

An MPC Approach for LPV Systems in Input-Output Form

Hossam S. Abbas, *Member, IEEE*, Roland Tóth, *Member, IEEE*, Nader Meskin, *Member, IEEE*,
Javad Mohammadpour, *Member, IEEE*, and Jurje Hanema

Abstract—In this paper, a discrete-time model predictive control (MPC) design approach is proposed to control systems described by linear parameter-varying (LPV) models in input-output form subject to constraints. To ensure stability of the closed-loop system, a quadratic terminal cost along with an ellipsoidal terminal constraint are included in the control optimization problem. The proposed scheme is a robust LPV-MPC scheme, which considers future values of the scheduling variable being uncertain and varying inside a prescribed polytope. The MPC design problem is formulated as a linear matrix inequality (LMI) problem. The effectiveness of the proposed LPV-MPC design is demonstrated using a numerical example.

I. INTRODUCTION

Identifying *linear parameter-varying* (LPV) models in *input-output* (IO) form from data [1] has become well developed with several successful applications, e.g., [2]. However, most of the LPV controller synthesis techniques have been developed based on *state-space* (SS) models. Obtaining reliable SS realization of IO models is usually hindered by the so-called dynamic-dependency problem connected to LPV realization theory [1], which introduces a significant complexity increase of the realized models that grows beyond the applicable range of computational tools. Therefore, it is desired to synthesize controllers using IO models directly.

Model predictive control (MPC) has been developed to solve control problems that have constraints and time delay. In the SS setting, the MPC problem has received considerable research interest, see e.g., [3], also with various techniques developed in the LPV setting. The control law in most of these techniques, e.g., [4], is calculated by repeatedly solving a convex optimization problem based on *linear matrix inequalities* (LMIs) to minimize an upper bound on a worst-case function involving stability constraints. A common property of these approaches that they are based on an LPV-SS representation of the system and they depend on the availability of the corresponding states during implementation. The use of observers to access the state information may deteriorate significantly closed-loop performance in terms of input disturbance rejection when plant input constraints become activated, as in that case, the nonlinearities start to

dominate the behavior of the closed-loop system, see [5]. To cope with these issues, a subspace-based predictive control for LPV systems has been proposed in [6]. The critical issue in this approach is that no stability guarantee has been provided. Moreover, the complexity exponentially increases with the order and number of scheduling variables (p) of the system.

To cope with these issues, we develop in this work an MPC approach to control LPV-IO models subject to constraints. For the sake of simplicity, we focus here on the SISO case. To ensure stability of the proposed MPC technique, we utilize the stability framework of [3] along with a sufficient condition developed recently in [7] for stability of LPV-IO models. The proposed MPC design approach is formulated as an optimization problem subject to LMI constraints, where a robust MPC problem is solved at every sampling instant for the LPV-IO model such that future values of p over the prediction horizon are considered uncertain but confined inside a prescribed polytope. The bounds on the rate of change of p are exploited to reduce the conservatism of the design. The significance of the proposed control approach lies in the fact that it enables MPC control design directly based on LPV-IO representations. In addition, it offers reference tracking by integral action, has asymptotic stability guarantee, and does not require a state observer.

The paper is organized as follows: After some preliminaries in Section II, the proposed LPV-MPC scheme is developed in Section III. The extension to robust LPV-MPC is presented in Section IV. An illustrative example is given in Section V. Finally, the conclusions are drawn in Section VI.

II. PRELIMINARIES

Notations: For a sequence $z(k) : \mathbb{Z} \rightarrow \mathbb{R}$, let $z_{[k+i, k+j]} \in \mathbb{R}^{|i-j|+1}$ gather the values of z ordered from the sampling instant $k+i$ to $k+j$, $i, j \in \mathbb{Z}$. For a matrix $Z \in \mathbb{R}^{n \times m}$, let $Z_{i,j} \in \mathbb{R}^{(j-i+1) \times m}$ gather the rows of Z ordered from row i to j . An upper linear fractional transformation (LFT) is denoted by $\Delta \star \begin{bmatrix} L_{11} & L_{12} \\ \dots & \dots \\ L_{21} & L_{22} \end{bmatrix} = L_{22} + L_{21} \Delta (I - L_{11} \Delta)^{-1} L_{12}$.

An input-output representation of a SISO LPV system in discrete-time can be given by the difference equation

$$\mathcal{G} : \left(1 + \sum_{i=1}^{n_a} a_i(p_k) q^{-i} \right) y(k) = \sum_{j=0}^{n_b} b_j(p_k) q^{-j} u(k), \quad (1)$$

or $\mathcal{A}(q^{-1}, p_k) y(k) = \mathcal{B}(q^{-1}, p_k) u(k)$, where q^{-1} is the backward time-shift operator, $n_a, n_b \geq 0$, $u(k) : \mathbb{Z} \rightarrow \mathbb{R}$ and $y(k) : \mathbb{Z} \rightarrow \mathbb{R}$ are the control input and the measured output, respectively. Furthermore, the coefficients a_i and b_j

H. S. Abbas is with the Elec. Eng. Dept., Faculty of Eng., Assiut University, Assiut 71515, Egypt, R. Tóth and J. Hanema are with the Control Systems Group, Dept. of Elec. Eng., Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, N. Meskin is with the Dept. of Elec. Eng., College of Eng., Qatar University, PO Box 2713 Doha, Qatar and J. Mohammadpour is with the Complex Systems Control Lab, College of Eng., The University of Georgia, Athens, GA 30602, USA. Emails: hossam.abbas@aun.edu.eg, {r.toth, j.hanema}@tue.nl, nader.meskin@qu.edu.qa, javadm@uga.edu.

