An obstacle avoidance and motion planning Command Governor based scheme: the Qball-X4 Quadrotor case of study

Walter Lucia, Mario Sznaier and Giuseppe Franzè

Abstract—In this paper we develop an obstacle avoidance control scheme for autonomous vehicles. The strategy is based on Command Governor (CG) ideas that are here extended to take into account non-convex and time-varying constraints arising in path planning obstacle avoidance problems. As one of its main merits, the path planner module of the proposed architecture has the aim to select a finite set of intermediate locations between the initial and final points which are used by the CG unit as intermediate set-points until the prescribed target is reached. Experimental results on the quadrotor Qball-X4 show the effectiveness of the proposed approach.

I. INTRODUCTION

Various military and civil applications often require unmanned vehicles (UVs) to autonomously move in unknown environments subject to dynamic and physical constraints. The problem of motion planning and control for autonomous vehicles deals with finding appropriate control inputs such that the vehicle motion satisfies the requirements of a specified task. Avoidance of collisions with obstacles is a key component of the safe navigation where the primary objective is to reach a target through the obstacle-free part of the environment. To cope with this problem, two aspects must be taken into account: the generation of a safe trajectory capable in principle to reach the target by avoiding obstacle occurrences along the path and finding a control action whose the aim is to drive "as close as possible" the UV to the planned path under the satisfaction of the constraints arising from the specific vehicle dynamics [1].

In past years, much efforts have been devote to develop local obstacle avoidance schemes: most of them use reactive methods based on sensor data [8]; other contributions model the dynamical robot behaviour by means of velocity- or tuning-radius-based system descriptions that are too simple to be considered adequate for realistic scenarios, see e.g. [10]. Although computationally efficient, such approaches may lead to starvation phenomena because the heuristic solution of the underline optimizations may give rise to local minima that could indefinitely stop the robot navigation.

Recently, predictive active steering control schemes for AVs have been presented in [5], and [11]. Specifically in these noticeable contributions, the autonomous vehicle is forced to track a given path which is assumed to be effectively doable in the sense that along the path no collisions can occur and the controller is capable of generating the adequate actions to meet the tracking requirements. On the other hand, this approach cannot be used under unknown environments because safe pre-defined paths cannot be easily determined.

The main contribution of this paper is to develop an implementation of the command governor (CG) approach [2], [3], [9] for dealing with the path planning obstacle avoidance problem for UVs by preserving its basic properties. The proposed control architecture (see Fig. 1) consists of three main modules: a reference manager (CG) whose aim is to generate at each time instant a feasible set-point to be tracked during the on-line operations; a planner unit that determines a finite sequence of locations in order to allow obstacle avoidance during the vehicle navigation and a control module (SCG-OA) that is in charge to manage switching events when time-varying constraint are considered.

In principle a modification of the constraint set structure should prescribe a new CG design during the on-line operations, but this could be computationally very demanding and, therefore, not usable in practical situations. To overcome this drawback, the key idea here is to avoid the CG re-design by exploiting the following arguments: 1) time-varying constraints arise due to a shift with respect the current equilibrium; 2) different constraints sets are overlapped and their shapes are invariant w.r.t. any equilibrium shift. Finally, the experimental section is instrumental to show the capabilities of the proposed framework when it is used within a real-time environment. To this end, a quadrotor Qball-X4 jointly with a Vicon Motion Capture System has been used to validate the framework. Quanser’s real-time control software is used to embed the proposed CG-based algorithm on the on-board Gumstix architecture and the control problem consists of regulating the quadrotor state trajectory to a desired space location by avoiding collisions with three beams.

