Protecting Systems from Violating Constraints Using Reference Governors*

Ilya Kolmanovsky[1*] and Nan Li[1]

[1*]Department of Aerospace Engineering, The University of Michigan, 1320 Beal Avenue, Ann Arbor, 48109-2140, Michigan, USA.

*Corresponding author(s). E-mail(s): ilya@umich.edu;
Contributing authors: nanli@umich.edu;

Abstract

Reference governors are add-on predictive safety supervision algorithms that monitor and modify, if it becomes necessary, commands passed to a nominal system to ensure that pointwise-in-time state and input constraints are satisfied. After briefly surveying the basics of the reference governor schemes, this paper describes several recent extensions of the reference governors. These include reduced order reference governors with flexible error budget, reference governors for nonlinear systems that exploit the logarithmic norm for response bounding, stochastic reference governors, and controller state and reference governors. Learning reference governors that are capable of handling constraints in uncertain systems are also discussed.

Keywords: State and control constraints, Reference governors, Predictive control, Constrained control

1 Introduction

Constraints refer to limits imposed on state and control variables which must be satisfied during system operation. Examples of constraints include, but are not limited to, actuator range and rate limits, pressure and temperature

---

*On behalf of all authors, the corresponding author states that there is no conflict of interest.

safety limits and obstacle avoidance requirements. With the continuing trends towards growing system autonomy, improved performance and downsizing, constraint handling and limit protection functions are becoming increasingly important to enable engineered systems to operate safely at the "limits."

Figure 1 illustrates the basic arrangement of a closed-loop system consisting of a plant (system being controlled) and a controller that generates the control signal $u(t)$, where t denotes time, in response to the reference command (set-point) $r(t)$ and plant state or output measurements. A disturbance or uncertainty $w(t)$ may also affect the plant response. In traditional control design, the objective is to design a controller that generates a control signal $u(t)$ to track the reference command $r(t)$ by suitable chosen performance variables $z(t)$, so $z(t)$ must be approximately equal to $r(t)$ and this should happen despite disturbances/uncertainties $w(t)$ acting on the plant. When a system also has state and control constraints, what this means is that there are additional requirements that certain variables or outputs $y(t)$ of the closed-loop system belong to a given set Y for all times, i.e., $y(t) \in Y$ for all t.

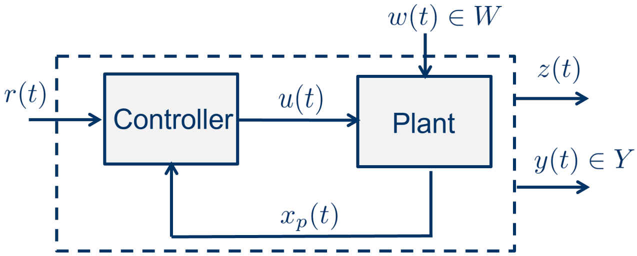

Fig. 1 Closed-loop system with constraints.

One of the prototypical examples of a constrained system is an aircraft gas turbine engine [1]. Such an engine has many constraints including low pressure compressor surge margin, high pressure compressor surge margin, overspeed and overtemperature limits, combustion lean blowout limit, flight idle limit, actuator range and rate limits, and so on. The control system of the gas turbine engine consists of two parts, power management and limit protection. It turns out that the limit protection subsystem, that takes care of constraints, is in many ways more extensive in terms of algorithmic and software footprint than the power management subsystem, which adjusts fueling rate to generate thrust. In other applications, by enforcing constraints through the control design, rollover for cars, trucks and ships, battery catching fire due to overheating, or aircraft disintegration in flight due to excessive structural loads induced by aggressive maneuvering can be avoided.

This article describes the reference governor, an add-on predictive safety supervision algorithm that monitors and modifies, if it becomes necessary, commands passed to the nominal system to ensure that constraints are satisfied. The basic arrangement with the reference governor augmenting the nominal closed-loop system is shown in Figure 2. The reference governor takes the reference command $r(t)$ and modifies it, if it becomes necessary, to a safe reference command $v(t)$. The reference governor is inactive, so $v(t) = r(t)$, if there is no danger of constraint violation, and it will *minimally* modify the command and make $v(t)$ different from $r(t)$ if it becomes necessary to prevent constraint violation. As it turns out, such reference governor schemes can be developed so that they are easy to implement and can operate with very fast online (onboard) computations. In a nutshell, the reference governor design problem boils down to defining a function that maps $r(t)$, the state of the closed-loop system $x(t)$ and possibly past modified commands to $v(t)$.

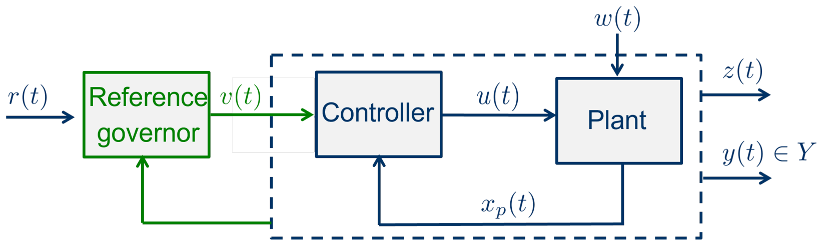

Fig. 2 Reference governor augmenting the nominal closed-loop system.

The reference governor is particularly helpful in a common situation, where one already has a controller providing satisfactory tracking performance, however, this legacy controller does not handle constraints. Due to changed performance requirements (e.g., demands for increased agility) or due to system downsizing the control system designers often find themselves in a situation that constraints now have to be dealt with. In this case, they could replace the legacy controller with an entirely different controller that handles constraints such as a Model Predictive Controller [2]. Another approach, which is appealing to practitioners as they are already familiar with the legacy controller and reluctant to introduce drastic changes, is to simply augment the existing controller with some kind of constraint enforcing mechanism. A reference governor is one such add-on scheme that monitors the reference commands to which the controller responds and modifies them if they create a danger of constraint violation to preserve safety.

A comprehensive survey of the existing reference governor theory and its various applications is contained in [3]. The objective of this paper is to review and comment on the basic reference governor ideas and describe several more recent extensions of the reference governor schemes that are not introduced in [3], including reduced order reference governors with flexible error budget, reference governors for nonlinear systems that exploit the logarithmic norm for response bounding, stochastic (controller state and) reference governors, and learning reference governors. For the latter, we will also mention several open problems. This paper follows a plenary talk by the first author given at 2021 International Conference on Informatics in Control, Automation and Robotics (ICINCO).

2 Reference Governor Basics

2.1 Safe sets and scalar reference governors

The key ingredient in constructing the reference governor is the notion of the *safe set* defined as follows. Suppose we have a discrete-time model of the nominal closed-loop system of the form,

$$x(t+1) = f(x(t), v(t), w(t)), \tag{1}$$
$$y(t) = h(x(t), v(t)) \in Y, \tag{2}$$

where $w(t) \in W$ for all $t \in \mathbb{Z}_{\geq 0} = \{0, 1, 2, \cdots\}$ is an unmeasured disturbance bounded in a specified set W. Then the safe set is the set of all constant commands and initial states that do not lead to constraint violation no matter what the disturbance w is acting on the system:

$$O_\infty = \{(v_r, x(0)) : \ v(t) \equiv v_r, \ w(t) \in W \ \forall t \geq 0 \Rightarrow y(t) \in Y \ \forall t \geq 0\}. \tag{3}$$

We refer to O_∞ as the safe set somewhat informally; more formally it is called the maximum output admissible set (MOAS) [4].