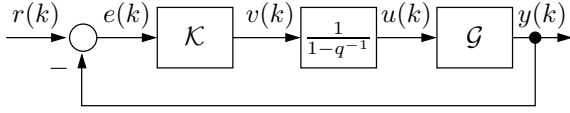


Fig. 1: Closed-loop interconnection: reference tracking.

are analytic and bounded functions of the scheduling variable $p_k = p(k) \in \mathbb{P}$, which is online measurable. For simplicity, we consider $b_0(p_k) \equiv 0$. Assume that \mathbb{P} is given by a convex set $\mathbb{P} := \text{Co}(\{p_1^v, \dots, p_{n_v}^v\})$, where each $p_i^v \in \mathbb{R}^{n_p}$ corresponds to a vertex of a polytope and Co denotes the convex hull. Moreover, let the rate of variation of the scheduling variable $dp(k) = p(k) - p(k-1)$ be bounded such that $dp(k) \in \mathbb{P}_d := \{dp \in \mathbb{R}^{n_p} \mid dp_{\min} \leq dp \leq dp_{\max}\}$. Consider the reference tracking problem depicted in Fig. 1 and assume that there exists a robust, *linear time-invariant* (LTI) controller \mathcal{K} which can stabilize the depicted closed-loop system for all $p \in \mathbb{P}$. \mathcal{K} together with the integral action can be written in an IO form

$$\mathcal{K}_I : \left(1 + \sum_{i=1}^{n_{\text{Ka}}} a_{\text{KI}i} q^{-i}\right) (1 - q^{-1}) u(k) = \sum_{j=0}^{n_{\text{Kb}}} b_{\text{KI}j} q^{-j} e(k), \quad (2)$$

or $\mathcal{A}_{\text{KI}}(q^{-1})u(k) = \mathcal{B}_{\text{KI}}(q^{-1})e(k)$, where $e(k) = r(k) - y(k)$; let $\mathcal{A}_{\text{KI}}(q^{-1}) = 1 + \sum_{i=1}^{n_{\text{Ka}}} a_{\text{KI}i} q^{-i}$ and $b_{\text{KI}0} = 0$. The closed-loop behavior of the system shown in Fig. 1 can be described implicitly in a so-called LPV kernel representation, see [7]:

$$\begin{bmatrix} \bar{A}(p_k) & -\bar{B}(p_k) & 0 \\ \bar{B}_{\text{K}} & \bar{A}_{\text{KI}} & -\bar{B}_{\text{K}} \end{bmatrix} \zeta(k) = D(p_k)\zeta(k) = 0, \quad (3)$$

where $\bar{A} = [1 \ a_1 \dots a_{n_{\text{dy}}}]$, $\bar{B} = [0 \ b_1 \dots b_{n_{\text{du}}}]$ are matrix valued functions, $\bar{A}_{\text{KI}} = [1 \ a_{\text{KI}1} \dots a_{\text{KI}n_{\text{du}}}]$, $\bar{B}_{\text{K}} = [0 \ b_{\text{KI}1} \dots b_{\text{KI}n_{\text{dy}}}]$ with $n_{\text{dy}} = \max(n_a, n_{\text{Kb}})$, $n_{\text{du}} = \max(n_b, n_{\text{Ka}} + 1)$ and $\zeta(k) = [y_{[k, k-n_{\text{dy}}]}^{\top} \ u_{[k, k-n_{\text{du}}]}^{\top} \ r_{[k, k-n_{\text{dy}}]}^{\top}]^{\top}$ with dimension $n_{\zeta} = 2n_{\text{dy}} + n_{\text{du}} + 3$. Based on a choice of a latent variable $x(k) = \Pi_1 \zeta(k)$ with dimension $n_x = 2n_{\text{dy}} + n_{\text{du}}$ for the closed-loop system (3), where $\Pi_1 = \text{diag}(\Pi_{1y}, \Pi_{1u}, \Pi_{1y})$, $\Pi_{1y} = [0 \ I_{n_{\text{dy}}}]$, $\Pi_{1u} = [0 \ I_{n_{\text{du}}}]$, it holds that $x(k+1) = \Pi_2 \zeta(k)$, with $\Pi_2 = \text{diag}(\Pi_{2y}, \Pi_{2u}, \Pi_{2y})$, $\Pi_{2y} = [I_{n_{\text{dy}}} \ 0]$, $\Pi_{2u} = [I_{n_{\text{du}}} \ 0]$, $\Pi_{iy} \in \mathbb{R}^{n_{\text{dy}} \times (n_{\text{dy}}+1)}$ and $\Pi_{iu} \in \mathbb{R}^{n_{\text{du}} \times (n_{\text{du}}+1)}$, $i = 1, 2$. Then, a sufficient condition for asymptotic stability in the Lyapunov sense and \mathcal{L}_2 -performance of the closed-loop system in Fig. 1 can be derived as shown in [7]. Consequently, the controller \mathcal{K} in Fig. 1 can be designed.

III. LPV-MPC SCHEME

Next, the proposed MPC technique is described under the temporary assumption that the future trajectory of p over the prediction horizon is available. This assumption will be relaxed later based upon a robust characterization of the future variations.

First a prediction equation, used for the MPC formulation, is required to express prediction of the future output sequence based on the past measurements generated by (1). The LPV system represented by (1) has an *infinite impulse response* (IIR) representation in the form $y(k) =$

$\sum_{i=0}^{\infty} h_i(p_{[k, k-i]})u(k-i)$, where $h_i(\cdot)$ are the Markov coefficients of the LPV system. Let us introduce the short form $h_i(k) = h_i(p_{[k, k-i]})$. Using (1), the Markov coefficients can be computed recursively as

$$h_i(k) = \begin{cases} b_i(p_k) - \sum_{j=1}^{\min(i, n_a)} a_j(p_k)h_{i-j}(k-j), & i \leq n_b \\ - \sum_{j=1}^{\min(i, n_a)} a_j(p_k)h_{i-j}(k-j) & \text{else.} \end{cases}$$

