

Constrained discrete model predictive control of an arm-manipulator using Laguerre function

Tarcisio Carlos F. Pinheiro¹ | Antonio S. Silveira

Laboratory of Control and Systems,
Federal University of Pará, Belém, Brazil

Correspondence

Tarcisio Carlos F. Pinheiro, Laboratory of Control and Systems, Federal University of Pará, Rua Augusto Corrêa, 01 - Guamá, Belém, Pará 66075-110, Brazil.
Email: tpinheiro@ufpa.br

Funding information

Brazilian National Council for Scientific and Technological Development (CNPq), Grant/Award Numbers: 142414/2018-2, 408559/2016-0

Summary

This work presents a multivariable predictive controller applied on a redundant robotic manipulator with three degrees of freedom. The article focuses on the design of a discrete model-based predictive controller (DMPC) using the Laguerre function as a control effort weighting method to enhance the solution of Hildreth's quadratic programming and to minimize the trade-off problem in constrained case. The Laguerre functions are used to simplify and enhance the control horizon effect through parsimonious control trajectory, thus reducing the computational load required to find the optimal control solution. Furthermore, these results can be confirmed by simulations and experimental tests on the manipulator and comparing it to the traditional DMPC approach and the discrete linear quadratic regulator.

KEYWORDS

constrained control, discrete model predictive control, Hildreth's quadratic programming, Laguerre function, parsimonious future control

1 | INTRODUCTION

In last years, the industrial automation has been used in many strategies to improve the production levels. The main strategies are the use of robotic arm-manipulators in industrial processes for applications in painting, welding, picking, handling, cleaning, and palletizing.¹ Today, automation systems and robotic arms are making tasks that were considered impossible 10 years ago, possible and common.² However, due to robotic manipulators nonlinear dynamics, the use of classic controllers, such as the proportional-integral-derivative (PID) controller, to solve extremely complex tasks, cannot be as efficient as state-of-the-art control techniques can.

In recent years, the traditional PID controller is gradually being changed by advanced control techniques, such as discrete model-based predictive controller (DMPC), capable of dealing with nonlinear multivariable systems more efficiently. This means that applications with the DMPC have increased in the industry in the last years. It is because the DMPC improves the closed-loop performance for multivariable systems with minimum control effort. Although, some industries still prefer to use the PID controller rather than predictive control due to the latter complex implementation and hard understanding.³

Even though the algorithm complexity and computational load limit the widespread of predictive controllers, implementations using advanced control techniques are increasing in the industrial control systems, for example: hybrid fuzzy DMPC,⁴ hybrid DMPC based on genetic algorithms,⁵ DMPC probabilistic neural network.⁶ These are gaining space in industrial processes where there is the need of constrained control effort, in which DMPC can give a strong advantage due

to the fact that its optimization procedure can be modified in order to handle constraints, ensuring the desired closed-loop performance subjected to the plant physical limitations.

Beyond the use of DMPCs in applications with slow dynamic systems, its use has also increased in the field of rapid sampling frequency control-loops, such as in power electronics,⁷ and aerospace systems,⁸ such that the need for long prediction horizons in MPC methods might be required since the transitory dynamic comprises dozens, hundreds, or even thousands of samples to describe the complete transitory behavior.

The classical DMPC approach is not computationally efficient for large prediction horizons and can lead to poorly numerically conditioned solutions and heavy computational effort when implemented online, in the case of fast sampling frequency control-loops. It is because the traditional approach uses the forward shift operator for the representation of the control trajectory.⁹ Then, it would be interesting to have a predictive control technique where it is not needed a high computational cost, especially when applied in the constrained control energy cases. It can be achieved through Laguerre networks applied in Hildreth's quadratic programming solution method used to solve the constrained optimization problem.

