A COMPUTATIONALLY IMPROVED APPROACH FOR DYNAMIC OUTPUT FEEDBACK ROBUST MODEL PREDICTIVE CONTROL

José Roberto Colombo Junior*, Roberto Kawakami Harrop Galvão*, Edvaldo Assunção†

*ITA - Instituto Tecnológico de Aeronáutica
Divisão de Engenharia Eletrônica
Departamento de Sistemas e Controle
Praça Marechal Eduardo Gomes, 50, 12228-900, Vila das Acácias, São José dos Campos, SP, Brazil

†UNESP - Universidade Estadual Paulista
Departamento de Engenharia Elétrica, Lab. de Pesquisa em Controle
Av. José Carlos Rossi, 1370, 15835-000, Ilha Solteira, SP, Brasil

Emails: colombojrj@gmail.com, kawakami@ita.br, edvaldo@dee.feis.unesp.br

Resumo— Este artigo propõe uma simplificação na formulação LMI (do inglês "Linear Matrix Inequality") de uma lei de controle preditivo robusta com realimentação de saída para sistemas lineares incertos com descrição politópica. A simplificação proposta conduz a uma formulação LMI equivalente, preservando assim as garantias originais de atendimento das restrições e estabilidade robusta. O ganho computacional resultante é ilustrado em um exemplo numérico. Mais especificamente, um número reduzido de iterações é requerido para obter uma solução factível para o problema de otimização.

Palavras-chave— Realimentação dinâmica de saída, controle preditivo robusto, desigualdades matriciais lineares, carga computacional.

Abstract— This paper proposes a simplification in the LMI (Linear Matrix Inequality) formulation of an output feedback robust model predictive control law for linear uncertain systems with polytopic description. The proposed simplification leads to an equivalent LMI formulation, thus retaining the original guarantees of constraint satisfaction and robust stability. The resulting computational gain is illustrated in a numerical example. More specifically, a reduced number of iterations is required to find a feasible solution for the optimization problem.

Keywords— Dynamic output feedback, robust model predictive control, linear matrix inequalities, computational workload.

1 Introduction

Robust Model Predictive Control (RMPC) is concerned with the receding-horizon control of uncertain systems subject to operational constraints (Bemporad and Garulli, 2000). A popular approach consists of minimizing an upper bound for an infinite horizon cost function, with model uncertainties expressed in terms of Linear Matrices Inequalities (LMIs), as in (Kothare et al., 1996), (Schuurmans and Rossiter, 2000), (Cuzzola et al., 2002) and (Huang et al., 2011). Such formulations typically employ full state feedback, which may be an inconvenience if only partial state information is available. In this case, state observers could be used to estimate the state variables that are not directly measured. However, the constraint satisfaction and stability guarantees may not hold if the controller and observer are designed separately (Wan and Kothare, 2002).

In order to retain these guarantees, a suitable Output Feedback Robust Model Predictive Control (OFRMPC) formulation is required. A common strategy is to employ a Dynamic Output Feedback Controller (DOFC) in an inner control loop, with the OFRMPC acting in the outer loop (Granado et al., 2003), (Lee and Park, 2007), (Ding et al., 2008), (Ding, 2010), (Li et al., 2013). With appropriate tuning (Colombo Junior et al., 2015), recent OFRMPC formulations such as that proposed in (Li et al., 2013) may achieve similar closed loop performance compared to the original state feedback RMPC approach (Kothare et al., 1996).

In this context, the present paper proposes a simplification of the LMI formulation presented in (Li et al., 2013) in order to reduce the computational effort required for on-line implementation of the OFRMPC control law. The proposed simplification leads to an equivalent LMI formulation, thus retaining the original guarantees of constraint satisfaction and robust stability.

The remainder of this paper is organized as follows. Section 2 describes the problem statement and OFRMPC formulation proposed in (Li et al., 2013). Section 3 brings the main contribution of the present work. Section 4 presents an illustrative simulation example. Finally, concluding remarks are presented in Section 5.

