

# Stabilizing Model Predictive Control of Stochastic Constrained Linear Systems

Daniele Bernardini, *Member, IEEE*, and Alberto Bemporad, *Fellow, IEEE*

**Abstract**—This paper investigates stochastic stabilization procedures based on quadratic and piecewise linear Lyapunov functions for discrete-time linear systems affected by multiplicative disturbances and subject to linear constraints on inputs and states. A stochastic model predictive control (SMPC) design approach is proposed to optimize closed-loop performance while enforcing constraints. Conditions for stochastic convergence and robust constraints fulfillment of the closed-loop system are enforced by solving linear matrix inequality problems off line. Performance is optimized on line using multistage stochastic optimization based on enumeration of scenarios, that amounts to solving a quadratic program subject to either quadratic or linear constraints. In the latter case, an explicit form is computable to ease the implementation of the proposed SMPC law. The approach can deal with a very general class of stochastic disturbance processes with discrete probability distribution. The effectiveness of the proposed SMPC formulation is shown on a numerical example and compared to traditional MPC schemes.

**Index Terms**—Constrained linear systems, model predictive control (MPC), stochastic control.

## I. INTRODUCTION

MODEL predictive control (MPC) is a popular strategy which has been widely adopted in industry as an effective mean of dealing with multivariable constrained control problems [1]. The idea behind MPC is to obtain the control signal by solving at each sampling time an open-loop finite-horizon optimal control problem based on a given prediction model of the process, by taking the current state of the process as the initial state. The control inputs are implemented in accordance with a receding horizon scheme.

Classical MPC formulations do not provide a systematic way to deal with model uncertainties and disturbances. Many predictive control schemes have been proposed to guarantee stability and constraint fulfillment in the presence of disturbances. Most works are based on the min-max approach, where the performance index to be minimized is computed over the worst possible disturbance realization [2]–[6]. However, min-max policies are often computationally demanding, and the resulting

control law may be too conservative, as no statistical properties about the disturbance are taken into account.

A different approach is addressed by stochastic MPC (SMPC), where convergence in probability and expected values of constraints and/or performance indices are considered (see, e.g., [7]–[10] and references therein). A common assumption when facing uncertainty with values on a continuous domain is to model the disturbance signal as a Gaussian noise, with given mean and covariance matrix [11]–[13]. Although such an assumption facilitates analytical computations of statistical properties and the statement of fundamental theoretical results, it can be restrictive in many practical applications, in which process uncertainty is better described by more general, even time-varying, statistical properties that are not satisfactorily captured by normal and time-invariant distributions.

When dealing with optimization problems in the presence of stochastic data, the approximation of continuous uncertainty to a discrete domain is often used, and constructed in a way to preserve the main properties of the underlying continuous process [14]–[16]. In this framework, the control problem formulation involves the setup of a scenario-based optimization tree, where only the most relevant disturbance patterns are modeled.

In this paper, we propose a SMPC formulation based on scenario generation for linear systems with discrete multiplicative disturbances. Our main goal is to obtain a less conservative control action with respect to standard robust MPC [2], [3] by restricting ourselves to consider stochastic convergence to the origin and a stochastic performance index. By virtue of the separation between stabilization and feasibility issues on one side, and stochastic performance optimization on the other, the proposed control scheme deals with an extremely general class of discrete stochastic disturbance processes. Stability and feasibility properties do not depend on the stochastic process model that generates the scenarios. Such a scenario generation engine can therefore change from application to application, e.g., from human driver's power and velocity demand learnt by a cognitive system in automotive power management and adaptive cruise control [17], [18], to network-induced uncertainty models in networked control systems [19], to a stock price model with stochastic volatility in option hedging [20].

The paper is organized as follows. The class of stochastic dynamic models dealt with in the paper is described in Section II. In Section III, two SMPC schemes based on quadratically constrained quadratic programming are proposed, for both the unconstrained and the constrained case. Section IV presents a procedure to recast the SMPC problem as a standard quadratic program, for which an explicit form is derived. Results of simulation tests are reported in Section V, and

Manuscript received October 29, 2010; revised May 25, 2011; accepted October 25, 2011. Date of publication November 16, 2011; date of current version May 23, 2012. This work was supported in part by the European project E-PRICE: Price-based Control of Electrical Power Systems, FP7-IST contract 249096. Recommended by Associate Editor P. Shi.

The authors are with the IMT Institute for Advanced Studies Lucca, 55100 Lucca, Italy (e-mail: daniele.bernardini@imtlucca.it; alberto.bemporad@imtlucca.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2011.2176429

conclusions are drawn in Section VI. A preliminary version of this paper was presented in [21].

*Notation:* In this manuscript,  $I_n$  is the  $n \times n$  identity matrix and  $^T$  denotes transposition. By  $\mathbf{0}$  and  $\mathbf{1}$  we denote column vectors or matrices of appropriate dimensions with, respectively, all zero and one entries. The minimum and maximum eigenvalues of a matrix  $A$  are denoted by  $\lambda_{\min}(A)$  and  $\lambda_{\max}(A)$ , respectively, and the expression  $A \succ 0$  ( $A \succeq 0$ ) is used to indicate that  $A$  is (semi)definite positive. Given a set  $\mathcal{A}$ ,  $\text{co}\{\mathcal{A}\}$  denotes the convex hull of  $\mathcal{A}$ . If  $\mathcal{A}$  is a polytope,  $\text{vert}\{\mathcal{A}\}$  is the set of the vertices of  $\mathcal{A}$ . Let  $\{w(k)\}_{k=0}^{\infty}$  be a sequence of random variables. We denote by  $\mathbb{E}_{[k_1, k_2]}(\cdot)$  the conditional expectation taken with respect to  $\{w(k_1), w(k_1+1), \dots, w(k_2)\}$ , given the sequence  $\{w(0), w(1), \dots, w(k_1-1)\}$ , where  $k_1, k_2 \in \mathbb{N}$  and  $k_2 \geq k_1$ . We also write  $\mathbb{E}_k(\cdot)$  for  $\mathbb{E}_{[k, k]}(\cdot)$  and use the short form  $\mathbb{E}(\cdot)$  to denote  $\mathbb{E}_{[0, +\infty)}(\cdot)$ .

## II. MODEL DESCRIPTION

Consider the discrete-time linear system

$$x(k+1) = A(w(k))x(k) + B(w(k))u(k) \quad (1)$$

where  $k \in \mathbb{N}$  is the time index,  $x(k) \in \mathbb{R}^{n_x}$  is the state,  $u(k) \in \mathbb{R}^{n_u}$  is the input,  $w(k) \in \mathcal{W}$  is the disturbance, and  $\mathcal{W} = \{\bar{w}_1, \bar{w}_2, \dots, \bar{w}_s\} \subset \mathbb{R}$  is a finite set (we consider scalar values for simplicity, the same ideas can be applied if  $w(k)$  has more components). By enumerating all the  $s$  possible realizations of  $w(k)$ , system (1) can be rewritten as

$$x(k+1) = \begin{cases} A_1x(k) + B_1u(k), & \text{if } w(k) = \bar{w}_1 \\ A_2x(k) + B_2u(k), & \text{if } w(k) = \bar{w}_2 \\ \vdots & \vdots \\ A_sx(k) + B_su(k), & \text{if } w(k) = \bar{w}_s \end{cases} \quad (2)$$

where  $(A_i, B_i)$ ,  $i = 1, 2, \dots, s$ , are given matrices of appropriate dimensions. To model the evolution of the disturbance  $w(k)$ , we introduce the (possibly time-varying) probability distribution

$$p(k) = [p_1(k), p_2(k), \dots, p_s(k)]^T$$

where  $p_j(k) = \Pr[w(k) = \bar{w}_j]$ , for all  $j \in \{1, 2, \dots, s\}$ , with  $p(k) \in \mathcal{D}$ , for all  $k \in \mathbb{N}$ , and  $\mathcal{D} = \{p \mid \mathbf{1}^T p = 1, p \geq \mathbf{0}\}$  is the standard probability simplex. In order to characterize the available information on  $p(k)$  we make the following assumptions.

*Assumption 1:* The probability distribution  $p(k) \in \mathcal{P}$  for all  $k \in \mathbb{N}$ , where  $\mathcal{P} \subseteq \mathcal{D}$  is a known polytope with vertices  $v^1, v^2, \dots, v^m \in \mathbb{R}^s$ , and  $v^j = [v_1^j, v_2^j, \dots, v_s^j]^T$ .

*Assumption 2:* The probability distribution  $p(k)$  is observable at each time step  $k \in \mathbb{N}$ .

In addition, a (possibly time-varying) model of the time evolution of  $p(k)$  may be available. Such a model will be exploited in the presented approach to increase the accuracy of the predicted system trajectory, hence improving performance. Moreover, note that Assumption 1 can always be satisfied by setting  $\mathcal{P} = \mathcal{D}$ , and this can either be obtained from the model of  $p(k)$ , or when no model of  $p(k)$  exists. However, as shown

in Section III, the size of  $\mathcal{P}$  affects the conservativeness of the control action, so any information that restricts  $\mathcal{P}$  is useful to speed up convergence of the closed-loop system.

Hereafter, we briefly review a few popular stochastic models that are suitable to be used, within the presented framework, as dynamics models describing the disturbance  $w(k)$  and the distribution  $p(k)$ .

### A. Stochastic Volatility

Stochastic volatility models are models where the variance of the considered random variable is driven by a stochastic process. An example is given by generalized autoregressive conditional heteroskedasticity (GARCH) models, in which the evolution of the variance is described by an autoregressive moving average (ARMA) model. Stochastic volatility models are commonly employed in econometrics, to model financial time-series. The quantized disturbance  $w(k) \in \mathcal{W}$  can be seen as a piecewise approximation of a random variable  $z(k) \in \mathbb{R}$ , which is described by the time-varying probability density function (pdf)  $f_z(k) = \mathcal{N}(0, \sigma^2(k))$  (other pdfs can be also considered). By partitioning the real domain in  $s$  cells  $C_1, C_2, \dots, C_s$ , we compute the probability distribution  $p(k)$  at time  $k$  as  $p_i(k) = \int_{C_i} f_z(k) dz$ , and the values  $\bar{w}_i = \int_{C_i} z f_z dz$ , for all  $i \in \{1, 2, \dots, s\}$ . Assuming that the evolution of the variance  $\sigma^2(k)$  is modeled by

$$\sigma^2(k) = \alpha_0 + \sum_{i=1}^q \alpha_i w^2(k-i) + \sum_{j=1}^p \beta_j \sigma^2(k-j) \quad (3)$$

where  $\alpha_0 > 0$ ,  $\alpha_i \geq 0$ ,  $\beta_j \geq 0$ , for all  $i \in \{1, 2, \dots, q\}$  and  $j \in \{1, 2, \dots, p\}$ , are given parameters, we can derive (a conservative approximation of) the bounding set  $\mathcal{P}$  from the standard deviation bounds  $\underline{\sigma}, \bar{\sigma}$  such that  $\sigma(k) \in [\underline{\sigma}, \bar{\sigma}]$ , for all  $k \in \mathbb{N}$ . We define

$$\mathcal{P} = \{p \in \mathbb{R}^s \mid \mathbf{1}^T p = 1, p^- \leq p \leq p^+\}$$

where  $p^- = [p_1^-, p_2^-, \dots, p_s^-]$ ,  $p^+ = [p_1^+, p_2^+, \dots, p_s^+]$ , and

$$p_i^- = \inf_{\sigma \in [\underline{\sigma}, \bar{\sigma}]} \frac{1}{\sigma \sqrt{2\pi}} \int_{C_i} e^{-z^2/2\sigma^2} dz$$

$$p_i^+ = \sup_{\sigma \in [\underline{\sigma}, \bar{\sigma}]} \frac{1}{\sigma \sqrt{2\pi}} \int_{C_i} e^{-z^2/2\sigma^2} dz$$

for all  $i \in \{1, 2, \dots, s\}$ . An example of the values of  $p^+$  and  $p^-$  for a specific choice of the parameters is shown in Fig. 1. In the case of  $p = q = 1$  it is easy to show that  $\underline{\sigma} = \alpha_0^{1/2}$  and  $\bar{\sigma} = ((\alpha_0 + \alpha_1 \bar{w})(1 - \beta_1)^{-1})^{1/2}$ , where  $\bar{w} = \sup_{w \in \mathcal{W}} w^2$ , provided that  $\beta_1 < 1$ .