As is well-known, reducing the computational effort in MPC is an established area of research. There are several ways to reduce the complexity and thus the computational burden in model predictive control. The methods that can be highlighted are move blocking,^{10,11} warm-starting in interior-point method,^{12,13} active-set approach,^{14,15} and fast gradient method.¹⁶ In addition, the Laguerre function also has potential benefits to achieve low computational loads with good feasibility and good performance in MPC design.^{17,18} It is because of its orthogonal property, making it an universal approximator which results in a parsimonious representation of the control trajectory.¹⁹

The purpose of this article is to use the Laguerre function within the DMPC design, achieving a control design called Laguerre DMPC or LDMPC. The aim is to control the angular position for each robotic arm joint with constrained control, lowering the computational cost and keeping a better trade-off between the control effort and closed-loop performance, in comparison with traditional DMPC. It is possible because the Laguerre function is an orthonormal function used to improve the horizon effect on the future control sequence. In other words, the set of Laguerre functions are used to capture the future control trajectory with a small number of parameters. Then, it can reduce the computational load when a large control horizon is needed.¹⁸

The article describes the DMPC design using Laguerre function applied to an arm-manipulator with three degrees of freedom (3-DOF). Experimental and simulation results are shown based on a test of angular position control of each joint of the manipulator. The results are compared with other classical discrete model-based predictive control (CDMPC) with receding horizon control, for both the constrained and unconstrained cases. Moreover, for unconstrained case, the LDMPC and CDMPC are benchmarked against the discrete linear quadratic regulator (DLQR).

Undoubtedly, the main contribution of the work is in the application side, because the LDMPC uses a multivariate approach to control a robot arm through a simple Arduino-based interface hardware, showing that the method presented can be applied in real systems with multiple inputs and multiple outputs (MIMO). Furthermore, articles with similar experimental results using LDMPC and a robot arm-manipulator, to the best of authors' knowledge, was not found in the control systems literature. The article with the nearest results was found in Reference 20, however, this work present a robot arm with only 2-DOF.

Beyond this introductory part, the article is organized as it follows: in Section 2, Laguerre function in DMPC design is presented; in Section 3, constrained control in DMPC design using Laguerre function is shown; in Section 4, the arm robot manipulator characteristics, experimental and simulation results of the control system are presented. In addition, the closed-loop stability analysis is described; conclusions and future works are shown in Section 5.

2 | LAGUERRE DMPC DESIGN

Initially, in dynamic systems theory, the Laguerre orthogonal functions have been used for system identification, because these functions could improve the numerical accuracy of the corresponding linear regression estimation problem.²¹⁻²⁴ Furthermore, a Laguerre model is produced by the discrete-time impulse response of a dynamic system and the same discrete-time impulse response that leads to the LDMPC design using Laguerre functions as shown by Reference 25.

In the literature of predictive control design, a set of Laguerre function is used to express the future incremental control trajectory through an orthonormal expansion. Moreover, there exist other types of orthonormal functions that have been used to capture the control trajectory, where can be mentioned the Kautz functions^{26,27} and the Legendre functions.^{28,29} However, the Laguerre functions can be better than the other mentioned approaches because the set of Laguerre functions

form a group of candidate functions with appropriate orders and with an increase in the number of its terms, such that the orthonormal expansion may have a better chance converging to the underlying optimal control trajectory.²⁵

The main goal in the design of DMPC is to optimize the future trajectory of a given system based on its dynamic model. In this article, this design model is in the state-space form because it is considered to be more effective for handling multivariable plants than the transfer function form.⁹

The nominal design model must be controllable and observable. Considering a system without dead times, described in the backward shift operator domain z^{-1} (ie, $F(z)z^{-1} = f(k-1)$), its state-space model is given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ y(k) &= Cx(k), \end{aligned} \quad (1)$$

where $u(k)$, $y(k)$, $x(k)$, are the vectors of q -inputs, q -outputs, and n_1 state variables, respectively.

To eliminate steady state errors, it is necessary to embed integrators into (1) in order to work with the control increment $\Delta u(k) = u(k) - u(k-1)$. By multiplying Equation (1) by $\Delta = (1 - z^{-1})$ and rewriting it in the augmented state vector form, $x_a(k) = [\Delta x(k) \quad y(k)]^T$, the following augmented state-space model, for the regulatory control case, is given by:

$$\begin{aligned} \underbrace{\begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix}}_{x_a(k+1)} &= \underbrace{\begin{bmatrix} A & 0_m^T \\ CA & I_{q \times q} \end{bmatrix}}_{A_a} \underbrace{\begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}}_{x_a(k)} \\ &+ \underbrace{\begin{bmatrix} B \\ CB \end{bmatrix}}_{B_a} \Delta u(k) \\ y(k) &= \underbrace{\begin{bmatrix} 0_m & I_{q \times q} \end{bmatrix}}_{C_a} \underbrace{\begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}}_{x_a(k)}, \end{aligned} \quad (2)$$

where $I_{q \times q}$ is identity matrices, 0_m is a zero matrix with dimensions $q \times n_1$ and A_a, B_a, C_a , are the augmented model state-space matrices.²⁵