In case of no additional disturbances, given $p_{[k, k+N]}$ and $u_{[k, k+N-1]}$, the future output sequence of the system can be computed by $y(k+j) = \theta^{\top}(k+j)\phi(k) + \sum_{i=0}^j h_i(k+j)u(k+j-i)$, $j \in \mathbb{I}_1^N$, where N is the prediction horizon, $\phi(k) \in \mathbb{R}^{n_a+n_b}$ is the regressor vector given as $\phi(k) = [y(k-1) \ \dots \ y(k-n_a) \ u(k-1) \ \dots \ u(k-n_b)]^{\top}$ and $\theta(k+j) \in \mathbb{R}^{n_a+n_b}$ is computed recursively via

$$\theta(k+j) = \vec{I}^j \bar{\theta}(k+j) - \sum_{i=1}^{\min(j, n_a)} a_i(p(k+j))\theta(k+j-i), \quad (4)$$

$j \in \mathbb{I}_1^N$, with $\bar{\theta}(\tau) = [-a_1(p_{\tau}) \ \dots \ -a_{n_a}(p_{\tau})b_1(p_{\tau}) \ \dots \ b_{n_b}(p_{\tau})]^{\top}$, $\tau = k+j$ and $\vec{I}^j = \text{diag}(I_{n_a}^j, I_{n_b}^j)$, where $I_{n_a}^j \in \mathbb{R}^{n_a}$, $I_{n_b}^j \in \mathbb{R}^{n_b}$ are obtained by shifting identity matrices of the corresponding dimensions with j column to the right. In order to provide a controller with integral action, an incremental IO model can be defined by introducing a new input signal as $v(k) = u(k) - u(k-1)$. Therefore, the LPV model can be rewritten as

$$\mathcal{G}_I : \mathcal{A}(q^{-1}, p_k)y(k) = \mathcal{B}(q^{-1}, p_k)(v(k) + u(k-1)). \quad (5)$$

Now, given the current value and the future trajectory of the scheduling variable and the input of the system, the future output of \mathcal{G}_I in (5) can be computed as follows:

$$y(k+j) = \tilde{\theta}^{\top}(k+j)\phi(k) + \sum_{i=0}^j \sum_{l=0}^i h_l(k+j)v(k+j-i), \quad (6)$$

$j \in \mathbb{I}_1^N$, where the vector $\tilde{\theta}(k+j) \in \mathbb{R}^{n_a+n_b}$ is computed as in (4) except its $(n_a+1)^{\text{th}}$ element that is given by $\tilde{\theta}_{n_a+1}(k+j) = \theta_{n_a+1}(k+j) + \sum_{i=0}^j h_i(k+j)$. Note that this is the coefficient of $u(k-1)$ in (6). Therefore, the key prediction equation for \mathcal{G}_I with $h_0(k) \equiv 0$ is given by

$$y_{[k, k+N]} = H(k)v_{[k, k+N-1]} + \Theta(k)\phi(k), \quad (7)$$

where $y_{[k+1, k+N]} \in \mathbb{R}^N$ is a vector of future values of the output, $H(k) \in \mathbb{R}^{N \times N}$ is a lower triangular Toeplitz matrix with the Markov coefficients of the system:

$$H(k) = \begin{bmatrix} h_1(k+1) & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^N h_i(k+N) & \dots & h_1(k+N) \end{bmatrix}, \Theta(k) = \begin{bmatrix} \tilde{\theta}_1(k+1) \\ \tilde{\theta}_2(k+2) \\ \vdots \\ \tilde{\theta}_N(k+N) \end{bmatrix} \quad (8)$$

$\Theta(k) \in \mathbb{R}^{N \times (n_a+n_b)}$; H and Θ are functions of $p_{[k, k+N]}$.

Next, the proposed LPV-MPC scheme with stability guarantees is formulated. Define the cost function

$$V_N = \sum_{i=0}^{N_e} \underbrace{\mu_i e^2(k+i-n_{dy})}_{\ell_e} + \sum_{j=1}^{N_v} \underbrace{\rho_j v^2(k+j-n_{du})}_{\ell_v} + V_f, \quad (9)$$

where $V_N = V_N(x_0, e_0, v[k, k+N-1], r[k+1, k+N], p[k, k+N])$ with $x_0 \in \mathbb{R}^{n_x}$ is the state vector at instant k and $e_0 = e(k) = r(k) - y(k)$ is the error signal as shown in Fig. 1, $V_f = V_f(x(k+N+1))$ defines a terminal cost with $x(k+N+1) = [y_{[k+N, k+N+1-n_{dy}]}^\top u_{[k+N, k+N+1-n_{du}]}^\top r_{[k+N, k+N+1-n_{dy}]}^\top]^\top$ (to simplify the notation, we drop the arguments) and $N_e = N + n_{dy} - 1$, $N_v = N + n_{du} - 1$, $N_v \leq N_e$. The terminal cost penalizes the states of the closed-loop system at the end of the prediction horizon, whereas the stage cost given by $\ell(e, v) = \ell_e + \ell_v$ specifies the desired control performance via arbitrary values for $N, \mu_i > 0$ and $\rho_j > 0$, see (9). The proposed MPC control problem is given by

$$\min_{v[k, k+N-1]} V_N, \quad (10a)$$

$$\text{subject to } u(k+i) \in \mathbb{U}, \quad \forall i \in \mathbb{I}_0^{N-1}, \quad (10b)$$

$$x(k+N+1) \in \mathbb{X}_f, \quad (10c)$$

with the LPV system dynamics represented by (5), where \mathbb{U} is a compact input set constraint defined as

$$\mathbb{U} := \{u(k) \in \mathbb{R} \mid u_{\min} \leq u(k) \leq u_{\max}\} \quad (11)$$

and $\mathbb{X}_f \in \mathbb{R}^{n_x}$ specifies a terminal set constraint to enforce the states at the end of the prediction horizon to lie in \mathbb{X}_f . Note that the constraints (10b-c) are implicit constraints on $v[k, k+N-1]$. The MPC control law is obtained by solving (10) at each sampling time instant and applying it to the system in a receding horizon manner.

