2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

Constrained Control of UAVs in Geofencing Applications

Elie Hermand, Tam W. Nguyen, Mehdi Hosseinzadeh, and Emanuele Garone

Abstract—**This paper focuses on the constrained control of UAVs in geofencing applications. Although geofence systems are becoming more attractive as a research topic, most works are focusing on defining the boundaries of the admissible geographical region without addressing the control issues and boundary-handling problems. In this paper, we propose a constrained control scheme to steer an UAV to the desired position while ensuring constraints satisfaction at all times. To do so, we make use of the recently introduced Explicit Reference Governor framework. The proposed scheme is validated through extensive experimental studies carried out in a laboratory environment.**

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have attracted the attention of several researchers in the last decade as their applications have significantly widened following technological and theoretical improvements. In particular, UAVs are nowadays widely used in *e.g.,* surveillance [1], monitoring [2], photography [3], agriculture [4], and search and rescue missions [5]. As UAVs are increasingly used in our daily life, regulations on flight operations have become crucial to enforce safety and security of aerial missions. This necessitates the development of technologies that allow UAVs to safely navigate according to the regulations.

Geofencing is a technique that defines virtual boundaries in a specific geographical area (see Fig. 1), which has recently attracted some interest in the research community [6], [7]. In the presented geofence systems, once the defined boundaries are violated, the reaction is limited to alerting the pilot or cutting the power of the UAV [8]. This obviously hampers the efficiency of the geofence system.

From a technical viewpoint, preventing UAVs from violating the boundaries defined by the geofence system can be considered as a constrained control problem. Constrained control addresses the problem of enforcing constraints satisfaction at all times while ensuring that control objectives are achieved.

Recently, the Explicit Reference Governor (ERG) scheme was introduced as a solution to constrained control problems [9]. The ERG scheme is an *add-on* unit which suitably modifies the derivative of the applied reference such that constraints are enforced at all times. This scheme has been proved to be effective in the field of aerial robotics [10]–[13], due to its simplicity and low computational efforts compared to optimization-based schemes (*e.g.,* Model Predictive Control [14]).

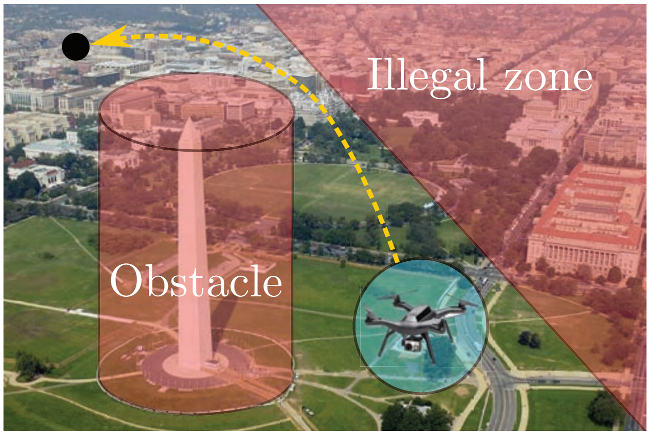

Fig. 1. Illustration of geofencing applications for UAVs.

The main goal of this paper is to bridge the gap between geofencing applications for UAVs and the constrained control concept. More specifically, this paper proposes a constrained control scheme based on the ERG framework, which will be applied to a UAV evolving in a constrained environment. The scheme consists in two parts: (i) pre-stabilizing the UAV, and (ii) using the Lyapunov theory to enforce constraints satisfaction. To the best of the authors' knowledge, no effective work is presented in the literature on how to control UAVs so to prevent them from violating the boundaries defined by geofence systems. One of the most remarkable features of ERG is that its very low computational efforts, which fits well with geofencing applications.

This paper is organized as follows. First, the problem is stated, the typical constraints in geofence systems are formulated, and the main control objectives are defined. Afterwards, an ERG-based constrained control scheme is proposed. Finally, the proposed scheme is validated through extensive experimental studies carried out on an *AR Drone*.

II. PROBLEM STATEMENT

Consider a fixed propeller quadrotor of mass $m > 0 \in \mathbb{R}$ flying in a 3-D constrained environment. The UAV is subject to the gravity force $g\hat{z}$[1], and the generalized coordinates are the position of the UAV $p := \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T \in \mathbb{R}^3$, the

This research has been funded by the FNRS MIS "Optimization-free Control of Nonlinear Systems subject to Constraints", Ref. F.4526.17, and by the FRIA "COPTERS project", Ref. F-3-5-5/FRIA/FC-12687. This work has also been supported by the European Commission under the grant agreement number 774571 Project PANTHEON.