Walter Lucia and Giuseppe Franzè are with DIMES, Università degli Studi della Calabria, Via Pietro Bucci, Cubo 42-C, Rende (CS), 87036, ITALY {wlucia,franze}@dimes.unical.it
Mario Sznaier is with Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA, msznaier@coe.neu.edu
II. PROBLEM FORMULATION

Let us consider autonomous vehicles whose dynamics or their linearization around an equilibrium point can be described as an LTI system

\[
\begin{align*}
    x_v(t+1) &= Ax_v(t) + Bu(t) + B_dd(t) \\
    p(t) &= Cx_v(t)
\end{align*}
\]

(1)

where \(p(t) \in \mathbb{R}^3\) represents the position of the vehicle in the 3D space, \(t \in \mathbb{Z}^+ := \{0, 1, \ldots\}\), \(x_v(t) \in \mathbb{R}^{n_v}\) denotes the plant state, \(u(t) \in \mathbb{R}^{m_u}\) the control input and \(d(t) \in D \subset \mathbb{R}^{n_d}\) an exogenous disturbance, with \(D\) a convex and compact set such that \(0_{n_d} \in D\). Moreover, (1) is subject to the following set-membership state and input constraints:

\[
u(t) \in U, \; \forall t \geq 0, \; x_v(t) \in X, \; \forall t \geq 0,
\]

(2)

with \(U\) and \(X\) compact and convex subsets of \(\mathbb{R}^{m_u}\) and \(\mathbb{R}^{n_v}\). We will focus on the behaviour of this autonomous vehicle within an environment where obstacles lie.

Definition 1: Let \(Ob_1\) be a convex obstacle. Then an obstacle scenario \(O\) is defined as

\[O := \{Ob_1, \ldots, Ob_{n_o}\}\]

(3)

where \(n_o\) denotes the number of objects involved. Moreover, the exterior non-convex region corresponding to \(O\) can be characterized as follows

\[O_{free} := \{x_v \in \mathbb{R}^{n_v} : h_i(x) > 0\}\]

(4)

where \(h_i : \mathbb{R}^{n_v} \to \mathbb{R}^{n_i}\) characterizes the admissibility state space region.

Then, the problem we want to solve can be stated as follows:

**Obstacle Avoidance Motion Planning (OAMP) Problem**

- Given an obstacle scenario \(O\) and a target location \(p_{goal}\), determine a state-feedback control policy

\[
u(t) = \eta(x_v(t), p_{goal})
\]

(5)

compatible with (2) and (4), such that starting from an initial condition \(x_v(0)\) the state trajectory \(x_v(t)\) is asymptotically driven to a target \(x_{goal}\) such that \(p_{goal} = Cx_{goal}\).

In the sequel, the problem will be addressed by using an extension of the command governor strategy developed in e.g. [2], [3].

III. BASIC COMMAND GOVERNOR (CG) DESIGN

Consider the following discrete-time linear time-invariant system:

\[
\begin{align*}
x(t+1) &= \Phi x(t) + Gg(t) + G_d d(t) \\
y(t) &= H_y x(t) \\
c(t) &= H_c x(t) + Lg(t) + L_d d(t)
\end{align*}
\]

(6)

where \(x(t) \in \mathbb{R}^n\) is the augmented state including both plant and primal controller components, \(g(t)\) the CG action, i.e. a suitably modified version of the reference signal \(r(t) \in \mathbb{R}^m\), \(y(t) \in \mathbb{R}^m\) the plant output which is required to track \(r(t)\) and \(c(t) \in \mathbb{R}^{n_c}\) the constrained output vector

\[
c(t) \in C, \; \forall t \in \mathbb{Z}^+
\]

(7)

with \(C\) a specified convex and compact set. In the sequel, the following assumptions are made:

A1. \(\Phi\) is a Schur matrix

A2. The model plant (6) is offset-free, i.e. \(H_y(I_n - \Phi)^{-1}G = I_m\)

Observe that the standard structure of a CG-equipped control system consists of two nested loops. The inner loop is designed without taking into account the prescribed constraints and allows to specify relevant system properties, e.g. stability of \(\Phi\), disturbance rejection. The outer loop consists of the CG unit which is in charge to modify the reference to be applied to the closed-loop system so as to avoid constraint violation.

