DOI: 10.48011/asba.v2i1.1516

Command Governor strategy based on region of attraction control switching

Álan C. e Sousa * **Luís F. P. Silva** ** **Walter Lucia** ***
Valter J. S. Leite *,**

* *Graduate Program on Electrical Engineering CEFET-MG & UFSJ (e-mail: acristoffers@gmail.com, valter@ieee.org).*
** *Department of Mechatronics Engineering / CEFET-MG / campus Divinópolis (e-mail:valter@ieee.org, luis@cefetmg.br)*
*** *Concordia Institute for Information Systems Engineering (CIISE), Montreal, Quebec, Canada (walter.lucia@concordia.ca)*

Abstract The command governor (CG) is a useful strategy to avoid constraints violation in case of control reconfiguration or operational conditions changes, like switched systems mode changes. In general, the structure of the CG specializes in producing an alternative reference signal to drive a closed-loop system to the desired reference while avoiding constraints violations in the control signal, state-space, or combinations of them. In this paper, we propose a simple strategy that modifies the conventional structures of the CG and supervisor, yielding a switched rule based on the estimated controllers' region of attraction. We present some simulations to illustrate the proposal's potential and compare it with a competitor scheme exploiting a dwell-time approach. The results suggest that our approach adds new possibilities of CG and supervisor design, reducing the transition time between system modes and improving the closed-loop performance indexes.

Keywords: Command Governor, Discrete-time systems, Switching systems, Lyapunov stability, Region of attraction

1. INTRODUCTION

The fundamental frameworks of classical and modern control theory are conceived for unconstrained systems. Therefore, if constraints are of interest, ad-hoc solutions must used. Some approaches handle saturating actuators by using a polytopic representation or generalized sector conditions (Tarbouriech et al., 2011). However, these techniques cannot handle the state's constraints directly, requiring the inclusion of state-space constraints (Klug et al., 2015). Some techniques started to be developed that enforces constraints, like the model predictive control (MPC) (Zhang, 2016; Wang, 2009). However, they require the design of a new controller following the framework. Another line of research tried to solve the same problem by using existing controllers. Instead of trying to compute the optimal control path that keeps the system constrained, it uses model prediction to change the reference given to existing controllers to keep the system constrained. The first of such techniques were reference filters that also imposed only soft-constraints (Vahidi et al., 2007).

The idea then evolved into Reference Governors (RG), which makes use of optimization to find the best reference to follow (Gilbert et al., 1995). The RG gives a virtual reference $g(k)$ to the controller, which is a scaled version of the real reference $r(k)$, read $g(k) = \delta r(k)$. The optimization problem finds the best $\delta r(k)$ that minimizes the distance between $g(k)$ and $y(k)$ without violating the constraints. Because of the simplicity of the optimization problem, this approach has an easy implementation but suffers from loss of dimensions. Such a loss comes from the fact that variation is one-dimensional while $r(k)$ may be multidimensional (Gilbert and Kolmanovsky, 1999).

Building on this idea, Bemporad et al. (1997) and Casavola et al. (2000) developed what is known today as the Command Governor (CG) approach. The difference to RG is that the CG technique optimizes $g(k)$ directly, requiring more computational processing power but yielding better system performance, especially when the reference has dimension greater than 1. They also built on the work of Kapasouris et al. (1988), which explores the ideas of the Lyapunov Theorem and Invariant Sets Theorem.

Switching systems are composed of many subsystems, called modes, which switch according to a switching rule (Liberzon, 2003; Liberzon and Morse, 1999). Only one subsystem can be active at a given time. The switching can cause instability even when all subsystems are stable, which leads to the notion of a dwell-time: how long a subsystem must remain active after switching to avoid instability (Liberzon and Morse, 1999). Different approaches have been proposed to compute the minimum dwell time (see (Chesi et al., 2010) and reference therein) and stabilizing controller (see (Lin and Antsaklis, 2009) for switched linear systems). Fewer solutions exist to deal with constrained switching systems, see e.g. (Lucia and Franzè, 2017; Franzè et al., 2017) and references therein.

In (Lucia and Franzè, 2017; Franzè et al., 2017), the CG framework is used to supervise the system mode switches and assure both stability and constraint satisfaction.

In this work, considering the class of switched systems (switching systems with controlled switching signals), we propose a novel switching rule based on the switching CG's region of attraction. It aims at improving the control performance obtained in the previous work of Franzè et al. (2017) where mode-switches occur after a worst-case dwell-time elapses.

The presented approach consists of a two-step switch, which uses an intermediate CG. This intermediate CG has the constraints of the source CG and the controller of the target CG. To change from CG_i to CG_j, you first switch from CG_i to a CG_{ij} when the system enters the region of attraction of CG_{ij}'s controller, and then changes to CG_j when the system enters the intersection of the constraint regions between CG_{ij} and CG_j. Thus, if it is verified that the controller of CG_j has better performance than that of CG_1, it can be used from an earlier stage.

This paper organization is as follows: in the next section, we define the command governor and supervisor structures. In section 3, we give two examples to illustrate the proposal and to compare its performance with other approaches found in the literature. The finds suggest our method outperforms the previous ones, being an excellent alternative to the CG switching policy.