Let the reference trajectory r be a piecewise constant signal with target steady-state values defined by the following set $\mathcal{R}_s := \{r_s \in \mathbb{R} \mid r_{\min} \leq r_s \leq r_{\max}\}$, where r_{\min}, r_{\max} are, respectively, the minimum and maximum admissible values of the reference trajectory. For a target steady-state value $r_s \in \mathcal{R}_s$ so that $y = r_s$, let $u_s \in \mathbb{U}$ be the corresponding steady-state input, which can be computed at a frozen scheduling variable $p_s \in \mathbb{P}$ by solving the linear equation $(1 + \sum_{i=1}^{n_a} a_i(p_s))r_s = (\sum_{i=1}^{n_b} b_i(p_s))u_s$. Therefore, the associated target steady-state of the system becomes $x_s = [r_s \ \dots \ r_s \ u_s \ \dots \ u_s \ r_s \ \dots \ r_s]^\top$. Then, let $\mathcal{U}_s := \{u_s \in \mathbb{U} \mid (1 + \sum_{i=1}^{n_a} a_i(p_s))r_s = (\sum_{i=1}^{n_b} b_i(p_s))u_s, \forall (r_s, p_s) \in \mathcal{R}_s \times \mathbb{P}\}$ define the set of all steady-state inputs u_s corresponding to all $r_s \in \mathcal{R}_s$, for all $p_s \in \mathbb{P}$. Moreover, let the set of all target steady states x_s associated with all $r_s \in \mathcal{R}_s$ be given as $\mathcal{X}_s := \{x_s \in \mathbb{R}^{n_x} \mid x_s = [r_s \ \dots \ r_s \ u_s \ \dots \ u_s \ r_s \ \dots \ r_s]^\top, \forall (r_s, u_s) \in \mathcal{R}_s \times \mathcal{U}_s\}$. Now, consider the following assumptions:

- A.1 There is no model error and no disturbances and the future trajectories of both r and p are known.
- A.2 The reference trajectory r is a piecewise constant signal, then, for any target output $y(k+N) = r_s$, $r_s \in \mathcal{R}_s$, all steady-states of the system, i.e., $x(k+N+1) = x_s$ (also corresponding to $u(k+N-1) = u_s$) should belong to the terminal set \mathbb{X}_f , namely, $\mathcal{X}_s \subset \mathbb{X}_f$.

A.3 The function $V_f(x(k))$ is continuous, positive definite for all $x(k)$ and $V_f(0) = 0$.

A.4 The set \mathbb{X}_f is closed.

In general, the closed-loop system can be asymptotically stabilized by the MPC control law $\kappa_N(\cdot)$ if there exists a terminal feedback controller $\kappa_f(x(k))$ such that the following conditions are satisfied [3]:

C.1 $V_f(\cdot)$ is a Lyapunov function on the terminal set \mathbb{X}_f under the controller $\kappa_f(\cdot)$ such that

$$V_f(x(k+1)) - V_f(x(k)) \leq -\ell(x(k), \kappa_f(x(k))) < 0, \quad (12)$$

$$\forall x(k) \in \mathbb{X}_f, \forall p_k \in \mathbb{P}, \forall k > N.$$

C.2 The set \mathbb{X}_f is positively invariant under the controller $\kappa_f(\cdot)$, i.e., if $x(k) \in \mathbb{X}_f$, then $x(k+1) \in \mathbb{X}_f, \forall p \in \mathbb{P}$.

C.3 $\kappa_f(\cdot) \in \mathbb{U}, \forall x \in \mathbb{X}_f$ (input constraint is satisfied in \mathbb{X}_f).

Under these conditions, the optimal cost function V_N^* is a Lyapunov function for the closed-loop system and its domain of attraction is the set of initial states x_0 , initial errors e_0 and future reference and scheduling trajectories, $r_{[k+1, k+N]}$, $p_{[k+1, k+N]}$, respectively, where the optimization problem is feasible; let such domain of attraction be denoted by \mathcal{X}_N . The invariance condition imposed on the terminal region makes the optimization problem feasible if the initial values are in the domain of attraction, c.f., [3] for more details.

Next, we show how $V_f(\cdot)$ and \mathbb{X}_f can be chosen to satisfy the above conditions. In terms of (12) the function $V_f(\cdot)$ can be chosen to be an upper bound on the value function of the unconstrained infinite horizon cost of the system states starting from \mathbb{X}_f and controlled by the terminal controller $\kappa_f(\cdot)$ [3]. Thus, we choose

$$V_f(x(k+N+1)) \geq \sum_{i=N+1}^{\infty} (\tilde{\mu} e^2(k+i-1) + \tilde{\rho} v^2(k+i-1)), \quad (13)$$

for all $x \in \mathbb{X}_f, \forall p \in \mathbb{P}$ where $\tilde{\mu} > 0$ and $\tilde{\rho} > 0$ are constants. To verify (13), we need to satisfy

$$\begin{aligned} & V_f(x(k+i+1)) - V_f(x(k+i)) \\ & \leq -(\tilde{\mu} e^2(k+i-1) + \tilde{\rho} v^2(k+i-1)) < 0, \end{aligned} \quad (14)$$

for all $e(k+i-1) \neq 0, v(k+i-1) \neq 0, i \geq N+1$ and $\forall p \in \mathbb{P}$. Then, summing (14) from $i = N+1$ to ∞ gives

$$\begin{aligned} & V_f(x(\infty)) - V_f(x(k+N+1)) \\ & \leq - \sum_{i=N+1}^{\infty} (\tilde{\mu} e^2(k+i-1) + \tilde{\rho} v^2(k+i-1)). \end{aligned} \quad (15)$$