The authors are with the Service d'Automatique et d'Analyse des Systèmes (SAAS), Université libre de Bruxelles (ULB), Brussels, Belgium, emails: ehermand@ulb.ac.be, tam.nguyen@ieee.org, mehdi.hosseinzadeh@ieee.org, egarone@ulb.ac.be

[1] The gravity acceleration is $g = 9.81$ [m/s^2]. We define $\hat{z} := \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ as the unit vector aligned with the gravity and we conventionally define $\hat{z}$ as pointing downwards.

velocity of the UAV $\dot{p} := \begin{bmatrix}\dot{p}_x & \dot{p}_y & \dot{p}_z\end{bmatrix}^T \in \mathbb{R}^3$, and the attitude (quaternion) of the UAV $q \in \mathbb{H}$. We assume the inputs of the system to be the thrust $T \geq 0 \in \mathbb{R}$ and the desired quaternion $q_C := \begin{bmatrix}q_{C,0} & q_{C,v}\end{bmatrix}^T \in \mathbb{H}$, where $q_{C,0} \in \mathbb{R}$ and $q_{C,v} \in \mathbb{R}^3$ are the real and the imaginary parts of q_C, respectively. In this paper, we use the classical model of a UAV presented in [15]. Since in this model the dynamics of the attitude is independent of the position dynamics, without loss of generality we assume that the UAV attitude q is already pre-stabilized to reach q_C.

In general, in geofencing applications we have two types of constraints (or boundaries):

1) **Wall constraints**: The constraints are a convex polytope composed of the collection of linear inequalities as

$$W = \{p \mid c^T p + d \geq 0\}, \tag{1}$$

where $c \in \mathbb{R}^{3\times n_w}$ and $d \in \mathbb{R}^{n_w}$, with $n_w \in \mathbb{N}_0$ the number of wall constraints. This type of constraint represents the boundaries of the authorized flying zone to avoid UAVs entering illegal zones.

2) **Obstacle constraints**: The constraints defined for the i-th obstacle are vertical infinite cylinders

$$O = \{p \mid (p_x - x_{o,i})^2 + (p_y - y_{o,i})^2 \geq r_i^2\}, \tag{2}$$

where $i = 1, \ldots, n_o$ with $n_o \in \mathbb{N}_0$ the number of obstacles, $\begin{bmatrix}x_{o,i} & y_{o,i}\end{bmatrix}^T \in \mathbb{R}^2$ represents the center of the i-th obstacle and $r_i \in \mathbb{R}_0^+$ is its radius. This type of constraints represents objects or zones that should be avoided, such as skyscrapers, antennas, jails, and military installations. Note that constraints (2) are located inside the boundaries defined by constraints (1).

The main goal of this paper is to utilize the so-called ERG framework so to handle the constraints that exist in geofencing applications. In particular, the UAV has to be controlled such that, starting from a suitable initial position $p_0 := \begin{bmatrix}p_{0,x} & p_{0,y} & p_{0,z}\end{bmatrix}^T \in \mathbb{R}^3$, it safely tends to the desired position $p_d := \begin{bmatrix}p_{d,x} & p_{d,y} & p_{d,z}\end{bmatrix}^T \in \mathbb{R}^3$ without violating constraints (1)-(2).

III. CONTROL ARCHITECTURE

This section discusses the development of a constrained control scheme for geofencing applications in detail. The first step is to pre-stabilize the system; then augmenting it with the ERG to enforce constraints satisfaction. The complete scheme is illustrated in Fig. 2.

A. Pre-stabilizing the UAV

This section proposes a control law for the inputs T and q_C. The position of the UAV can be stabilized using the control law proposed in [12]. In particular, the thrust T is designed as

$$T = \|F\|, \tag{3}$$

where $F := \begin{bmatrix}F_x & F_y & F_z\end{bmatrix}^T \in \mathbb{R}^3$ is the total required force obtained through a Proportional-Derivative (PD) control law with gravity compensation as

$$F = -K_p(p - p_d) - K_d\dot{p} - mg\hat{z}, \tag{4}$$

where $K_p > 0 \in \mathbb{R}^{3\times 3}$ and $K_d > 0 \in \mathbb{R}^{3\times 3}$ are the proportional and derivative gains, respectively.

For what concerns the tracking of the desired attitude q_C, we use the following control law [12]:

$$q_{C,0} = \cos\left(\frac{\alpha}{2}\right), \quad q_{C,v} = \frac{\sin\left(\frac{\alpha}{2}\right)}{\sqrt{F_x^2 + F_y^2}} \begin{bmatrix}F_y & -F_x & 0\end{bmatrix}^T, \tag{5}$$

where $\alpha = \arctan\left(\sqrt{F_x^2+F_y^2}/F_z\right)$.

Note that for sufficiently fast inner loop dynamics and using the control law (3)-(5), it is possible to prove that the system is asymptotically stable and that it becomes a linear system [12]. In the following, (A, B, C, D) is a state-space realization of the aforementioned pre-stabilized linear system.