Notations: The set of real numbers is denoted by $\mathbb{R}$. $\mathbb{R}^n$ denotes a vector of n real elements. $\mathcal{V}$, $\mathcal{W}$ and $\mathcal{C}$ represent sets. $M^\top$ represents the transpose of M. $g(k)$ represents a discrete-time signal. We define $x^\top \Psi x$ with $x \in \mathbb{R}^n$ and semidefinite positive matrix $\Psi \in \mathbb{R}^{n\times n}$ as the norm $\|x\|^2_\Psi$.

2. PROBLEM FORMULATION AND PRELIMINARIES

A switched discrete-time system given in the state-space representation has the form

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k), \\ y(k) &= C_i x(k) + D_i u(k), \\ c(k) &= E_i x(k) + F_i u(k), \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the state vector, $y(k) \in \mathbb{R}^p$ is the output, $c(k) \in \mathbb{R}^{n_c}$ is the constrained or weight output, the matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$, $C \in \mathbb{R}^{n_y\times n}$, and $D \in \mathbb{R}^{n_y\times n_u}$ concern the system's dynamic and output, matrices $E \in \mathbb{R}^{n_c\times n}$ and $F \in \mathbb{R}^{n_c\times n_u}$ concern the constrained output and are chosen by the designer to take into account the constraints associate with the state and control signal. The sub-index $i = 1,\ldots,s$ refers to the active model. In many cases the mode i is switched depending on the state-vector, and thus on its trajectory.

We exploit the state-feedback structure to propose a mode-dependent proportional-integral (PI) action for the system (1). The controller is designed to ensure null steady-state error for piecewise constant references at each mode i, and thus the integral action is applied over the tracking error (Lopes et al., 2020)

$$e(k) = r(k) - y(k), \tag{2}$$

where $r(k)$ is the desired output of the system. The proportional action comes from the system's state deviation with respect to the equilibrium point. Because we suppose the state values are not measured, we use an observer to estimate them. Fig. 1 depicts the topology of the considered controller with the observer.

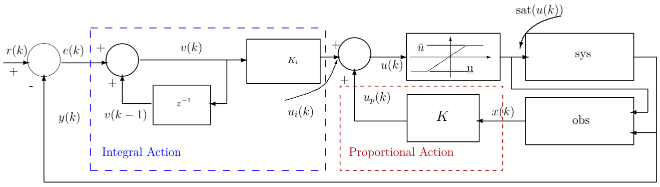

Figure 1. PI-like controller.

By defining an augmented state vector

$$\xi(k) = \left[\hat{x}(k)^\top \; v(k)^\top\right]^\top,$$

where $v(k) \in \mathbb{R}^{n_u}$ is the vector of added integrators, the closed-loop system shown in Fig. 1 can be rewritten as

$$\begin{aligned} \xi(k+1) &= \mathcal{A}_i \xi(k) + \mathcal{B}_i u(k), \\ y_k &= \mathcal{C}_i \xi(k) + \mathcal{D}_i u(k), \end{aligned} \tag{3}$$

where $\mathcal{A}_i = \begin{bmatrix} A_i & \mathbf{0} \\ -C_i A_i & \mathbf{I} \end{bmatrix}$, $\mathcal{B}_i = \begin{bmatrix} B_i \\ -C_i B_i \end{bmatrix}$, $\mathcal{C}_i = \begin{bmatrix} C_i & \mathbf{0} \end{bmatrix}$, $\mathcal{D}_i = \begin{bmatrix} \mathcal{D}_i^\top & \mathbf{0} \end{bmatrix}^\top$.

The design of each controller gain $K_i \in \mathbb{R}^{n_u\times(n+p)}$ may use standard LMI based techniques, such as, for instance, pole placement (Yu, 2013), LPV design (Briat, 2014), or robust control (Boyd et al., 1994). In this case, the main issue is to ensure the stability between the switching instants, and our approach concerns such an aspect, as presented in the next section. It is important to note that, in practice, the controller may be designed independently for each mode.

Thus, for each mode we design a observer gain $L_i \in \mathbb{R}^{n\times n_y}$ ensuring the asymptotic convergence of the estimation error $(\hat{x}(k) - x(k))$ to zero. Such gains yield the following mode-dependent observers:

$$\hat{x}(k+1) = A_i \hat{x}(k) + B_i u(k) + L_i(y(k) - C_i \hat{x}(k)). \tag{4}$$

Conventional methods in the literature can calculate the gain L_i (Chen, 2012-11; Hespanha, 2018-02). Therefore, the linear control law becomes

$$u(k) = K_i \hat{\xi}(k) = \left[K_{Pi} \; K_{Ii}\right] \hat{\xi}(k), \tag{5}$$

with $\hat{\xi}(k) = \left[\hat{x}(k)^\top \; v(k)^\top\right]^\top$, for $i = 1,\ldots,s$.