### B. Markov Chains

Markov chains are used in a wide area of applications, such as physics, statistics, biology, and economics. A distinctive characteristic of Markov chains is that the next state depends only on the current state, and not on the history of transitions that lead to the current state. They are defined by a discrete set of

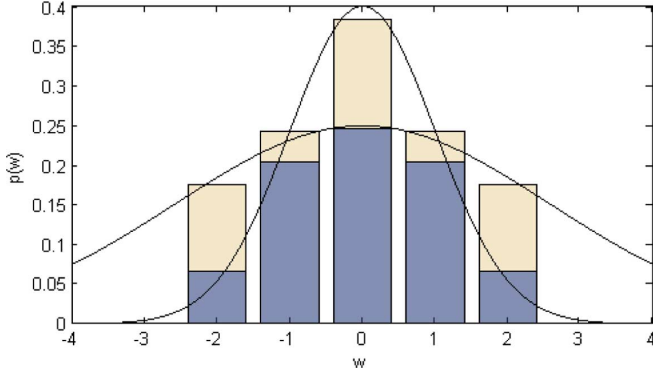


Fig. 1. Values of  $p^-$  (blue, lower level) and  $p^+$  (yellow, upper level) obtained for  $\mathcal{W} = \{-2, -1, 0, 1, 2\}$  and  $f_z(k) = \mathcal{N}(0, \sigma^2(k))$ , where  $\sigma^2(k)$  is modeled by (3) with  $p = q = 1$  and parameters  $\alpha_0 = 1$ ,  $\alpha_1 = 0.2$ ,  $\beta_1 = 0.3$ . Plots of  $\mathcal{N}(0, \underline{\sigma}^2)$  and  $\mathcal{N}(0, \bar{\sigma}^2)$  are also reported.

$n_M$  states  $\{\bar{z}_1, \bar{z}_2, \dots, \bar{z}_{n_M}\}$ , a discrete set of  $s$  outputs or emissions  $\{\bar{w}_1, \bar{w}_2, \dots, \bar{w}_s\}$ , a transition probability matrix  $T \in \mathbb{R}^{n_M \times n_M}$ , and an emission matrix  $E \in \mathbb{R}^{n_M \times s}$ , such that

$$t_{ij} = \Pr[z(k+1) = \bar{z}_j \mid z(k) = \bar{z}_i] \quad (4a)$$

$$e_{ij} = \Pr[w(k) = \bar{w}_j \mid z(k) = \bar{z}_i] \quad (4b)$$

where  $z(k)$  and  $w(k)$  are the state and the output of the Markov chain at time  $k$ , and  $t_{ij}$  and  $e_{ij}$  are the elements of  $T$  and  $E$ , respectively. If  $p(k)$  is driven by a Markov chain, we define  $\mathcal{P} = \text{co}\{E_1^T, E_2^T, \dots, E_s^T\}$ , where  $E_i$  is the  $i$ th row of  $E$ . In this case, system (2) belongs to the well studied class of Markov jump linear systems [22], [23].

### III. STOCHASTIC MPC DESIGN

Consider the problem of regulating the state  $x$  of system (1) to the origin. Our goal is to design a stochastic MPC control scheme which solves this problem by exploiting the available information on the disturbance. Stochastic control is intended here with respect to both the performance index to be minimized, and the type of stability considered. We aim at enforcing exponential stability in the mean-square sense, in accordance with the following definition.

*Definition 1:* System (1) is said *Uniformly Globally Mean-Square Exponentially Stable* (UGMSES) if there exist  $c \geq 0$  and  $\lambda \in [0, 1)$  such that for any initial condition  $x(0) \in \mathbb{R}^{n_x}$  it holds that

$$\mathbb{E}(x(k)^T x(k)) \leq c\lambda^k x(0)^T x(0), \quad \forall k \in \mathbb{N}. \quad (5)$$

Moreover, system (1) is said *Uniformly Locally Mean-Square Exponentially Stable* (ULMSES) in  $\mathcal{X}_0$  if (5) holds for any initial condition  $x(0) \in \mathcal{X}_0 \subseteq \mathbb{R}^{n_x}$ .

*Lemma 1:* Consider the stochastic autonomous discrete-time linear system

$$x(k+1) = \tilde{A}(w(k))x(k) \quad (6)$$

where the disturbance  $w(k)$ , for all  $k \in \mathbb{N}$ , is an independent and identically distributed (i.i.d.) random process. Define the

function  $V : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_+$  as  $V(x) = x^T P x$ , with  $P = P^T \succ 0$ , and assume a matrix  $L = L^T \succ 0$  exists such that

$$\mathbb{E}_k(V(x(k+1))) - V(x(k)) \leq -x(k)^T L x(k), \quad \forall k \in \mathbb{N}. \quad (7)$$

Then system (6) is UGMSES.

*Proof:* The proof follows in spirit the proof of [24, Lemma 1]. Since  $x(k)$  is independent of  $w(k)$  and  $\tilde{A}(w(k))$  is independent of  $\{w(0), w(1), \dots, w(k-1)\}$ , for all  $k \in \mathbb{N}$ , by (7) we have

$$\begin{aligned} \mathbb{E}(V(x(k+1))) &= \mathbb{E}(V(x(k))) + \mathbb{E}(V(x(k+1)) \\ &\quad - V(x(k))) \\ &= \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &\quad + \mathbb{E}_{[0, k]}(V(x(k+1)) - V(x(k))) \\ &= \mathbb{E}_{[0, k-1]}(V(x(k))) + \mathbb{E}_{[0, k-1]} \\ &\quad (\mathbb{E}_k(V(x(k+1))) - V(x(k))) \\ &\leq \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &\quad + \mathbb{E}_{[0, k-1]}(-x(k)^T L x(k)) \\ &\leq \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &\quad - \lambda_{\min}(L) \mathbb{E}_{[0, k-1]}(x(k)^T x(k)) \\ &\leq \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &\quad - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)} \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &= \left(1 - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)}\right) \mathbb{E}_{[0, k-1]}(V(x(k))) \\ &= \left(1 - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)}\right) \mathbb{E}(V(x(k))). \end{aligned}$$

By induction, this leads to

$$\mathbb{E}(V(x(k+1))) \leq \left(1 - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)}\right)^{k+1} V(x(0)) \quad (8)$$

and hence

$$\begin{aligned} \mathbb{E}(x(k)^T x(k)) &\leq \frac{1}{\lambda_{\min}(P)} \mathbb{E}(V(x(k))) \\ &\leq \frac{1}{\lambda_{\min}(P)} \left(1 - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)}\right)^k V(x(0)) \\ &\leq \frac{\lambda_{\max}(P)}{\lambda_{\min}(P)} \left(1 - \frac{\lambda_{\min}(L)}{\lambda_{\max}(P)}\right)^k x(0)^T x(0). \end{aligned}$$

Define  $\lambda = 1 - \lambda_{\min}(L)/\lambda_{\max}(P)$ . Clearly  $\lambda \geq 0$ , as  $\mathbb{E}(V(x(k+1)))$  is always a nonnegative quantity and therefore (8) imposes  $\lambda^k$  to be nonnegative, and  $\lambda < 1$  because the conditions  $L \succ 0$ ,  $P \succ 0$  imply  $\lambda_{\min}(L)/\lambda_{\max}(P) > 0$ . Hence, by also setting  $c = \lambda_{\max}(P)/\lambda_{\min}(P) > 0$ , property (5) is satisfied.  $\blacksquare$

The presence or the absence of state and/or input constraints poses different issues. These two cases, which are handled separately in the rest of this section, share the same methodological approach: Off line, a Lyapunov function and a linear feedback control law which provide mean-square stability are obtained by solving a semidefinite programming (SDP) problem. On line,

a stochastic MPC controller based on scenario enumeration is applied to optimize a given objective function by exploiting the current measurements and the stochastic information on the disturbance, using the previously computed Lyapunov function to retain stability.

### A. The Unconstrained Case

Our first goal is to compute a Lyapunov function and a control law which provide mean-square stability of the controlled system in the absence of constraints. We restrict ourselves to consider quadratic Lyapunov candidates  $V(x) = x^T P x$ , and impose a linear feedback structure on the input, i.e.,  $u(k) = Kx(k)$ ,  $\forall k \in \mathbb{N}$ . Although such constant control law will not be used to compute the actual input signal, the existence of the matrix  $K$  will serve to prove the recursive feasibility of the receding horizon policy that will be presented later.

1) *Lyapunov Function Synthesis*: We begin considering an instance of the mentioned regulation problem where the probability distribution is constant over time, and then extend the result to time-varying distributions.

*Lemma 2*: Consider a process modeled by system (1), where the disturbance  $w(k) \in \mathcal{W}$  is described by an i.i.d. random process with constant probability distribution  $p(k) = \bar{p}$ , for all  $k \in \mathbb{N}$ . Let  $Q \in \mathbb{R}^{n_x \times n_x}$ ,  $W \in \mathbb{R}^{n_x \times n_x}$ ,  $Y \in \mathbb{R}^{n_u \times n_x}$  be a solution of the LMI

$$\begin{bmatrix} Q & Q & M^T \\ Q & W & \mathbf{0} \\ M & \mathbf{0} & \Delta \end{bmatrix} \succeq 0 \quad (9)$$

where  $Q = Q^T \succ 0$ ,  $W = W^T \succ 0$ ,  $\Delta = \text{diag}\{Q, Q, \dots, Q\}$ ,  
 $\underbrace{\hspace{10em}}_{s \text{ times}}$

and

$$M = \begin{bmatrix} \sqrt{\bar{p}_1}(A_1 Q + B_1 Y) \\ \sqrt{\bar{p}_2}(A_2 Q + B_2 Y) \\ \vdots \\ \sqrt{\bar{p}_s}(A_s Q + B_s Y) \end{bmatrix}.$$

Then, system (1) in closed loop with  $u(k) = Kx(k)$  for all  $k \in \mathbb{N}$ , with  $K = YQ^{-1}$ , is UGMSES.

*Proof*: Letting  $V(x) = x^T P x$ ,  $P = P^T \succ 0$ , we have

$$\mathbb{E}_k(V(x(k+1))) = \sum_{j=1}^s p_j(k) x(k+1 | k, j)^T P x(k+1 | k, j) \quad (10)$$

where  $x(k+1 | k, j) = A_j x(k) + B_j u(k)$ ,  $j \in \{1, 2, \dots, s\}$ . By using (10), substituting  $P = Q^{-1}$ ,  $L = W^{-1}$ ,  $p(k) = \bar{p}$ ,  $u(k) = YQ^{-1}x(k)$  in (7), and applying Schur complements, we have that (7) is equivalent to (9). Hence, if (9) is feasible, then by Lemma 1 the constant feedback control law  $u(k) = YQ^{-1}x(k)$  makes the closed-loop system UGMSES. ■

In the following, we will denote the LMI (9) as  $\mathcal{L}(\bar{p}) \succeq 0$ , stressing the dependance of the solution on the value of  $\bar{p}$ . The result of Lemma 2 can be easily extended to the case in which the probability distribution  $p(k)$  varies over time, as shown in the following lemma.

*Lemma 3*: Consider a process modeled by system (1), where the disturbance  $w(k) \in \mathcal{W}$  is described by a i.i.d. random process with probability distribution  $p(k) \in \mathcal{P}$ , for all  $k \in \mathbb{N}$ , such that Assumption 1 is satisfied. Let  $v^1, v^2, \dots, v^m$  be the vertices of  $\mathcal{P}$ , and let  $Q \in \mathbb{R}^{n_x \times n_x}$ ,  $W \in \mathbb{R}^{n_x \times n_x}$ ,  $Y \in \mathbb{R}^{n_u \times n_x}$  be a solution of the SDP problem

$$\min_{Q, W, Y} \text{trace}(W) \quad (11a)$$

$$\text{s.t. } \text{trace}(Q) = 1 \quad (11b)$$

$$W - W_0 \succeq 0 \quad (11c)$$

$$\text{diag}\{\mathcal{L}(v^1), \mathcal{L}(v^2), \dots, \mathcal{L}(v^m)\} \succeq 0 \quad (11d)$$

where  $Q = Q^T \succ 0$ ,  $W = W^T \succ 0$  and  $W_0 \succ 0$ . Then, system (1) in closed loop with  $u(k) = Kx(k)$  for all  $k \in \mathbb{N}$ , with  $K = YQ^{-1}$ , is UGMSES.