By noticing that, under a constant command \(g(t) = \omega \forall t\), the disturbance-free steady state solution of (6) is

\[
\begin{align*}
\bar{x}_\omega &= (I_n - \Phi)^{-1}G \omega \\
\bar{y}_\omega &= H_y(I_n - \Phi)^{-1}G \omega \\
\bar{c}_\omega &= H_c(I_n - \Phi)^{-1}G \omega + L \omega
\end{align*}
\]

(8)

and by considering the following Minkowski difference recursions on the constrained set \(C\)

\[
\begin{align*}
C_0 &= C \sim L_0 D, \; C_k := C_{k-1} \sim H e \Phi^k - 1 G D \\
C_\infty &= \bigcap_{k=0}^{\infty} C_k
\end{align*}
\]

(9)

it has been proved in [2] that the CG action \(g(\cdot)\) is computed according to the following convex optimization problem over a finite prediction horizon \(k_0 \in \mathbb{Z}^+\):

\[
g(t) = \arg \min_{\omega \in \mathcal{V}(x(t))} ||\omega - r(t)||_\Psi, \; \Psi = \Psi^T > 0
\]

(10)

Then

\[
\mathcal{V}(x(t)) = \{\omega \in W^\delta : c(k, x, \omega) \in C_k, k=0, \ldots, k_0\}
\]

(11)

with \(c(k, x, \omega) = H_c \left( \Phi^k x + \sum_{i=0}^{k-1} \Phi^{k-i-1} G \omega \right) + L \omega\),

\[
W^\delta = \{\omega \in \mathbb{R}^m : c_{\omega} \in C^\delta\}, \; C^\delta = C_\infty \sim B_\delta
\]

(12)

and \(B_\delta\) a ball of radius \(\delta\) centered at the origin, is the set of all constant virtual commands whose state evolutions starting from \(x\) satisfies all the constraints also during the transients.

IV. AN OBSTACLE AVOIDANCE COMMAND GOVERNOR STRATEGY

The basic CG framework requires that the constrained set \(C\) must satisfy the following requirements: 1) compact and convex; 2) time-invariant. The consequence of these restrictions is that the use of the CG strategy in its classical version for solving the proposed OAMP problem is no longer viable because the obstacles give rise to non-convex geometrical constraints that could change in principle at each time instant. The goal of this section is then to overcome such drawbacks by adequately extending the CG design so that it can comply with the OAMP requirements.

To this end, the key point is to convexify at each time instant an active (to be specified) region around the current vehicle position by means of a suitable obstacle constraints updating procedure. The next sections will be devoted to formally develop this idea.
Fig. 2. Viability retention under constraints change

A. Viability retention under time-varying constraints

Without loss of generality, consider two constraint sets

\[ C_i = \{ c \in \mathbb{R}^{n_c} : z_i(c) \leq 0 \}, \quad \text{compact and convex} \]

\[ C_j = \{ c \in \mathbb{R}^{n_c} : z_j(c) \leq 0 \}, \quad \text{compact and convex} \]

such that \( C_i \neq C_j \) and \( C_i \cap C_j \neq \emptyset \), then the following property holds true

\[ W^\delta_i \cap W^\delta_j \neq \emptyset \]

where \( W^\delta_i = \{ \omega \in \mathbb{R}^m : c_w \in C^\delta_s \} \), and \( C^\delta_s = C_s \sim B^3 \), \( s = i, j \).

By making use of the following definition:

**Definition 2:** The state \( x(t) \) is defined \( C^\delta \)-admissible if there exist \( \omega \in W^\delta_i \) such that \( c(t, x, \omega) \in C(\omega), \forall t \geq \tilde{t} \). The next result proves the constraint scenario switching admissibility.

