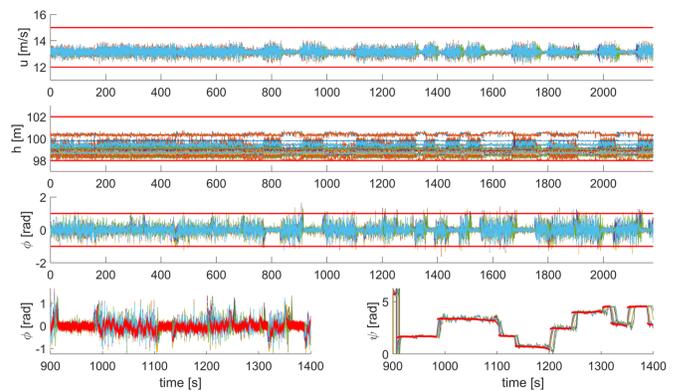


Fig. 4. Airdspeed, altitude and roll angle variation along 20 trajectories plus a zoom-in on both roll and heading angles.

to follow. This is confirmed by Fig. 4 where the tracking capabilities are represented with respect to four state variables: longitudinal component of the velocity  $u$ , altitude  $h$ , roll angle  $\phi$ , and heading angle  $\psi$ , within the corresponding constraint boundaries (red bold lines). As we can notice, only few values of the roll angle exceed the constraints, which is still acceptable with respect to the probability of constraint violation considered in this study.

In order to provide a preliminary assessment of the SMPC control effort, the battery discharge has been evaluated, and is represented in Fig. 5 (upper graph) as a percentage of the total battery energy, i.e. 37 kJ, considering an initial electric charge of 5000 mAh and a nominal voltage of 7.4 V. The actual current and voltage required by the system to the batteries have been estimated as a function of the torque provided by the propeller, following the Glauert's blade element theory, with respect to the throttle input control, see [3] for technicalities.

Moreover, the computational cost has been estimated for each simulation, by comparing the Elapsed Real (wall-clock) Time, obtained exploiting the toolbox [26], with respect to the simulated time (2180 s). As we can notice from Fig. 5 (lower scatter plot), the elapsed time is lower than the simulated in the majority of cases, with an average percentage less than 75%, thus implying real-time implementability of the proposed scheme on processors with computational capabilities similar to those adopted in the simulations, except for two extreme cases, where the use of a non-dedicated hardware affected the results. Future software-in-the-loop simulations will show if the proposed scheme is real-time implementable on the targeted hardware or if it would be required to add an onboard companion computer.

To complete the SMPC scheme validation, the proposed controller effectiveness has been compared with a classical MPC, still considering the presence of a fixed-direction wind turbulence of 1 m/s. Fig. 6 represents both the 3D trajectories obtained, i.e red for SMPC and black for LQMPC, considering the same MPC parameters in terms of Q and R matrices as well as prediction horizon. Both control approaches show an effective response in the presence of wind turbulence. However, the LQMPC control capabilities are significantly more affected by the presence of disturbance than the SMPC

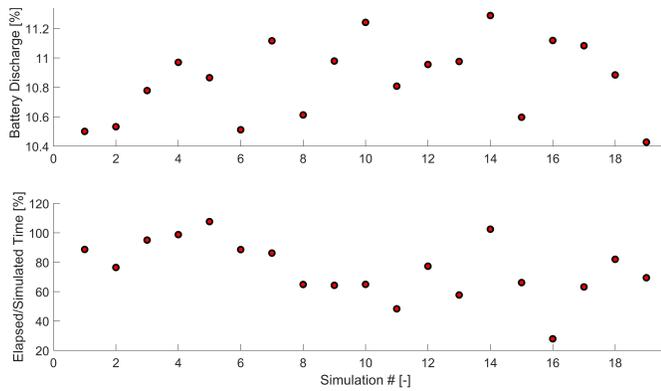


Fig. 5. Battery discharge (upper) and computational effort (lower). We observe a high variability, due to the fact that simulations were not performed on a dedicated hardware, running a real-time code: the results were hence affected by the different external computational loads requested to the computer used to perform the simulations.

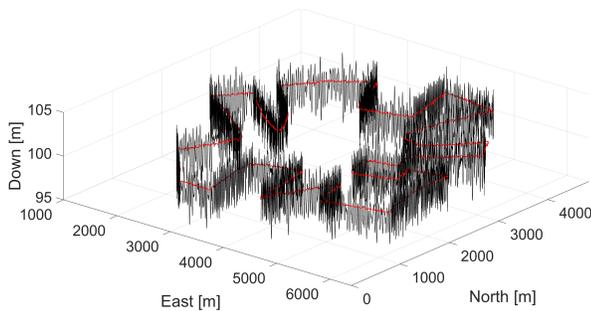


Fig. 6. LQMP (black line) and SMPC (red line) 3D trajectories.

ones. Moreover, to properly control the FW-UAV during the trajectory, it was necessary to reduce the LQMP sample time to 0.01 s instead of 0.1 s. Hence, the computation cost raised to 140%, i.e. the elapsed time is about 3057 s. Analogously, also the control effort results to be much higher, with a residual battery energy of about 46%.

## V. CONCLUSIONS

In this research, a combination of waypoint guidance algorithm and a tracking SMPC based on offline sampling is proposed for FW-UAVs, in the presence of wind turbulence and parametric uncertainties. In particular, the effectiveness of the strategy proposed has been validated via simulations considering a patrolling mission over a Miami's Downtown urban area. The results obtained shows the SMPC capability of guaranteeing good stability performance of the platform and remarkable tracking capabilities. Moreover, the effectiveness of the SMPC approach has been demonstrated through its comparison with a classical MPC scheme.