*Proof*: Easily follows by Lemma 2, noting that if  $(Q, W, Y)$  is a solution of (11), then it is also a solution of  $\mathcal{L}(p(k))$ , for any  $k \in \mathbb{N}$ , and thus it satisfies (5). To prove this, see that for any  $p \in \mathcal{P}$  there exists a vector  $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_m]^T$  such that  $\alpha \geq \mathbf{0}$ ,  $\mathbf{1}^T \alpha = 1$ , and  $p = \sum_{i=1}^m \alpha_i v^i$ . Since  $(Q, W, Y)$  satisfies (11d), we have

$$\sum_{j=1}^s v_j^i (A_j + B_j K)^T P (A_j + B_j K) \preceq P - L \quad (12)$$

for all  $i \in \{1, 2, \dots, m\}$ , where  $P = Q^{-1}$ ,  $L = W^{-1}$ ,  $K = YQ^{-1}$ . For  $(Q, W, Y)$  to be a solution of  $\mathcal{L}(p) \succeq 0$ ,  $p \in \mathcal{P}$ , it must satisfy

$$\sum_{j=1}^s p_j (A_j + B_j K)^T P (A_j + B_j K) \preceq P - L$$

or, equivalently,

$$\sum_{i=1}^m \alpha_i \sum_{j=1}^s v_j^i (A_j + B_j K)^T P (A_j + B_j K) \preceq P - L$$

which is satisfied as  $\mathbf{1}^T \alpha = 1$  and because of (12). ■

Constraint (11c) imposes a bound on the decay rate of  $V(x)$ : instead of looking for the fastest mean-square convergence to the origin, we aim at obtaining a Lyapunov function with satisfactory decay rate; the available extra margin will be used on line to optimize with respect to a given objective function.

*Remark 1*: It can be easily checked that, if  $\mathcal{P} = \mathcal{D}$ , problem (11) is a *robust* control problem (e.g., is analogous to what is proposed in [2]), and its solution enforces robust convergence to the origin for all the possible sequences of  $[A(w(k)) \ B(w(k))] \in \text{co}\{[A_1 \ B_1], [A_2 \ B_2], \dots, [A_s \ B_s]\}$ . In other words, in this framework robust stability can be seen a special case of stochastic stability, where no *a priori* information on probability bounds is exploited. On the other hand, when  $p(k) = \bar{p}$  is constant over time (as in Lemma 2), we have  $\mathcal{P} = \{\bar{p}\}$ , i.e., constraint (11d) is imposed for a single distribution  $p(k)$ , thus providing a non-conservative solution which enforces stochastic convergence only for the particular distribution considered. Hence, we see that the size of  $\mathcal{P}$  affects the conservativeness of the resulting control law, which ranges from

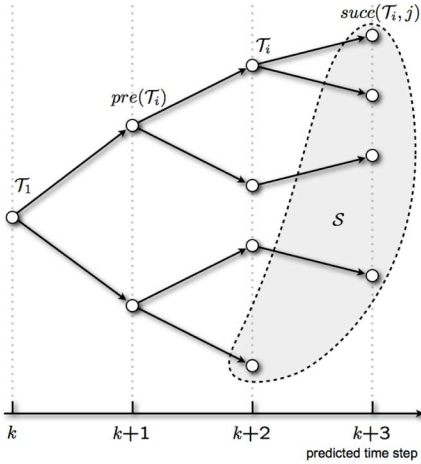


Fig. 2. Example of a *multiple-horizon* optimization tree. The root node  $\mathcal{T}_1$  is defined by the measured state  $x(k)$ . The set  $\mathcal{S}$  includes all the leaf nodes, that may have different prediction horizons.

purely stochastic (=deterministic probability distribution) to totally robust (=unknown probability distribution).

2) *Optimization Tree Design*: In order to optimize over a stochastic measure of the predicted system trajectory, we need to construct an optimization tree which describes the possible evolution in time of the state. We assume here that a model of the time dynamics of the probability distribution  $p(k)$  is available. Such a model, which may have any form (deterministic, stochastic, etc.), is exploited to generate the predicted scenarios over which the stochastic performance is optimized. Note that the stability properties granted by solving problem (11) do not involve any dynamic model of  $p(k)$ , except for the bounds expressed by  $\mathcal{P}$ . In other words the dynamic model that we assume here for predicting the future values of  $p(k)$  does not affect closed-loop convergence or control problem feasibility. As a consequence, its exactness is not critical for stability. However, the better is the dynamic model, the better the performance of the closed-loop system will be. In this sense we claim that, in the proposed control policy, stability/feasibility issues are decoupled from performance optimization.

We next present a tree design scheme based on a maximum-likelihood approach, where the optimization tree is updated every time step using the available information on the state and the disturbance. Let us introduce the following quantities to formally define the tree design procedure.

- $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n\}$ : the set of the tree nodes. Nodes are indexed progressively as they are added to the tree (i.e.,  $\mathcal{T}_1$  is the root node and  $\mathcal{T}_n$  is the last node added).
- $pre(\mathcal{N})$ : the predecessor of node  $\mathcal{N}$ .
- $succ(\mathcal{N}, j)$ : the successor of node  $\mathcal{N}$  generated with mode  $j \in \{1, 2, \dots, s\}$ .
- $\delta(\mathcal{N}) \in \{1, 2, \dots, s\}$ : the mode leading to node  $\mathcal{N}$ .
- $\pi_{\mathcal{N}}$ : the realization probability of node  $\mathcal{N}$ , i.e., the probability of reaching node  $\mathcal{N}$  from  $\mathcal{T}_1$ .
- $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_c\}$ : the set of candidate nodes, defined as  $\mathcal{C} = \{\mathcal{N} \notin \mathcal{T} \mid \exists (i, j) : \mathcal{N} = succ(\mathcal{T}_i, j)\}$ .
- $\mathcal{S} \subset \mathcal{T}$ : the set of leaf nodes,  $\mathcal{S} = \{\mathcal{T}_i \in \mathcal{T} \mid succ(\mathcal{T}_i, j) \notin \mathcal{T}, \forall j \in \{1, 2, \dots, s\}\}$ .

An illustrative optimization tree is shown in Fig. 2. The tree design procedure is listed in Algorithm 1 and described next. Each

node of the tree represents a future state which will be taken into account in the optimization problem. Starting from the root node  $\mathcal{T}_1$ , which is defined by the current value  $p(k)$ , a list of candidate nodes  $\mathcal{C}$  is evaluated by considering all the  $s$  possible dynamics in (2) and their probabilities  $p_1(k), p_2(k), \dots, p_s(k)$ . The candidate with maximum probability  $\mathcal{C}_{i^*}$  is added to the tree and removed from  $\mathcal{C}$ . Then, the list of candidates is updated by adding all the successors of the last node added to the tree, i.e.,  $succ(\mathcal{C}_{i^*}, j)$ , for all  $j \in \{1, 2, \dots, s\}$ . The algorithm is repeated until a desired number of nodes  $n_{\max}$  is reached. This procedure leads to a flexible tree structure, where different paths may in general have different prediction horizons, and as such we refer to it as a *multiple-horizon* approach (see Fig. 2).

---

#### Algorithm 1 Optimization tree design procedure at time $k$

---

- 1: set  $\mathcal{T} = \mathcal{T}_1, \pi_{\mathcal{T}_1} = 1, n = 1, c = s$ ;
- 2: set  $\mathcal{C} = \cup_{j=1}^s \{succ(\mathcal{T}_1, j)\}$
- 3: **while**  $n < n_{\max}$  **do**
- 4:   **for all**  $i \in \{1, 2, \dots, c\}$ , **do**
- 5:     compute  $\pi_{\mathcal{C}_i}$  according to the dynamic model of  $p(k)$ ;
- 6:   **end for**
- 7:   set  $i^* = \arg \max_{i \in \{1, 2, \dots, c\}} \pi_{\mathcal{C}_i}$ ;
- 8:   set  $\mathcal{T}_{n+1} = \mathcal{C}_{i^*}$ ;
- 9:   set  $\mathcal{T} = \mathcal{T} \cup \{\mathcal{T}_{n+1}\}$ ;
- 10:   set  $\pi_{\mathcal{T}_{n+1}} = \pi_{\mathcal{C}_{i^*}}$ ;
- 11:   set  $\mathcal{C} = (\mathcal{C} \setminus \mathcal{C}_{i^*}) \cup \cup_{j=1}^s \{succ(\mathcal{C}_{i^*}, j)\}$ ;
- 12:   set  $c = c + s - 1$ ;
- 13:   set  $n = n + 1$ ;
- 14: **end while**

3) *Control Problem Formulation*: We propose a formulation where the objective function to be minimized relies on an approximation of the expected value of the closed-loop performance, evaluated as a quadratic function of the state and the input. This is an arbitrary choice, however, and other kinds of criteria could also be considered, e.g., by optimizing over higher order moments. The approximation can be made arbitrarily accurate by increasing the number of nodes  $n_{\max}$ , at the expense of a higher computational load (see Remarks 2–3 for further details).

For ease of notation, in the following the abbreviate forms  $\pi_i, \delta(i), pre(i)$ , will be used to denote  $\pi_{\mathcal{T}_i}, \delta(\mathcal{T}_i), pre(\mathcal{T}_i)$ , respectively. Moreover, with  $x_i, u_i$  we denote, respectively, the state and the input associated with node  $\mathcal{T}_i$ . Let  $x(k)$  be the measured state and  $p(k)$  the probability distribution of  $w(k)$  at time  $k$ . Then, the unconstrained stochastic MPC problem can be formulated as

$$\min_u \sum_{i \in \mathcal{T} \setminus (\{\mathcal{T}_1\} \cup \mathcal{S})} \pi_i x_i^T Q_x x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{S}} \pi_i u_i^T Q_u u_i + \sum_{i \in \mathcal{S}} \pi_i x_i^T Q_S x_i \quad (13a)$$

$$\text{s.t. } x_1 = x(k) \quad (13b)$$

$$x_i = A_{\delta(i)} x_{pre(i)} + B_{\delta(i)} u_{pre(i)}, \forall i \in \mathcal{T} \setminus \{\mathcal{T}_1\} \quad (13c)$$

$$\begin{aligned} & \sum_{i=1}^s p_i(k) (A_i x_1 + B_i u_1)^T P (A_i x_1 + B_i u_1) \\ & \leq x_1^T (P - L) x_1 \end{aligned} \quad (13d)$$

where  $Q_x$ ,  $Q_u$ , and  $Q_S$  are weight matrices of appropriate dimensions. Problem (13) is a quadratically constrained quadratic programming (QCQP) problem. Constraints (13b) and (c) define the initial condition and the state dynamics, respectively. Constraint (13d) recursively enforces closed-loop stability in the mean-square sense, as shown in the following theorem.

*Theorem 1:* Consider a process modeled by system (1), where the disturbance  $w(k) \in \mathcal{W}$  is described by a random process with probability distribution  $p(k) \in \mathcal{P}$ , for all  $k \in \mathbb{N}$ , such that Assumptions 1–2 are satisfied. Let problem (11) admit a solution  $(Q, W, Y)$  and let  $P = Q^{-1}$ ,  $L = W^{-1}$ . Then, system (1) in closed loop with  $u(k) = u_1$ , where  $u_1$  is given by the receding horizon solution of problem (13), is UGMSES.