And now with the safe set defined, the simplest reference governor that has been proposed is the scalar reference governor or SRG, in which the modified reference $v(t)$ is a linear interpolation between the previous value of the modified reference, $v(t-1)$, and desired reference, $r(t)$, with the interpolating factor denoted by $\kappa(t)$ and determined as a solution to the following optimization problem:

$$\begin{aligned} &\text{Maximize } \kappa(t) \\ &\quad \text{subject to} \\ &\qquad\qquad v(t) = v(t-1) + \kappa(t)(r(t) - v(t-1)), \\ &\qquad\qquad 0 \leq \kappa(t) \leq 1, \\ &\qquad\qquad \begin{bmatrix} v(t) \\ x(t) \end{bmatrix} \in P \subseteq O_\infty. \end{aligned} \tag{4}$$

If $\kappa(t)$ is set to 1, $v(t)$ is equal to $r(t)$ and reference command is passed through. If $\kappa(t)$ is chosen to be less than 1, the reference command is modified. To ensure that constraints are satisfied and reference modification is kept to a minimum, $\kappa(t)$ is chosen as the maximum value from the interval $[0, 1]$ subject to the condition that the updated reference, $v(t)$, and the system state, $x(t)$, are safe, that is, they are in the subset called P of MOAS O_∞. This subset P can be quite general. For instance, it does not need to be invariant under constant commands, and, in fact, simpler choices of P lead to simpler optimization problems. For non-invariant (under constant commands) sets P, the SRG optimization problem may become infeasible for some t; for these t then $\kappa(t)$ is set to zero. So to implement SRG, we need to be able to compute the set P offline and then use it to determine $\kappa(t)$ online.

2.2 Implementation of reference governors based on linear models

The offline computations of P are simplest in the case of linear models of the form:

$$x(t+1) = Ax(t) + Bv(t), \tag{5}$$

$$y(t) = Cx(t) + Dv(t) \in Y, \tag{6}$$

where A is stable (i.e., a Schur matrix with all eigenvalues strictly inside the unit disk of the complex plane), and when the constraints are imposed on the output which is a linear function of the state x and the command v, while the constraint set Y is polyhedral and given by affine inequalities

$$Y = \{y : \; Hy \le h\},$$

with $h > 0$, where the inequality is understood as componentwise. One advantage of linear models is that it is very easy to predict future response of $y(t)$ to an initial state $x(0)$ and constant command, v; this prediction is given by

$$v(t) = v \; \forall t \Rightarrow y(t) = CA^t x(0) + C(I - A^t)(I - A)^{-1}Bv + Dv,$$

which is what the state transition formula reduces to in the case of constant input. Then one can construct a set, $\tilde{O}_\infty$, which is an inner approximation of O_∞ and is defined by linear inequalities:

$$\tilde{O}_\infty = \Bigg\{ (v, x(0)) : \begin{bmatrix} HD & HC \\ HCB + HD & HCA \\ \vdots & \vdots \\ HC(I - A^{t^*})(I-A)^{-1}B + HD & HCA^{t^*} \\ HC(I-A)^{-1}B + HD & 0 \end{bmatrix} \begin{bmatrix} v \\ x(0) \end{bmatrix} \le \begin{bmatrix} h \\ h \\ \vdots \\ h \\ (1-\epsilon)h \end{bmatrix} \Bigg\}.$$

In this expression, each row corresponds to the requirement that the predicted output t steps ahead is safe, i.e., satisfies the constraints [3]. It turns out that one only needs to build these rows up to some finite time instant $t^* = t^*(\epsilon)$ (which can be computed or estimated a priori) if the constraints on the steady-state output, represented by the last row, are slightly tightened with $1 \gg \epsilon > 0$. Finally, the set P can be generated from $\tilde{O}_\infty$ by eliminating redundant and almost redundant inequality constraints and using a tightening procedure [5]. Thus the offline part of constructing P is quite straightforward.

The online computations of κ are also easy, as κ is a scalar. The conditions that $v(t)$ and $x(t)$ belong to P, when $P = \{(v,x) : H_v v + H_x x \leq h\}$ is a polytope, reduce to solving a system of linear inequalities where the only unknown is the scalar variable κ:

$$\begin{bmatrix} v(t-1)+\kappa(r(t)-v(t-1)) \\ x(t) \end{bmatrix} \in P \Leftrightarrow \begin{array}{r} a_i + b_i\kappa \leq c_i, i = 1, \cdots, n_P, \\ 0 \leq \kappa \leq 1, \end{array}$$

where $a_i = H_{v,i}v(t-1) + H_{x,i}x(t)$, $b_i = H_{v,i}(r(t) - v(t-1))$, $c_i = h_i$, $i = 1, \cdots, n_P$. Solving such a system can be done explicitly without needing any iterative solver, that is, one can write down an explicit formula for the solution:

$$\kappa_U = \min\{\min_{i:\ b_i>0}\{\frac{c_i - a_i}{b_i}\}, 1\},$$

$$\kappa_L = \max\{\max_{i:\ b_i<0}\{\frac{c_i - a_i}{b_i}\}, 0\},$$

$$\kappa = \begin{cases} \kappa_U & \text{if } \kappa_L \leq \kappa_U \text{ and } a_i \leq c_i \text{ for all } i \text{ such that } b_i = 0, \\ 0 & \text{otherwise.} \end{cases}$$

Hence is very easy to compute $\kappa(t)$ online while software implementing such computations is quite simple.

Much of the same offline and online procedure extends to the case of linear systems with set-bounded disturbances, $w(t) \in W$. The extension relies on the use of Pontryagin (P)-difference [6] between sets

$$X \sim W = \{x :\ x + w \in X\ \forall w \in W\},$$

and the superposition principle for linear systems which enables to account for the effect of the disturbance on the system response with easy-to-mechanize set operations. The details are given in [5] and elsewhere.

Example 1 To provide an illustration of the reference governor operation, we consider a simple example of single axis reorientation of a spacecraft with a flexible appendage. The equations of motion in continuous-time are given by

$$J\ddot{\theta} + \sqrt{2}\delta\ddot{q} = u$$

$$\ddot{q} + \sigma^2 q + \sqrt{2}\delta\ddot{\theta} = 0$$

where the angle θ specifies spacecraft orientation and q is the modal coordinate of the first elastic mode that informs the appendage deflection. The values of the parameters are $J = 3026$, $\delta = 35.865$, $\sigma = 1.112$. The nominal controller is of Proportional-Derivative (PD) type and generates the control moment, u, according to

$$u = -K_p(\theta - v) - K_d\dot{\theta},$$

$$K_p = J\omega_n^2, \ K_d = 2J\zeta\omega_n,$$

where v is the target spacecraft orientation. The controller is purposefully made quite aggressive to enable agile maneuvering with the desired closed-loop natural frequency $\omega_n = 0.05$ and damping ratio $\zeta = 0.4$. Suppose now the constraints are imposed on the orientation, θ, elastic deflection, q, and control signal, u:

$$|\theta| \leq \frac{\pi}{2}, \ |q| \leq 0.01, \ |u| \leq 0.5.$$

Figure 3 shows that without the reference governor the nominal closed-loop system violates the constraints when responding to larger orientation change commands. On the other hand, the reference governor designed on a discrete-time model obtained from the continuous-time closed-loop system model assuming an update period of $T_s = 0.1$ sec can successfully prevent constraint violations (Figure 4). It does so by modifying the reference command from steps to ramp-like profiles. Note that in this example we assumed a continuous-time (analog) PD controller. Of course, a digital PD controller can also be treated as easily.