If (14) is satisfied, then with Assumption A.3, we have $V_f(x(k+N+1)) \geq V_f(x(\infty))$, hence, (13) holds. Therefore, Condition C.1 can be verified, if there exists a function $V_f(\cdot)$ that satisfies Assumption A.3 along with (14). Note that such a $V_f(\cdot)$ can serve as a Lyapunov function for the closed-loop system shown in Fig. 1. On the other hand, this also implies the existence of a control law $\kappa_f(\cdot)$ that can drive a state in \mathbb{X}_f into a steady-state point $x_s \in \mathbb{X}_f$, i.e., $\lim_{k \rightarrow \infty} \|x(k) - x_s\| = 0$. Therefore, we need to derive a controller such that (14) holds for all $i \geq N+1$, and

consequently, it guarantees that $x(k)$ converges to x_s . In other words, we employ (14) to design the controller $\kappa_f(\cdot)$, the existence of which implies that $V_f(\cdot)$ is a Lyapunov function for the closed-loop system. This suggests that $V_f(\cdot)$ could be a quadratic function as

$$V_f(x(k)) = x^\top(k)P_f x(k), \quad P_f = P_f^\top \succ 0, \quad (16)$$

$P_f \in \mathbb{R}^{n_x \times n_x}$. Then, based on (14), (16) and the application of the S-procedure and Finsler's Lemma, we have the following sufficient condition.

Theorem 1: The closed-loop system described by (3), is asymptotically internally stable and satisfies the \mathcal{L}_2 -performance constraint $\zeta^\top(k+i)Q\zeta(k+i) \geq 0$, where $Q = \text{diag}(Q_1, Q_2)$, $Q_1 = \text{diag}(-1, 0, \dots)$, $Q_2 = \text{diag}(\gamma^2, 0, \dots)$, $\gamma > 0$, if there exist a controller $\kappa_f(\cdot)$, $F \in \mathbb{R}^{n_\xi \times 2}$ and

$$S = \begin{bmatrix} S_1 & 0 & -S_1 \\ 0 & S_2 & 0 \\ -S_1 & 0 & S_1 \end{bmatrix}, \quad S_1 = \begin{bmatrix} \tilde{\mu} & 0 \\ 0 & 0 \end{bmatrix}, \quad S_2 = \begin{bmatrix} \tilde{\rho} & -\tilde{\rho} & 0 \\ -\tilde{\rho} & \tilde{\rho} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$\tilde{\mu} > 0$, $\tilde{\rho} > 0$, where $S_1 \in \mathbb{R}^{n_{dy} \times n_{dy}}$ and $S_2 \in \mathbb{R}^{n_{du} \times n_{du}}$ such that

$$P_f = P_f^\top \succ 0, \quad (17a)$$

$$\Pi_2^\top P_f \Pi_2 - \Pi_1^\top (P_f + S) \Pi_1 + Q + FD(p) + D^\top(p)F^\top < 0 \quad (17b)$$

hold for all $p \in \mathbb{P}$.

The proof is omitted due to lack of space. Therefore, existence of the controller $\kappa_f(\cdot)$ satisfying (17a-b) for all $p \in \mathbb{P}$ guarantees that $V_f(x(k))$ is a Lyapunov function satisfying (14), which implies Condition C.1. The inequalities (17a-b) can be solved as a feasibility optimization problem with bilinear matrix inequality (BMI) constraints using the approach in [7].

To guarantee asymptotic internal stability of the proposed MPC controller, we further need to verify Conditions C.2 and C.3. For C.2, it is required to specify \mathbb{X}_f to be a positive invariant set with the controller $\kappa_f(\cdot)$ [3]. One way to achieve this is to choose \mathbb{X}_f as a sub-level set of $V_f(\cdot)$ [3], as follows:

$$\mathbb{X}_f := \{x(k) \in \mathbb{R}^{n_x} \mid x^\top(k)P_f x(k) \leq \alpha\}, \quad \alpha > 0. \quad (18)$$

By this choice, \mathbb{X}_f is an ellipsoidal terminal set constraint, which can be enlarged by α . It is positive invariant for the closed-loop system with the controller $\kappa_f(\cdot)$ if $K_f \mathbb{X}_f \subset \mathbb{U}$. This provides that condition C.3 holds. Usually, the constant α is chosen as the largest value such that $K_f x \in \mathbb{U}$, $\forall x \in \mathbb{X}_f$.

In the proposed MPC scheme, we follow the same strategy and we further restrict \mathbb{X}_f to be an ellipsoidal terminal set \mathbb{X}_f . The above described conditions can be attained by solving the convex optimization problem

$$\max_{\tilde{\alpha}} \tilde{\alpha} \quad \text{subject to} \quad -u_{\max} \leq \tilde{\alpha} \|P_f^{-1} K_f\| \leq u_{\max}, \quad (19)$$

where $\tilde{\alpha} = \sqrt{\alpha}$. Hence, \mathbb{X}_f in (18) can be redefined as

$$\mathbb{X}_f := \{x(k) \in \mathbb{R}^{n_x} \mid x^\top(k)P_f x(k) \leq \alpha_m\}, \quad (20)$$

where α_m is the solution of (19). To ensure feasibility of the proposed MPC, the terminal set \mathbb{X}_f should also satisfy Assumption A.2. This can be satisfied by verifying that \mathbb{X}_f in

(20) contains all target steady-states $x_s \in \mathcal{X}_s$ of the system, i.e., $\mathbb{X}_f \supset \mathcal{X}_s$. Next, we show how this can be fulfilled. Introduce

$$\alpha_s = \max_{x_s \in \mathcal{X}_s} x_s^\top P_f x_s, \quad (21)$$

then if $\alpha_s \leq \alpha_m$, see (20), we can verify that $\mathcal{X}_s \subset \mathbb{X}_f$. It can be shown that the optimization problem (21) is a constrained nonlinear optimization problem, which can be solved using any gradient-based optimization method.

Finally, we can summarize the previous results.

Theorem 2: Suppose that

- (a) Assumptions A.1, A.2, A.3 and A.4 are satisfied, and
- (b) there exists a terminal cost, given by (16), such that (17) is satisfied and a terminal set, given by (20), such that $\alpha_m \geq \alpha_s$, where α_m and α_s are scalars being the solutions of the optimization problems (19) and (21), respectively,

then, conditions C.1, C.2 and C.3 are satisfied. Consequently, the MPC controller derived by solving (10) asymptotically internally stabilizes the system (5) for all initial values of x_0 , e_0 , $r_{[k+1, k+N]}$ and $p_{[k, k+N]}$ in the set \mathcal{X}_N .