B. ERG Implementation

Once the position and attitude of the UAV is stabilized, the next step is to add the constraint-handling capability to enforce constraints (1)-(2). This will be done by using the ERG framework to generate the auxiliary reference $v(t) \in \mathbb{R}^3$ so that the trajectories of the pre-stabilized system are always contained in a suitable invariant set. As shown in [16], an intuitive choice for the invariant set is the invariant level set defined by the Lyapunov theory. Thus, to ensure satisfaction of constraints (1)-(2) at all times, it is sufficient to manipulate the auxiliary reference $v(t)$ so that the Lyapunov function is always smaller than a suitably defined upper-bound. This can be done by manipulating the auxiliary reference $v(t)$ according to the following differential equation

$$\dot{v}(t) = \Delta(x(t), v(t))\rho(p_d(t), v(t)), \tag{6}$$

with

$$\Delta(x(t), v(t)) = \kappa \min_i \left(\Gamma_i(v(t)) - V_i(x(t), v(t))\right), \tag{7}$$

where $\kappa > 0$ is a tuning parameter, $x(t) = \begin{bmatrix}p_x(t) & p_y(t) & p_z(t) & \dot{p}_x(t) & \dot{p}_y(t) & \dot{p}_z(t)\end{bmatrix}^T \in \mathbb{R}^6$ is the state of the pre-stabilized system, and $i = 1, 2$ corresponds to constraints (1) and (2), respectively. Also, $\Delta(x(t), v(t))$ and $\rho(p_d(t), v(t))$ are the two fundamental components of the ERG scheme, called the Dynamic Safety Margin (DSM) and the Attraction Field (AF), respectively (see Fig. 2). Note that $\Gamma_i(v(t))$, $i = 1, 2$ is determined such that $V_i(x(t), v(t)) \leq \Gamma_i(v(t))$ implies that the corresponding constraint is satisfied.

Using the invariant Lyapunov level set to build the DSM, $\Gamma(v(t))$ can be interpreted as the maximum value of the Lyapunov function such that the Lyapunov level set touches the constraints (1)-(2) but does not violate them.

As shown in [9], determining $\Gamma(v(t))$ at each time is an optimization problem which usually needs to be solved online. In general, an analytic solution does not exist for the optimization problem. However, for some special cases of Lyapunov functions and constraints, one can find an analytic parameterized solution. In particular, for linear systems subject

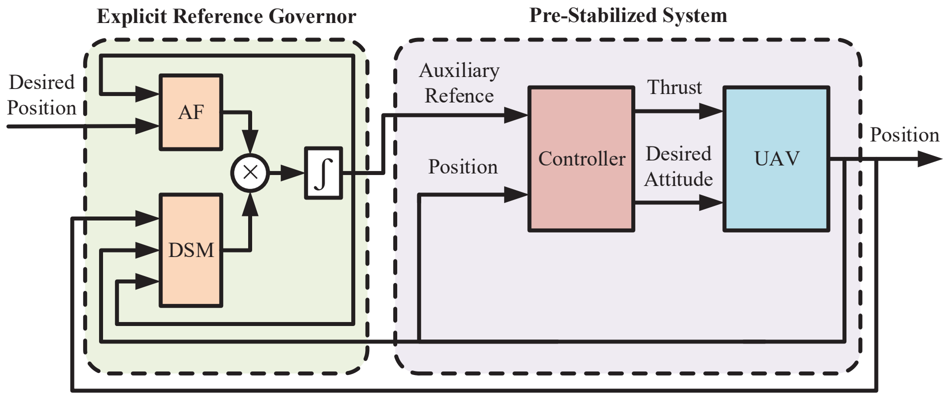

Fig. 2. Proposed ERG-based scheme for geofence applications.

to linear constraints (1) with a quadratic Lyapunov function in the following form:

$$V(x(t),v(t)) = \left(x(t)-\bar{x}_{v(t)}\right)^T P \left(x(t)-\bar{x}_{v(t)}\right), \tag{8}$$

where $P = P^T > 0$, and $\bar{x}_{v(t)} \in \mathbb{R}^n$ denotes the equilibrium point associated to $v(t)$, it can be shown [16] that the optimal $\Gamma(v(t))$ can be determined as

$$\Gamma(v(t)) = \frac{\left(c^T \bar{x}_{v(t)} + d\right)^2}{c^T P_w^{-1} c}. \tag{9}$$

where P_w is the solution of the following optimization problem:

$$\begin{cases} P_w = & \arg\min \log\det(P) \\ \text{s.t.} & A^T P + PA \leq 0 \\ & P \geq cc^T \\ & P > 0 \end{cases} \tag{10}$$