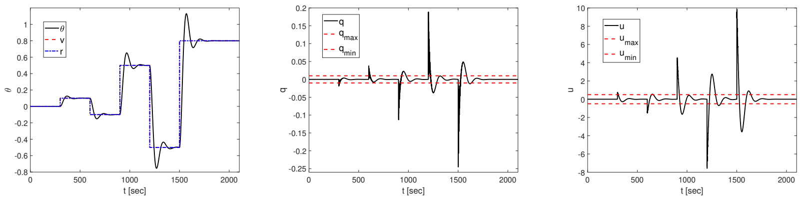

Fig. 3 Time histories of θ (left), q (middle) and u (right) without the reference governor ($v = r$) in response to command steps in r shown by dash-dotted blue lines (left). Constraints are shown by horizontal dashed red lines.

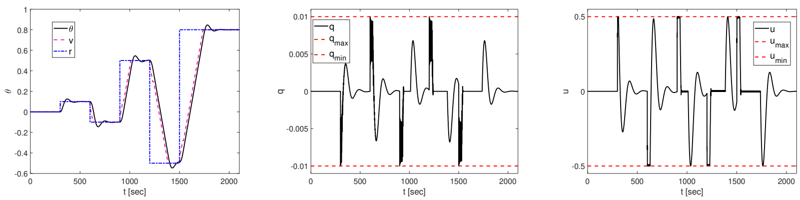

Fig. 4 Time histories of θ (left), q (middle) and u (right) with the reference governor in response to command steps in r shown by dash-dotted blue lines (left). The time history of $v \neq r$ is shown by magenta dashed line on the left plot. Constraints are shown by horizontal dashed red lines.

2.3 Implementation of reference governors based on nonlinear models

It is possible to extend the reference governor construction to nonlinear systems with set-bounded disturbances and parametric uncertainties [7]. For nonlinear systems, we generally look for a set P as a sub-level set of a continuous function $\tilde{V}$:

$$P = \{(v, x(0)) : \tilde{V}(v, x(0)) \leq 0\}.$$

This subset should be safe and strongly returnable, that is, any state trajectory corresponding to a constant command and initial state in this set should firstly satisfy constraints (that is, be safe) and secondly it should eventually enter the interior of the set P (which means strongly returnable). There are several methods of constructing such a $\tilde{V}$ function.

One approach is to construct it from a Lyapunov function of the closed-loop system if such a Lyapunov function is known. Specifically, $\tilde{V}(v, x) = V(v, x) - \Gamma(v)$ where V is a Lyapunov function parameterized by the reference. This approach is illustrated and experimentally validated in [8] for Electromagnetically Actuated Mass Spring Damper (EAMSD) system.

Another approach is data-driven and involves machine-learning applied to train a classifier from simulation results or experimental data. With this approach, the system response to different initial conditions and constant commands is determined and a classifier is trained to distinguish between safe and unsafe state-command combinations. This corresponds to

$$\tilde{V}(v, x) = \min_j \phi_j(v, x),$$

where ϕ_j is one of possibly several trained classifiers such that

$$\phi_j(v, x) \leq 0 \Rightarrow (v, x) \text{ is safe.}$$

Finally, it is possible to avoid constructing $\tilde{V}$ explicitly at all, and simply use the online prediction of the response of (1)-(2) to a given initial condition and constant command over a sufficiently long prediction horizon [9]. Specifically, suppose all constraints are of the form, $h_i(x, v) \leq 0$, $i = 1, \cdots, n_Y$, and $x(t;\ v, x(0))$ denote the predicted response of the state at time t to a constant command v and an initial state $x(0)$ at the initial time of 0. Then, $\tilde{V}$ is the maximum predicted constraint violation, i.e.,

$$\tilde{V}(v, x(0)) = \max_{t=0,\cdots,T} \left\{ \max_{i=1,\cdots,n_Y} \{h_i(x(t;\ v, x(0)), v)\} \right\},$$

where T is the chosen (sufficiently long) time horizon.

The online computation of κ in the SRG case for nonlinear systems is very similar to the linear case, except that the constraints on κ are imposed in terms of a nonlinear function, $\tilde{V}$:

$$\begin{aligned}&\text{Maximize } \kappa(t)\\&\quad\text{subject to}\\&\qquad v(t) = v(t-1) + \kappa(t)(r(t) - v(t-1)),\\&\qquad 0 \le \kappa(t) \le 1,\\&\qquad \tilde{V}(v(t), x(t)) \le 0.\end{aligned}$$

If no feasible solution exists, $\kappa(t)$ is set to 0 and some extra logic is added to accept small increments $v(t) - v(t-1)$, only if the value of $\tilde{V}$ function is sufficiently decreased, specifically, $\tilde{V}(v(t-1), x(t)) \le -\epsilon$, where $\epsilon > 0$ is chosen consistently with the assumptions in [7]. To determine κ, one can use bisections [9], or in some cases, for instance, if $\tilde{V}$ is quadratic, the solution can be given by an explicit formula.

Furthermore, when a $\tilde{V}$ function as introduced above is known, it is possible to use an explicit feedback law to dynamically modify the reference $v(t)$ instead of relying on online optimizations to determine $v(t)$. Reference governors in that form are called the *Explicit Reference Governor* [10–12], which is particularly suitable for applications where computational power is too limited or system dynamics are too fast to implement optimization-based strategies.

Example 2 To provide a simple illustration of the ease with which the reference governor can be designed, we consider the following example of a continuous-time closed-loop system,

$$\dot{x}_1 = x_1^2 + u, \tag{7}$$

$$u = \mathtt{sat}_{[-20,20]}\big(-v^2 - k_P(x_1 - v) - k_I x_2\big), \tag{8}$$

$$\dot{x}_2 = x_1 - v, \tag{9}$$

which is a first order nonlinear system (7) under Proportional-plus-Integral (PI) control (8)-(9) with $k_P = 5.6$ and $k_I = 16$ plus feedforward. The state x_2 is the integrator state (part of the controller rather than the plant). The k_P and k_I values are designed to locally stabilize the system. As there are range limits on the actuator, the control signal is saturated to the interval $[-20, 20]$; this saturation is treated as a nonlinearity (i.e., not handled as a constraint). The system is highly nonlinear and its trajectories can have a finite escape time due to x_1^2 term in (7); for instance, the response to $x_1(0) = -4$, $x_2(0) = 0$ and the reference command $v = 2.56$ exhibits such instability. The PI controller is able to locally pre-stabilize the equilibria of the form $[v, 0]^\mathsf{T}$ for $-4 \le v \le 2.56$. Safe combinations of v and $x(0)$ must satisfy two requirements: stability, which is tested through simulations by checking whether $|x_1(30) - v| < 0.01$, and overshoot being less than 3.5 which is tested through simulations by checking $x_1(t) \le 3.5$ for all $0 \le t \le 30$. Safe and unsafe combinations are shown in Figure 5 along with the set P chosen as a union of 4 polyhedral sets, $A^{(i)} z \le b^{(i)}$, $z = [v, x^\mathsf{T}]^\mathsf{T}$ each well within the region of safe points and

$$\tilde{V}(v, x) = \min_{i=1,2,3,4}\{\max_{j=1,\cdots,6}\{A_j^{(i)} z - b_j^{(i)}\}\},$$