2 Output feedback robust model predictive control formulation

Consider the following polytopic uncertain system:

$$x(k+1) = Ax(k) + Bu(k) \tag{1}$$
$$y(k) = Cx(k) \tag{2}$$
$$(A,\ B,\ C) \in \Omega \tag{3}$$

where $x \in \mathbb{R}^{n_x}$ is the state vector, $u \in \mathbb{R}^{n_u}$ is the control input, $y \in \mathbb{R}^{n_y}$ is the measurable system output and Ω is an uncertainty polytope with known vertices $(A_i,\ B_i,\ C_i)$, where $A_i \in \mathbb{R}^{n_x \times n_x}$, $B_i \in \mathbb{R}^{n_x \times n_u}$ and $C_i \in \mathbb{R}^{n_y \times n_x}$, $i = 1, 2, ..., n_p$.

The OFRMPC formulation adopted in (Li et al., 2013) uses a DOFC given by

$$\hat{x}(k+1) = A_0\hat{x}(k) + B_0 y(k) + F_1\hat{x}(k), \tag{4}$$
$$u(k) = \mathrm{sat}(C_0\hat{x}(k) + D_0 y(k) + F_2\hat{x}(k)). \tag{5}$$

where “sat” is a ± 1 saturation function, $\hat{x}(k) \in \mathbb{R}^{n_x}$ represents the estimated state vector and A_0, B_0, C_0 and D_0 are matrices describing the DOFC dynamics. The matrices $F_1 \in \mathbb{R}^{n_x \times n_x}$ and $F_2 \in \mathbb{R}^{n_u \times n_x}$ are obtained as solution of the OFRMPC optimization problem.

Remark 1 *This OFRMPC formulation assumes normalized ± 1 bounds on the control variables. Different values for these bounds can be taken into account by scaling the columns of the input matrix B appropriately, as discussed in (Cao and Lin, 2005).*

Let $z(k)$ be an augmented state vector defined as

$$z(k) = \begin{bmatrix} \hat{x}(k) \\ e^T(k) \end{bmatrix} \tag{6}$$

with $e(k) = x(k) - \hat{x}(k)$. The system (1) – (3) controlled by (4) and (5) without the saturation function can be described as

$$z(k+1) = \tilde{A}z(k) + \tilde{B}\begin{bmatrix} F_1\hat{x}(k) \\ F_2\hat{x}(k) \end{bmatrix} \tag{7}$$

where

$$\tilde{A} = \begin{bmatrix} A_0 + B_0C & B_0C \\ A - A_0 + BC_0 - B_0C + BD_0C & A - B_0C + BD_0C \end{bmatrix} \tag{8}$$

$$\tilde{B} = \begin{bmatrix} I & 0 \\ -I & B \end{bmatrix}. \tag{9}$$

The effect of saturation on the control in (5) is taken into account in the LMI formulation by using the strategy originally proposed in (Cao and Lin, 2005).

Although the state is not measured, it is assumed that the initial state estimation error $e(0)$ is bounded by the expression $e^T(0)\hat{S}(0)^{-1}e(0) \leq 1$, for a given $\hat{S}(0)$.

The OFRMPC formulation involves an infinite-horizon cost function $J_\infty(k)$ given by

$$J_\infty(k) = \sum_{i=0}^{\infty}\left(||z(k+i|k)||^2_{P_z} + ||u(k+i|k)||^2_{P_u}\right) \tag{10}$$

where $P_z \in \mathbb{R}^{2n_x \times 2n_x}$ and $P_u \in \mathbb{R}^{n_u \times n_u}$ are symmetric, positive-definite weight matrices and $z(k+i|k)$, $u(k+i|k)$ denote the future values of z and u predicted on the basis of the information available at time k.

In view of the uncertainty on the A, B, C matrices, the optimization problem can be formulated in a min-max framework as

$$\min_{F_1,F_2} \max_{(A,B,C)\in\Omega} J_\infty(k), \tag{11}$$

subject to

$$|u_p(k+i|k)| \leq 1,\ p = 1, 2, ..., n_u,\ i \geq 0 \tag{12}$$
$$|y_q(k+i|k)| \leq y_{q,max},\ q = 1, 2, ..., n_y,\ i > 0 \tag{13}$$

where $y_{q,max}$ denotes the bound on the magnitude of the qth output variable.