## REFERENCES

[1] K. P. Valavanis and G. J. Vachtsevanos, *UAV Applications: Introduction*. Dordrecht: Springer Netherlands, 2015, pp. 2639–2641.  
 [2] —, *Handbook of Unmanned Aerial Vehicles*. Springer Publishing Company, Incorporated, 2014.  
 [3] M. Mammarella, E. Capello, and F. Dabbene, “Sample-based SMPC tracking of fixed-wing UAV: multi-scenario mapping,” in *arXiv:cs.SY/2018*.

[4] R. Garca, F. Rubio, and M. Ortega, “Robust PID control of the quadrotor helicopter,” in *2nd IFAC Conf. on Advances in PID Control*, 2012, pp. 229 – 234.  
 [5] E. Capello, D. Sartori, G. Guglieri, and F. Quagliotti, “Robust assessment for the design of multi-loop proportional integrative derivative autopilot,” *IET Control Theory Applications*, vol. 6, no. 11, pp. 1610–1619, July 2012.  
 [6] B. J. Bialy, J. Klotz, K. Brink, and W. E. Dixon, “Lyapunov-based robust adaptive control of a quadrotor UAV in the presence of modeling uncertainties,” in *2013 American Control Conf.*, pp. 13–18.  
 [7] Z. T. Dydek, A. Annaswamy, and E. Lavretsky, “Adaptive control of quadrotor UAVs: A design trade study with flight evaluations,” *IEEE Trans. on Control Systems Technology*, vol. 21, no. 4, pp. 1400–1406, 2013.  
 [8] E. L. Pastor, J. Lopez, and P. Royo, “A hardware/software architecture for UAV payload and mission control,” *IEEE/AIAA Digital Avionics Systems Conf.*, pp. 1–8, 2006.  
 [9] H. Chao, Y. Cao, and Y. Chen, “Autopilots for small unmanned aerial vehicles: A survey,” *Int. J. of Control, Automation and Systems*, vol. 8, no. 1, pp. 36–44, 2010.  
 [10] F. Gavilan, R. Vazquez, and E. F. Camacho, “An iterative model predictive control algorithm for uav guidance,” *IEEE Trans. on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2406–2419, 2015.  
 [11] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes, “Robust model predictive flight control of unmanned rotorcrafts,” *J. of Intelligent & Robotic Systems*, vol. 81, no. 3, pp. 443–469, Mar 2016.  
 [12] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart, *Model Predictive Control for Trajectory Tracking of Unmanned Aerial Vehicles Using Robot Operating System*. Springer Int. Publishing, 2017, pp. 3–39.  
 [13] M. Kamel, K. Alexis, M. Achteik, and R. Siegwart, “Fast nonlinear model predictive control for multicopter attitude tracking on so(3),” in *2015 IEEE Conf. on Control Applications (CCA)*, Sept 2015, pp. 1160–1166.  
 [14] T. Stastny, A. Dash, and R. Siegwart, “Nonlinear MPC for fixed-wing UAV trajectory tracking: Implementation and flight experiments,” in *AIAA Guidance, Navigation, and Control Conf.*, 01 2017.  
 [15] E. Capello, G. Guglieri, and G. Ristorto, “Guidance and control algorithms for mini-UAV autopilots,” *Aircraft Engineering and Aerospace Technology*, vol. 89, no. 1, pp. 133–144, 2017.  
 [16] M. Mammarella, E. Capello, M. Lorenzen, F. Dabbene, and F. Allgower, “A general sampling-based SMPC approach to spacecraft proximity operations,” in *2017 IEEE 56th Annual Conf. on Decision and Control (CDC)*, Dec 2017, pp. 4521–4526.  
 [17] M. Mammarella, M. Lorenzen, E. Capello, H. Park, F. Dabbene, F. Allgower, G. Guglieri, and M. Romano, “An offline-sampling SMPC framework with application to automated space maneuvers,” in *arXiv:cs.SY/1803.01652*, 2018.  
 [18] E. Capello, G. Guglieri, P. Marguerettaz, and F. Quagliotti, “Preliminary assessment of flying and handling qualities for mini-UAVs,” *J. of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 43–61, Jan 2012.  
 [19] B. Etkin and L. Reid, *Dynamics of Flight: Stability and Control*. New York: John Wiley and Sons, 1996.  
 [20] B. Stevens and F. Lewis, *Aircraft Control and Simulation*. New York: John Wiley and Sons, 2003.  
 [21] E. Capello, G. Guglieri, F. Quagliotti, and D. Sartori, “Design and validation of an  $l_1$  adaptive controller for mini-UAV autopilot,” *J. of Intelligent & Robotic Systems*, vol. 69, no. 1, pp. 109–118, Jan 2013.  
 [22] B. Kouvaritakis and M. Cannon, *Model predictive control*. Springer, 2016.  
 [23] M. Vidyasagar, *Learning and generalisation: with applications to neural networks*. Springer Science & Business Media, 2013.  
 [24] M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgower, “Stochastic MPC with offline uncertainty sampling,” *Automatica*, vol. 81, pp. 176–183, 2017.  
 [25] M. Mammarella and E. Capello, “A robust MPC-based autopilot for mini UAVs,” in *2018 Int. Conference on Unmanned Aircraft Systems (ICUAS’18)*, June 2018.  
 [26] G. Vallabha, *Real-Time Pacer MATLAB/Simulink Toolbox (ver. 1.0.0.1)*, 2017.