where $j = 1, \cdots, 6$ indexes the rows of $A^{(i)}$. The thinnest of these sets, shown by red color, contains the equilibrium manifold for $-4 \leq v \leq 2.56$ in its interior and hence P is strongly returnable; the other sets are thicker and surround the equilibrium manifold for $-4 \leq v \leq 1$, $-4 \leq v \leq 0$ and $-4 \leq v \leq -1$, respectively. The cross section of each of these sets in $x_1 - v, x_2$ directions is square and maximized subject to the constraint that each of these sets is not containing any unsafe points. The response of x_1 and u to the command profile r with the resulting reference governor generating v is given in Figure 6. Note that the closed-loop response is stable and tracks the reference command, while the overshoot stays below the imposed constraint of 3.5. The reference governor was updating the reference each $T_s = 0.1$ sec. Note that since the set P is defined based on continuous-time simulations, the same set P can be used to implement the reference governor with any other update period, T_s, while intersample constraint violations are avoided. Clearly, the reference governor has been able to extend the safe domain of attraction of the closed-loop system. Other choices of the set P, e.g., using machine learning based classifiers trained from the set of safe and unsafe points in Figure 5 can further improve the speed of the closed-loop response.

A simpler first order filter between r and v, $\dot{v} = -\frac{1}{\tau}(v - r)$ with a fixed time constant (e.g., $\tau = 0.5$ sec tuned by trial and error) could also be used to preserve closed-loop stability in transitions between different commands and avoid overshoot constraint violation, see Figure 7. Such a filter unnecessarily slows down system response, which is visible in particular on steps down in command; however, unlike a typical reference governor such a first order filter is purely a feedforward (open-loop) solution and does not require any state measurements. The use of such sensorless control for stabilization is interesting in that feedforward control is normally not thought of as being able to provide stabilization, with a notable exception of vibration control [13]. Sensorless feedforward reference governors can also be designed [14].

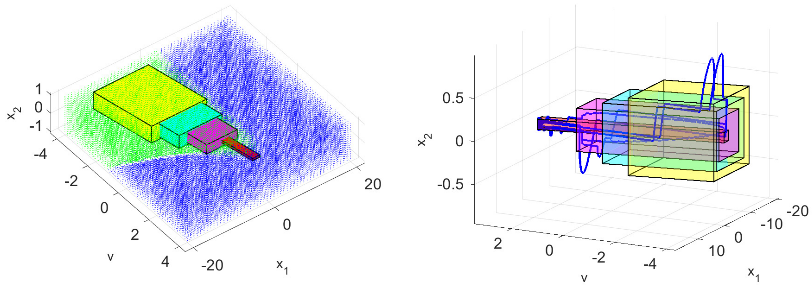

Fig. 5 Left: The set P defined as the union of four polyhedral sets, unsafe points (blue) and safe points (green). Right: The trajectory with the reference governor superimposed over the set P.

2.4 Reference governor theoretical properties

There is much known about reference governor theoretical properties for linear and nonlinear systems with set-bounded disturbances and parameter uncertainties [3]. In particular, feasibility at the initial time, that is, the ability to

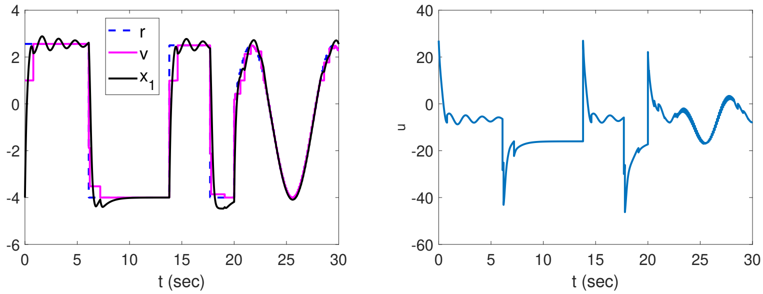

Fig. 6 Left: Time histories of r, v and x_1 with the reference governor. Right: Time history of u.

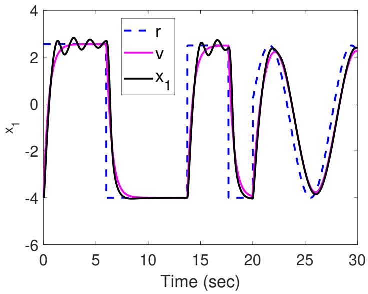

Fig. 7 Time histories of r, v and x_1 with the constant time constant pre-filter.

find $v(0)$ for the given initial state $x(0)$ such that $(v(0), x(0)) \in P$ (or even just $(v(0), x(0)) \in O_\infty$) implies constraint adherence for all future time instants, a property called recursive feasibility. Furthermore, the modified reference, $v(t)$, can be guaranteed to converge to strictly steady-state constraint admissible constant reference commands, r, in finite-time. Similar convergence results hold for nearly constant and slowly-varying commands $r(t)$ under suitable assumptions. Finally, the reference governor enlarges the constrained domain of attraction, i.e., the set of safely recoverable initial states, as compared to the case when the reference is just passed through. This has been illustrated for system (7)-(9) as, for instance, the response to $x_1(0) = -4$, $x_2(0) = 0$ and the reference command $v = r = 2.56$ exhibits instability and violates the overshoot constraint but not if the reference governor is added.

3 Recent Reference Governor Extensions

Several dozens of extensions to the basic reference governor schemes above have been developed in the published literature. Many of them are surveyed in [3]. In the remainder of this paper we comment on a few that have been developed more recently.

3.1 Reduced order reference governor with flexible error budget

The reduced order reference governor applies to closed-loop systems that can be decomposed into a slow and fast subsystems and that after appropriate coordinate transformations admit the following form,

$$x_s(t+1) = A_s x_s(t) + B_s v(t), \tag{10}$$
$$x_f(t+1) = A_f x_f(t) + B_f v(t), \tag{11}$$
$$y(t) = C_s x_s(t) + C_f x_f(t) + Dv(t) \in Y, \tag{12}$$

where Y is a compact and convex set with the origin in its interior. The matrices A_s for the slow subsystem and A_f for the fast subsystem are assumed to be Schur. The objective is to design a reference governor to protect against constraint violations in (12) without relying on the information about the fast states, x_f. That is, ideally, one would design the reference governor based on the reduced order model,

$$x_s(t+1) = A_s x_s(t) + B_s v(t),$$
$$y_{\mathsf{s}}(t) = C_s x_s(t) + (C_f \Gamma_f + D)v(t) \in Y,$$

where $\Gamma_f = (I - A_f)^{-1} B_f$ and fast variables in the output equation are replaced by their steady-state values. Since in transients, fast variables deviate from the steady-state, this is clearly not enough.

We proceed as follows. Define $e(t) = x_f(t) - \Gamma_f v(t-1)$, which is the error between the steady-state of the fast subsystem corresponding to $v(t-1)$ and current value of the fast state, $x_f(t)$. Suppose that $v(t-1)$ changes to $v(t)$ and stays constant afterwards, $v(t+k) = v(t)$, $k \geq 0$. Then, by straightforward algebraic manipulations based on the definition of $e(t)$ and the dynamic equation of $x_f(t)$ above, it follows that

$$e(t+1) = A_f e(t) - A_f \Gamma_f (v(t) - v(t-1)), \tag{13}$$
$$e(t+k) = A_f e(t+k-1) = A_f^{k-1} e(t+1), \ k > 1. \tag{14}$$

Equations (13)-(14) reveal that the deviation of the fast states from the steady-state can be managed by changing v slowly so that $v(t) - v(t-1)$ is small in magnitude.