Note that the constrained output can also be expressed in terms of the augmented state $\xi(k)$, taking into account the state of the integrator, i.e., $c(k)$ in (1) can be given by

$$c(k) = \mathcal{E}_i \hat{\xi}(k) + \mathcal{F}_i u(k), \tag{6}$$

with the mode-dependent matrices $\mathcal{E}_i$ and $\mathcal{F}_i$ with adequate dimensions.

2.1 Command Governor

The Command Governor (CG) is an add-on technique that extends existing controllers with constraint enforcement. It uses the system model to predict states given a reference $y(k)$ and computes the virtual reference $g(k)$ closest to $y(k)$ that keeps the system constrained. Fig. 2 presents its block diagram, which shows that the CG is aware of the system and controller states. The constraint itself is an integral part of the CG.

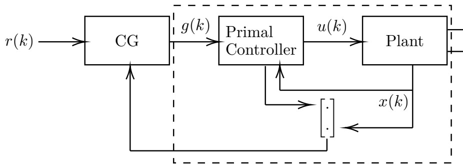

Figure 2. Command Governor Block Diagram.

In what follows we use the sets $\mathcal{C}$, $\mathcal{W}$, and $\mathcal{V}$ defined as follows:

(1) The set of values allowed to the constrained output, $c(k)$, defines the set $\mathcal{C}$, which is the output restriction.
(2) Let $\omega \equiv g(k)$ for a constant $g(k)$. $\mathcal{W}$ is the set of all values ω that keep $c(k)$ constrained on the steady-state.
$$\mathcal{W} = \{\omega \in \mathbb{R}^{n_y} : c(k) \in \mathcal{C}, k \to \infty\}.$$
(3) $\mathcal{V}$ is the set of ω values ensuring the constrained output to belong to $\mathcal{C}$ in the next k_0 samples
$$\mathcal{V} = \{\omega \in \mathcal{W} : c(k) \in \mathcal{C}, 0 < k \leq k_0\},$$

We write the set $\mathcal{V}$ in terms of convex constraints on the virtual reference, the k future states, the minimization of the distance to the desired reference, and actuator saturation.

The output of the Command Governor is given by the following convex optimization procedure:

$$g(k) = \arg\min_{\omega \in \mathcal{V}} \|\omega - r(k)\|^2_{\Psi} \tag{7}$$

where Ψ is a semi-definite positive weighing matrix and $g(k)$ is the virtual reference closest to the desired reference, not allowing the system to violate the constraints. Observe that the set $\mathcal{V}$ is defined by evolving the system in closed-loop, without disturbances, a number k_0 of samples ahead, and testing if the state sequence remains inside the constraints.

2.2 Supervisor

The supervisor is responsible for selecting which i-th CG is currently active. Fig. 3 shows its block diagram, in which we see that all CGs are continuously updated, but only one is active to send control signals to the system. The policy for switching CGs is problem-dependent. It might be another controller giving better performance, according to some metric, or being able to access a state-space region.

However, even if the policy allows the change, it will not be admissible if $x(k) \notin \mathcal{X}_i \cap \mathcal{X}_j$, where

$$\mathcal{X}_i = \left\{ x \in \mathbb{R}^n | \begin{bmatrix} \omega \\ x \end{bmatrix} \in \mathcal{Z}_i \text{ for at least one } \omega \in \mathbb{R}^{n_y} \right\} \tag{8}$$

is the set of all states that can be steered to an equilibrium point without constraint violation, with

$$\mathcal{Z}_i = \left\{ \begin{bmatrix} r \\ x \end{bmatrix} \in \mathbb{R}^{n_y} \times \mathbb{R}^n | c(k) \in \mathcal{C}_i, \forall k \in \mathbb{Z}^+ \right\} \tag{9}$$

being the admissible output set.

A system can have many CG units, each of them with its constraint region. A generic CG switch $CG_i \to CG_j$ is admissible (with respect to the plant constraints) if the two CG's domains have a nonempty intersection. If one wants to go from CG_1 to CG_3 and there is no intersection, one must find a path through other CGs with nonempty intersections. Therefore the union of all CG's domains cannot be a disconnected domain. To change the active CG, one needs to go to a point interior to the CG' domains intersection, called way-point. This point can be chosen arbitrarily inside the intersection of constraints. However, it is better to use points in the central area, not so close to the borders, to avoid problems involving disturbances and controller sensitivity (Keel and Bhattacharyya, 1997). All the paths from one CG to another can be calculated offline, for example, using graph theory(Ahuja et al., 1990-04; Pettie, 2004-01).

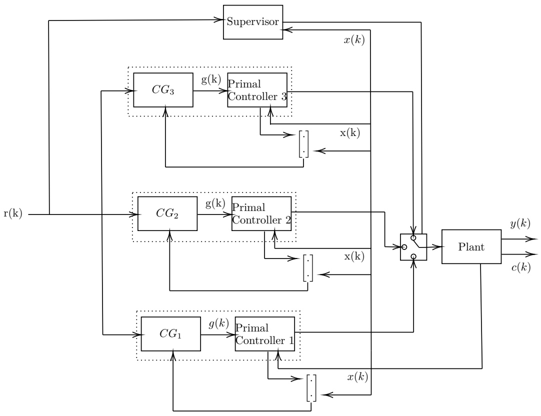

Figure 3. Supervisor Block Diagram.