As shown in (Li et al., 2013), this min-max problem can be replaced with the following semidefinite programming (SDP) problem (14) - (29), which involves the minimization of an upper bound 2γ for the cost J_∞:

$$\min_{\gamma,\gamma_1,c,d,Q,\hat{Q},Y_1,Y_2,Y_H,M,W} \beta\gamma + \gamma_1 + \beta_1 c + \beta_2 d, \tag{14}$$

subject to

$$\gamma \geq \gamma_1 \tag{15}$$

$$\begin{bmatrix} M & \Upsilon_i Q + \hat{Y}_H \\ \star & Q \end{bmatrix} \geq 0 \tag{16}$$

$$M_{pp} \leq 1/2 \tag{17}$$

$$\begin{bmatrix} W & C_i\begin{bmatrix} I & I \end{bmatrix}(\tilde{A}_{i,j}Q + \tilde{B}_i Z_l) \\ \star & Q \end{bmatrix} \geq 0 \tag{18}$$

$$W_{qq} \leq 0,5y^2_{q,max} \tag{19}$$

$$\begin{bmatrix} 1 & \star \\ \hat{x}(k) & Q_1 \end{bmatrix} \geq 0 \tag{20}$$

$$\begin{bmatrix} Q & \star & \star & \star \\ \tilde{A}_{i,j}Q + \tilde{B}_i Z_1 & \hat{Q} & \star & \star \\ P_z^{1/2}Q & 0 & \gamma_1 I & \star \\ P_u^{1/2}(\Upsilon_i Q + \hat{Y}_2) & 0 & 0 & \gamma_1 I \end{bmatrix} \geq 0 \tag{21}$$

$$\begin{bmatrix} Q & \star & \star & \star \\ \tilde{A}_{i,j}Q + \tilde{B}_i Z_m & \hat{Q} & \star & \star \\ P_z^{1/2}Q & 0 & \gamma I & \star \\ P_u^{1/2}(\Upsilon_i Q + V_m\hat{Y}_2 + V_m^-\hat{Y}_H) & 0 & 0 & \gamma I \end{bmatrix} \geq 0 \tag{22}$$

$$Q_1 \geq Q_3 \tag{23}$$

$$Q_2 \geq Q_4 \tag{24}$$

$$Q_1 \geq q_1(k) \tag{25}$$

$$Q_2 \geq \hat{S}(k) \tag{26}$$

$$Q_3 \leq cI \tag{27}$$

$$Q_4 \leq dI \tag{28}$$

$$\begin{bmatrix} Q & \star \\ \begin{bmatrix} I & 0 \end{bmatrix} (\tilde{A}_{i,j} Q + \tilde{B}_i Z_l) & \frac{1}{2} Q_3 \end{bmatrix} \geq 0 \tag{29}$$

where $q_1(0) = 0$, $\Upsilon_i = [C_0 + D_0 C_i \;\; D_0 C_i]$, $\hat{Y}_2 = [Y_2 \; 0]$, $\hat{Y}_H = [Y_H \; 0]$ and

$$Z_l = \begin{bmatrix} Y_1 & 0 \\ V_l Y_2 + V_l^- Y_H & 0 \end{bmatrix}, \tag{30}$$

$$Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}, \tag{31}$$

$\hat{Q} = \begin{bmatrix} Q_3 & 0 \\ 0 & Q_4 \end{bmatrix}$, $Q_1 = Q_1^T$, $Q_2 = Q_2^T$, $Q_3 = Q_3^T$, $Q_4 = Q_4^T$, $q_1(k) \in \mathbb{R}^{n_x \times n_x}$, $p = 1, 2, ..., n_u$, $i = 1, 2, ..., n_p$, $j = 1, 2, ..., n_p$ (or $j = i$, if $D_0 = 0$), $l = 1, 2, ..., 2^{n_u}$, $m = 2, 3, ..., 2^{n_u}$ and $q = 1, 2, ..., n_y$. Blocks above or below the main diagonal of a symmetric matrix are denoted by a $\star$ symbol. The matrices V_l and V_l^- are given in Definition 1 below. The matrices $\tilde{A}_{i,j}$ and $\tilde{B}_i$ are defined as

$$\tilde{A}_{i,j} = \begin{bmatrix} A_0 + B_0 C_j & B_0 C_j \\ A_i - A_0 + B_i C_0 - B_0 C_j + B_i D_0 C_j & A_i - B_0 C_j + B_i D_0 C_j \end{bmatrix} \tag{32}$$

$$\tilde{B}_i = \begin{bmatrix} I & 0 \\ -I & B_i \end{bmatrix}. \tag{33}$$

In this formulation, $\beta > 0$, $\beta_1 > 0$ and $\beta_2 > 0$ are additional design parameters, which can be tuned to improve closed-loop performance. More details can be found in (Huang et al., 2011) and (Li et al., 2013).