*Proof:* By similar reasonings of Lemma 3, we have that mean-square stability is provided by the receding-horizon satisfaction of constraint (13d), which now depends explicitly on the measured state  $x(k)$  and on the decision variable  $u(k)$ . We only need to show that problem (13) is recursively feasible at every time step. This follows by noting that (11d) implies (13d) if a state-feedback structure is imposed on the input  $u(k)$ . Hence,  $u_i = Kx_i$ , for all  $i$  such that  $\mathcal{T}_i \in \mathcal{T} \setminus \mathcal{S}$ , where  $K = YQ^{-1}$ , is always a feasible solution of (13). ■

*Remark 2:* In the borderline case where  $\mathcal{T}$  is a complete tree, i.e., a tree in which all the leaf nodes are at some depth  $N$  and all nodes but the leaf nodes have exactly  $s$  successors, we have that minimizing the objective function (13a) at time  $k$  is equivalent to minimizing

$$\begin{aligned} & \mathbb{E}_{[k, k+N-1]} \left( \sum_{j=0}^{N-1} \left( x(k+j)^T Q_x x(k+j) \right. \right. \\ & \left. \left. + u(k+j)^T Q_u u(k+j) \right) + x(k+N)^T Q_S x(k+N) \right). \end{aligned} \quad (14)$$

Otherwise, if the tree is not complete, (13a) is an approximation of (14), where the nodes related to the cut branches can be seen as terms in the cost function with null weight.

*Remark 3:* Even though only mean-square stability of the state is explicitly guaranteed by Theorem 1, it is easy to see that also the state covariance matrix is asymptotically driven to zero. Let  $\mu(k) = \mathbb{E}(x(k))$  and  $\Sigma(k) = \mathbb{E}((x(k) - \mu(k))(x(k) - \mu(k))^T)$  be the mean and the covariance matrix of  $x$  at time  $k$ , respectively, and  $\sigma_{ij}^2 \geq 0$  the elements of  $\Sigma$ , with  $i, j \in \{1, 2, \dots, n_x\}$ . The trace of  $\Sigma(k)$  is  $\sum_{i=1}^{n_x} \sigma_{ii}^2(k) = \mathbb{E}((x(k) - \mu(k))^T (x(k) - \mu(k)))$ . We have that (5) implies  $\lim_{k \rightarrow \infty} \mathbb{E}(x(k)^T x(k)) = 0$  and hence  $\lim_{k \rightarrow \infty} \mu(k) = \mathbf{0}$ . Combining these equations we obtain  $\lim_{k \rightarrow \infty} \text{trace}(\Sigma(k)) = 0$ . Since all the elements of  $\Sigma$  are nonnegative, we have  $\lim_{k \rightarrow \infty} \sigma_{ii}^2(k) = 0, \forall i$ . Semidefinite positiveness and symmetry of  $\Sigma$  yield  $\lim_{k \rightarrow \infty} \Sigma(k) = \mathbf{0}$ . Moreover, by similar reasonings, if we set  $Q_u = \mathbf{0}$  and  $Q_x = Q_S = I_{n_x}$  in (13a), problem (13) recursively minimizes

an approximation of  $\text{trace}(\Sigma)$  over the prediction tree. Hence, the weight matrices  $Q_x, Q_u, Q_S$  can be seen as tuning knobs which trade variance of the state trajectory for user-defined performance requirements.

The rest of this section is devoted to illustrate possible extensions of the proposed approach, intended to deal with more general problem formulations.

4) *Stochastic Weight Matrices:* In problem (13), the weight matrices  $Q_x, Q_u, Q_S$  are assumed to be constant over time. However, in general they can explicitly depend on the realization of the stochastic disturbance  $w$ . For instance, in financial applications the objective function is often an economic quantity where the weights represent prices of goods or assets, which are in general time-varying. Within the presented framework we can easily account for stochastic weight matrices by defining a set of  $s$  possible weight values

$$\mathcal{Q} = \{(Q_x^1, Q_u^1, Q_S^1), (Q_x^2, Q_u^2, Q_S^2), \dots, (Q_x^s, Q_u^s, Q_S^s)\}$$

such that

$$(Q_x(k), Q_u(k), Q_S(k)) = (Q_x^j, Q_u^j, Q_S^j)$$

if  $w(k) = \bar{w}_j, j \in \{1, 2, \dots, s\}$ . The availability of such a stochastic model which includes uncertainty on the weight matrices does not affect stability of the closed-loop system, but only performance. Hence, the computation of the Lyapunov function (11) needs not to be modified, and the cost function (13a) can be recast as

$$\sum_{i \in \mathcal{T} \setminus (\{\mathcal{T}_1\} \cup \mathcal{S})} \pi_i x_i^T Q_{x,j} x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{S}} \pi_i u_i^T Q_{u,j} u_i + \sum_{i \in \mathcal{S}} \pi_i x_i^T Q_{S,j} x_i$$

where now  $(Q_{x,j}, Q_{u,j}, Q_{S,j})$  are the weights related to the disturbance realization associated to the  $j$ th node of the tree. For clarity of presentation, in the following we will assume constant weight matrices.

5) *Soft Probabilistic Constraints:* As described in Section III-A2, at every time step the optimization tree provides information on the realization probability of every modeled node. This knowledge can be exploited to add soft constraints to the control problem to be satisfied in a probabilistic fashion, while retaining stochastic convergence to the origin. As a simple example, consider a soft constraint on a scalar state  $x \in \mathbb{R}$  (more complex constraints can be handled similarly). Let  $\tilde{\mathcal{X}} = \{x \mid |x| \leq \tilde{x}^+, \tilde{x}^+ > 0\}$ , and let

$$\Pr[x(k+1) \in \tilde{\mathcal{X}} \mid k] \geq \tilde{p}(k) \quad (15)$$

for all  $k \in \mathbb{N}$ , the probabilistic condition that is desirable to achieve, where  $\tilde{p}(k) \in (0, 1)$  can be time-varying. We can impose the constraint  $x_l \in \tilde{\mathcal{X}}$  as a soft constraint on every node  $\mathcal{T}_l \in \mathcal{T} \setminus \mathcal{T}_1$ , by recasting the control problem (13) as

$$\begin{aligned} \min_{u, \rho} & \sum_{i \in \mathcal{T} \setminus (\{\mathcal{T}_1\} \cup \mathcal{S})} \pi_i x_i^T Q_x x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{S}} \pi_i u_i^T Q_u u_i \\ & + \sum_{i \in \mathcal{S}} \pi_i x_i^T Q_S x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{T}_1} \rho_i^T Q_\rho \rho_i \end{aligned} \quad (16a)$$

$$\text{s.t.} \quad (13b), (13c), (13d) \quad (16b)$$

$$|x_i| \leq \tilde{x}^+ + \rho_i, \rho_i \geq 0, \forall i: \mathcal{T}_i \in \mathcal{T} \setminus \mathcal{T}_1 \quad (16c)$$

where  $\rho$  is a vector of slack variables weighted in the cost function by means of the matrix  $Q_\rho$ . By using this formulation, we aim at fulfilling (15) with a minimum probability

$$\tilde{p}(k) = \sum_{j \in \mathcal{H}_1} \pi_j, \quad \forall k \in \mathbb{N} \quad (17)$$

where  $\mathcal{H}_1$  is the set of successors of the root node  $\mathcal{T}_1$  at time  $k$ , i.e.,  $\mathcal{H}_1 = \{\mathcal{T}_i \in \mathcal{T} \mid \text{pre}(\mathcal{T}_i) = \mathcal{T}_1\}$ . By suitably modifying the control problem formulation, the value of the probability bound  $\tilde{p}(k)$  can be tuned as needed, according to (17). Additional nodes can be added to the tree in order to increase  $\tilde{p}(k)$ , or the set of nodes where (16c) is imposed can be restricted to decrease it. Indeed, satisfaction of the probabilistic condition (15) is not guaranteed by the receding horizon solution of (16), and it holds at time  $k$  if  $\sum_{j \in \mathcal{H}_1} \rho_j = 0$ . Even though soft probabilistic constraints have been introduced here for the case of unconstrained systems, analogous considerations hold as well for the constrained case, which is addressed below.

### B. The Constrained Case

Let us take now into account constraints on state and input vectors. We consider the component-wise bounds

$$x \in \mathcal{X} = \{x \mid |x_i| \leq \bar{x}_i, i = 1, 2, \dots, n_x\} \quad (18a)$$

$$u \in \mathcal{U} = \{u \mid |u_i| \leq \bar{u}_i, i = 1, 2, \dots, n_u\}. \quad (18b)$$

Other kinds of constraints, such as polytopic or ellipsoidal constraints, can be managed in a similar way, see, e.g., [2]. While the optimization tree design described in Section III-A3 is not affected by the presence of constraints (18), the offline Lyapunov function computation and the online control problem formulation presented, respectively, in Sections III-A1 and III-A4 need to be modified.

1) *Lyapunov Function Synthesis:* We present a solution derived from [2]. Our goal here is not to enlarge the feasible state set with respect to the robust controller presented in [2], but to exploit the stochastic model of the disturbance in order to provide a less conservative control action. The basic idea is to obtain, offline, a Lyapunov function and a feedback control law that guarantee constraints fulfillment and robust convergence to the origin, i.e.,

$$V(x(k+1)) - V(x(k)) \leq -x(k)^T L x(k) \quad (19)$$

for all  $k \in \mathbb{N}$ , and then relax (19) to its mean-square counterpart (7) in the online control problem, using the available information on  $p(k)$ . By using Schur complements (19) can be expressed as the LMIs

$$\begin{bmatrix} Q & (L^{1/2}Q)^T & (A_jQ + B_jY)^T \\ L^{1/2}Q & \gamma I_{n_x} & \mathbf{0} \\ A_jQ + B_jY & \mathbf{0} & Q \end{bmatrix} \succeq 0 \quad (20)$$

for all  $j \in \{1, 2, \dots, s\}$ , in the variables  $Q \in \mathbb{R}^{n_x \times n_x}$ ,  $Y \in \mathbb{R}^{n_u \times n_x}$ ,  $\gamma \in \mathbb{R}^+$ , where  $P = \gamma Q^{-1}$ ,  $Q = Q^T \succ 0$ ,  $L = L^T \succ 0$ , and a constant feedback structure has been imposed on the input, i.e.,  $u(k) = Kx(k)$ ,  $\forall k \in \mathbb{N}$ , with  $K = YQ^{-1}$ . In order to take into account constraints, we define the ellipsoid

$$\mathcal{E}_{\max} = \{x \mid x^T P x \leq \gamma\} = \{x \mid x^T Q^{-1} x \leq 1\}.$$

$\mathcal{E}_{\max}$  is an invariant ellipsoid for the closed-loop trajectories of system (1) controlled by  $u(k) = Kx(k)$ , i.e.,  $x(k) \in \mathcal{E}_{\max}$  implies  $x(k+t) \in \mathcal{E}_{\max}$ ,  $\forall t \in \mathbb{N}$ . A sufficient condition for the satisfaction of (18a) and (b) is given, respectively, by the LMIs

$$\begin{bmatrix} Q & (I_{n_x}^i (A_jQ + B_jY))^T \\ I_{n_x}^i (A_jQ + B_jY) & \bar{x}_i^2 \end{bmatrix} \succeq 0 \quad (21)$$

for all  $i \in \{1, 2, \dots, n_x\}$ ,  $j \in \{1, 2, \dots, s\}$ , and

$$\begin{bmatrix} X & Y \\ Y^T & Q \end{bmatrix} \succeq 0, \quad X_{ii} \leq \bar{u}_i^2, \quad \forall i \in \{1, 2, \dots, n_u\} \quad (22)$$

where  $I_{n_x}^i$  is the  $i$ th row of the  $n_x \times n_x$  identity matrix, and  $X_{ii}$  are the diagonal elements of the symmetric matrix  $X \in \mathbb{R}^{n_u \times n_u}$  (see [2] for more details). Finally, given a state  $x_0$ , we can express the condition  $x_0 \in \mathcal{E}_{\max}$  as

$$\begin{bmatrix} 1 & x_0^T \\ x_0 & Q \end{bmatrix} \succeq 0. \quad (23)$$

Hence, the problem of computing a quadratic Lyapunov function and a constant feedback control law to fulfill constraints (18) and robust stability (19) can be cast as the semidefinite programming problem

$$\max_{Q, X, Y, \gamma} \log \det(Q) \quad (24a)$$

$$\text{s.t.} \quad (20), (21), (22), (23) \quad (24b)$$

where  $x(0) = x_0$  is the initial state. The purpose of maximizing the objective function  $\log \det(Q)$  is to maximize the volume of the ellipsoid  $\mathcal{E}_{\max}$ , to find the largest set of states that contains the initial condition  $x(0)$  [25].