In (Lucia and Franzè, 2017; Franzè et al., 2017) it has been shown that any CG switch, e.g. $CG_i \to CG_j$ can be safely accomplished (preserving the constraints) form any point belonging to the intersection between the two CGs' domains. Therefore, a possible way to achieve a safe switch is to define a waypoint reference in the intersection set. Once the plant trajectory, under the action of CG_i is confined in the intersection region, then CG_j is activated. Moreover, the minimum waiting time assuring that the state of the system enters CG_j's domain is defined resorting to the concept of guaranteed dwell-time.

This approach is a way of guaranteeing stability, as the dwell-time is calculated not to allow the switching to occur neither too soon nor too often, giving enough time to the current controller to stabilize the system before switching again. It is, however, a very conservative approach that assumes a worst-case scenario. Although it might be necessary to stabilize some systems, it is not always needed and may lead the convergence time to take longer.

3. A NEW SUPERVISOR POLICY

We propose a new switching policy that does not use dwell time, allowing switching modes earlier, by taking advantage of the concept of the region of attraction of the controllers, and yielding better closed-loop performance indexes, as, for instance, minimizing the settling time. The switch occurs in two steps: first, the CG is changed to a temporary CG (CG_{ij}), consisting of the previous CG's constraint and the target CG's controller. This switch happens when the system enters the CG_j' region of attraction. CG_j is activated when the constrained output is inside CG_{ij} and CG_j domain intersection.

3.1 Offline computations

When switching controllers earlier, it is necessary to compute an estimate of the region of attraction of the i-th controller. A region of attraction (RoA) is a region of initial conditions yielding trajectories with guaranteed convergence to the equilibrium condition. In such a case, we consider (1) under input saturation. Thus, even though the input constraints are usually included in the set $\mathcal{C}$ and are handled by the optimization machinery, at this point, we explicitly consider them, by using a saturating allowance approach (Tarbouriech et al., 2011). In such a case, we assume decentralized saturation function, $\texttt{sat}\,(u(k))$, instead of $u(k)$ as the input signal in (1), where $\bar{u}$ is the saturation value and

$$\texttt{sat}\,(u) = \texttt{sign}(U_\ell)\min\{|u_{(\ell)}|, \bar{u}_{(\ell)}\}, \quad \ell = 1, \dots, m,$$

where $u_{(\ell)}$ means the ℓ-th entry of u. An useful Lyapunov candidate function for each CG is $V(\hat{\xi}(k)) = \hat{\xi}(k)^\top P_i \hat{\xi}(k)$, with $\mathbf{0} < P_i = P_i^\top \in \mathbb{R}^{n_a \times n_a}$ and $i = 1, \dots, s$, where n_a is the system dimension plus the number of controller states (e.q. integrators). Such an estimate can be computed by solving a suitable optimization procedure for each mode, such as

$$\begin{aligned} &\min_{P_i, G_i, S_i} \texttt{Trace}(P_i) \\ &\text{s. t.} \begin{bmatrix} \mathcal{A}_i^\top P_i \mathcal{A}_i - P_i & -\mathcal{B}_i S_i + G_i^\top \\ \star & -2S_i \end{bmatrix} < \mathbf{0} \qquad (10) \\ &\begin{bmatrix} -P_i & (K_i - G_i)^\top \\ \star & -u^2 \end{bmatrix} < \mathbf{0} \end{aligned}$$

with $G_i \in \mathbb{R}^{n_u \times n_a}$, $S_i \in \mathbb{R}^{n_u \times n_u}$ is a positive diagonal matrix, for $\ell = 1, \dots, m$. This optimization procedure allows to compute a contractive level set associate with the Lyapunov function $V(\hat{\xi}(k))$, given by

$$\mathcal{L}_V(P_i) = \{\hat{\xi}(k) \in \mathbb{R}^{n_a} \ : \ \hat{\xi}(k)^\top P_i \hat{\xi}(k) \leq 1\}. \qquad (11)$$

Note that (10) uses control signal saturation ($\bar{u}$) to generate an estimate of RoA. Because we can run the optimization procedure (10) offline, it does not change the proposed switching scheme.

3.2 Implementation summary

Algorithm 1 describes how the switch should occur between the current CG and the target one. We suppose that the target CG is known, and its constraint set has a no empty intersection with the constraint set of the actual one. Thus, the next CG should be decided in a higher instance, calculated offline, as stated in the Section 2.2.

The algorithm gives the possibility of operating in a hybrid condition, i.e., with constraints of the current CG and the controller of the target CG. Thus, a vital part of our switching policy is allowing switching controllers earlier, mixing the CG operational conditions. Switching earlier is a good option when one knows that the next command governor's controller has better performance than the controller of the current one. Note that better performance can mean, but is not limited to, faster convergence.