Also, in the case of quadratic Lyapunov functions, it can be shown that for each obstacle given in constraints (2), $\Gamma(v(t))$ can be determined as [17]

$$\Gamma(v(t)) = \frac{\left(\left(\nabla O_i|_{\tilde{x}_{v(t)}}\right)^T \left(\bar{x}_{v(t)} - \tilde{x}_{v(t)}\right)\right)^2}{\left(\nabla O_i|_{\tilde{x}_{v(t)}}\right)^T P^{-1} \left(\nabla O_i|_{\tilde{x}_{v(t)}}\right)}. \tag{11}$$

where O_i represents the i-th obstacle given in (2), and

$$\tilde{x}_{v(t)} = -\frac{x_{c,i} - \bar{x}_{v(t)}}{\|x_{c,i} - \bar{x}_{v(t)}\|} r + x_{c,i} \tag{12}$$

with $x_{c,i} = \begin{bmatrix} x_{o,i} & y_{o,i} & \bar{x}_{v(t),3} & 0 & 0 & 0 \end{bmatrix}^T$ in which $\bar{x}_{v(t),3}$ is the third entry of $\bar{x}_{v(t)}$.

For what regards the AF, it can be designed by decoupling into an attraction and a repulsion term as

$$\rho(p_d(t),v(t)) = \rho_a(p_d(t),v(t)) + \rho_r(p_d(t),v(t)), \tag{13}$$

where $\rho_a(p_d(t),v(t))$ is a vector field which points towards the desired position, and $\rho_r(p_d(t),v(t))$ is a vector field which points away from the constraints. For the attraction term $\rho_a(p_d(t),v(t))$, the most intuitive choice is

$$\rho_a(p_d(t),v(t)) = \frac{p_d(t)-v(t)}{\max\{\|p_d(t)-v(t)\|,\eta\}}, \tag{14}$$

where $\eta > 0$ is a smoothing factor. The repulsion term $\rho_r(p_d(t),v(t))$ is split into two terms as

$$\rho_r(p_d(t),v(t)) = \rho_{r,1}(p_d(t),v(t)) + \rho_{r,2}(p_d(t),v(t)), \tag{15}$$

where $\rho_{r,1}(\cdot)$ and $\rho_{r,2}(\cdot)$ are repulsion terms associated to constraints (1) and (2), respectively. Repulsion terms can be considered as

$$\rho_{r,1}(\cdot) = \max\left\{\frac{\zeta - c^T p - d}{\zeta - \delta}, 0\right\} \frac{c}{\|c\|}, \tag{16}$$

$$\rho_{r,2}(\cdot) = \sum_{i=1}^{n_o} \max\left\{\frac{\zeta - \phi_i + r_i}{\zeta - \delta}, 0\right\} \begin{bmatrix} \frac{p_x - x_{0,i}}{\phi_i} \\ \frac{p_y - y_{0,i}}{\phi_i} \\ 0 \end{bmatrix}, \tag{17}$$

where $\zeta > \delta > 0$ and $\phi_i = \sqrt{(p_x - x_{o,i})^2 - (p_y - y_{o,i})^2}$. Note that the repulsion terms (16)-(17) guarantee that $c^T p + d \geq \delta$ and $(p_x - x_{o,i})^2 + (p_y - y_{o,i})^2 \geq (r_i + \delta)^2$ at all times.

IV. EXPERIMENTAL RESULTS

The proposed scheme is applied to an *AR Drone* of mass $m = 0.47$ [kg] using the control law (3) and (5) combined with (6). The gains K_p and K_d chosen for the *AR Drone* are given in Appendix. Furthermore, for the experiments we have chosen $\kappa = 2$, $\zeta = 0.5$, and $\delta = 0.4$.

The general setup of the experimental environment is illustrated in Fig. 3. It consists of the optical motion capture environment *Optitrack*, where the software *Motive* computes the position and the attitude of the flying object with the help of eight *Flex 13* cameras updated with a frequency of 120 Hz. These cameras can cover a flying space of 3.8 m × 3.8 m × 3 m in x, y, and z coordinates, respectively. The communication between *Motive* and other client softwares is done through UDP communication using the *NatNet* service. The packets

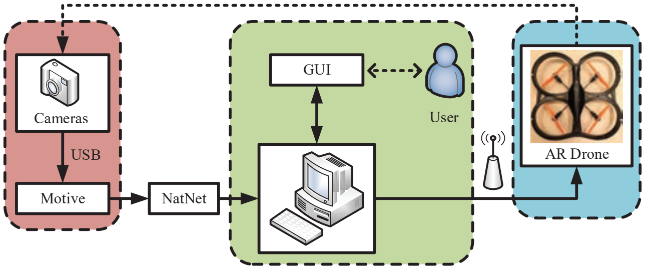

Fig. 3. Experimental environment.