The theory of reduced order reference governor is developed in [15, 16] essentially based on this insight. Here we illustrate the reduced order reference governor construction in a slightly more general form, with the flexible error budget, motivated by [17].

Formally, the reduced order reference governor design process is based on selecting a compact and convex set E_x, with the origin in its interior, and adjusting v to always maintain the error bounded as

$$e(t) \in c(t)E_x \text{ for all } t \geq 0, \quad 0 < c(t) \leq 1.$$

The set E_x must be contractive under A_f, i.e.,

$$A_f E_x \subseteq \lambda E_x \quad 0 \leq \lambda < 1. \tag{15}$$

The flexible error budgeting refers to choosing $c(t)$ online differently for different t (of course, subject to certain conditions) rather than using a constant value. Based on (13)-(14), $e(t+1) \in c(t+1)E_x$ can be ensured by choosing $v(t)$ and $c(t+1)$ so that the following condition is satisfied

$$-A_f\Gamma_f(v(t)-v(t-1)) \in (c(t+1)-\lambda c(t))E_x, \quad 1 \geq c(t+1) \geq \lambda c(t). \tag{16}$$

To complete the reference governor design, we need to satisfy two main properties: safety (the application of $v(t+k) = v(t)$ should result in no future constraint violations) and invariance (same safety property must hold for the pair $(v(t), x_s(t+1))$.

Firstly, note that based on (13)-(14) and (15), $e(t+1) \in c(t+1)E_x$ implies for $k \geq 1$ that $e(t+k) \in A_f^{k-1}c(t+1)E_x \subseteq \lambda^{k-1}c(t+1)E_x \subseteq c(t+1)E_x$. Hence

$$\begin{aligned}
y(t+k) &= C_s x_s(t+k) + C_f(e(t+k) + \Gamma_f v(t+k-1)) + Dv(t+k) \\
&\in \left\{C_s x_s(t+k) + (C_f\Gamma_f + D)v(t)\right\} \bigoplus C_f c(t+1)E_x, \quad k \geq 1 \\
y(t) &= C_s x_s(t) + C_f(e(t) + \Gamma_f v(t-1)) + Dv(t) \\
&\in \left\{C_s x_s(t) + (C_f\Gamma_f + D)v(t-1) + D(v(t)-v(t-1))\right\} \bigoplus C_f c(t)E_x
\end{aligned}$$

Define for $0 < c < 1$ the set

$$\tilde{Y}^c = \left\{y: \ y \bigoplus C_f c E_x \subseteq Y\right\} = Y \sim (C_f c E_x)$$

and suppose this set is nonempty with the origin in the interior. Then let

$$\begin{aligned}
\tilde{O}_\infty^c = \Big\{&(v, x_s): \ C_s A_s^k x_s + C_s(I-A_s)^{-1}(I-A_s^k)B_s v \\
&+(C_f\Gamma_f + D)v \in \tilde{Y}^c, \ k = 0, 1, \cdots, \\
&\text{and } C_s(I-A_s)^{-1}B_s v + (C_f\Gamma_f + D)v \in (1-\epsilon)\tilde{Y}^c\Big\},
\end{aligned} \tag{17}$$

where $1 \gg \epsilon > 0$. The last constraint ensures strict steady-state admissible and that the set $\tilde{O}_\infty$ is finitely determined. To ensure that the pair $(v(t), x_s(t))$

is safe based on the above, it is sufficient to ensure that

$$(v(t), x_s(t)) \in \tilde{O}_\infty^{c(t+1)}. \tag{18}$$

Suppose now that the command-state pair at $t-1$ was feasible, i.e., $(v(t-1), x_s(t-1)) \in \tilde{O}_\infty^{c(t)}$. By definition of $\tilde{O}_\infty^{c(t)}$, this ensures that $C_s x_s(t) + (C_f\Gamma_f + D)v(t-1) + C_f c(t)E_x \subseteq Y$ but may not be sufficient to guarantee that $y(t) = C_s x_s(t) + (C_f\Gamma_f + D)v(t-1) + D(v(t) - v(t-1)) + C_f c(t)E_x \subseteq Y$ if $D \neq 0$. Hence, an extra condition,

$$C_s x_s(t) + (C_f\Gamma_f + D)v(t-1) + D(v(t) - v(t-1)) \in \tilde{Y}^{c(t)} \tag{19}$$

needs to be imposed on the selection of $v(t)$. Clearly, $v(t) = v(t-1)$ and $c(t+1) = c(t)$ is always a feasible choice.

Finally, the reduced order reference governor can be defined as the solution to the following optimization problem:

$$\text{Minimize } \|r(t) - v(t)\|^2 \text{ with respect to } v(t),\ c(t+1)$$

$$\text{subject to (18), (16), (19)}\ .$$

Note that this reduced order reference governor is of non-SRG type, and that a simplified implementation with constant $c(t) = c$ where $0 < c < 1$ is possible, which corresponds to the fixed error budget [15, 16].

From the computational standpoint, the constraint (16) appears to be more problematic, as it involves a potentially high dimensional set E_x and matrices A_f and Γ_f. However, note that $v(t)$ and $c(t+1)$ are low-dimensional and hence many constraints in (16) are redundant and can be eliminated offline, greatly simplifying (16).

Extensions of the reduced order reference governor to more flexible schemes such as *Extended Command Governors* [18] and the use of state observers is done in [15]. Extensions of the reduced order reference governor to the use of subsets $P^c \subset \tilde{O}_\infty^c$ and to nonlinear models remain open problems.

3.2 Reference governors based on overbounding the nonlinearity using logarithmic norms

The mechanism for determining the safety of a state and constant command pair ultimately relies on the prediction and bounding of the response of the system. The prediction of the response based on linear/linearized models is particularly easy but in the case of nonlinear systems additional bounding of the response deviation from that of the linear system is required. In the approach presented in [19], this bounding is performed using logarithmic norms. A logarithmic norm for an $n \times n$ matrix F induced by the conventional matrix norm

$\|\cdot\|$ is defined as

$$\mu(F) = \lim_{h \to 0^+} \frac{\|I_n + hF\| - 1}{h}, \quad F \in \mathbb{R}^{n \times n}.$$

For instance, if $\|\cdot\| = \|\cdot\|_2$ is the 2-norm, it can be shown that $\mu(F) = \lambda_{\mathtt{max}}\big(\frac{1}{2}(F + F^\top)\big)$ [20].