Algorithm 1 New Supervisor Policy

```
Input: $CG_i \leftarrow$ current CG, $CG_j \leftarrow$ target GG.
let $P_1$ and $P_2$ be the solution of optimization procedure (10), for $CG_i$ and $CG_j$, respectively.
if should switch controllers earlier then
    while $\hat{\xi}(k) \notin \mathcal{L}_V(P_2)$ do
        calculate $g(k)$
        execute controller
    end while
    change current controller and model to $CG_j$'s ones
    reset integrators
end if
while $x(k) \notin \mathcal{X}_i \cap \mathcal{X}_j$ do
    calculate $g(k)$
    execute controller
end while
change to $CG_j$
reset integrators if not already done
change $r(k)$ to next way-point
restart algorithm
```

The implementation of the Algorithm 1 deserves some attention concerning the next steps.

Should controllers switch earlier? The user may implement several strategies to answer such a question. The main issue here is to determine when the controller of the target CG may perform better. Such a controller activation must happen only when the closed-loop state belongs to its RoA. Moreover, it must achieve some performance requirements, justifying an earlier activation. One case is a controller that gives a more aggressive control action, for instance, by accelerating the closed-loop convergence response. The designer must carefully evaluate the ideal criteria for each controlled process.

Hybrid CG changing: In case of changing to the next CG controller but remaining with the current CG constraints, we call such a case hybrid or partial change. Both model and controller switch. Therefore, the new model and controller should attain the same output values as its previous one.

Reset integrators: After hybrid or complete CG switching, it is necessary to adequately set the controller state ensuring the continuity of the system output. In this paper, we chose to simulate the target CG's model and controller, evolving the closed-loop to the same output value of the actual non-linear system.

Another way of dealing with integrators is to run the next CG in parallel, with some modifications: instead of minimizing the distance to the reference, minimize the distance to the actual state and use only $\omega \in \mathcal{C}$ as a constraint. Also, do not use an observer. This way, the next CG's system's internal states will evolve to a valid point that lies in its constraint. When the switch occurs, the CG will already have a valid state.

Another approach is to find the values of the integrators that make the control output the same as the actual output. This method might work well with stable systems that have continuity from one mode to another but will not work if the modes require completely different control signals (for example, switched electrical systems).

4. NUMERICAL EXAMPLES

In this section, we apply our approach to two systems and compare the achievements with the approaches presented by Franzè et al. (2017). The first example is physically motivated, and the second one concerns an unstable academic model.

4.1 Level Control System

Consider an interactive tank system as indicated in Figure 4. It consists of two coupled tanks, namely $T1$ and $T2$, that are feed by two with controlled outflows u_1 and u_2, measured in $\text{cm}^3\,\text{s}^{-1}$. The levels of each tank, h_1 and h_2 (cm), are the control objective variables.

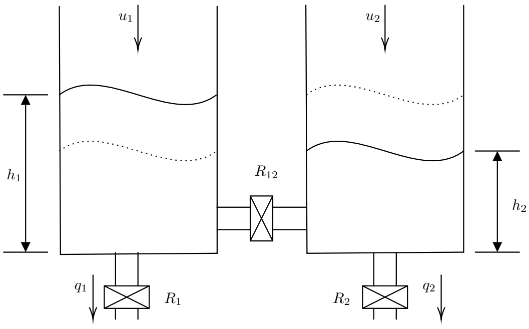

Figure 4. System of Coupled Tanks

The output flow of the $T1$ and $T2$ are denoted by q_1 and q_2 ($\text{cm}^3\,\text{s}^{-1}$), respectively, and the flow between them is noted by q_{12} ($\text{cm}^3\,\text{s}^{-1}$).

Both tanks have the same cross-section area, denoted as A (cm^2), as well as the cross-section areas of the restrictions in the outputs of the tanks, a (cm^2); g ($\text{cm}\,\text{s}^{-2}$) is the gravity acceleration. By using Bernoulli's equations, we have:

$$\begin{aligned} \dot{h}_1(t) &= \frac{u_1(t) - q_1(t) \pm q_{12}}{A}, \\ \dot{h}_2(t) &= \frac{u_2(t) - q_2(t) \mp q_{12}}{A}, \end{aligned} \tag{12}$$

where the flows are given by $q_1(t) = a\sqrt{2gh_1(t)}$, $q_2(t) = a\sqrt{2gh_2(t)}$, and $q_{12}(t) = a\sqrt{2g\,|h_2(t) - h_1(t)|}$.

This is a nonlinear and switching system, as the model changes depending on the height of the tanks' water column. At each mode, $h_1 > h_2$ or $h_1 \leq h_2$, equation (12) can be linearized around an equilibrium operational point (x_{eq}, u_{eq}) by using Jacobian matrices. In what follows, we use $a = 5.9$, $A = 961\pi$, and $g = 980.665$.