For the servo control problem, that is, for the reference tracking case, the augmented state vector changes to $x_a(k) = [\Delta x(k) \quad e(k)]^T$, where $e(k)$ is the vector of reference tracking errors. However, it is important to remark that when $e(k)$ is included, the augmented design model is no longer observable and this state must be forced into $x_a(k)$.

In Reference 21, the z -transform of the discrete-time Laguerre polynomials are defined as follows:

$$\begin{aligned} \Gamma_1(z) &= \frac{\sqrt{(1-a^2)}}{1-az^{-1}} \\ \Gamma_2(z) &= \frac{\sqrt{(1-a^2)}}{(1-az^{-1})(1-az^{-1})} \frac{z^{-1}-a}{(1-az^{-1})} \\ &\vdots \\ \Gamma_N(z) &= \sqrt{(1-a^2)} \frac{(z^{-1}-a)^{N-1}}{(1-az^{-1})^N}. \end{aligned} \quad (3)$$

The parameters a and N shown in (3) can be chosen separately for each input signal in MIMO case systems, where a is the discrete pole of the Laguerre function, called *scaling factor* in the literature, that can be selected by the user and need to be in the range of $0 \leq a < 1$ to guarantee the stability of the system. Generally, a is chosen as an estimate of the real part (absolute value) of the closed-loop dominant eigenvalue, where a large value for a and a small number of N , correspond to an incremental control with a slow decay rate.²⁵ N is the *order of the Laguerre network* and it is used to capture the control signal. When N increases, the degrees of freedom regarding the control trajectory also increases.⁹ The parameter

N has a similar role as the control horizon (N_c) in the DMPC, with $N = N_c$ when $a = 0$.²⁵ However, the advantage is to use a nonzero a and to work with $N \leq N_c$ in the LDMPC design.

$l_N(k)$ is the inverse z -transform of (3), given by:

$$l_N(k) = \mathcal{Z}^{-1}\{\Gamma_N(z)\}. \quad (4)$$

Then, the discrete-time Laguerre function can be rewritten in a vector form, for $i = 1, \dots, q$ inputs, as it follows:

$$L_i(k) = \begin{bmatrix} l_1(k) & l_2(k) & \dots & l_N(k) \end{bmatrix}^T, \quad (5)$$

$$L(k)^T = \begin{bmatrix} L_1(k) & L_2(k) & \dots & L_i(k) \end{bmatrix}. \quad (6)$$

The initial condition of (5) can be obtained from Reference 9

$$L_i(0)^T = \sqrt{\beta} \begin{bmatrix} 1 & -a & a^2 & \dots & (-1)^{N-1} a^{N-1} \end{bmatrix}, \quad (7)$$

where $\beta = (1 - a^2)$.

Their orthonormality can be expressed by $\sum_{k=0}^{\infty} l_i(k)l_j(k) = 0$ for $i \neq j$, and $\sum_{k=0}^{\infty} l_i(k)l_j(k) = 1$ for $i = j$. It will be used in LDMPC design.

According to Reference 9, $L_i(k)$ can also be solved recursively by

$$L_i(k) = A_i L_i(k-1), \quad (8)$$

where A_i is a Toeplitz matrix with dimensions $N \times N$ and it is a function of a and β parameters. This matrix is given by Reference 9:

$$A_i = \begin{bmatrix} a & 0 & 0 & 0 & \dots & 0 \\ \beta & a & 0 & 0 & \dots & 0 \\ -a\beta & \beta & a & 0 & \dots & 0 \\ a^2\beta & -a\beta & \beta & a & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ (-1)^{N-2} a^{N-2} \beta & (-1)^{N-3} a^{N-3} \beta & \dots & \dots & \beta & a \end{bmatrix}. \quad (9)$$