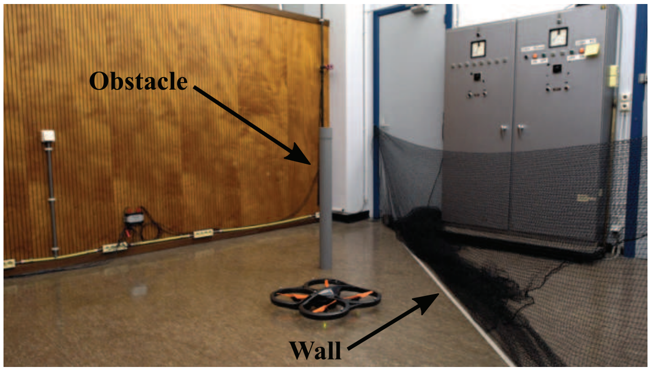

Fig. 4. Overview of the constraints.

are sent from the computer to the UAV through Wi-Fi using the UDP protocol.

The UAV used in this paper is the *Parrot AR Drone 2.0*. The drone is fitted with five reflective markers in an asymmetric way so that *Motive* can distinguish the front from the back. Regarding the constraints, a net and a PVC pipe are used to represent constraints (1) and (2), respectively (see Fig. 4). For the implementation of the proposed scheme, the programming language *Python* is chosen because of its extensive use and the availability of a great number of libraries. The source code is available at [18].

Note that $\dot{p}$ in (4) is not directly measured during the experiments; so, a velocity estimator needs to be implemented. For this purpose, we utilized a linear Kalman Filter (KF) [19] with covariance matrices of process and observation noises Q and R, respectively. Since *Motive* provides the precise position of the *AR Drone* with an error of about 1 [mm], the covariance matrix of observation noise R is selected diagonal whose elements are equal to 10^{-6}. As for the covariance matrix of process noise Q, by assuming a diagonal structure, the entries are tuned heuristically as 0.01. To validate the developed KF experimentally, a pendulum with reflective markers (to measure its position by *Motive*) was set up in the capture arena, such that it can oscillate in the yz-plane (see [20] for details). As seen in Fig. 5, the estimated position derived from the estimated velocity obtained by the developed KF follows properly the trend of the real position of the pendulum.

Since for linear systems one can easily find a quadratic Lyapunov function in the form of (8), we use a black box modeling approach to approximate the pre-stabilized *AR Drone* with a linear model to take advantage of the parametrized $\Gamma(v(t))$ given in (9) and (11). For this purpose, we apply several consecutive steps in all directions to the pre-stabilized *AR Drone*. Then, the N4SID algorithm [21] is utilized to obtain a linear approximation of the drone dynamics. This results in a 6*th*-order state-space model with A, B, C, and D matrices as given in Appendix. To validate the approximated model, the real position of the *AR Drone* is compared with the one obtained through the derived linear model. As seen in Fig. 6, the variables obtained through the identified linear model follow the trend of the real ones.

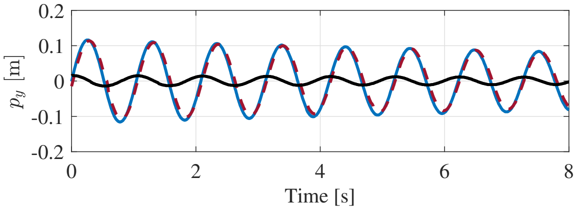

Fig. 5. Experimental validation of the developed KF. Blue line: Real y-position, Red line: Estimated y-position, Black line: Estimated velocity.

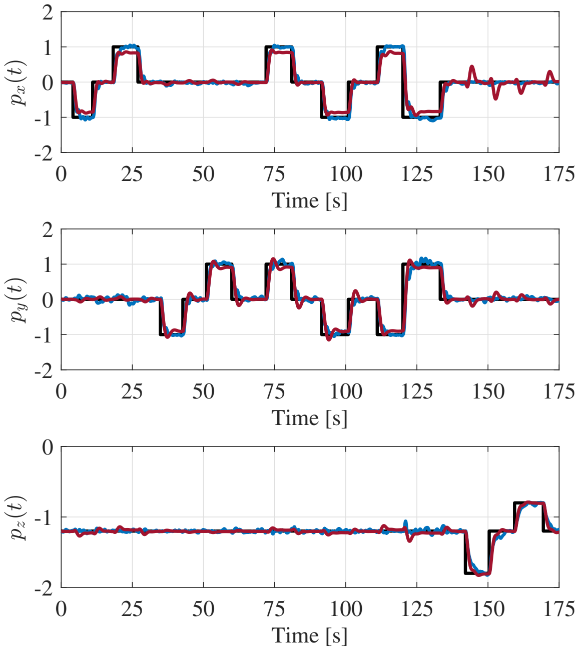

Fig. 6. Validation of the approximated linear model for the AR Drone. Black line: Desired position, Blue line: Real position, Red line: Position obtained via identified linear model.

Regarding the construction of the Lyapunov function for constraint (1), the problem (10) is solved with the toolbox provided in [22]. For constraint (2), the corresponding Lyapunov function is constructed by solving the Lyapunov inequality $A^T P_o + P_o A + I_6 \leq 0$, where I_6 is the identity matrix of dimension 6×6. The matrices P_w and P_o are given in Appendix.