Two operational conditions with $x(k) = \begin{bmatrix} h_1(k) & h_2(k) \end{bmatrix}^\top$, are such that:

$$x_{eq}^1 = \begin{bmatrix} 57.5 \\ 43.61 \end{bmatrix}, u_{eq}^1 = \begin{bmatrix} 744 \\ 2960 \end{bmatrix}, x_{eq}^2 = \begin{bmatrix} 43.61 \\ 57.5 \end{bmatrix}, u_{eq}^2 = \begin{bmatrix} 2960 \\ 744 \end{bmatrix},$$

yielding two operational modes, each of them corresponding to a CG. After the linearization, we discretized the continuous-time model using a sample time of 5 s and Euler equations to get discrete-time model given by (1) with matrices:

$$A_1 = \begin{bmatrix} 0.92 & 0.053 \\ 0.053 & 0.91 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.91 & 0.053 \\ 0.053 & 0.92 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0.0016 & 4.5 \cdot 10^{-5} \\ 4.5 \cdot 10^{-5} & 0.0016 \end{bmatrix}, B_2 = \begin{bmatrix} 0.0016 & 4.5 \cdot 10^{-5} \\ 4.5 \cdot 10^{-5} & 0.0016 \end{bmatrix},$$

$$C_1 = C_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

and $s = 2$. Using Lyapunov's and Kalman's approaches (Chen, 2012-11; Hespanha, 2018-02) we got the following controller and observer gains:

$$K_1 = \begin{bmatrix} -875.384 & -9.217 & -297.447 & 7.982 \\ -8.505 & -849.853 & 8.514 & -279.434 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} -849.853 & -8.505 & -279.434 & 8.514 \\ -9.217 & -875.384 & 7.982 & -297.447 \end{bmatrix},$$

$$L_1 = \begin{bmatrix} -0.919 & -0.053 \\ -0.053 & -0.915 \end{bmatrix}, L_2 = \begin{bmatrix} -0.915 & -0.053 \\ -0.053 & -0.919 \end{bmatrix},$$

for the operational modes 1 and 2.

We simulated our approaches described by algorithm 1 to switch between the CGs 1 and 2. In both cases the system starts in $x(0) = \begin{bmatrix} 43.61 & 57.5 \end{bmatrix}$. The references are $r = \begin{bmatrix} 57.5 & 43.61 \end{bmatrix}$, for $0 \leq k \leq 10 \cup 50 \leq k \leq \infty$, and $r = \begin{bmatrix} 43.61 & 57.5 \end{bmatrix}$, for $11 \leq k \leq 49$. Therefore, the system must perform a closed path.

Fig. 5 shows the path taken by the closed-loop system over the space of the system's state. The trajectory marked with red-dashed line concerns the results achieved without early-switch, and the solid-blue line is related to ones with early-switch. The shadow regions concern the constraints of each mode. Also, Fig. 5 shows the borders of the estimates of the regions of attraction, RoA, for each mode, with dot-dashed lines.

It is clear that the path taken under each algorithm are almost the same. The respective control signals are given in Fig. 6, pointing to no relevant difference between the methods.

4.2 Unstable System

Consider the unstable switching system (1) with matrices:

$$A_1 = \begin{bmatrix} 1 & 0.003 \\ 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 0.0074 \\ 0 & 1.1 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0.0005 & 1.2 \cdot 10^{-6} \\ 0 & 0.0008 \end{bmatrix}, B_2 = \begin{bmatrix} 0.0019 & 3.6 \cdot 10^{-5} \\ 0 & 0.011 \end{bmatrix},$$

$$C_1 = C_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

and two operational points were selected:

$$x_{eq}^1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, u_{eq}^1 = \begin{bmatrix} -2 \\ \frac{-5}{4} \end{bmatrix}, x_{eq}^2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, u_{eq}^2 = \begin{bmatrix} \frac{-30}{19} \\ -10 \end{bmatrix}.$$

The sample-period was 0.1 s, and controller and observer gains are given by:

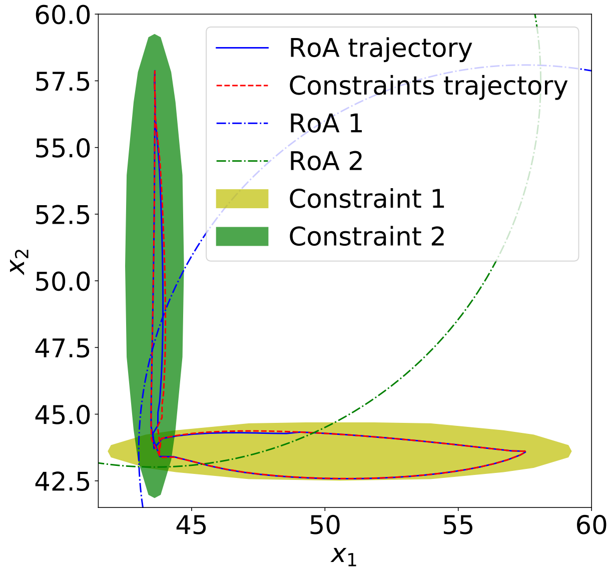

Figure 5. States trajectory for Example 1.