**Proposition 1:** Given the system (6) and the constraint scenarios \( C^\delta_i \) and \( C^\delta_j \) satisfying (13). Then, there always exists a finite concatenation of virtual constant commands \( \omega \in W^\delta_i \cup W^\delta_j \) such that, starting from an arbitrary \( x(0) \) \( C^\delta_i \) or \( C^\delta_j \)-admissible initial state, the state trajectory of (6) is driven to \( x_\omega \) with \( \omega \in W^\delta_i \cup W^\delta_j \). Moreover under a constraint scenario change there exists a finite switching time \( t_{sw} < \infty \) and an arbitrary small scalar \( \Delta_x > 0 \) such that \( |x(t_{sw}) - x_\omega| < \Delta_x \), where \( \omega \in W^\delta_i \cap W^\delta_j \), and \( x(t_{sw}) \) is \( C^\delta_i \) and \( C^\delta_j \)-admissible (see (2) for a graphical illustration).

**Proof -** Follows by resorting to viability arguments [2] and it has been omitted for the sake of space. \( \square \)

B. Time-varying constraints for obstacle avoidance purposes

Here, we show that the results of Proposition 1 can be specialized in order to provide a solution to the proposed OAMP problem.

To this end by exploiting the fact that the vehicle state can be split as geometrical \( (x_g) \) and non-geometrical \( (x_{ng}) \) components, model (6) can be re-written as

\[
\begin{align*}
    x(t+1) = & \begin{bmatrix} x_g(t+1) \\ x_{ng}(t+1) \\ x_u(t+1) \end{bmatrix} = \Phi x(t) + G g(t) + G_d d(t) \\
    p(t) = y(t) = & H y x(t) \\
    c(t) = & \begin{bmatrix} c_g \\ c_{ng} \\ c_u \end{bmatrix} = H_c x(t) + L g(t) + L_d d(t)
\end{align*}
\]

(14)

where \( x_g \in \mathbb{R}^3 \) are the spatial components, \( x_u \in \mathbb{R}^{n_u} \) the primal controller state and \( x_{ng} \in \mathbb{R}^{n-3-n_u} \) the non-geometrical state variables. Moreover, \( c_g \), \( c_{ng} \) and \( c_u \) take care of the constraints prescribed for each state component.

Let \( x(0) \) and \( C^\delta \) be the initial condition and the constrained set, respectively. Under the action of the CG action, the vehicle will asymptotically reach the best feasible approximation of the desired state space position \( p_{goal} \), i.e. \( \hat{g} \).

If \( \hat{g} \neq p_{goal} \), the following arguments can be used:

- Compute the equilibrium condition corresponding to \( \hat{g} \):
  \[
  \hat{x}_{eq} = (I - \Phi)^{-1} G \hat{g} + \hat{c}_{eq} = H_c \hat{x}_{eq} + L \hat{g}
  \]

(15)

- Due to linearity of (6), the state space is shifted according to \( \hat{x}_{eq} \):
  \[
  \begin{align*}
  \hat{x}(t+1) = & \hat{\Phi} \hat{x}(t) + G \hat{g}(t) + G_d d(t) \\
  \hat{p}(t) = & H y \hat{x}(t) \\
  \hat{c}(t) = & H_c \hat{x}(t) + L \hat{g}(t) + L_d d(t)
  \end{align*}
  \]

(16)

- A new shifted (w.r.t. \( \hat{c}_{eq} \)) constrained set results:
  \[
  \hat{c}(t) \in \hat{C}^\delta := \text{shift}(C^\delta, \hat{c}_{eq})
  \]

(17)

Therefore, the OAMP problem can be solved by using a suitable sequence of shifts on the initial constrained set \( C \) until the target position \( p_{goal} \) is reached. A possible difficulty to pursue such an idea is that it could be necessary to modify shapes or dimensions of \( C^\delta \) because of the presence of obstacles along the path.