2) *Control Problem Formulation:* With the same notation used in Section III-A6, the constrained stochastic MPC problem at time  $k$  is formulated as the quadratically constrained quadratic programming (QCQP) problem

$$\min_u \sum_{i \in \mathcal{T} \setminus (\{\mathcal{T}_1\} \cup \mathcal{S})} \pi_i x_i^T Q_x x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{S}} \pi_i u_i^T Q_u u_i + \sum_{i \in \mathcal{S}} \pi_i x_i^T Q_S x_i \quad (25a)$$

$$\text{s.t.} \quad x_1 = x(k) \quad (25b)$$

$$x_i = A_{\delta(i)} x_{\text{pre}(i)} + B_{\delta(i)} u_{\text{pre}(i)}, \quad \forall i \in \mathcal{T} \setminus \{\mathcal{T}_1\} \quad (25c)$$

$$x_i \in \mathcal{X}, \quad \forall i \in \mathcal{T} \setminus \{\mathcal{T}_1\} \quad (25d)$$

$$u_i \in \mathcal{U}, \quad \forall i \in \mathcal{T} \setminus \mathcal{S} \quad (25e)$$

$$(A_i x_1 + B_i u_1) \in \mathcal{X}, \quad \forall i : \text{succ}(\mathcal{T}_1, i) \notin \mathcal{T} \text{ and } p_i(k) > 0 \quad (25f)$$

$$(A_i x_1 + B_i u_1)^T P (A_i x_1 + B_i u_1) \leq \gamma, \quad \forall i : p_i(k) > 0 \quad (25g)$$

$$\sum_{i=1}^s p_i(k) (A_i x_1 + B_i u_1)^T P (A_i x_1 + B_i u_1) \leq x_1^T (P - L) x_1. \quad (25h)$$

Constraint (25f) guarantees that the next state  $x(k+1)$  satisfies (18a), even if the disturbance realization  $w(k)$  at time  $k$  is not modeled in the current tree  $\mathcal{T}$ . Constraint (25g) recursively enforces the state  $x$  to lie in  $\mathcal{E}_{\max}$ . Imposing this constraint is

necessary because, in general, there exist states such that  $x \in \mathcal{X}$  but  $x \notin \mathcal{E}_{\max}$ ; hence, (25f) alone is not sufficient to provide recursive feasibility.

*Theorem 2:* Consider a process modeled by system (1) with initial condition  $x(0) = x_0$ , where the disturbance  $w(k) \in \mathcal{W}$  is described by a random process with probability distribution  $p(k) \in \mathcal{P}$ , for all  $k \in \mathbb{N}$ , such that Assumptions 1–2 are satisfied. Let problem (24) admit a solution  $(Q, X, Y, \gamma)$  and let  $P = \gamma Q^{-1}$ . Then, system (1) in closed loop with  $u(k) = u_1$ , where  $u_1$  is given by the receding horizon solution of problem (25), is ULMSES in  $\mathcal{E}_{\max}$  and satisfies constraints (18) for all  $k \in \mathbb{N}$ .

*Proof:* Satisfaction of state constraints (18a) at time  $k+1$  is provided by (25d), which concerns the state trajectories modeled in the tree, and (25f), which accounts for the unmodeled (but possible) scenarios. Satisfaction of input bounds (18b) at time  $k$  is provided by (25e) for  $i = 1$ . Mean-square stability is enforced by constraint (25h), where the only variable is  $u_1$ . Hence, it remains to prove that problem (25) is recursively feasible at every time step. Following the same lines of Theorem 1, our goal is to show that the input sequence  $u_i = Kx_i$ , for all  $i$  such that  $T_i \in \mathcal{T} \setminus \mathcal{S}$ , where  $K = YQ^{-1}$ , is always feasible for problem (25). Since  $(Q, X, Y, \gamma)$  is a solution of (24), the matrix  $P = \gamma Q^{-1}$  satisfies the robust stability condition (19) with  $u = Kx$ , and consequently also its stochastic counterpart (7), of which (25h) is a particular case for a given distribution  $p(k)$ . Moreover, if  $x(k) \in \mathcal{E}_{\max}$  then the control law  $u(k) = Kx(k)$  guarantees that constraints (18) are fulfilled, i.e., it satisfies (25d)–(f), and recursively provides  $x(k+1) \in \mathcal{E}_{\max}$ . Since we have  $x(0) \in \mathcal{E}_{\max}$  by construction, and the constraint (25g) enforces the next state  $x(k+1)$  to lie in  $\mathcal{E}_{\max}$ , then  $u_i = Kx_i$ , for all  $i$  such that  $T_i \in \mathcal{T} \setminus \mathcal{S}$ , is a feasible solution of problem (25) at every time step  $k$ , and this completes the proof. ■

#### IV. QP FORMULATION OF STOCHASTIC MPC AND ITS EXPLICIT SOLUTION

In this section, we recast the QCQP-based SMPC presented in Section III-B as a standard quadratic programming (QP) problem.<sup>1</sup> This has a twofold advantage: first, it can be solved with a reduced computational load; second, it allows to obtain the SMPC law in explicit form by employing standard multiparametric programming solvers. To this end, we construct an alternative formulation of the QCQP (25) where the quadratic inequality constraints (25g) and (h) are replaced by affine inequalities that are linearly dependent on the vector of parameters, that is the current state  $x(k)$ . In order to use piecewise linear (PWL) functions as candidate Lyapunov functions to prove stochastic convergence of state trajectories, we introduce a definition of stability different from the mean-square exponential stability given in Definition 1.

*Definition 2:* System (1) is said *Uniformly Globally Mean-PWL Exponentially Stable* (UGMPES) if there exist

<sup>1</sup>We only focus on the constrained problem. The unconstrained case can be derived similarly and is omitted here for the sake of brevity.

$c \geq 0$  and  $\lambda \in [0, 1)$  such that for any initial condition  $x(0) \in \mathbb{R}^{n_x}$  it holds that

$$\mathbb{E}(\Psi(x(k))) \leq c\Psi(x(0))\lambda^k, \quad \forall k \in \mathbb{N} \quad (26)$$

where  $\Psi : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$  is defined by

$$\Psi(x) = \max_i R_i x \quad (27)$$

and satisfies the condition

$$\Psi(x) > 0, \quad \forall x \neq \mathbf{0}$$

where  $R_i$ ,  $i \in \{1, 2, \dots, n_r\}$ , is the  $i$ th row of a matrix  $R \in \mathbb{R}^{n_r \times n_x}$ . Moreover, system (1) is said *Uniformly Locally Mean-PWL Exponentially Stable* (ULMPES) in  $\mathcal{X}_0$  if (26) holds for any initial condition  $x(0) \in \mathcal{X}_0 \subseteq \mathbb{R}^{n_x}$ .

*Lemma 4:* Condition (26) holds if

$$\mathbb{E}_k(\Psi(x(k+1))) \leq \lambda\Psi(x(k)) \quad (28)$$

with  $\lambda \in (0, 1)$ , is satisfied for all  $k \in \mathbb{N}$ .

*Proof:* The proof follows by similar reasonings of Lemma 1. Using (28) we have that

$$\begin{aligned} \mathbb{E}(\Psi(x(k))) &= \mathbb{E}_{[0, k-1]}(\Psi(x(k))) \\ &= \mathbb{E}_{[0, k-2]}(\mathbb{E}_{k-1}(\Psi(x(k)))) \\ &\leq \mathbb{E}_{[0, k-2]}(\lambda\Psi(x(k-1))) \\ &= \lambda\mathbb{E}_{[0, k-2]}(\Psi(x(k-1))) \\ &= \lambda\mathbb{E}(\Psi(x(k-1))) \\ &\leq \dots \leq \lambda^k\Psi(x(0)) \end{aligned}$$

which provides (26) with  $c = 1$ . ■

Indeed, any Minkowski function of a polyhedral C-set is a valid candidate for  $\Psi(x)$  [26], [27]. A special case is given by  $\Psi(x) = \|x\|_{\infty}$ , where  $R = [I_{n_x}, -I_{n_x}]^T$ .

*Remark 4:* The stability condition (26), that will be referred to in short as *mean-PWL stability*, is as good as condition (5) to ensure convergence of the state to the origin. In fact, (26) implies that  $\lim_{k \rightarrow \infty} \mathbb{E}(\Psi(x(k))) = 0$ , and consequently  $\lim_{k \rightarrow \infty} \mathbb{E}(x(k)) = \mathbf{0}$ , since  $\Psi(x(k)) > 0$  if  $x(k) \neq \mathbf{0}$ . By similar reasonings as in Remark 3, it also implies that  $\lim_{k \rightarrow \infty} \Sigma(k) = \mathbf{0}$ . Hence, the same convergence properties of mean and covariance of the state vector to the origin granted by stability in the mean-square sense (5) are retained also in the case of mean-PWL stability (26).

##### A. SMPC as a QP

Some additional ingredients are needed to formulate the SMPC problem as a quadratic programming problem and prove its properties. Define the scalar

$$\lambda = \inf\{z \in (0, 1) \mid zP - P + L \succeq 0\} \quad (29)$$

where  $L \succ 0$  is given and  $P = \gamma Q^{-1}$  is obtained by solving (24). The parameter  $\lambda$  describes the guaranteed contracting factor of the quadratic Lyapunov function  $V(x)$  from one time step to another, i.e.,  $V(x(k+1)) \leq \lambda V(x(k))$ . The definition



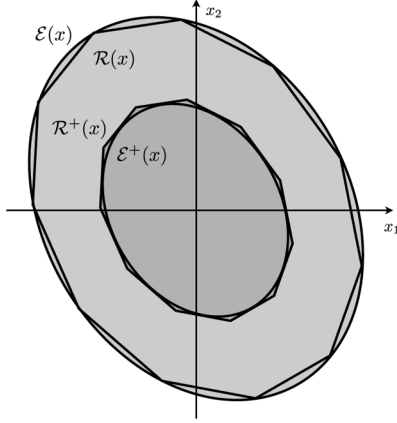


Fig. 3. Ellipsoidal and polyhedral sets used for QP formulation of SMPC.

of  $\lambda$  in (29) is well posed, as a feasible solution of the SDP problem (29) can always be obtained by an opportune choice of  $L$ , e.g.,  $L = \alpha P$ , with  $\alpha \in (0, 1)$ . Moreover, define the ellipsoids

$$\begin{aligned} \tilde{\mathcal{E}}_{\max}^+ &= \{x \mid x^T P x \leq \sup_{z \in \mathcal{E}_{\max}} z^T (P - L) z\} \\ \mathcal{E}_{\max}^+ &= \{x \mid x^T P x \leq \lambda \gamma\} \end{aligned}$$

where  $\mathcal{E}_{\max}^+ \subset \mathcal{E}_{\max}$  as  $\lambda \in (0, 1)$ . Then, construct the polytope  $\mathcal{R}_{\max} = \{x \mid R x \leq \mathbf{1}\}$  such that

$$\mathcal{E}_{\max}^+ \subset \mathcal{R}_{\max} \subset \mathcal{E}_{\max}. \quad (30)$$

The matrix  $R$  can be obtained using one of the existing methods for computing polyhedral approximations of ellipsoids, see, e.g., [28]. For convenience we choose  $\mathcal{R}_{\max}$  so that its vertices lie on the boundary of the ellipsoid  $\mathcal{E}_{\max}$ , i.e.,  $v^T P v = \gamma$ ,  $\forall v \in \text{vert}\{\mathcal{R}_{\max}\}$ . The polytope  $\mathcal{R}_{\max}$  will be used to grant the feasibility of the control problem and to induce a Minkowski function  $\Psi$  of the form (27). The level sets of  $\Psi$  are denoted as

$$\mathcal{R}(x) = \{z \mid R z \leq \Psi(x) \mathbf{1}\}. \quad (31)$$

Define also the scalar

$$c(x) = \inf\{z \mid v^T P v \leq \gamma z, v \in \text{vert}\{\mathcal{R}(x)\}\} \quad (32)$$

where  $c(x) > 0$  for all  $x \neq 0$ , and the sets

$$\mathcal{E}(x) = \{z \mid z^T P z \leq \gamma c(x)\} \quad (33)$$

$$\mathcal{E}^+(x) = \{z \mid z^T P z \leq \lambda \gamma c(x)\}. \quad (34)$$

Finally, define the polytope

$$\mathcal{R}^+(x) = \{z \mid R z \leq \eta \Psi(x) \mathbf{1}\} \quad (35)$$

where

$$\begin{aligned} \eta &= \inf\{y \mid R z \leq y \Psi(x) \mathbf{1}, x \neq 0, x \in \mathcal{R}_{\max}, \\ &\quad \forall z : z^T P z \leq \lambda \gamma c(x)\}. \end{aligned} \quad (36)$$

According to the notation introduced above,  $\mathcal{R}_{\max} = \mathcal{R}(\bar{x})$  and  $\mathcal{E}_{\max} = \mathcal{E}(\bar{x})$ , for any  $\bar{x}$  such that  $\Psi(x) = 1$ . An illustration of the sets (31), (33), (34) and (35) is shown in Fig. 3.