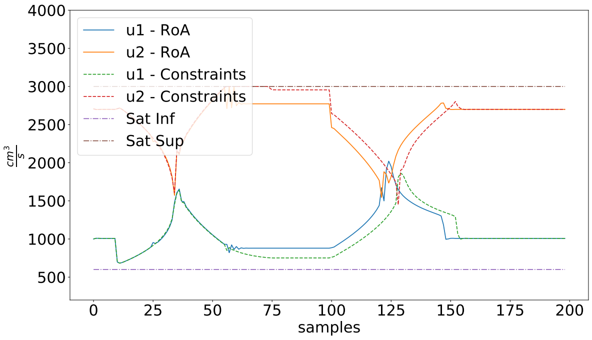

Figure 6. Control signals for Example 1.

$$K_1 = \begin{bmatrix} -2.669 \cdot 10^3 & -1.993 & -6.741 \cdot 10^2 & 1.010 \\ 3.582 \cdot 10^{-4} & -1.669 \cdot 10^3 & -3.103 \cdot 10^{-4} & -4.210 \cdot 10^2 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} -7.034 \cdot 10^2 & -1.268 & -1.769 \cdot 10^2 & 6.097 \cdot 10^{-1} \\ -3.903 \cdot 10^{-6} & -1.370 \cdot 10^2 & -1.292 \cdot 10^{-5} & -3.202 \cdot 10^1 \end{bmatrix},$$

$$L_1 = \begin{bmatrix} 0.89374311 & 0.01193574 \\ 0.00922275 & 0.91488323 \end{bmatrix}, L_2 = \begin{bmatrix} 0.64130679 & 0.0107965 \\ 0.00533561 & 0.8518838 \end{bmatrix}.$$

With the same procedures and considerations of the previous example, including the used color codes, we simulated the unstable system. Figure 7 shows the same system trajectory for both methods.

The second method shows better performance under control signal restrictions. Figure 8 reveals a difference in the control signals, where the first method, displayed in red dashed-line and green dashed-line, have higher control signal outputs than the second method, shown by blue solid-line and orange solid-line.

Both methods results in the same settling time, but with much lower control effort with the strategy of early switching.

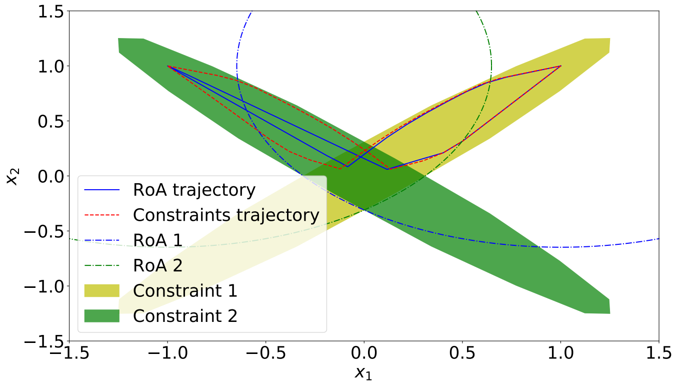

Figure 7. States trajectory for Example 2

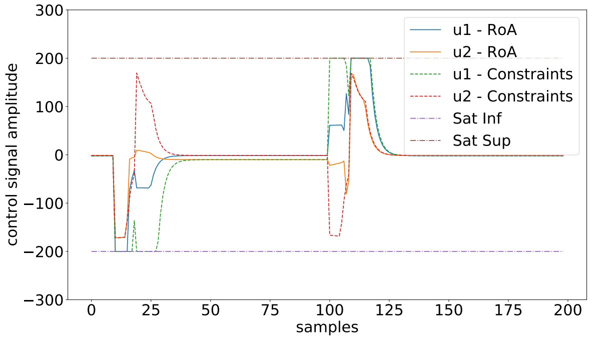

Figure 8. Control signals for Example 2

4.3 Dwell-time comparison

We simulated the system described in in (Franzè et al., 2017, Equation 25) using the approach proposed here. We used the same controllers and simulated with the same changes and times as described by Franzè et al. (2017). Our method took 15 s to converge after the first reference change, as opposed to 20 s achieved by the dwell-time method, making our approach 25% faster.

5. RESULTS AND DISCUSSIONS

We presented two new schemes for Control Governor switching that do not use dwell-time, leading to faster closed-loop responses. When the control's objective is to take the system from one operational condition to another, it might not be necessary to calculate a dwell-time. A dwell-time is only necessary to prevent the system from switching back and forth and becoming unstable, which is not the case when the system will switch and move away from the switching condition.

In the case of the tanks presented, the switch can only occur at the intersection of the constraints. The supervisor will move the reference to the next way-point as soon as the CGs are swapped, disqualifying any further CG change. It is then only necessary to show the stability of the switch.

It is possible to calculate the region of attraction of the controller designed using the Lyapunov theorem. It is then possible to guarantee that, once switched, the system will converge to the new reference. Because of this, both methods avoid waiting for the system to converge to a way-point. The advantage is then a shorter time to reach the real reference.

6. CONCLUSIONS

The proposed method allow for faster mode switching when the system can not, by design, go back to the previous mode. The approach allow us to bypass the dwell-time and achieve better closed-loop performance indexes, such as the settling time. The use of the region of attraction can be further investigated to easy the work done by the command governor, by exploring the contractivity properties of the estimate region of attraction.