In [19], a class of nonlinear continuous-time constrained systems with bounded disturbances $w(t)$ is considered of the form,

$$\dot{x}(t) = f(x(t), v(t)) + w(t), \quad \|w(t)\| \leq w_{\mathtt{max}}, \tag{20}$$

with pointwise-in-time state constraints

$$x(t) \in X, \tag{21}$$

where X is a compact set. For this system, the linearized model at a given state-command pair $(x_v(\bar{v}), \bar{v})$ has the form,

$$\delta\dot{x}(t) = f_x(x_v(\bar{v}), \bar{v})\delta x(t) + f_v(x_v(\bar{v}), \bar{v})\delta v(t) + w(t), \tag{22}$$

where f_x and f_v denote the Jacobian matrices, and it is assumed that

$$\mu(f_x(x, \bar{v})) \leq \mu_e < 0 \text{ for all } x \in X. \tag{23}$$

The assumption (23) implies that the nominal closed-loop system possesses desirable contractivity characteristics between trajectories. Then the error, defined as,

$$e(t) = x(t) - \big(x_v(\bar{v}) + \delta x(t)\big), \tag{24}$$

between the response of (20) and (22) to a constant $\delta v(t)$, can be shown to satisfy

$$D_t^+ \|e(t)\| \triangleq \lim_{h \to 0^+} \frac{\|e(t+h)\| - \|e(t)\|}{h} \leq \mu_e \|e(t)\| + \eta_x \|\delta x(t)\| + \eta_v \|\delta v\|,$$

and be bounded [19] as

$$\|e(t)\| \leq \Gamma_v(\bar{v})\|\delta v\| + \Gamma_w(\bar{v}) w_{\mathtt{max}}, \text{ for all } t \geq 0, \tag{25}$$

where [19]:

$$\Gamma_v(\bar{v}) = \frac{\eta_x \|f_v(x_v(\bar{v}), \bar{v})\| - \eta_v \mu(f_x(x_v(\bar{v}), \bar{v}))}{\mu_e \mu(f_x(x_v(\bar{v}), \bar{v}))}, \quad \Gamma_w(\bar{v}) = \frac{\eta_x}{\mu_e \mu(f_x(x_v(\bar{v}), \bar{v}))},$$

and $\|f_x(x, \bar{v}) - f_x(x_v(\bar{v}), \bar{v})\| \leq \eta_x$, $\|f_v(x, v) - f_v(x_v(\bar{v}), \bar{v})\| \leq \eta_v$ for all $x \in X$. Exploiting the above bound, and under suitable assumptions, the reference governor operation reduces to solving a quadratic programming (QP) problem online. Furthermore, guarantees of recursive feasibility, constraint enforcement (without intersample constraint violation) and finite-time convergence to constant strictly steady-state constraint admissible references are available.

In a recent paper [21] the logarithmic norm bounding is employed for a class of continuous-time systems with unknown but bounded delays and with set-bounded disturbances. An explicit reference governor in the form of a discrete-time update law, that corresponds to one iteration of an optimization algorithm per discrete-time step, is considered. The results guarantee finite-time convergence to constant strictly steady-state constraint admissible references.

The technique of overbounding the nonlinear system response is quite general. In particular, opportunities to base the reference governor design on more general functional series expansions [22] remain to be exploited. In cases without disturbances such expansions reduce to a Taylor series in δv; the first term in this series can be computed by solving the standard sensitivity differential equations.

3.3 Controller State and Reference Governor

Another recent extension of the reference governor to a more flexible scheme is called the *Controller State and Reference Governor* or CSRG. This extension first appeared in [23].

CSRG is used for systems controlled by dynamic controllers, for example, controllers that incorporate integrators and have an integral action. In the case when a controller is dynamic, its state, denoted by x_c, is something one can modify or reset in order to help satisfy constraints. An example of such a system is (7)-(9) where $x_c = x_2$.

CSRG resets the controller state x_c along with the reference modification. For linear systems/models, CSRG operates by minimizing a quadratic function in a way that preserves desirable response properties and enlarges the constrained domain of attraction. More specifically, if

$$x = \begin{bmatrix} x_p \\ x_c \end{bmatrix}$$

is the closed-loop system state where x_p is the state of the plant and x_c is the state of the dynamic controller, CSRG operates by solving the following optimization problem online:

$$(v(t), x_c(t)) =$$

$$\arg \min_{v(t), x_c(t)} \Big\{ (v(t) - r(t))^\mathsf{T} S (v(t) - r(t)) + (x(t) - \bar{x}_{v(t)})^\mathsf{T} F (x(t) - \bar{x}_{v(t)}) \Big\},$$
$$\text{subject to } \big(v(t), x(t)\big) \in \tilde{O}_\infty,$$

where $\bar{x}_v$ denotes the steady-state corresponding to v, $S = S^\mathsf{T} \succ 0$ and the matrix $F = F^\mathsf{T} \succ 0$ satisfies the closed-loop Lyapunov equation.

An application of CSRG to aircraft gas turbine engines with constraints on compressor surge margins and fuel rate is considered in [23]. The fuel flow in such an engine is adjusted using a PI controller which tracks the fan speed set-point which is informed by the pilot PLA setting. CSRG is used to adjust the fan speed set-point and reset the state of the integrator of the PI Controller. It turns out that in this example CSRG very significantly extends the set of safely recoverable states, that is, initial states for which subsequent operation is possible without constraint violations. See Figure 8.

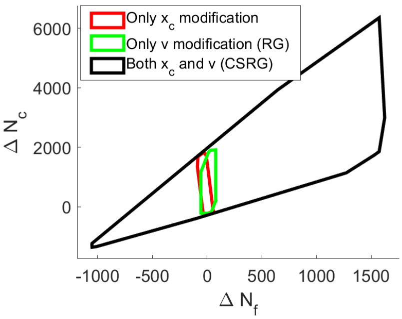

Fig. 8 The sets of recoverable deviations in fan speed and core speed in the gas turbine engines by CSRG versus alternatives based on linear model.

Extensions of CSRG to the use of subsets $P \subset \tilde{O}_\infty$ and to nonlinear models remain open problems.

3.4 Chance Constrained Reference Governor and Controller State and Reference Governor

Another recently developed scheme is called the *Chance Constrained Reference Governor* [24]. This reference governor scheme can handle stochastic disturbance inputs, $w(t)$. This scheme was motivated by applications to the flying wind turbine where there are disturbances due to wind and turbulence and constraints on angle of attack and tether tension. The chance constrained reference governor applies to system models with normally distributed i.i.d disturbance inputs $w(t) \sim \mathcal{N}(0, \Sigma_w)$:

$$\begin{aligned} x(t+1) &= Ax(t) + Bv(t) + B_w w(t), \\ y(t) &= Cx(t) + Dv(t) + D_w w(t) \in Y. \end{aligned}$$

The chance constrained reference governor enforces chance constraints on the system output imposed over the prediction horizon as

$$\beta \leq \texttt{Prob}\left[y(t+\tau) \in Y\right], \quad \tau = 0, 1, 2, \cdots,$$

where $y(t+\tau)$ denotes the predicted output at the time instant $t+\tau$ over the prediction horizon. Given that the model is linear, the predicted output is normally distributed at each time instant $t+\tau$ and its mean and variance can be easily predicted allowing to reformulate the chance constraint as a deterministic constraint. The predicted mean and variance can be used to construct a β-level confidence ellipsoid, which describes an ellipsoidal region within which the output $y(t+\tau)$ has a probability of β to realize. Then, the chance constraint can be enforced by enforcing this β-level confidence ellipsoid to be contained entirely in the safe set Y. The above confidence ellipsoid approach is the approach adopted in [24]. However, in [25] it has been shown that in most cases "(equal) risk allocation + inverse cumulative distribution function" is a better (same form while less conservative) approach than this confidence ellipsoid approach (see Fig. 2 of [25]). For the latter approach, one first applies Boole's inequality to separate the joint chance constraint into multiple individual chance constraints (for the case where $Y = \{y : \ Hy \leq h\}$ has multiple linear inequalities). Then, one uses the predicted mean and variance of $y(t+\tau)$ and the inverse of the cumulative distribution function of the standard normal distribution to re-express each individual chance constraint as a deterministic constraint.