By recalling that the set (7) can be written w.l.o.g. as

\[
C = \{ c \in \mathbb{R}^{nc} : T c \leq b \}
\]

(18)

where \( T \in \mathbb{R}^{z \times nc}, \ g \in \mathbb{R}^z, \ z \geq nc, \ \text{rank}(T) = nc \)

with \( T \) a shaping matrix and \( b \) accounting for the set dimensions, and the bounded disturbance set \( D \)

\[
D = \{ d \in \mathbb{R}^{nd} : Q d \leq [h_1, \ldots, h_{n_q}]^T \}, \ Q \in \mathbb{R}^{n_q \times n_q}
\]

(19)

we have that the admissibility region \( \mathcal{V}(x) \) is characterized by the following inequalities

\[
T H_c (I - \Phi)^{-1} + L \leq b^k, \ k = 0, \ldots, \hat{k}
\]

(20)

\[
T(H_c (I - \Phi)^{-1} + L) \leq b^{k_\epsilon} - \delta \left[ \sqrt{T^T T_i} \right]
\]

(21)

where \( b^k \) is computed using the following recursion

\[
\begin{align*}
    \text{row}_p(b^0) &= \text{row}_p(b) \\
    \text{row}_p(b^k) &= \text{row}_p(b^{k-1}) - \sup_{d \in D} \text{row}_p(T) H_c \Phi^{k-1} G_d d
\end{align*}
\]

where \( \text{row}_p(\cdot) \) denotes the \( p-th \) row of its argument,

\[
k_c = \left[ \log \epsilon + \log(1 - \lambda) \right] [\sigma(H_c) \sigma(G_d) M d_{max}]
\]

(22)

\( \lambda \in (0, 1), \ \epsilon > 0 \) and \( \delta \left[ \sqrt{T^T T_i} \right] \) is the support function describing the ball \( B_{\hat{k}} \), see [2] for technical details.

Then, from (20)-(22) it clearly follows that a change on the constrained set shape \( (T) \) will require a complete CG redesign (for example the computation of the control horizon \( \hat{k} \)), while a variation on its dimensions \( (b) \) does not affect the CG structure.
V. SWITCHING COMMAND GOVERNOR OBSTACLE
AVOIDANCE ALGORITHM (SCG-OA)

Let \( p_{\text{start}}, p_{\text{goal}}, O \) and \( C \) be starting and target positions, the obstacle configuration and the constraint set, respectively. In the sequel, the following assumptions are made:

- There exists an admissible path connecting \( p_{\text{start}} \) and \( p_{\text{goal}} \);
- There exists a path module, hereafter named planner, whose task is to determine a sequence of intermediate target positions to pass through so that the obstacles can be avoided and the goal location reached, i.e.
\[
p = \text{planner}(O, p(t), p_{\text{goal}})
\]
which, by using as inputs the obstacle configuration, the current vehicle position and the target, returns the next location \( p \) to be reached;
- There exists a vision module (cameras or group of sensors) used to measure the obstacle-free region dimensions around the current vehicle position.

These assumptions lead to the following computationally tractable algorithm:

**Basic-SCG-OA algorithm**

0. **Initialization:** \( \bar{x}_{eq} = 0_x, \bar{c}_{eq} = 0_n, \bar{g} = 0_m \) and \( \bar{C} = C \).

1. Compute:\n\[
p = \text{planner}(O, p(t), p_{\text{goal}})\]
\[
p_{\text{reach}} = \arg\min_{w \in W_x} \| w - (p - \bar{g}) \|^2 \quad (23)
\]
\[
x_p^{\text{reach}} = (I - \Phi)^{-1} G p_{\text{reach}} + \bar{x}_{eq} \quad (24)
\]