The LDMPC design uses a set of Laguerre functions, $l_1(m), l_2(m), \dots, l_N(m)$, to represent the future control trajectory, $\Delta u(k_i), \Delta u(k_i + 1), \dots, \Delta u(k_i + m)$, for each sample instant k , where k_i is the initial time of the moving horizon window, while m is an arbitrary future sample instant. This strategy of control using orthonormal functions can be used to express the control increment signal,

$$\Delta u(k_i + m|k_i) = \sum_{j=1}^N p_j(k_i) l_j(m) = L_i(m)^T \eta, \quad (10)$$

where in the SISO case, η is the parameter vector with dimension $1 \times N$,

$$\eta = \begin{bmatrix} p_1 & p_2 & \dots & p_N \end{bmatrix}^T, \quad (11)$$

$p_j, j = 1, 2, \dots, N$, are the coefficients, and they are functions of initial time of the moving horizon window, k_i . Extend to MIMO case, η is represented by,

$$\eta^T = \begin{bmatrix} \eta_1^T & \eta_2^T & \dots & \eta_i^T \end{bmatrix} \quad (12)$$

with dimensions $1 \times N_{\text{sum}}$, $N_{\text{sum}} = N_1 + \dots + N_i$, $i = 1, \dots, q$ inputs, where each input signal is designated to have an independent η_i . The coefficient vector η will be optimized and calculated in design through the minimization of cost function (13).

The cost function of the LDMPC is based on the minimization of the error between the set-point signal and the output signal,²⁵ where the objective is to find the vector η that minimizes the cost function

$$J = \eta^T \Omega \eta + 2\eta^T \Psi x_a(k) + \sum_{m=1}^{N_p} x_a(k)^T (A_a^T)^m Q A_a^m x_a(k). \quad (13)$$

The matrices Ω and Ψ are, respectively,

$$\Omega = \sum_{m=1}^{N_p} S_c(m)^T Q S_c(m) + R_L, \quad (14)$$

$$\Psi = \sum_{m=1}^{N_p} S_c(m)^T Q A_a^m. \quad (15)$$

The matrix $S_c(m)$, $m = 1, \dots, N_p$, is the convolution sum to compute the prediction of the system shown in (2) with respect to the output prediction horizon N_p , weighted by the Laguerre network and given by Reference 9

$$S_c(m) = \sum_{j=0}^{m-1} A_a^{m-j-1} B_a L(j)^T. \quad (16)$$

The solution for the convolution sum can be obtained through linear algebraic equation or recursively by

$$S_c(m) = A_a^{m-1} S_c(m-1) + \begin{bmatrix} B_1 L_1(m-1)^T & B_2 L_2(m-1)^T & \dots & B_i L_i(m-1)^T \end{bmatrix}. \quad (17)$$

Observe in (17) that the B_a matrix was divided into the form

$$B_a = \begin{bmatrix} B_1 & B_2 & \dots & B_i \end{bmatrix} \quad (18)$$

to represent each column of the matrix B_a and for $m = 2, 3, 4, \dots, N_p$, $S_c(1) = B_a L(0)^T$, using the relation $L_i(m) = A_i L_i(m-1)$ with the initial condition $L(0)^T$ as given in (7).

In (14), $R_L = RI$ is a $N \times N$ diagonal matrix and R is a scalar acting as a control effort weighting factor to tune the closed-loop response speed. The lower the value of R , less weight is put on the control, consequently leading to a faster closed-loop response. Furthermore, the matrix $Q = C_a^T C_a$ is used for the purpose of minimizing the reference tracking errors.

The minimization of the cost function in (13) is done by taking its partial derivative to find the optimal solution for the control vector η , without constraints, resulting in Reference 25

$$\eta = -\Omega^{-1} \Psi x_a(k). \quad (19)$$

Finally, static Laguerre terms are included in (19) and this last equation is rewritten based on the augmented design model input, which is the optimal control increment (without constraints), say it the receding horizon control law, given by:

$$\Delta u(k) = -K_{\text{mpc}} x_a(k). \quad (20)$$

The contribution of the Laguerre terms in the controller gain, K_{mpc} , is defined by Reference 25:

$$K_{\text{mpc}} = \overbrace{\begin{bmatrix} L_1(0)^T & \sigma_2^T & \dots & \sigma_i^T \\ \sigma_1^T & L_2(0)^T & \dots & \sigma_i^T \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_1^T & \sigma_2^T & \dots & L_i(0)^T \end{bmatrix}}^{L_{\text{Mat}}} \Omega^{-1} \Psi \quad (21)$$

σ_i^T elements are all zero row vectors with dimensions equal to N_i , $i = 1, \dots, q$ inputs, and $x_a(k) = [\Delta x(k) \quad e(k)]^T$. Where, $e(k) = y(k) - y_r(k)$, and $y_r(k)$ is the set-point.