Definition 1 (Hu and Lin, 2001) *The set $\{V_l \in \mathbb{R}^{n_u \times n_u}, l = 1, 2, ..., 2^{n_u}\}$ comprises all the diagonal matrices with diagonal elements equal to either 1 or 0. For example, if $n_u = 2$, then*

$$V_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, V_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$
$$V_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, V_4 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Moreover, $V_l^- = I - V_l$, where I represents a identity matrix with suitable dimensions.

Henceforth the SDP problem (14) - (29) will be denoted by $\mathbb{P}(k)$.

The gain matrices F_1, F_2 to be employed in the calculation of $u(k)$ in (4), (5) are given by

$$F_1 = Y_1 Q_1^{-1} \tag{34}$$
$$F_2 = Y_2 Q_1^{-1} \tag{35}$$

and the matrices q_1 and $\hat{S}$ are updated as

$$q_1(k+1) = Q_1 \tag{36}$$
$$\hat{S}(k+1) = Q_4 \tag{37}$$

with Y_1, Y_2, Q_1 and Q_4 obtained as solution of $\mathbb{P}(k)$. As demonstrated in (Li et al., 2013), if $\mathbb{P}(k)$ is feasible at $k = 0$, then $\mathbb{P}(k)$ will remain feasible for $k > 0$ and the closed-loop system will be robustly stable.

3 Proposed simplification

With the aim of reducing the computational workload, the following theorem presents a simplification of the LMI constraint (29).

Theorem 1 *The LMI constraint* (29) *is equivalent to*

$$\begin{bmatrix} Q & \star \\ \begin{bmatrix} (A_0 + B_0 C_j) Q_1 + Y_1 & B_0 C_j Q_2 \end{bmatrix} & \frac{1}{2} Q_3 \end{bmatrix} \geq 0 \tag{38}$$

with $j = 1, 2, \ldots, n_p$.

Proof: See Appendix A.

In view of Theorem 1, the OFRMPC optimization problem can be reformulated as

$$\min_{\gamma, \gamma_1, c, d, Q, \hat{Q}, Y_1, Y_2, Y_H, M} \quad \beta\gamma + \gamma_1 + \beta_1 c + \beta_2 d, \tag{39}$$

subject to (15)-(28), (38).

It is worth noting that constraint (29) involves $n_p^2 2^{n_u}$ vertices (if $D_0 \neq 0$), whereas (38) involves only n_p vertices. Moreover, if matrix C is known, the constraint (38) will involve only a single LMI.

If this optimization problem is feasible, the gain matrices F_1 and F_2 are calculated as in (34) and (35) at each sampling time. The constraint satisfaction and robust stability guarantees from (Li et al., 2013) are preserved because the proposed formulation is equivalent to the original one.

4 Simulation example

Consider the following model for a DC motor (Lordelo and Fazzolari, 2014):

$$\dot{x}(t) = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & -B_{eq}/J \end{bmatrix}}_{A_c} x(t) + \underbrace{\begin{bmatrix} 0 \\ K/J \end{bmatrix}}_{B_c} u(t) \tag{40}$$

$$y(t) = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} x(t) \tag{41}$$

$$x(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix} \tag{42}$$

where $\theta(t)$ and $\dot{\theta}(t)$ correspond to angular position (rad) and velocity (rad/s), respectively, $B_{eq} = 0.08404$ Nms/rad is the viscous friction coefficient, $J = 2.1 \times 10^{-3}$ kgm^2 is the moment of inertia and K is an actuator gain with nominal value $K_{nom} = 0.1285$ Nm/V. Herein, K is assumed to be uncertain, with possible values in the range $K \in [0.5K_{nom},\ 2K_{nom}]$, which leads to a polytopic structure model with two vertices $(A_c, B_{c,1})$ and $(A_c, B_{c,2})$, where

$$B_{c,1} = \begin{bmatrix} 0 \\ 0.5K_{nom}/J \end{bmatrix}, \quad B_{c,2} = \begin{bmatrix} 0 \\ 2K_{nom}/J \end{bmatrix} \tag{43}$$