It is easy to show that  $c(x) \in (0, 1)$  for all  $x \in \mathcal{E}_{\max}$ , and  $\eta \in (0, 1)$  since  $\mathcal{R}_{\max} \subset \mathcal{E}_{\max}$ . Moreover, the value of  $\eta$  is not dependent on the particular state  $x \in \mathcal{R}_{\max}$  chosen to compute (36), as shown by the following lemma.

*Lemma 5:* Let  $x \neq 0$  such that  $x \in \mathcal{R}_{\max}$ , and let  $\eta = \inf\{y \mid R z \leq y \Psi(x) \mathbf{1}, \forall z : z^T P z \leq \lambda \gamma c(x)\}$ . Then,

$$\inf\{y \mid R z \leq y \Psi(\tilde{x}) \mathbf{1}, \forall z : z^T P z \leq \lambda \gamma c(\tilde{x})\} = \eta \quad (37)$$

for all  $\tilde{x} \neq 0$  such that  $\tilde{x} \in \mathcal{R}_{\max}$ .

*Proof:* Let us parametrize  $\tilde{x} = ah$ , where  $a > 0$  and  $h$  is such that  $\Psi(h) = \Psi(x)$ , i.e.,  $h$  lies on the boundary of  $\mathcal{R}(x)$ . We have  $\mathcal{R}(\tilde{x}) = \mathcal{R}(ah) = \{z \mid R z \leq a \Psi(x) \mathbf{1}\}$ . If  $v \in \text{vert}\{\mathcal{R}(x)\}$ , then  $av \in \text{vert}\{\mathcal{R}(\tilde{x})\}$ . Hence, it holds

$$\begin{aligned} c(\tilde{x}) &= c(ah) \\ &= \inf\{z \mid v^T P v \leq \gamma z, v \in \text{vert}\{\mathcal{R}(ah)\}\} \\ &= \inf\{z \mid a^2 v^T P v \leq \gamma z, v \in \text{vert}\{\mathcal{R}(x)\}\} \\ &= a^2 c(x). \end{aligned}$$

By using  $\tilde{x} = ah$  and  $c(\tilde{x}) = a^2 c(x)$ , we can rewrite the LHS of (37) as

$$\begin{aligned} &\inf\{y \mid R z \leq y \Psi(\tilde{x}) \mathbf{1}, \forall z : z^T P z \leq \lambda \gamma c(\tilde{x})\} \\ &= \inf\{y \mid R z \leq ya \Psi(x) \mathbf{1}, \forall z : z^T P z \leq \lambda \gamma a^2 c(x)\} \\ &= \inf\{y \mid R \zeta \leq y \Psi(x) \mathbf{1}, \forall \zeta = \frac{z}{a} : \zeta^T P \zeta \leq \lambda \gamma c(x)\} \\ &= \eta \end{aligned}$$

which completes the proof.  $\blacksquare$

We are now ready to state the main theorem of this section.

*Theorem 3:* Consider a process modeled by system (1) with initial condition  $x(0) = x_0$ , where the disturbance  $w(k) \in \mathcal{W}$  is described by a random process with probability distribution  $p(k) \in \mathcal{P}$ , for all  $k \in \mathbb{N}$ , such that Assumptions 1–2 are satisfied. Let problem (24) admit a solution  $(Q, X, Y, \gamma)$ , and let  $P = \gamma Q^{-1}$ . Consider the following QP problem

$$\begin{aligned} \min_{u, y} \quad & \sum_{i \in \mathcal{T} \setminus (\{\mathcal{T}_1\} \cup \mathcal{S})} \pi_i x_i^T Q_x x_i + \sum_{i \in \mathcal{T} \setminus \mathcal{S}} \pi_i u_i^T Q_u u_i \\ & + \sum_{i \in \mathcal{S}} \pi_i x_i^T Q_S x_i + \rho y^T y \end{aligned} \quad (38a)$$

$$\text{s.t.} \quad x_1 = x(k) \quad (38b)$$

$$x_i = A_{\delta(i)} x_{\text{pre}(i)} + B_{\delta(i)} u_{\text{pre}(i)}, \forall i \in \mathcal{T} \setminus \{\mathcal{T}_1\} \quad (38c)$$

$$x_i \in \mathcal{X}, \forall i \in \mathcal{T} \setminus \{\mathcal{T}_1\} \quad (38d)$$

$$u_i \in \mathcal{U}, \forall i \in \mathcal{T} \setminus \mathcal{S} \quad (38e)$$

$$\begin{aligned} &(A_i x_1 + B_i u_1) \in \mathcal{X}, \\ &\forall i : \text{succ}(\mathcal{T}_1, i) \notin \mathcal{T} \text{ and } p_i(k) > 0 \end{aligned} \quad (38f)$$

$$R(A_i x_1 + B_i u_1) \leq \mathbf{1}, \forall i : p_i(k) > 0 \quad (38g)$$

$$G y \leq 0, \quad (38h)$$

$$\begin{aligned} &R x_1 \leq y_{s+1} \mathbf{1}, R(A_i x_1 + B_i u_1) \leq y_i \mathbf{1}, \\ &\forall i \in \{1, 2, \dots, s\}, \end{aligned} \quad (38i)$$

where  $\rho > 0$  and  $G = [p(k)^T, -\eta]$ . Then, system (1) in closed loop with  $u(k) = u_1$ , where  $u_1$  is given by the receding horizon solution of problem (38), is ULMPEs in  $\mathcal{X} \cap \mathcal{R}_{\max}$  and satisfies constraints (18) for all  $k \in \mathbb{N}$ .

*Proof:* In order to prove the theorem we will show that, if (24) has a solution, then 1) constraint (38g) admits a solution  $u_1 \in \mathcal{U}$  for all  $x_1 \in \mathcal{X} \cap \mathcal{R}_{\max}$ , 2) the recursive satisfaction of (38h) and (i) grants mean-PWL stability, and finally 3) constraints (38h) and (38i) are always feasible.

- 1) Note that (38g) is equivalent to  $x(k+1) \in \mathcal{R}_{\max}$ , for any  $w(k) \in \mathcal{W}$  which is possible at time  $k$ . Since  $(Q, X, Y, \gamma)$  is given by the solution of (24), it satisfies (19) and by Theorem 2 we have that for all  $x(k) \in \mathcal{E}_{\max}$  and  $w(k) \in \mathcal{W}$ , there exists  $u(k) \in \mathcal{U}$  such that  $x(k+1) \in \mathcal{E}_{\max}^+$ . Of course  $\mathcal{E}_{\max}^+ \subset \mathcal{E}_{\max}$ , because  $L \succ 0$ . Moreover,  $\lambda P - P + L \succeq 0$  implies that

$$\begin{aligned} \sup_{z \in \mathcal{E}_{\max}} z^T (P - L) z &\leq \sup_{z \in \mathcal{E}_{\max}} z^T \lambda P z \\ &= \lambda \sup_{z \in \mathcal{E}_{\max}} z^T P z \\ &= \lambda \gamma \end{aligned}$$

i.e.,  $\tilde{\mathcal{E}}_{\max}^+ \subseteq \mathcal{E}_{\max}^+$ . Combining this with (30), we have that  $\tilde{\mathcal{E}}_{\max}^+ \subseteq \mathcal{E}_{\max}^+ \subset \mathcal{R}_{\max} \subset \mathcal{E}_{\max}$ . Hence, for all  $x(k) \in \mathcal{E}_{\max}$  and  $w(k) \in \mathcal{W}$ , there exists  $u(k) \in \mathcal{U}$  such that  $x(k+1) \in \mathcal{R}_{\max}$ , which proves the feasibility of (38g).

- 2) As stated in Lemma 4, mean-PWL stability is granted by the satisfaction of the following condition

$$\mathbb{E}_k(\Psi(x(k+1))) \leq \eta \Psi(x(k)) \quad (39)$$

at all time steps  $k$ , where  $\eta \in (0, 1)$  is defined by (36). Condition (39) can be rewritten as

$$\sum_{i=1}^s p_i(k) \max_j R_j(A_i x(k) + B_i u(k)) \leq \eta \max_j R_j x(k). \quad (40)$$

Now, introducing the vector of slack variables  $y = [y_1, y_2, \dots, y_{s+1}]^T \in \mathbb{R}^{s+1}$  to compute the  $s+1$  max operators in (40), adding the term  $\rho y^T y$  in the objective function (38a), and substituting  $x(k) = x_1$ ,  $u(k) = u_1$ , we have that mean-PWL stability is yielded by the constraints

$$\begin{aligned} \sum_{i=1}^s p_i(k) y_i &\leq \eta y_{s+1} \\ R x_1 &\leq y_{s+1} \mathbf{1} \\ R(A_i x_1 + B_i u_1) &\leq y_i \mathbf{1} \end{aligned}$$

for all  $i \in \{1, 2, \dots, s\}$ , which are equivalent to (38h) and (38i).

- 3) It remains to prove that (38h) and (38i) admit a feasible solution  $u_1 \in \mathcal{U}$  for all  $x_1 \in \mathcal{X} \cap \mathcal{R}_{\max}$ . By definition of  $c(x)$  in (32) we have that  $\mathcal{R}(x) \subset \mathcal{E}(x)$ . Moreover,  $\mathcal{R}^+(x) \subset \mathcal{R}(x)$  as  $\eta > 0$ . Finally, by definition of  $\eta$  in (36) we have that  $\mathcal{E}^+(x) \subset \mathcal{R}^+(x)$ . Hence, it holds that

$$\mathcal{E}^+(x) \subset \mathcal{R}^+(x) \subset \mathcal{R}(x) \subset \mathcal{E}(x) \quad (41)$$

for all  $x \in \mathcal{R}_{\max}$ ,  $x \neq \mathbf{0}$  (for  $x = \mathbf{0}$ , we have  $\mathcal{E}^+(\mathbf{0}) = \mathcal{R}^+(\mathbf{0}) = \mathcal{R}(\mathbf{0}) = \mathcal{E}(\mathbf{0}) = \{\mathbf{0}\}$ ). Now, by Theorem 2 we have that there exists  $u(k) \in \mathcal{U}$  such that  $x(k+1) \in \mathcal{E}^+(x(k))$ , for all  $x(k) \in \mathcal{E}(x(k))$  and  $w(k) \in \mathcal{W}$ , since

$\lambda P - P + L \succeq 0$ . Combining this with (41) we have that, for all  $x(k) \in \mathcal{R}_{\max}$  and  $w(k) \in \mathcal{W}$ , there exists  $u(k) \in \mathcal{U}$  such that  $x(k+1) \in \mathcal{R}^+(x(k))$ , i.e.,

$$R(A_i x(k) + B_i u(k)) \leq \eta \Psi(x(k)) \mathbf{1}, \quad \forall i \in \{1, 2, \dots, s\}. \quad (42)$$

Now, letting  $j_i^* = \arg \max_j R_j(A_i x(k) + B_i u(k))$ , by (42) we have that

$$\begin{aligned} R_{j_i^*}(A_i x(k) + B_i u(k)) &\leq \eta \max_j R_j x(k), \\ &\quad \forall i \in \{1, 2, \dots, s\}. \end{aligned} \quad (43)$$

Substituting (43) in the left-hand side of (40), we obtain

$$\sum_{i=1}^s p_i(k) \eta \max_j R_j x(k) \leq \eta \max_j R_j x(k) \quad (44)$$

which holds as  $\mathbf{1}^T p(k) = 1$ . Hence, (40) is always feasible, and so are the equivalent constraints (38h) and (38i). This completes the proof.  $\blacksquare$

Note that the feasible region of the SMPC based on QP (i.e., the set of states that fulfill constraints and admit a feasible solution of the predictive control problem) is given by  $\mathcal{X} \cap \mathcal{R}_{\max}$ , while the feasible region of the SMPC based on QCQP is  $\mathcal{X} \cap \mathcal{E}_{\max}$ . Being  $\mathcal{R}_{\max} \subset \mathcal{E}_{\max}$ , the QP formulation has in general a smaller feasible region than the QCQP one (except in the case  $\mathcal{X} \subset \mathcal{R}_{\max}$ , where the regions are both equal to  $\mathcal{X}$ ). This difference can be reduced by increasing the number of vertices which define the polytope  $\mathcal{R}_{\max}$ , at the cost of an increased complexity of the QP problem (38), since every vertex adds  $2s+1$  inequality constraints to the control problem.