2. If \( \| x(t) - x_p^{\text{reach}} \|_2 < \Delta_x \)
   2.1. \( \hat{g} \leftarrow p_{\text{reach}} \): compute \( \bar{x}_{eq}, \bar{c}_{eq} \) by (15) and update \( \bar{C} \) according to (17);
   2.2 Compute a new reference \( p \)
   \[
p = \text{planner}(O, x(t), p_{\text{goal}})\]
   2.3. Update \( x_p^{\text{reach}} \) according to (23)-(24);

3. Compute the command governor action \( \hat{g}(t) \)
\[
\hat{g}(t) = \arg\min_{w \in \mathcal{V}(x(t))} \| w - (p - \bar{g}) \|^2 \quad (25)
\]

4. Apply \( g(t) = \hat{g} + \hat{\hat{g}}(t) ; t = t + 1 \) and goto 2.

It is worth noticing that the Basic-SCG-OA algorithm may give rise to a certain level of conservativeness because the Step 2 (due to the CG viability property conditions [2]) could impose a stop-and-go phase until the vehicle reaches the pre-assigned target. A possible strategy to overcome this drawback is to exploit knowledge about the future (at the next time instant) candidate equilibrium point, i.e. \( x_p^{\text{reach}} \) computed by means of (24), with \( \bar{C}_{\text{reach}} \) the corresponding constraint window. Specifically by denoting with \( V_{\text{reach}}(x) \) the set of feasible commands pertaining to \( x_{\text{reach}} \), then Step 2. of the Basic-SCG-OA algorithm becomes:

**New-step 2:**

2. If \( x(t) \in V_{\text{reach}}(x(t)) \), then
  2.1. \( \hat{C} \leftarrow \bar{C}_{\text{reach}} \); and \( \hat{\hat{g}} \leftarrow p_{\text{reach}} \);
  2.2. Compute \( \bar{x}_{eq}, \bar{c}_{eq} \) by (15) and a new reference

![Fig. 3. Qball-X4 axis and sign convention](image-url)

The thrust generated by each propeller is modeled using first-order systems:
\[
F = K - \frac{\omega}{s + \omega} u_v
\]
while the roll and pitch angles are described by the following first order systems:
\[
J \hat{\theta} = \Delta FL
\]
where \( J = J_r = J_p \) and \( \Delta u = F_1 - F_2 \). The height model is described by
\[
M \ddot{z}_v = 4F \cos r \cos p - Mg
\]
while the \( X_v - Y_v \) position models are governed by
\[
M \dot{X}_v = 4F \sin p, \quad M \dot{Y}_v = -4F \sin r
\]
Thus, the corresponding state space and control input vectors are as follows:
\[
x_v = \begin{bmatrix} r & \dot{r} & p & \dot{p} & x_v & \dot{x}_v & y_v & \dot{y}_v & z_v & \dot{z}_v \end{bmatrix}^T
\]
\[
u = \begin{bmatrix} \Delta u_r & \Delta u_p & u_v \end{bmatrix}^T
\]
B. QBall-X4 data acquisition and Vicon cameras system

A Quanser’s on board avionics data acquisition card (DAQ), an HiQ and an embedded Gumstix computer are in charge of acquiring measurements from the on-board sensors and to drive the Qball-X4 motors. Moreover, the Qball-X4 spatial Cartesian coordinates are obtained by means of an indoor high speed Vicon© cameras system [14]. Within the working environment, this infrared cameras system has the capability to quickly identify and track reflective markers with high accuracy, see Fig. (4).

C. On board controller implementation

The Quanser’s real-time control software allows to develop and evaluate controllers performance on the Qball-X4 by using Matlab/Simulink. This software is capable to access and evaluate controllers performance on the Qball-X4 by exploiting the on-board Gumstix computer architecture. Finally, recall that the main computational task of the SCG-OA is the solution of the constrained quadratic optimization problem (10) that has been solved by using an ADMM (alternating direction method of multipliers) procedure [4].