The state equation (40) needs to be discretized into form (1) for use in the OFRMPC control law. Assuming that the motor voltage is kept constant between sampling times by a Zero Order Hold, an exact discretization can be carried out as (Franklin et al., 1998)

$$A = e^{A_c T} = \sum_{n=0}^{\infty} \frac{(A_c T)^n}{n!} \tag{44}$$

$$B = \left(\int_0^T e^{A_c \tau} d\tau \right) B_c = \left(\sum_{n=0}^{\infty} \frac{(A_c T)^n}{(n+1)!} \right) B_c T \tag{45}$$

Since matrix B is related in a linear manner with the uncertain matrix B_c, the discretization results in a polytopic model with two vertices (A, B_1) and (A, B_2), where

$$B_1 = \left(\int_0^T e^{A_c \tau} d\tau \right) B_{c,1} \tag{46}$$

$$B_2 = \left(\int_0^T e^{A_c \tau} d\tau \right) B_{c,2} \tag{47}$$

The sampling period was set to $T = 10$ ms, which is an order of magnitude smaller than the settling time of the open-loop impulse response of the system (117 ms for a ± 1 % criterion).

The DOFC parameters were set to

$$A_0 = \begin{bmatrix} 1.00 & 0.016 \\ 0 & 0.37 \end{bmatrix} \tag{48}$$

$$B_0 = \begin{bmatrix} 1.17 \\ 4.55 \end{bmatrix} \tag{49}$$

$$C_0 = \begin{bmatrix} -0.32 & -0.12 \end{bmatrix} \tag{50}$$

$$D_0 = 0 \tag{51}$$

The OFRMPC was tuned with $\beta = 1$, $\beta_1 = 100$, $\beta_2 = 10$, $P_z = \text{diag}([1000,\ 0.1])$ and $P_u = 1$. The bounds on the control variable were set to ± 6 V, corresponding to the nominal voltage of the DC motor employed in (Lordelo and Fazzolari, 2014), as indicated in (Quanser Consulting, 2015). The initial system state was set to $x(0) = [\pi/4 \quad 0]^T$, and the OFRMPC algorithm was initialized with $\hat{x}(0) = [2\pi/9 \quad 0]^T$ and $\hat{S}(0) = \text{diag}([\pi/18, \pi/18])$.

The original and proposed formulations of the SDP problem involved in the OFRMPC control law were solved employing the Robust Control Toolbox of MATLAB 2012a. The default settings of the SDP solver were used. The simulations were carried out with the (A, B_2) vertex of the plant model, which corresponds to $K = 2K_{nom}$.

Figures 1 and 2 present the resulting output and control signals, respectively. As can be seen, the DC motor position was driven back to the origin in approximately 10 sampling periods, with proper enforcement of the control constraint. The results obtained with the original and proposed implementations of the OFRMPC control law are almost indistinguishable, which corroborates the equivalence between the two implementations. The minor differences observed in the results can be ascribed to numerical issues.

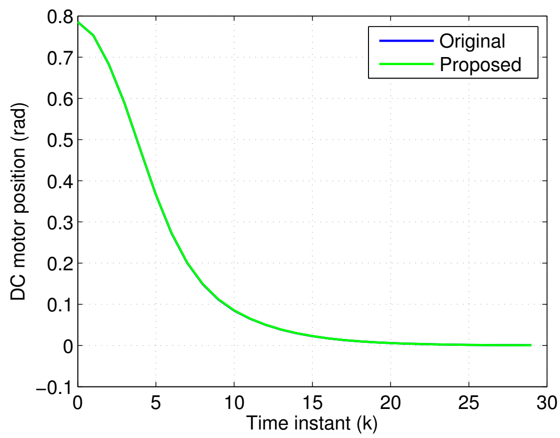

Figure 1: DC motor position.