Next, the MPC optimization problem (10) is represented as an optimization problem with LMI constraints, for which LMI solvers can be utilized. Moreover, this is the key step to formulate the robust LPV-MPC scheme in the next section. The cost function (9) can be rewritten as follows:

$$V_N = V_0 + (*)^\top M (r_{[k+1, k+N-1]} - y_{[k+1, k+N-1]}) + (*)^\top R v_{[k, k+N-1]} + (*)^\top \tilde{P}_f \tilde{x}(k+N+1), \quad (22)$$

where V_0 is a constant term given by $\sum_{i=0}^{n_{dy}} \mu_i e^2(k+i - n_{dy}) + \sum_{j=1}^{n_{du}-1} \rho_j v^2(k+j - n_{du})$, $M = \text{diag}\{\mu_{n_{dy}+1}, \mu_{n_{dy}+2}, \dots, \mu_{N_e}\} \in \mathbb{R}^{(N-1) \times (N-1)}$, $R = \text{diag}\{\rho_{n_{du}}, \rho_{n_{du}+1}, \dots, \rho_{N_v}\} \in \mathbb{R}^{N \times N}$, $\tilde{x}(k+N+1) = T_x^{-1} x(k+N+1)$ with $T_x \in \mathbb{R}^{n_x \times n_x}$ being a state transformation $T_x = \text{diag}(T_{x1}, T_{x2}, T_{x1})$, $T_{x1} \in \mathbb{R}^{n_{dy} \times n_{dy}}$, $T_{x2} \in \mathbb{R}^{n_{du} \times n_{du}}$ are anti-diagonal matrices with all nonzero entries equal to one, which rearranges the state vector as \tilde{x} , and $\tilde{P}_f = T_x^\top P_f T_x$.

Now, given $p_{[k, k+N]}$ and $r_{[k+1, k+N]}$, the optimization problem (10) can be rewritten as

$$\min \beta \quad (23a)$$

$$\beta, v_{[k, k+N-1]}$$

$$\text{subject to} \quad V_N \leq \beta, \quad (23b)$$

$$u(k+i) \in \mathbb{U}, \quad \forall i \in \mathbb{I}_0^{N-1}, \quad (23c)$$

$$x(k+N+1) \in \mathbb{X}_f. \quad (23d)$$

Substituting (8) into (23b), then, applying Schur complement provides the LMI form of (23b) as

$$\begin{bmatrix} M^{-1} & 0 & 0 & r_{[k+1, k+N-1]} - H_{1, N-1}(k)v_{[k, k+N-1]} \\ & 0 & R^{-1} & 0 \\ & & & -\Theta_{1, N-1}(k)\phi(k) \\ & & & v_{[k, k+N-1]} \\ & & & \dots \\ & & & H_{N+1-n_{dy}, N}(k)v_{[k, k+N-1]} \\ & & & + \Theta_{N+1-n_{dy}, N}(k)\phi(k) \\ & 0 & 0 & \tilde{P}_f^{-1} \\ & & & T_u v_{[k, k+N-1]} + \mathbf{1}_{n_{du}} u(k-1) \\ & & & r_{[k+N+1-n_{dy}, k+N]} \\ & *^\top & *^\top & *^\top \\ & & & \beta - V_0 \end{bmatrix} \succeq 0 \quad (24)$$

where $1_x = [1 \ 1 \ \dots \ 1]^\top \in \mathbb{R}^x$ and $T_u \in \mathbb{R}^{n_{du} \times Nc}$ is given by

$$T_u = \begin{bmatrix} T_{u1} & T_{u2} \\ 1_{N-n_{du}+1}^\top & 1_{n_{du}-1}^\top \end{bmatrix}, \quad T_{u1} \in \mathbb{R}^{(n_{du}-1) \times (Nc-n_{du}+1)}, \\ T_{u2} \in \mathbb{R}^{(n_{du}-1) \times (n_{du}-1)}$$

with $T_{u1} \in \mathbb{R}^{(n_{du}-1) \times (Nc-n_{du}+1)}$ being a matrix whose entries are all one and $T_{u2} \in \mathbb{R}^{(n_{du}-1) \times (n_{du}-1)}$ is a lower triangular matrix whose non-zero entries are one. Next, the control input constraint (23c) can be written as

$$Ev_{[k,k+N-1]} \preceq c, \quad E = \begin{bmatrix} T \\ -T \end{bmatrix}, \quad c = \begin{bmatrix} 1_N u_{\max} - 1_N u(k-1) \\ -1_N u_{\min} + 1_N u(k-1) \end{bmatrix} \quad (25)$$

with $T \in \mathbb{R}^{N \times N}$ a lower triangular matrix whose non-zero entries are all one. Finally, the terminal set constraint (23d) using (20) can be written as an LMI:

$$\begin{bmatrix} \tilde{P}_f^{-1} & & & & & \\ & H_{N+1-n_{dy},N}(k)v_{[k,k+N-1]} & & & & \\ & + \Theta_{N+1-n_{dy},N}(k)\phi(k) & & & & \\ & T_u v_{[k,k+N-1]} + 1_{n_{du}} u(k-1) & & & & \\ & & r_{[k+N+1-n_{dy},k+N]} & & & \\ *^\top & & & \alpha_m & & \end{bmatrix} \succeq 0. \quad (26)$$

Therefore, solving the MPC problem (10) for an LPV-IO model can be presented as an optimization problem with LMI constraints as follows: At any time instant k , given $x_0, e_0, p_{[k,k+N]}, r_{[k+1,k+N]}, \tilde{P}_f, \alpha_m$ and appropriate values for N, M and R , solve

$$\min_{\beta, v_{[k,k+N-1]}} \beta \quad \text{subject to} \quad (24), (25), (26). \quad (27)$$

This problem is solved online at each time instant k , where N, M, R are tuning parameters chosen by the user. Also, \tilde{P}_f and α_m should be obtained offline by solving the feasibility problem (17) and the optimization problem (19), respectively.