VII. EXPERIMENTS

Experiments are instrumental to show the benefits of the proposed approach when critical scenarios are considered. The obstacle configuration is the following:

- a room: \( x_v \times y_v \times z_v = 4 \times 1.2 \times 3 \text{m}^3 \);
- Obstacle positions: summarized in Table II and shown in Fig. 5.

<table>
<thead>
<tr>
<th>Obstacle</th>
<th>width(m)</th>
<th>depth(m)</th>
<th>height(m)</th>
<th>center of gravity(m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ob(^t)</td>
<td>0.7</td>
<td>0.7</td>
<td>0.7</td>
<td>[0.6; 0.2; 0.30]</td>
</tr>
<tr>
<td>Ob(^s)</td>
<td>0.7</td>
<td>0.7</td>
<td>0.7</td>
<td>[1.8; -0.4; 0.35]</td>
</tr>
</tbody>
</table>

TABLE II

The experimental task is:

Starting from the initial condition \( p_{start} = (0, 0, 0.12) \text{m} \), see Fig. 5, the Qball-X4 quadrotor must reach the target \( p_{goal} = (2.9, -0.3, 0.50) \text{m} \) and come back to \( p_{start} \) by avoiding the obstacles along the path.

The Qball-X4 model (25)-(28) has been linearized around the following equilibria:

\[
x_v = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0.12 & 0 \end{bmatrix}^T (31)
\]

\[
\bar{u} = \begin{bmatrix} 0 & 0 & 0.031 \end{bmatrix}^T (32)
\]

Then, a discrete time linear invariant description is obtained by using the Euler forward differences method with \( \Delta t = 0.001 \text{sec} \) the sampling time interval. The set of constraints is summarized below:

- Working environment (room):
  \[-1.2 \leq x_v \leq 1.2, \quad -1 \leq y_v \leq 3.5, \quad 0 \leq z_v \leq 2.5\]
- Roll and pith angle (validity of the linearized model):
  \[|r| \leq 0.3, \quad |p| \leq 0.3\] (33)
- Control command effort (rotor saturation):
  \[|\Delta u_r| \leq 0.025, \quad |\Delta u_p| \leq 0.025, \quad 0.011 \leq |u_v| \leq 0.051\] (34)

The CG parameters are: \( \delta = 10^{-5} \) and \( \Psi = I_3 \). The constraint horizon \( k_0 = 120 \) was computed via the numerical procedure proposed in [6]. The primal compensator has been designed as a two-degree of freedom LQ controller.

All the relevant results are reported in Figs. 6-11. Fig. 6 depicts the Qball-X4 state trajectories under the action of the Basic SCG-OA (blue line) and SCG-OA (red line) algorithms in the 3D space, whereas Fig. 7 accounts for the dynamical behaviours along the \( x \) and \( y \) axes. As expected, the SCG-OA algorithm shows remarkably lower settling times (about 43 sec) than its Basic SCG-OA competitor (about 60 sec). Figs. 8-9 show the dynamical behaviours of the planner and command governor modules. There, it is interesting to remark that the CG unit is always able to modify the reference provided by the planner module in order to ensure constraint satisfaction at each future time instant.

Moreover, Fig. 10 shows roll and pitch angles behaviors, while Fig. 11 reports the applied control inputs, i.e. \( \Delta r, \Delta p \) and \( u_v \). As it is clearly highlighted, the prescribed constraints (33)-(34) are always satisfied.

VIII. CONCLUSIONS

An extension of the basic CG strategy for solving the obstacle avoidance motion planning problem for autonomous vehicles has been presented. Linearity and constraint sets overlapping properties are the key ingredients to prove that
feasibility retention and constraints fulfilment are preserved despite of any constraint structure modification. The experimental results on the Qball-X4 quadrotor allow to clearly show the effectiveness of the proposed SCG-OA algorithm when real obstacle scenarios are taken into consideration.

ACKNOWLEDGEMENT

This paper is partially funded by European Commission, European Social Grant and Regione Calabria.

REFERENCES