Figure 3 presents the optimization cost (see (14)) *versus* number of SDP solver iterations at $k = 0$. It is worth noting that the initial point of each curve corresponds to the first feasible solution found by the solver. As can be seen, the

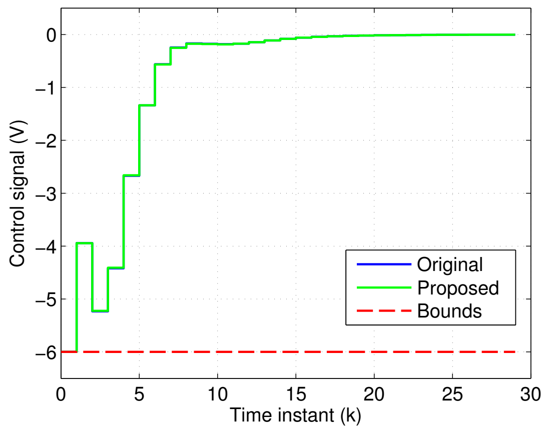

Figure 2: Control voltage and associated bound.

proposed implementation required 58 iterations to find a feasible solution, whereas the original formulation required 74 iterations. Similar results were obtained for $x_1(0) = 0.5, 0.6, \ldots, 1.0$ rad, as well as for $\hat{x}_2(0) = 1$ rad/s and -1 rad/s.

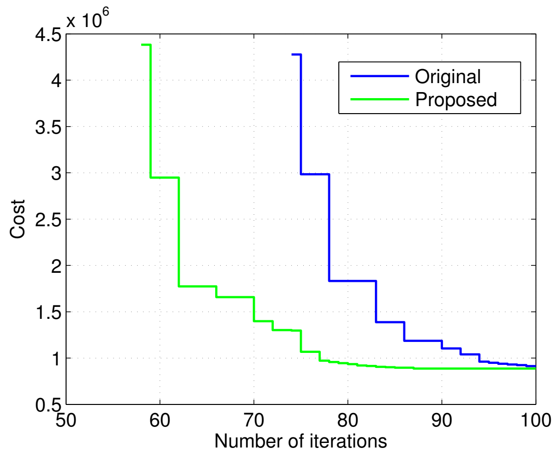

Figure 3: Optimization cost *versus* number of solver iterations.

It may be argued that the computational advantage of the proposed formulation would be less significant at subsequent time instants, because the solver can be initialized with the solution obtained at the previous sampling time. However, non-modelled issues such as sensor noise, nonlinearities or exogenous disturbances may cause the previous solution to be infeasible in practice. In this case, the proposed implementation would be useful to reduce the computational workload in the search for a new feasible solution.

5 Concluding remarks

This paper presented a simplified formulation for the OFRMPC control law originally proposed in (Li et al., 2013). More specifically, an equivalent LMI constraint was derived in order to simplify the SDP problem that needs to be solved at each sampling time. In a simulation example, the number of iterations required by the SDP solver to find a feasible solution was reduced from 74 to 58.

Work is being conducted to validate the proposed OFRMPC implementation in a laboratory system, by using the experimental procedures described in (Colombo Junior et al., 2016).

Acknowledgments

The authors acknowledge the support of FAPESP (Research Grant 2011/17610-0), CNPq (Research Fellowships 303714/2014-0 and 300703/2013-9) and CAPES (PhD scholarship).

Appendix A – Proof of Theorem 1

By replacing (32) and (33) for $\tilde{A}_{i,j}$ and $\tilde{B}_i$, the $(2,1)$ block in (29) can be rewritten as

$$
\begin{aligned}
&\begin{bmatrix} I & 0 \end{bmatrix}\left(\tilde{A}_{i,j}Q + \tilde{B}_i Z_l\right) = \\
&\begin{bmatrix} I & 0 \end{bmatrix}\left(\begin{bmatrix} A_0 + B_0C_j & B_0C_j \\ A_i - A_0 + B_iC_0 - B_0C_j + B_iD_0C_j & A_i - B_0C_j + B_iD_0C_j \end{bmatrix} Q + \begin{bmatrix} I & 0 \\ -I & B_i \end{bmatrix} Z_l\right) = \\
&\begin{bmatrix} A_0 + B_0C_j & B_0C_j \end{bmatrix} Q + \begin{bmatrix} I & 0 \end{bmatrix} Z_l \qquad (52)
\end{aligned}
$$