To validate the proposed scheme experimentally, we consider the two following case scenarios.

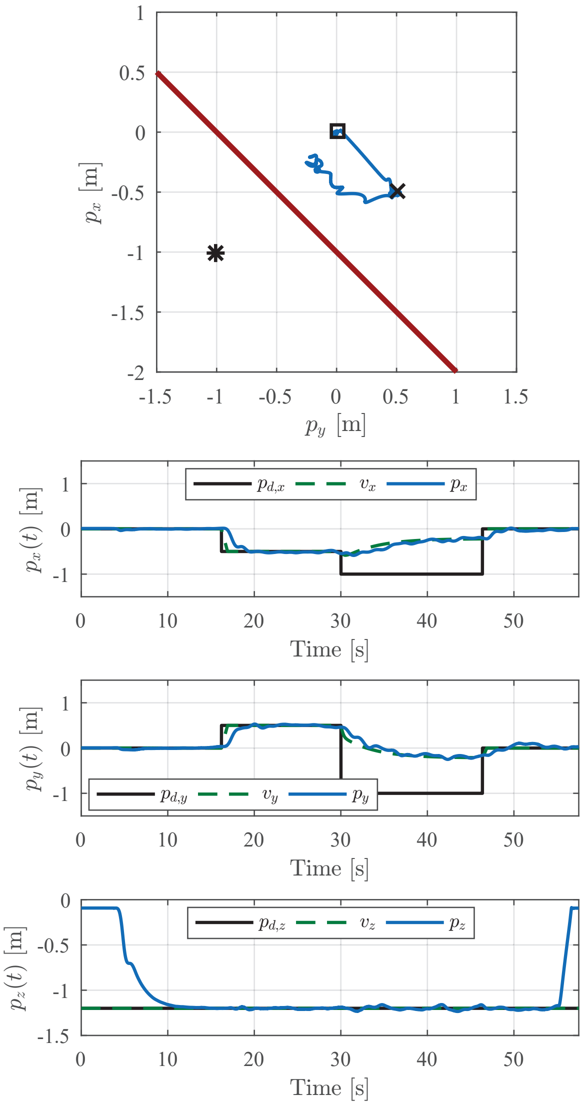

Fig. 7. Experimental results for Scenario 1. The square, cross, and star marks represent the take-off, admissible, and non-admissible desired positions, respectively.

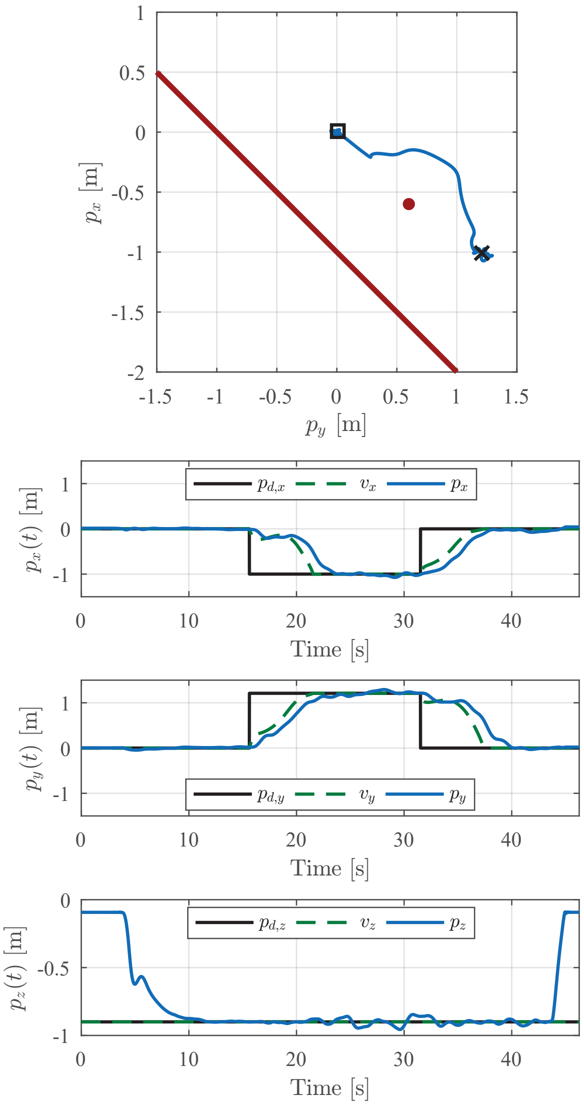

Fig. 8. Experimental results for Scenario 2. The square and cross marks represent the take-off and the desired positions, respectively.