Unfortunately, since disturbances do not have compact support, recursive feasibility even of the chance constraint cannot be guaranteed. The chance constrained reference governor uses a very simple mechanism for infeasibility handling by setting $\kappa(t) = 0$ that maintains the applied reference command constant and equal to the previous value. It is proven in [24] that under initial and strict steady-state feasibility assumptions, the chance constraint guarantee is maintained in closed loop with the infeasibility handling mechanism, i.e.,

$$\beta \leq \texttt{Prob}\left[y(t) \in Y\right] \text{ for all } t \geq 0.$$

Furthermore, the modified reference command is shown to converge to the desired constant strictly steady-state admissible reference command with probability one:

$$\texttt{Prob}\left[\lim_{t_f \to \infty}(v(t) = r_s \quad \forall t \geq t_f)\right] = 1.$$

It is also possible [25] to develop a *Chance Constrained Controller State and Reference Governor (CSRG)* which is able to not only modify the reference command but also reset the states of the dynamic controller, x_c. For instance, it can reset the integrator states if the controller uses integral action to help deal with the constraints. As in the chance constrained reference governor case, one complication caused by the possibility of the stochastic disturbance input taking arbitrary large values is that recursive feasibility cannot be guaranteed and so occasional infeasibility of CSRG optimization problem cannot be avoided. A simple infeasibility handling mechanism is added, which maintains the last reference command and propagates controller dynamics without reset in the event of infeasibility. For the closed-loop system with this infeasibility handling mechanism it is shown in [25] that the original (that is, the design-intent) chance constraints are satisfied. In addition, almost sure finite-time convergence of the modified reference to strictly steady-state constraint admissible constant references

$$\mathtt{Prob}\left[\exists t_f \in \mathbb{Z}_{\geq t_s} \text{ such that } v(t) = r_s \ \forall t \in \mathbb{Z}_{\geq t_f}\right] = 1$$

and mean square exponential boundedness for the closed-loop system state have been demonstrated as well. For a number of examples, for instance, of aircraft longitudinal and lateral dynamics with gusts we showed that the set of recoverable states from which CSRG is able to enforce the chance constraints is expanded by CSRG versus the conventional chance constrained reference governor.

3.5 Learning Reference Governor

Another approach for applying reference governors to uncertain systems involves the so called *Learning Reference Governor* or LRG. See Figure 9. For instance, a system may have changed due to damage, loading, or in-field modifications, e.g., a truck could be loaded by some bulky unknown load, and even if one had a model for the system originally, it becomes no longer accurate. What LRG does in this situation is that it experiments with the system applying different piecewise constant command profiles, sees how the system responds, and tries to learn the safe set based on the distance to constraint violation boundary. Three variants of LRG have been proposed: machine learning-based LRG which is analogous to our approach in Example 2, non-safety critical version of LRG that learns from constraint violations during experimentation and tightens an estimate of the safe set [12, 26], and a safety critical version [27, 28] which ensures safe learning, that is, no constraint violations occur during learning and after learning is completed.

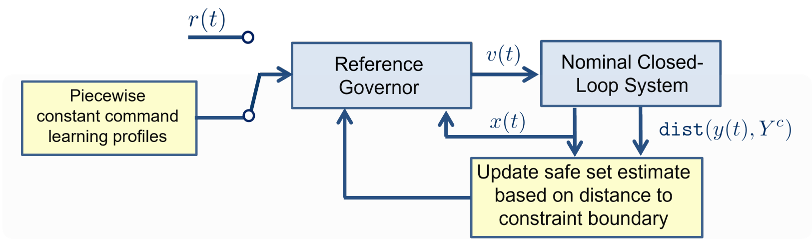

Fig. 9 Learning Reference Governor.

The safety-critical LRG is arguably the most interesting of these variants. To implement it, it suffices to measure two characteristics of system response to *step inputs*. The first one is the steady state distance to constraint boundary, denoted by $d(v)$. The second one is the maximum transient response deviation from steady-state, which is a function $D(v, \Delta v, \Delta x)$ of the nominal value of the reference command v, change in the reference command Δv, and the state deviation from steady-state at the beginning of the step, Δx. The dependence on Δx accounts for the initial state deviation from steady-state if the time for system to settle down between steps is insufficient. The definitions of these functions are illustrated in Figure 10.

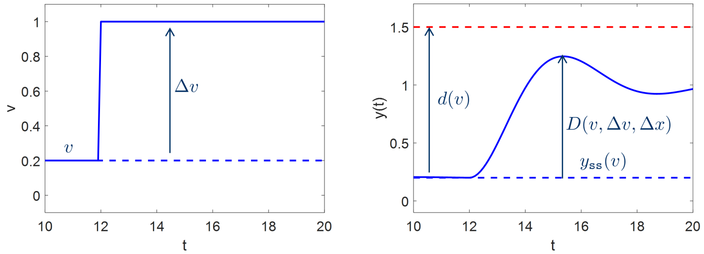

Fig. 10 Definitions of functions D and d.

Now, if the Lipschitz constant of the function D is known *a priori*, this knowledge can be combined with collected step response experimental measurement data to compute an upper bound $\bar{D}(v, \Delta v, \Delta x)$ on $D(v, \Delta v, \Delta x)$. Then the condition,

$$\bar{D}(v, \Delta v, \Delta x) \leq d(v),$$

defines a safe set which can be used for implementation of the reference governor. Specifically, the upper bound can be defined as

$$\bar{D}(v, \Delta v, \Delta x) = \min\left(\min_{i \in \mathcal{D}}\left(D_i + L\left\|\begin{bmatrix} v \\ \Delta v \\ \Delta x \end{bmatrix} - \begin{bmatrix} v_i \\ \Delta v_i \\ \Delta x_i \end{bmatrix}\right\|\right), L\left\|\begin{bmatrix} \Delta v \\ \Delta x \end{bmatrix}\right\|\right)$$

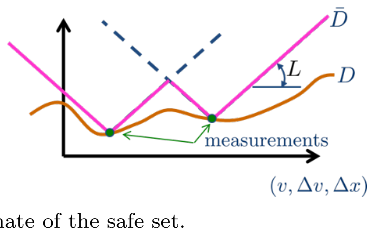

Fig. 11 Forming an estimate of the safe set.

Then the reference governor finds $v(t) = v(t-1) + \kappa(t)\big(r(t) - v(t-1)\big)$, where $\kappa(t) \in [0, 1]$ is maximized based on the condition:

$$\bar{D}\big(v(t-1), v(t) - v(t-1), x(t) - x_{ss}(v(t-1))\big) \leq d(v(t-1)).$$

In [27, 28], monotonic performance improvement and convergence properties of learning reference governor are established. It is shown that learning can be stopped at any time and the system can be deployed on the mission and is guaranteed to be safe. Furthermore, finite-time convergence of modified reference $v(t)$ to constant strictly steady-state admissible commands, r, is ensured. Of course, more prolonged learning phases can ensure that the operation of the reference governor is less conservative.

We have applied LRG to a variety of case studies, one of which is the rollover avoidance for a truck [28]. In this case, the command for the truck is the steering wheel angle and the constraints are imposed on the load transfer ratio to avoid vehicle rollover. Note that learning can be performed in physical world on an actual vehicle or in virtual world on high fidelity models if such models are available and quickly reconfigurable. Such a learning reference governor can be integrated into an autonomous motion planning and control algorithm as in Figure 12. In [29], the learning reference governor is applied to the spacecraft autonomous rendezvous, proximity operation and docking maneuvers.