## B. Explicit Solution of QP-Based SMPC

The complexity of the online control problem (38) grows with the state and input dimensions, and this may result in impractical application to fast processes where the available computation time is small, especially if a high number of nodes  $n_{\max}$  is used to approximate the expected system trajectory with satisfying accuracy. An efficient approach to evaluate MPC laws was proposed in [29]: rather than solving the control problem on line for the current state vector, by employing multiparametric programming techniques the problem is solved off line for a given range of parameters, providing the explicit dependence of the control input on the parameters (for a survey on explicit MPC the reader is referred to [30]). Currently available tools to compute the explicit solution, such as the Hybrid Toolbox for Matlab [31], require the control problem to have affine equality and inequality constraints, with linear dependence on the vector of parameters. Provided that this is the case, the control law obtained in explicit form can be shown to be piecewise affine (PWA).

The time-varying nature of the SMPC problem (38), due to the underlying optimization tree, prevents it from being directly recast in a fashion suitable for multiparametric programming. However, the structure of the tree at time  $k$ , designed according to Algorithm 1, depends only on the current value of the distribution  $p(k)$ . Hence, if  $p(k)$  takes value in a finite set, and this set is known in advance, we can construct *offline* all the optimization trees that will be needed on line, one for every possible value of  $p(k)$ , thus remapping the time-varying control problem

(25) in a set of time-invariant problems, each of them depending only on the measured state  $x(k)$ . Then, we are able to enumerate all the possible probability distributions  $p(k)$ , construct the corresponding optimization trees, and compute for each of them the explicit solution of the SMPC control problem via multiparametric programming.

In order to consider a finite set of probability distributions we rely on the following assumption.

*Assumption 3:* At each time step  $k$ , the probability distribution  $p(k) \in \mathcal{Q} = \{q^1, q^2, \dots, q^n\}$ , where  $n$  is finite and  $q^i \in \mathcal{D}$  for all  $i \in \{1, 2, \dots, n\}$ .

Although restrictive on the values that  $p(k)$  can assume, Assumption 3 is often satisfied in practice, as, for instance, in the case of Markov chains reviewed in Section II-B. Now, the explicit SMPC control law can be defined as the PWA function

$$u^*(x(k), p(k)) = F_{i,j}x(k) + g_{i,j}$$

if  $x(k) \in \mathcal{A}_{i,j}$  and  $p(k) = q^i$ , where  $F_{i,j} \in \mathbb{R}^{n_u \times n_x}$  and  $g_{i,j} \in \mathbb{R}^{n_u}$  are the gains associated to every polyhedral region  $\mathcal{A}_{i,j} = \{x \in \mathbb{R}^{n_x} \mid C_{i,j}x \leq d_{i,j}\}$ ,  $i \in \{1, 2, \dots, n\}$ ,  $j \in \{1, 2, \dots, r_i\}$ . The gains  $F_{i,j}$ ,  $g_{i,j}$  and the regions  $\mathcal{A}_{i,j}$  are given by solving with respect to the parameter  $x(k) \in \mathcal{X} \cap \mathcal{R}_{\max}$  the  $n$  multiparametric quadratic programming (mp-QP) problems obtained by substituting  $p(k) = q^i$  in (38), for all  $i \in \{1, 2, \dots, n\}$ . Online, first the active explicit SMPC controller is selected based on the current value of  $p(k)$ , then the active region of the selected controller is found based on the measurement  $x(k)$ , and finally the control move associated to the selected region is evaluated. The online execution of the explicit SMPC scheme is summarized in Algorithm 2.

---

#### Algorithm 2 Explicit SMPC

---

- 1: **At time step  $k$ :**
- 2: find  $i \in \{1, 2, \dots, n\}$  such that  $p(k) = q^i$ ;
- 3: find  $j \in \{1, 2, \dots, r_i\}$  such that  $C_{i,j}x(k) \leq d_{i,j}$ ;
- 4: compute  $u(k) = F_{i,j}x(k) + g_{i,j}$ ;

### V. EXAMPLE

In this section, we test the performance of the proposed approach on a numerical example, simple enough to concentrate on highlighting the various features of SMPC. For more advanced applications to real-life control problems the reader is referred to [17] and [18], which address the problem of optimal power splitting among the available sources in hybrid electric vehicles and of adaptive cruise control; to [19], which introduces an SMPC approach for networked control systems subject to time-varying delays and time-varying transmission intervals; and to [20] for applications of SMPC to dynamic option hedging.

Consider the second-order discrete-time uncertain linear system of the form (1), with  $A_i = \begin{bmatrix} -0.8 & 1 \\ 0 & \bar{w}_i \end{bmatrix}$ ,  $B_i = [0, 1]^T$ ,  $i \in \{1, 2, 3\}$ , and  $w(k) \in \mathcal{W} = \{0.8, 1.2, -0.4\}$ . The spectrum of the matrix  $A(w(k))$  can be either contained inside the unit circle or not, depending on the value  $w(k)$ . We assume that  $p(k)$  is described by a time-homogeneous Markov chain

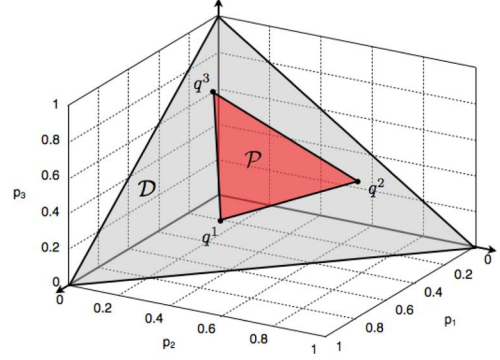


Fig. 4. Plot of the polytopes  $\mathcal{D}$  and  $\mathcal{P}$  for the considered test case.

defined by (4) with  $T = E = \begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0.1 & 0.6 & 0.3 \\ 0.2 & 0.1 & 0.7 \end{bmatrix}$ , and that the

state  $z(k)$  of the Markov chain at time  $k$  is known. This model satisfies Assumptions 1, 2, and 3, as at time  $k$  we have  $p(k) = q^i$  if  $z(k) = \bar{z}_i$ ,  $i \in \{1, 2, 3\}$ , where  $q^1 = [0.5, 0.3, 0.2]^T$ ,  $q^2 = [0.1, 0.6, 0.3]^T$ , and  $q^3 = [0.2, 0.1, 0.7]^T$ . The plots of  $\mathcal{P}$  and  $\mathcal{D}$  are shown in Fig. 4.

The purpose of the control action is to steer the state  $x$  of the controlled plant to the origin, while satisfying state and input constraints (18), defined by  $\bar{x} = [10, 2]^T$  and  $\bar{u} = 1$ . First, using *cvx* [32], we found a solution of the semidefinite programming problem (24), and obtained the feasible ellipsoid  $\mathcal{E}_{\max}$ . In order to formulate the QP-based SMPC, we designed  $\mathcal{R}_{\max}$  as a polygonal approximation of  $\mathcal{E}_{\max}$  with eight vertices, such that (30) is fulfilled. Then, we run a set of  $N_s = 5000$  simulations, each of  $T_s = 15$  time steps, with random initial state sampled from the feasible region of the QP-based SMPC (see Fig. 6).

The weight matrices used in simulations are  $Q_x = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix}$ ,  $Q_u = 1$ , and  $L = 10^{-4} \begin{bmatrix} 1 & -1 \\ -1 & 25 \end{bmatrix}$ , and a number of nodes  $n_{\max} = 20$  has been chosen to design the optimization tree.

The proposed SMPC control scheme was compared to the following three controllers:

- The LMI-based *robust MPC* (RMPC) proposed in [2], which provides robust convergence and hard constraint fulfillment, but does not exploit the available stochastic information on the disturbance  $w$ .
- A linear deterministic *frozen-time MPC* (FTMPC) formulation with time-varying system model, where at every step  $k$  a MPC control problem is solved based on the dynamics mode  $i \in \{1, 2, \dots, s\}$  which is currently the most probable, i.e., such that  $p_i(k) \geq p_j(k)$ ,  $\forall j$ . FTMPC retains the constraints on mean-square stability and recursive feasibility; hence, stochastic convergence and state and input constraints fulfillment are guaranteed, but a cost function based on a single scenario is minimized, instead of building a stochastic optimization tree. In other words, FTMPC can be seen as a special case of SMPC where  $T = E = I_3$ .
- A linear deterministic *prescient MPC* (PMPC) formulation, where the complete knowledge of the disturbance realization for a given future horizon window is exploited. At time  $k$ , the PMPC solves an optimal control problem over a

TABLE I  
 SIMULATION RESULTS

Controller $c$	$\mu(c)$	$\sigma^2(c)$
RMPC	1.408	1.159
FTMPC	1.361	1.220
SMPC	1.268	1.133
PMPC	1	1

finite horizon of  $n_{\max} - 1$  steps by knowing the future evolution of  $w(k+j)$  in advance, for all  $j \in \{0, 1, \dots, n_{\max} - 1\}$ . The PMPC is optimal if  $n_{\max} = T_s$ , otherwise the solution is in general suboptimal. The resulting closed-loop trajectory can be seen as a benchmark to evaluate the performance obtained by the other controllers.

To compare the performances achieved by the aforementioned control schemes, we define the experimental cost function

$$J(i, c) = \sum_{k=1}^{T_s} (x^{i,c}(k)^T Q_x x^{i,c}(k) + u^{i,c}(k)^T Q_u u^{i,c}(k)) \quad (45)$$

where  $i \in \{1, 2, \dots, N_s\}$  indexes the values related to the  $i$ th simulation, and  $c \in \{\text{RMPC}, \text{FTMPC}, \text{SMPC}, \text{PMPC}\}$  refers to the implemented controller. Table I shows the simulation results in terms of the mean  $\mu(c)$  and the variance  $\sigma^2(c)$  over all the simulations of the experimental cost function (45) for every controller  $c$ , normalized with respect to the PMPC value, i.e.,

$$\mu(c) = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{J(i, c)}{J(i, \text{PMPC})}$$

$$\sigma^2(c) = \frac{\sum_{i=1}^{N_s} (J(i, c) - \frac{1}{N_s} \sum_{j=1}^{N_s} J(j, c))^2}{\sum_{i=1}^{N_s} (J(i, \text{PMPC}) - \frac{1}{N_s} \sum_{j=1}^{N_s} J(j, \text{PMPC}))^2}.$$

As we can see from the results in Table I, the proposed stochastic MPC policy achieves an average improvement in the closed-loop performance of 10% with respect to robust MPC, and of 7% with respect to frozen-time MPC. In the presented example, the SMPC based on the QCQP (25) and its QP-based counterpart (38) achieved indistinguishable results; hence, the mentioned simulation data hold for both formulations. On a Macbook 2.4 GHz, using Matlab 7.6 and Cplex 11.2, the average computation time needed to solve an instance of the QCQP (25) and of the QP (38) were 2.2 ms and 1.4 ms, respectively.

In order to evaluate the tradeoff between performance and complexity for both the QCQP and the QP formulations, the previous simulations have been repeated with different maximum numbers of nodes  $n_{\max}$  in the optimization tree. The results in terms of averaged normalized experimental cost  $\mu$  and computation time are shown in Fig. 5.