- **Scenario 1:** This scenario considers the case in which the *AR Drone* is flying in an environment constrained by a wall, where the admissible region is $p_x + p_y + 1 \geq 0$ (see Fig. 4). Starting from the initial position $p_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, we first apply the desired reference $p_d = \begin{bmatrix} 0 & 0 & -1.2 \end{bmatrix}^T$ to take off. Then, after the drone is stabilized, we apply $p_d = \begin{bmatrix} -0.5 & 0.5 & -1.2 \end{bmatrix}^T$, which is inside the admissible region. As seen in Fig. 7, the *AR Drone* can reach the desired position. Afterwards, we apply $p_d = \begin{bmatrix} -1 & -1 & -1.2 \end{bmatrix}^T$, which is outside the admissible region. As expected, it moves towards the desired position, but stops and hovers near the wall, while minimizing the distance with the desired reference.
- **Scenario 2:** This scenario adds an obstacle of diameter 0.09 [m] placed inside the admissible region of Scenario 1, *i.e.,* $\begin{bmatrix} x_o & y_o \end{bmatrix} = \begin{bmatrix} -0.6 & 0.6 \end{bmatrix}$ (see Fig. 4). Similarly to Scenario 1, we start from the initial position $p_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, then apply the desired reference $p_d = \begin{bmatrix} 0 & 0 & -0.9 \end{bmatrix}^T$ to take off. After the stabilization of the *AR Drone*, the next desired reference is $p_d = \begin{bmatrix} -1 & 1.2 & -0.9 \end{bmatrix}^T$. Note that this reference is chosen such that without using the proposed scheme, the drone would move in a straight line making it collide with the obstacle (see Fig. 4). As seen in Fig. 8, the *AR Drone* moves accordingly and avoids colliding the obstacle and the wall.

The videos of the aforementioned scenarios can be found on https://wp.me/p9eDF3-8I.

V. CONCLUSIONS

This paper has proposed a constrained control scheme based on the ERG framework for geofencing applications. To verify the effectiveness of the proposed scheme, we applied it to the *AR Drone* flying in a constrained environment. Experimental results demonstrated that the proposed scheme is effective for the case of UAVs evolving in a bounded space in the presence of obstacles. Future works will aim at extending the results to other types of constraints (*e.g.,* non-convex class of constraints and time-varying constraints).

ACKNOWLEDGEMENT

We would like to thank Mr. Laurent Catoire and Mr. Serge Torfs, who gave us an invaluable help for the setup of the experimental environment.

APPENDIX
COMPUTED MATRICES

$$K_p = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.8 \end{bmatrix}, K_d = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.4 \end{bmatrix}$$

$$A = \begin{bmatrix} -0.0703 & 0 & 0 & 0.9938 & 0 & 0 \\ 0 & -0.064 & 0 & 0 & 0.9980 & 0 \\ 0 & 0 & -0.0534 & 0 & 0 & 0.9923 \\ -1.7872 & 0 & 0 & -1.1057 & 0 & 0 \\ 0 & -1.6302 & 0 & 0 & -1.0044 & 0 \\ 0 & 0 & -1.2956 & 0 & 0 & -1.0573 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0616 & 0 & 0 \\ 0 & 0.0556 & 0 \\ 0 & 0 & 0.0532 \\ 1.5874 & 0 & 0 \\ 0 & 1.4382 & 0 \\ 0 & 0 & 1.2917 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$P_w = 10^{-1} \begin{bmatrix} 5.0869 & 5.0981 & 0.0001 & 0.3446 & 0.3610 & 0.0001 \\ 5.0981 & 5.1194 & 0.0001 & 0.4039 & 0.4261 & 0.0002 \\ 0.0001 & 0.0001 & 0.0005 & 0.0004 & 0.0004 & 0.0004 \\ 0.3446 & 0.4039 & 0.0004 & 1.3942 & 1.4610 & 0.0006 \\ 0.3610 & 0.4261 & 0.0004 & 1.4610 & 1.5416 & 0.0007 \\ 0.0001 & 0.0002 & 0.0004 & 0.0006 & 0.0007 & 0.0008 \end{bmatrix}$$

$$P_o = \begin{bmatrix} 1.0174 & 0 & 0 & 0.0689 & 0 & 0 \\ 0 & 1.0239 & 0 & 0 & 0.0852 & 0 \\ 0 & 0 & 0.0001 & 0 & 0 & 0.0001 \\ 0.0689 & 0 & 0 & 0.2788 & 0 & 0 \\ 0 & 0.0852 & 0 & 0 & 0.3083 & 0 \\ 0 & 0 & 0.0001 & 0 & 0 & 0.0002 \end{bmatrix}$$

REFERENCES

[1] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based iot platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, 2017.

[2] G. Bareth, J. Bendig, N. Tilly, D. Hoffmeister, H. Aasen, and A. Bolten, "A comparison of uav-and tls-derived plant height for crop monitoring: using polygon grids for the analysis of crop surface models (csms)," *Photogrammetrie-Fernerkundung-Geoinformation*, vol. 2016, no. 2, pp. 85–94, 2016.