Fig. 12 Reference Governor as a part of autonomous vehicle control system.

The above results are based on model simulations, but we have also tested such schemes experimentally [30]. The non-safety critical version of learning reference governor proposed in [12] has been tested experimentally on a running internal combustion engine. This non-safety critical learning reference governor can violate constraints during learning and it learns from such constraint violations. After a sufficiently informative learning phase the learning reference governor is able to avoid engine misfiring during engine decelerations that could be caused by high exhaust gas recirculation. The learning reference governor learns to slow down throttle closure in such a way as to leave enough air in intake manifold for combustion and avoid engine misfire.

4 Concluding Remarks

The reference governor is an add-on supervisor to a nominal closed-loop system which modifies the reference command in order to enforce the constraints. In this paper we reviewed and commented on the basic reference governor ideas and on several more recent extensions of the reference governor schemes. We have also mentioned several open problems.

Acknowledgments

The authors acknowledge their many collaborators, who are coauthors on joint publications, and, in particular, contributions of Dr. Uroš Kalabić to topics discussed.

Funding

This research is supported by the National Science Foundation grant number CMMI-1904394 and by the Air Force Office of Scientific Research under grant FA9550-20-1-0385.

Compliance with Ethical Standards

Declarations

Conflict of Interest: The authors have declared that no confict of interest exists.

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] Jaw, L., Mattingly, J.: Aircraft Engine Controls. American Institute of Aeronautics and Astronautics, Reston, VA (2009)

[2] Rawlings, J.B., Mayne, D.Q., Diehl, M.: Model Predictive Control: Theory, Computation, and Design. Nob Hill Publishing, Madison, Wisconsin (2017)

[3] Garone, E., Di Cairano, S., Kolmanovsky, I.: Reference and command governors for systems with constraints: A survey on theory and applications. Automatica **75**, 306–328 (2017)

[4] Gilbert, E.G., Tan, K.T.: Linear systems with state and control constraints: The theory and application of maximal output admissible sets. IEEE Transactions on Automatic control **36**(9), 1008–1020 (1991)

[5] Gilbert, E.G., Kolmanovsky, I.: Fast reference governors for systems with state and control constraints and disturbance inputs. International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal **9**(15), 1117–1141 (1999)

[6] Kolmanovsky, I., Gilbert, E.G.: Theory and computation of disturbance invariant sets for discrete-time linear systems. Mathematical Problems in Engineering **4**(4), 317–367 (1998)

[7] Gilbert, E., Kolmanovsky, I.: Nonlinear tracking control in the presence of state and control constraints: A generalized reference governor. Automatica **38**(12), 2063–2073 (2002)

[8] Miller, R.H., Kolmanovsky, I., Gilbert, E.G., Washabaugh, P.D.: Control of constrained nonlinear systems: A case study. IEEE Control Systems Magazine **20**(1), 23–32 (2000)

[9] Bemporad, A.: Reference governor for constrained nonlinear systems. IEEE Transactions on Automatic Control **43**(3), 415–419 (1998)

[10] Garone, E., Nicotra, M.M.: Explicit reference governor for constrained nonlinear systems. IEEE Transactions on Automatic Control **61**(5), 1379–1384 (2015)

[11] Nicotra, M.M., Garone, E.: The explicit reference governor: A general framework for the closed-form control of constrained nonlinear systems. IEEE Control Systems Magazine **38**(4), 89–107 (2018)

[12] Liu, K., Li, N., Rizzo, D., Garone, E., Kolmanovsky, I., Girard, A.: Model-free learning to avoid constraint violations: An explicit reference governor approach. In: 2019 American Control Conference (ACC), pp. 934–940 (2019). IEEE

[13] Meerkov, S.: Principle of vibrational control: Theory and applications. IEEE Transactions on Automatic Control **25**(4), 755–762 (1980)

[14] Garone, E., Tedesco, F., Casavola, A.: Sensorless supervision of linear dynamical systems: The feed-forward command governor approach. Automatica **47**(7), 1294–1303 (2011)

[15] Kalabić, U.V., Kolmanovsky, I.V., Gilbert, E.G.: Reduced order extended command governor. Automatica **50**(5), 1466–1472 (2014)

[16] Kalabić, U., Kolmanovsky, I., Buckland, J., Gilbert, E.: Reduced order reference governor. In: 51st Conference on Decision and Control (CDC), pp. 3245–3251 (2012). IEEE

[17] Di Cairano, S., Kalabić, U.V., Kolmanovsky, I.V.: Reference governor for network control systems subject to variable time-delay. Automatica **62**, 77–86 (2015)

[18] Gilbert, E.G., Ong, C.-J.: Constrained linear systems with hard constraints and disturbances: An extended command governor with large domain of attraction. Automatica **47**(2), 334–340 (2011)

[19] Li, N., Kolmanovsky, I.V., Girard, A.: A reference governor for nonlinear systems with disturbance inputs based on logarithmic norms and quadratic programming. IEEE Transactions on Automatic Control **65**(7), 3207–3214 (2019)

[20] Afanasiev, V.N., Kolmanovskii, V., Nosov, V.R.: Mathematical Theory of Control Systems Design vol. 341. Springer, Dordrecht (2013)

[21] Li, N., Geng, S., Kolmanovsky, I.V., Girard, A.: An explicit reference governor for linear sampled-data systems with disturbance inputs and uncertain time delays. IEEE Transactions on Automatic Control (2022)

[22] Gilbert, E.: Functional expansions for the response of nonlinear differential systems. IEEE Transactions on Automatic Control **22**(6), 909–921 (1977)

[23] McDonough, K., Kolmanovsky, I.: Controller state and reference governors for discrete-time linear systems with pointwise-in-time state and control constraints. In: 2015 American Control Conference (ACC), pp. 3607–3612 (2015). IEEE

[24] Kalabić, U.V., Li, N.I., Vermillion, C., Kolmanovsky, I.V.: Reference governors for chance-constrained systems. Automatica **109**, 108500 (2019)

[25] Li, N., Girard, A., Kolmanovsky, I.: Chance-constrained controller state and reference governor. Automatica **133**, 109864 (2021)

[26] Higgins, M.R., Burlion, L.: Implementation of a learning-based explicit reference governor for constrained control of a UAV. In: AIAA SCITECH 2022 Forum, p. 2042 (2022)

[27] Liu, K., Li, N., Kolmanovsky, I., Rizzo, D., Girard, A.: Model-free learning for safety-critical control systems: A reference governor approach. In: 2020 American Control Conference (ACC), pp. 943–949 (2020). IEEE

[28] Liu, K., Li, N., Kolmanovsky, I., Rizzo, D., Girard, A.: Safe learning reference governor: Theory and application to fuel truck rollover avoidance. Journal of Autonomous Vehicles and Systems, 1–19 (2021)

[29] Ikeya, K., Liu, K., Girard, A., Kolmanovsky, I.V.: Learning to satisfy constraints in spacecraft rendezvous and proximity maneuvering: A learning reference governor approach. In: AIAA SCITECH 2022 Forum, p. 2514 (2022)

[30] Maldonado, B.P., Li, N., Kolmanovsky, I., Stefanopoulou, A.G.: Learning reference governor for cycle-to-cycle combustion control with misfire

avoidance in spark-ignition engines at high exhaust gas recirculation–diluted conditions. International Journal of Engine Research **21**(10), 1819–1834 (2020)