Finally, we applied multiparametric programming techniques to obtain the QP-based SMPC law in explicit form, as described in Section IV-B. We solved  $n = 3$  mp-QPs based on (38), one for each possible value of the probability distribution  $p(k)$ . The corresponding state space partitions are shown in Fig. 6, and consist of 18 regions for  $z(k) = \bar{z}_1$  and  $z(k) = \bar{z}_3$ , and 20 regions for  $z(k) = \bar{z}_2$ .

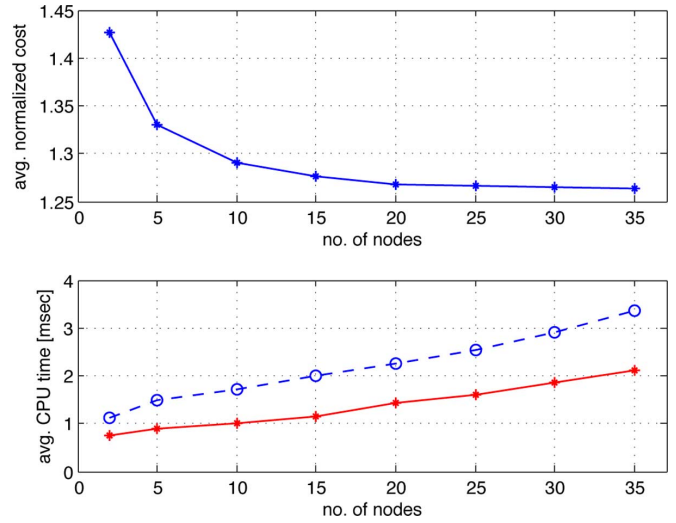


Fig. 5. Plot of average normalized experimental cost  $\mu$  and average CPU time of QCQP-based SMPC (blue, dashed line) and QP-based SMPC (red, solid line) over  $N_s = 5000$  runs, for varying number of tree nodes  $n_{\max}$ . Cost normalization is computed with respect to 20-nodes PMPC for comparison purposes.

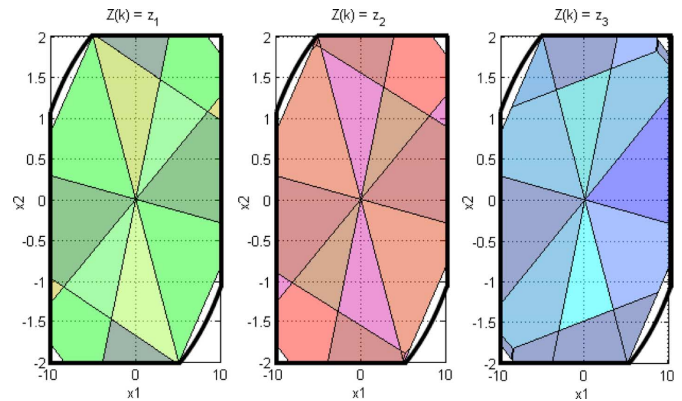


Fig. 6. State space partitions of the explicit solution of the QP-based SMPC (38) for every Markov chain state  $\bar{z}_i$ ,  $i \in \{1, 2, 3\}$ . The black solid lines describe the boundary of the feasible set of QCQP-based SMPC.

## VI. CONCLUSION

This paper has presented a stochastic model predictive control formulation based on scenario generation for linear systems affected by discrete multiplicative disturbances and state/input constraints. By separating the problems of performance optimization on one hand, and stochastic convergence to the origin with robust constraint fulfillment on the other, we set up a control scheme which requires the offline solution of an LMI problem, and the receding horizon implementation of a QCQP or a QP problem to obtain the control action. For the latter case, the SMPC law can also be obtained in explicit form. The proposed control algorithm is suitable for application to a wide class of discrete disturbance processes. Simulations on a numerical example have been run to evaluate performance in comparison to classic robust and deterministic MPC formulations. Although the approach relies on *discrete* stochastic models of the disturbance, it can be further extended to consider *continuously* distributed uncertainty, as presented in [19].

## ACKNOWLEDGMENT

The authors would like to thank P. Colaneri for pointing out fundamental references on mean-square stability of discrete-time systems, and D. M. de la Peña for fruitful discussions.

## REFERENCES

- [1] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Practice*, vol. 11, no. 7, pp. 733–764, 2003.
- [2] M. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.
- [3] P. O. M. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Trans. Autom. Control*, vol. 43, no. 8, pp. 1136–1142, Aug. 1998.
- [4] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Proc. Robust. Ident. Control*, 1999, vol. 245, pp. 207–226.
- [5] A. Bemporad, F. Borrelli, and M. Morari, "Min-max control of constrained uncertain discrete-time linear systems," *IEEE Trans. Autom. Control*, vol. 48, no. 9, pp. 1600–1606, Sep. 2003.
- [6] D. M. de la Peña, T. Alamo, A. Bemporad, and E. F. Camacho, "A decomposition algorithm for feedback min-max model predictive control," *IEEE Trans. Autom. Control*, vol. 51, no. 10, pp. 1688–1692, Oct. 2006.
- [7] A. Schwarm and M. Nikolaou, "Chance-constrained model predictive control," *AICHE J.*, vol. 45, no. 8, pp. 1743–1752, 1999.
- [8] D. M. de la Peña, A. Bemporad, and T. Alamo, "Stochastic programming applied to model predictive control," in *Proc. 44th IEEE Conf. Decision Control*, Sevilla, Spain, 2005, pp. 1361–1366.
- [9] P. Couchman, M. Cannon, and B. Kouvaritakis, "Stochastic MPC with inequality stability constraints," *Automatica*, vol. 42, pp. 2169–2174, 2006.
- [10] J. Primbs and C. Sung, "Stochastic receding horizon control of constrained linear systems with state and control multiplicative noise," *IEEE Trans. Autom. Control*, vol. 54, no. 2, pp. 221–230, Feb. 2009.
- [11] D. H. van Hessem, C. W. Scherer, and O. H. Bosgra, "LMI-based closed-loop economic optimization of stochastic process operation under state and input constraints," in *Proc. 40th IEEE Conf. Decision Control*, Orlando, FL, 2001, pp. 4228–4233.
- [12] M. Ono and B. C. Williams, "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint," in *Proc. 47th IEEE Conf. Decision Control*, Cancun, Mexico, 2008, pp. 3427–3432.
- [13] I. Batina, A. A. Stoorvogel, and S. Weiland, "Optimal control of linear, stochastic systems with state and input constraints," in *Proc. 41th IEEE Conf. Decision Control*, Las Vegas, NV, 2002, pp. 1564–1569.
- [14] K. Høyland and S. W. Wallace, "Generating scenario trees for multistage decision problems," *Manage. Sci.*, vol. 47, no. 2, pp. 295–307, 2001.
- [15] J. Dupačova, G. Consigli, and S. W. Wallace, "Scenarios for multistage stochastic programs," *Ann. Operat. Res.*, vol. 100, pp. 25–53, 2004.
- [16] G. C. Goodwin, J. Østergaard, D. E. Quevedo, and A. Feuer, "A vector quantization approach to scenario generation for stochastic NMPC," in *Int. Workshop Assess. Future Direct. Nonlinear Model Predict. Control*, Pavia, Italy, 2008.
- [17] G. Ripaccioli, D. Bernardini, S. Di Cairano, A. Bemporad, and I. V. Kolmanovskiy, "A stochastic model predictive control approach for series hybrid electric vehicle power management," in *Proc. Amer. Contr. Conf.*, Baltimore, MD, 2010, pp. 5844–5849.
- [18] M. Bichi, G. Ripaccioli, S. Di Cairano, D. Bernardini, A. Bemporad, and I. Kolmanovskiy, "Stochastic model predictive control with driver behavior learning for improved powertrain control," in *Proc. 49th IEEE Conf. Decision Control*, Atlanta, GA, 2010, pp. 607–6082.
- [19] D. Bernardini, M. C. F. Donkers, A. Bemporad, and W. P. M. H. Heemels, "A model predictive control approach for stochastic networked control systems," in *Proc. 2nd IFAC Workshop Estimation and Control Netw. Syst.*, Annecy, France, 2010, pp. 7–12.
- [20] A. Bemporad, T. Gabbiellini, L. Puglia, and L. Bellucci, "Scenario-based stochastic model predictive control for dynamic option hedging," in *Proc. 49th IEEE Conf. Decision Control*, Atlanta, GA, 2010, pp. 6089–6094.

- [21] D. Bernardini and A. Bemporad, "Scenario-based model predictive control of stochastic constrained linear systems," in *Proc. 48th IEEE Conf. Decision Control*, Shanghai, China, 2009, pp. 6333–6338.
- [22] O. L. V. Costa, M. D. Fragoso, and R. P. Marquez, *Discrete-Time Markov Jump Linear Systems*. London, U.K.: Springer-Verlag, 2005.
- [23] P. Bolzern, P. Colaneri, and G. De Nicolao, "Markov jump linear systems with switching transition rates: Mean square stability with dwell-time," *Automatica*, vol. 46, no. 6, pp. 1081–1088, 2010.
- [24] T. Morozan, "Stabilization of some stochastic discrete-time control systems," *Stochast. Anal. Applicat.*, vol. 1, no. 1, pp. 89–116, 1983.
- [25] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory Studies in Applied Mathematics*. Philadelphia, PA: SIAM, 1994.
- [26] D. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1969.
- [27] F. Blanchini, "Nonquadratic Lyapunov functions for robust control," *Automatica*, vol. 31, no. 3, pp. 451–461, 1995.
- [28] A. Alessio, M. Lazar, A. Bemporad, and W. P. M. H. Heemels, "Squaring the circle: An algorithm for generating polyhedral invariant sets from ellipsoidal ones," *Automatica*, vol. 43, pp. 2096–2103, 2007.
- [29] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.
- [30] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Proc. Int. Workshop Assess. Future Direct. Nonlinear Model Predict. Control*, Pavia, Italy, 2008.
- [31] A. Bemporad, "Hybrid Toolbox—User's Guide," 2003.
- [32] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming (Web Page and Software)," 2009 [Online]. Available: <http://stanford.edu/~boyd/cvx>



**Daniele Bernardini** (M'11) was born in 1982. He received the M.S. degree in computer engineering and the Ph.D. degree in information engineering from the University of Siena, Siena, Italy, in 2007 and 2010, respectively.

In spring 2010, he visited the Department of Electrical Engineering, Stanford University, Stanford, CA. In 2011, he was with the Department of Mechanical and Structural Engineering, University of Trento, Trento, Italy. In October 2011, he joined the IMT Institute for Advanced Studies Lucca, Lucca, Italy, where he is a Postdoctoral Research Fellow. His research interests include model predictive control, stochastic control, networked control, and applications to automotive and energy systems.



**Alberto Bemporad** (S'93–M'99–SM'06–F'10) received the M.S. degree in electrical engineering and the Ph.D. degree in control engineering from the University of Florence, Florence, Italy, in 1993 and 1997, respectively.

He spent the academic year 1996/1997 at the Center for Robotics and Automation, Department of Systems Science and Mathematics, Washington University, St. Louis, MO, as a Visiting Researcher. From 1997 to 1999, he held a postdoctoral position at the Automatic Control Laboratory, ETH Zurich, Zurich, Switzerland, where he collaborated as a Senior Researcher (2000–2002). From 1999 to 2009, he was with the Department of Information Engineering, University of Siena, Siena, Italy, becoming an Associate Professor in 2005. From 2010 to 2011, he was with the Department of Mechanical and Structural Engineering, University of Trento, Trento, Italy. Since 2011, he has been a Full Professor at the IMT Institute for Advanced Studies Lucca, Lucca, Italy. He has published more than 220 papers in the areas of model predictive control, hybrid systems, automotive control, multiparametric optimization, computational geometry, robotics, and finance. He is the author or coauthor of various MATLAB toolboxes for model predictive control design, including the *Model Predictive Control Toolbox* (The Mathworks, Inc.).

He was an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 2001 to 2004 and Chair of the Technical Committee on Hybrid Systems of the IEEE Control Systems Society from 2002 to 2010.