[3] S. Klosterman, E. Melaas, J. Wang, A. Martinez, S. Frederick, J. OKeefe, D. A. Orwig, Z. Wang, Q. Sun, C. Schaaf *et al.*, "Fine-scale perspectives on landscape phenology from unmanned aerial vehicle (uav) photography," *Agric. For. Meteorol.*, vol. 248, pp. 397–407, 2018.

[4] B. H. Y. Alsalam, K. Morton, D. Campbell, and F. Gonzalez, "Autonomous uav with vision based on-board decision making for remote sensing and precision agriculture," in *PRoc. 2017 IEEE Aerospace Conf.*, Big Sky, MT, USA, Mar. 4–11, 2017, pp. 1–12.

[5] M. Silvagni, A. Tonoli, E. Zenerino, and M. Chiaberge, "Multipurpose uav for search and rescue operations in mountain avalanche events," *Geomat. Nat. Haz. Risk*, vol. 8, no. 1, pp. 18–33, 2017.

[6] J. Downey, B. J. Michini, J. Moster, D. C. Weigel, and J. Ogden, "Unmanned aerial vehicle authorization and geofence envelope determination," Feb. 9 2016, uS Patent 9,256,994.

[7] S. Zhang, D. Wei, M. Q. Huynh, J. X. Quek, X. Ma, and L. Xie, "Model predictive control based dynamic geofence system for unmanned aerial vehicles," in *AIAA Information Systems-AIAA Infotech@ Aerospace*, 2017, p. 0675.

[8] M. N. Stevens, H. Rastgoftar, and E. M. Atkins, "Specification and evaluation of geofence boundary violation detection algorithms," in *Proc. 2017 Int. Conf. Unmanned Aircraft Systems (ICUAS)*, Miami, FL, USA, Jun. 13–16, 2017, pp. 1588–1596.

[9] E. Garone and M. M. Nicotra, "Explicit reference governor for constrained nonlinear systems," *IEEE Trans. Autom. Control*, vol. 61, no. 5, pp. 1379–1384, 2016.

[10] B. Convens, K. Merckaert, M. M. Nicotra, R. Naldi, and E. Garone, "Control of fully actuated unmanned aerial vehicles with actuator saturation," in *Proc. 20th IFAC World Congress*, Jul. 2017, pp. 12 715–12 720.

[11] M. M.Nicotra, M. Bartulovic, E. Garone, and B. Sinopoli, "A distributed explicit reference governor for constrained control of multiple uavs," in *Proc. 5th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Philadelphia, USA, Sep. 10–11, 2015, pp. 156–161.

[12] M. M. Nicotra, "Constrained control of nonlinear systems: The explicit reference governor and its application to unmanned aerial vehicles," Ph.D. dissertation, Université libre de Bruxelles (ULB), 2016.

[13] T. Nguyen and E. Garone, "Control of a uav and a ugv cooperating to manipulate an object," in *Proc. 2016 American Control Conference (ACC)*, Boston, MA, USA, Jul. 6–8, 2016, pp. 1347–1352.

[14] T. S. Lourenço, A. M. de Almeida Pinto, and R. V. Lopes, "Model predictive control applied to a quadrotor uav," *Revista Interdisciplinar de Pesquisa em Engenharia-RIPE*, vol. 2, no. 20, pp. 164–178, 2017.

[15] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "Robust global asymptotic attitude stabilization of a rigid body by quaternion-based hybrid feedback," in *Proc. 2009 48th IEEE Conf. Decision and Control held jointly with the 2009 28th Chinese Control Conf.*, Shanghai, China, Dec. 15–18, 2009, pp. 2522–2527.

[16] E. Garone, M. Nicotra, and L. Ntogramatzidis, "Explicit reference governor for linear systems," *International Journal of Control*, 2017.

[17] M. M. Nicotra and E. Garone, "The explicit reference governor-a general framework for the closed-form control of constrained nonlinear systems," *IEEE Control Syst. Mag.*, 2018, Pre-print available at: www.gprix.it/ERGplus.pdf.

[18] E. Hermand. saas-uav-ulb. [Online]. Available: https://github.com/ehermand/saas-uav-ulb

[19] G. Bishop, G. Welch *et al.*, "An introduction to the kalman filter," *Proc. SIGGRAPH 2001 Courses*, vol. 8, no. 27599-23175, p. 41, 2001.

[20] E. Hermand, "Constrained control of a UAV in a world with obstacles," Master's thesis, Université libre de Bruxelles (ULB), 2018.

[21] P. van Overschee and B. de Moor, *Subspace Identification of Linear Systems: Theory, Implementation, Applications.* Springer US, 1996.

[22] A. Cotorruelo Jiménez, D. Limón, M. Nicotra, and E. Garone, "Explicit reference governor toolbox," in *Proc. 4th Int. Forum on Research and Technologies for Society and Industry*, Palermo, Italy, Sep. 10–13, 2017.