REFERENCES

Ahuja, R.K., Mehlhorn, K., Orlin, J., and Tarjan, R.E. (1990-04). Faster algorithms for the shortest path problem. *Journal of the ACM (JACM)*, 37(2), 213–223. doi:10.1145/77600.77615. URL `https://doi.org/10.1145/77600.77615`.

Bemporad, A., Casavola, A., and Mosca, E. (1997). Nonlinear control of constrained linear systems via predictive reference management. *IEEE Transactions on Automatic Control*. doi:10.1109/9.557577.

Boyd, S., Ghaoui, L.E., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities In Systems And Control Theory*. SIAM Studies in Applied Mathematics.

Briat, C. (2014). *Linear parameter-varying and time-delay systems*, volume 3. Springer.

Casavola, A., Mosca, E., and Angeli, D. (2000). Robust command governors for constrained linear systems. *IEEE Transactions on Automatic Control*. doi:10.1109/9.887628.

Chen, C.T. (2012-11). *Linear System Theory and Design (The Oxford Series in Electrical and Computer Engineering)*. Oxford University Press. URL `https://www.xarg.org/ref/a/0199959579/`.

Chesi, G., Colaneri, P., Geromel, J.C., Middleton, R., and Shorten, R. (2010). Computing upper-bounds of the minimum dwell time of linear switched systems via homogeneous polynomial Lyapunov functions. In *Proceedings of the 2010 American Control Conference, ACC 2010*. doi:10.1109/acc.2010.5530590.

Franzè, G., Lucia, W., and Tedesco, F. (2017). Command governor for constrained switched systems with scheduled model transition dwell times. *International Journal of Robust and Nonlinear Control*. doi:10.1002/rnc.3841.

Gilbert, E.G. and Kolmanovsky, I. (1999). Fast reference governors for systems with state and control constraints and disturbance inputs. *International Journal of Robust and Nonlinear Control*. doi:10.1002/(SICI)1099-1239(19991230)9:15<1117::AID-RNC447>3.0.CO;2-I.

Gilbert, E.G., Kolmanovsky, I., and Tan, K.T. (1995). Discrete-time reference governors and the nonlinear control of systems with state and control constraints. *International Journal of Robust and Nonlinear Control*. doi:10.1002/rnc.4590050508.

Hespanha, J.P. (2018-02). *Linear Systems Theory*. Princeton Press. ISBN13: 9780691179575.

Kapasouris, P., Athans, M., and Stein, G. (1988). Design of feedback control systems for stable plants with saturating actuators. In *Proceedings of the IEEE Conference on Decision and Control*. doi:10.1109/cdc.1988.194356.

Keel, L.H. and Bhattacharyya, S.P. (1997). Robust, fragile, or optimal? *IEEE Transactions on Automatic Control*, 42(8), 1098–1105.

Klug, M., Castelan, E.B., Leite, V.J.S., and Silva, L.F.P. (2015). Fuzzy dynamic output feedback control through nonlinear takagi–sugeno models. *Fuzzy Sets and Systems*, 263, 92–111. doi:10.1016/j.fss.2014.05.019. URL `https://doi.org/10.1016/j.fss.2014.05.019`.

Liberzon, D. and Morse, A.S. (1999). Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine*, 19(5), 59–70.

Liberzon, D. (2003). *Switching in systems and control*. Birkhauser.

Lin, H. and Antsaklis, P.J. (2009). Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic Control*. doi:10.1109/TAC.2008.2012009.

Lopes, A.N.D., Leite, V.J.S., Silva, L.F.P., and Guelton, K. (2020). Anti-windup TS fuzzy PI-like control for discrete-time nonlinear systems with saturated actuators. *International Journal of Fuzzy Systems*, 22(1), 46–61. doi:10.1007/s40815-019-00781-0. URL `https://doi.org/10.1007/s40815-019-00781-0`.

Lucia, W. and Franzè, G. (2017). Stabilization and reference tracking for constrained switching systems: A predictive control approach. *International Journal of Adaptive Control and Signal Processing*. doi:10.1002/acs.2804.

Pettie, S. (2004-01). A new approach to all-pairs shortest paths on real-weighted graphs. *Theoretical Computer Science*, 312(1), 47–74. doi:10.1016/s0304-3975(03)00402-x. URL `https://doi.org/10.1016/s0304-3975(03)00402-x`.

Tarbouriech, S., Garcia, G., Gomes da Silva Jr., J.M., and Queinnec, I. (2011). *Stability and stabilization of linear systems with saturating actuators*. Springer Science.

Vahidi, A., Kolmanovsky, I., and Stefanopoulou, A. (2007). Constraint handling in a fuel cell system: A fast reference governor approach. *IEEE Transactions on Control Systems Technology*. doi:10.1109/TCST.2006.883242.

Wang, L. (2009). *Model Predictive Control System Design and Implementation Using MATLAB*. Advances in Industrial Control. Springer-Verlag London, 1 edition. doi:10.1007/978-1-84882-331-0. URL `https://goo.gl/czFKuq`.

Yu, Hai-Hua; Duan, G. (2013). *LMIs in control systems: analysis, design and applications*. CRC Press.

Zhang, X. (2016). Fast MPC Solvers for Systems with Hard Real-Time Constraints.