IV. ROBUST LPV-MPC SCHEME

We propose in this section an MPC scheme based on the above formulation to design a robust MPC controller for LPV-IO models in which only $p(k)$ is needed to be known at every control cycle k while the future values of p , i.e., $p(k+1), p(k+2), \dots, p(k+N)$, are uncertain. Therefore, in (22), the worst-case cost over all possible future scheduling values is considered. We then employ the full block multipliers introduced in [8], to provide an optimization problem with a finite number of LMI constraints, some of which are required to be verified only at the vertices of \mathbb{P} . Moreover, bounds on the rate of variation of p will be exploited to verify these LMIs at the vertices of a subset of \mathbb{P} , which can reduce the conservatism of the design. The robust MPC design introduced here is based on the full block \mathcal{S} -procedure (Theorem 8 in [8]), which can be used to convert an uncertain matrix inequality to a finite set of inequalities using full block multipliers.

At a sampling instant k , if each of the constraints (24) and (26) can be represented as a certain quadratic form, the full block \mathcal{S} -procedure can be used to transform each of them into a form which enables solving the optimization problem (27) without requiring the future values of p . The

first step is to formulate each of the constraints (24) and (26), respectively, as

$$F^\top(p)W_F(k)F(p) \succeq 0, \quad (28a)$$

$$G^\top(p)W_G(k)G(p) \succeq 0, \quad (28b)$$

where $F(p) \in \mathbb{R}^{n_F \times (2N+n_x)}$, $n_F = 4N + n_x + n_a + n_b + n_{du} + 1$, and $G(p) \in \mathbb{R}^{n_G \times (n_x+1)}$, $n_G = N + n_x + n_a + n_b + n_{dy} + n_{du} + 2$ are matrix valued functions of H and Θ and $W_F \in \mathbb{R}^{n_F \times n_F}$, $W_G \in \mathbb{R}^{n_G \times n_G}$ are matrix valued functions of r, v . These matrices are not given here due to space restrictions. As a consequence, (28a) and (28b) can replace (24) and (26), respectively, in the optimization problem (27). Next, we transform both $F(p)$ and $G(p)$ into an upper LFT form as follows:

$$F(p) = \Delta_F \star \begin{bmatrix} \vdots & \vdots \\ F_{11} & F_{12} \\ \vdots & \vdots \\ F_{21} & F_{22} \\ \vdots & \vdots \end{bmatrix}, \quad G(p) = \Delta_G \star \begin{bmatrix} \vdots & \vdots \\ G_{11} & G_{12} \\ \vdots & \vdots \\ G_{21} & G_{22} \\ \vdots & \vdots \end{bmatrix}, \quad (29)$$

such that

$$\Delta_F = \text{diag}\{p_1 I_{r_{F1}}, \dots, p_{n_p} I_{r_{F_{n_p}}}\}, \quad \Delta_F \in \mathbf{\Delta}_F, \quad (43)$$

$$\Delta_G = \text{diag}\{p_1 I_{r_{G1}}, \dots, p_{n_p} I_{r_{G_{n_p}}}\}, \quad \Delta_G \in \mathbf{\Delta}_G \quad (44)$$

where

$$\mathbf{\Delta}_F(k) = \{\Delta_F(k) \in \mathbb{R}^{n_{\Delta_F} \times n_{\Delta_F}} \mid \\ \underline{p}_i(k) \leq p_i \leq \bar{p}_i(k), \quad i = 1, 2, \dots, n_p\}, \quad (45a)$$

$$\mathbf{\Delta}_G(k) = \{\Delta_G(k) \in \mathbb{R}^{n_{\Delta_G} \times n_{\Delta_G}} \mid \\ \underline{p}_i(k) \leq p_i \leq \bar{p}_i(k), \quad i = 1, 2, \dots, n_p\}, \quad (45b)$$

$n_{\Delta_F} = \sum_{i=1}^{n_p} r_{F_i}$, $n_{\Delta_G} = \sum_{i=1}^{n_p} r_{G_i}$, $\bar{p}_i(k) = \max(N \cdot dp_{\max i} + p_i(k), p_{\min i})$ and $\underline{p}_i(k) = \min(N \cdot dp_{\min i} + p_i(k), p_{\max i})$.

Now, if the LFTs (29) are well-posed, i.e., $(I - F_{11}\Delta_F)^{-1}$ and $(I - G_{11}\Delta_G)^{-1}$ are well-defined for all $p \in \mathbb{P}$, then we can apply the results of [8] to the conditions (28a-b). Therefore, at the sampling instant k , given $x_0, e_0, r_{[k+1,k+N]}$, the parameters \tilde{P}_f and α_m , which can be computed offline, and the design parameters N, M and R , the optimization problem (27) associated with the robust MPC design considered here can be formulated using the full block \mathcal{S} -procedure, which is not given here due to space restrictions. Next, as \mathbb{P} is a convex polytope and the blocks Δ_F and Δ_G have linear dependence on p , the LMIs of the robust MPC design are only required to be solved at the vertices of \mathbb{P} .