By replacing (30) and (31) for Z_l and Q, the expression in (52) can be rewritten as

$$
\begin{aligned}
&\begin{bmatrix} A_0 + B_0C_j & B_0C_j \end{bmatrix}\begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} + \\
&\begin{bmatrix} I & 0 \end{bmatrix}\begin{bmatrix} Y_1 & 0 \\ V_lY_2 + V_l^{-}Y_H & 0 \end{bmatrix} = \\
&\begin{bmatrix} (A_0 + B_0C_j)Q_1 & B_0C_jQ_2 \end{bmatrix} + \begin{bmatrix} Y_1 & 0 \end{bmatrix} = \\
&\begin{bmatrix} (A_0 + B_0C_j)Q_1 + Y_1 & B_0C_jQ_2 \end{bmatrix} \qquad (53)
\end{aligned}
$$

which corresponds to the $(2,1)$ block in (38). Therefore, it can be concluded that the LMIs (29) and (38) are equivalent.

$\square$

References

Bemporad, A. and Garulli, A. (2000). Output-feedback predictive control of constrained linear systems via set-membership state estimation, *International Journal of Control* **73**(8): 655–665.

Cao, Y. Y. and Lin, Z. (2005). Min-max MPC algorithm for LPV systems subject to input saturation, *IEE Proceedings - Control Theory and Applications* **152**(3): 266–272.

Colombo Junior, J. R., Afonso, R. J. M., Galvão, R. K. H. and Assunção, E. (2016). Robust model predictive control of a benchmark electromechanical system, *Journal of Control, Automation and Electrical Systems* **27**: 119 – 131.

Colombo Junior, J. R., Galvão, R. K. H. and Assunção, E. (2015). Comparação multiobjetivo de controladores preditivos robustos empregando realimentação de estado e de saída, *XII Simpósio Brasileiro de Automação Inteligente, Natal, RN*, pp. 878–883.

Cuzzola, F. A., Geromel, J. C. and Morari, M. (2002). An improved approach for constrained robust model predictive control, *Automatica* **38**(7): 1183–1189.

Ding, B. (2010). Constrained robust model predictive control via parameter-dependent dynamic output feedback, *Automatica* **46**: 1517–1523.

Ding, B., Xi, Y., Cychowski, M. T. and O'Mahony, T. (2008). A synthesis approach for output feedback robust constrained model predictive control, *Automatica* **44**(1): 258–264.

Franklin, G. F., Workman, M. L. and Powell, J. D. (1998). *Digital control of dynamic systems*, 3 edn, Ellis-Kagle Press, Half Moon Bay, CA, USA.

Granado, E., Colmenares, W., Bernussou, J. and Garcia, G. (2003). Linear matrix inequality based model predictive controller, *IEE Proceedings – Control Theory and Applications* **150**(5): 528–533.

Huang, H., Li, D., Lin, Z. and Xi, Y. (2011). An improved robust model predictive control design in the presence of actuator saturation, *Automatica* **47**: 861–864.

Kothare, M. V., Balakrishnan, V. and Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities, *Automatica* **32**(10): 1361–1379.

Lee, S.-M. and Park, J. H. (2007). Output feedback model predictive control for LPV systems using parameter-dependent Lyapunov function, *Applied Mathematics and Computation* **190**(1): 671–676.

Li, D., Xi, Y. and Gao, F. (2013). Synthesis of dynamic output feedback RMPC with saturated inputs, *Automatica* **49**: 949–954.

Lordelo, A. D. S. and Fazzolari, H. A. (2014). On interval goal programming switching surface robust design for integral sliding mode control, *Control Engineering Practice* **32**: 136 – 146.

Quanser Consulting (2015). Rotary servo base unit, `http://www.quanser.com/products/rotary_servo`. Accessed 11/11/2015.

Schuurmans, J. and Rossiter, J. (2000). Robust predictive control using tight sets of predicted states, *IEE Proceedings – Control Theory and Applications* **147**(1): 13–18.

Wan, Z. and Kothare, M. V. (2002). Robust output feedback model predictive control using off-line linear matrix inequalities, *Journal of Process Control* **12**(7): 763–774.