V. NUMERICAL EXAMPLE

In this section, the performance of the proposed MPC scheme for LPV-IO models is demonstrated on a simulation example. The system to be controlled is an unstable 2nd-order LPV system represented in LPV-IO form according to (1) as

$$a_1(p_k) = -0.2 + 0.7p(k), \quad a_2(p_k) = 0.7 + 0.4p(k), \\ b_2(p_k) = 1.6 + 2.8p(k), \quad b_1(p_k) = 3.4 + 1.2p(k),$$

and $b_0(p_k) = 0$, with $\mathbb{P} = [0, 1]$, $u_{\max} = 2$ and the reference to be tracked is given in advance as shown in Fig. 2a. In

order to find $V_f(\cdot)$, the feasibility problem defined by (17a-b) has been solved to obtain the matrix $P_f \in \mathbb{R}^{6 \times 6}$ and the terminal controller $\kappa_f(\cdot)$, with $n_{K_a} = 1$ and $n_{K_b} = 2$. Next, the ellipsoidal terminal set \mathbb{X}_f in (20) is constructed offline by computing the value of the parameter α_m by solving (19) and $\mathbb{X}_f \supset \mathcal{X}_s$ has been verified by solving (21). Given P_f and α_m , which parameterize $V_f(\cdot)$ and \mathbb{X}_f , respectively, the proposed MPC schemes can be applied. The tuning parameters have been chosen as $\rho = 480$, $\mu = 600$, $N = 6$ and $N_c = 5$ to achieve some desired control objectives. Then, the robust LPV-MPC scheme has been implemented by solving its associated optimization problem at each sampling instant k to obtain the online optimal control law. The resulting control structure has been validated via a simulation study with a scheduling trajectory depicted in Fig. 2c. Stability of the closed-loop system over the entire operating region and feasibility of the optimization problem at all sampling instants have been achieved by the MPC design. The evolution of the output and the control input of the closed-loop system with the MPC controller are shown in Figs. 2a and 2b, respectively, which demonstrate a satisfactory tracking capability at different operating conditions besides of the integral action without violating the control input bounds. For comparison purpose, the proposed LPV-MPC scheme has been implemented by solving (27) where the future values $p_{[k+1, k+N]}$ have been provided at each instant k (p -anticipating LPV-MPC). Figures 2a and 2b show that both MPC schemes provide almost the same performance, which demonstrates the capability of the proposed robust MPC scheme.

VI. CONCLUSIONS

In this paper an MPC scheme has been proposed to control LPV-IO models subject to input constraints. The proposed LPV-MPC scheme characterizes a robust strategy to counteract the worst-case possible uncertainties of the scheduling variations. To guarantee closed-loop asymptotic stability, an appropriate quadratic terminal cost is added to the quadratic finite horizon cost function of the online MPC optimization problem and an ellipsoidal terminal set constraint is included. The full block S-procedure with an LFT formulation of the parameter dependent inequality constraints as well as information about the rate of change of p have been used to develop a robust MPC scheme with low conservativeness. The online optimization problem involved in such a scheme is convex and can be solved by semi-definite programming tools to compute the optimal control action at each sampling instant. Using a simulation example, the performance of the proposed scheme has been demonstrated.

ACKNOWLEDGMENT

This publication was made possible by the NPRP grant (No. 5-574-2-233) from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

The first author expresses his sincere thanks to Assiut University, Egypt, for supporting him in this work.

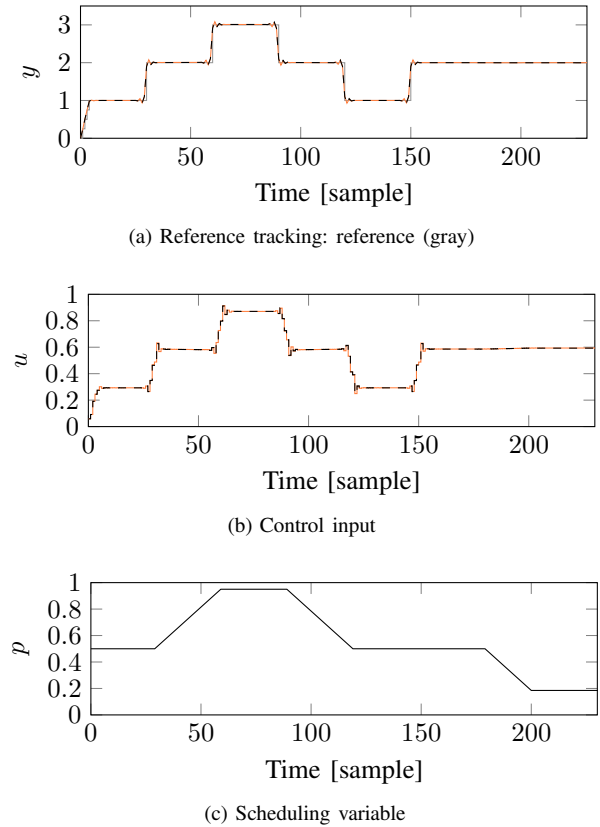


Fig. 2: Closed-loop performance with the robust (red) and the p -anticipating (black-dashed) LPV-MPC schemes.

REFERENCES

- [1] R. Tóth, H. Abbas, and H. Werner, "On the state-space realization of LPV input-output models: Practical approaches," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 1, pp. 139–153, 2012.
- [2] A. Bachnas, R. Tóth, J. Ludlage, and A. Mesbah, "A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study," *Journal of Process Control*, vol. 24, no. 4, pp. 272–285, 2014.
- [3] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [4] Y. Lu and Y. Arkun, "Quasi-min-max MPC algorithms for LPV systems," *Automatica*, vol. 36, no. 4, pp. 527–540, 2000.
- [5] L. Wang and P. Young, "An improved structure for model predictive control using non-minimal state space realization," *Journal of Process Control*, vol. 16, pp. 355–371, 2006.
- [6] J. Dong, B. Kulcsár, J. van Wingerden, and M. Verhaegen, "Closed-loop subspace predictive control for linear parameter-varying systems (I) - the nominal case," in *Proceedings of the European Control Conference*, Budapest, Hungary, 2009, pp. 2009–2014.
- [7] S. Wollnack, H. Abbas, H. Werner, and R. Tóth, "Fixed-structure LPV controller synthesis based on implicit input output representations," in *Proceedings of the 52nd IEEE Conference on Decision and Control*, Florence, Italy, 2013, pp. 2103–2108.
- [8] C. Scherer, "LPV control and full block multipliers," *Automatica*, vol. 27, no. 3, pp. 325–485, 2001.