The K_{mpc} gain can be rewritten and divided into two parts,

$$K_{\text{mpc}} = \begin{bmatrix} K_x & K_e \end{bmatrix} \quad (22)$$

being K_x the gain for state ($\Delta x(k)$) feedback and K_e is the gain for the error state feedback. Then, the closed-loop of LDMPC is given by,

$$\begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix} = (A_a - B_a K_{\text{mpc}}) \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} + B_a K_e y_r(k),$$

$$y(k) = C_a \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}. \quad (23)$$

Furthermore, to help understand the project of LDMPC, the flow chart and block diagram are shown in Figures 1 and 2, respectively.

3 | CONSTRAINED CONTROL IN LDMPC DESIGN USING LAGUERRE FUNCTION

The constraints on the amplitude of the control variable are solved through a quadratic programming method. This method uses the Laguerre functions in order to have the flexibility to define constraints on future control trajectories. It decreases the number of constraints solved online within the prediction horizon and reduces the computational load for high-order systems with long-range predictions.²⁵

The constraints are articulated as linear inequalities and are combined with the cost function

$$\bar{J} = \frac{1}{2} \eta^T \Omega \eta + (\Psi x_a(k))^T \eta, \quad (24)$$

used in the predictive control design. The optimization procedure is to minimize \bar{J} using quadratic programming to find an optimal predictive control and next, find the constrained minimum of a positive definite quadratic function subject to linear inequality constraints, expressed by Reference 25:

$$\begin{bmatrix} M_{\Delta U} \\ -M_{\Delta U} \\ M_U \\ -M_U \end{bmatrix} \eta \leq \begin{bmatrix} V_{\Delta U}^{\max} \\ -V_{\Delta U}^{\min} \\ V_U^{\max} \\ -V_U^{\min} \end{bmatrix}. \quad (25)$$

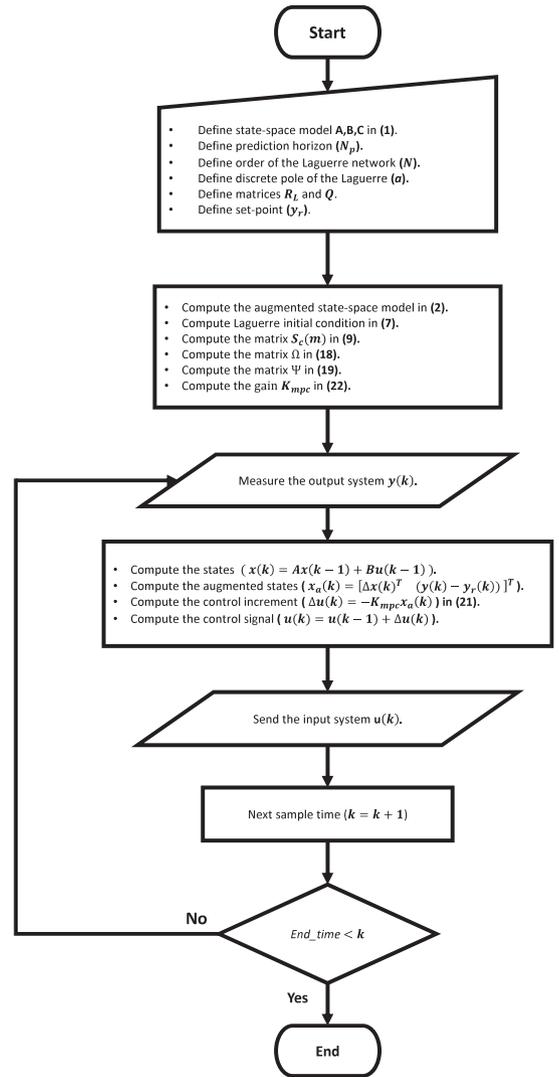


FIGURE 1 Flow chart of LDMPC unconstrained. LDMPC, Laguerre DMPC

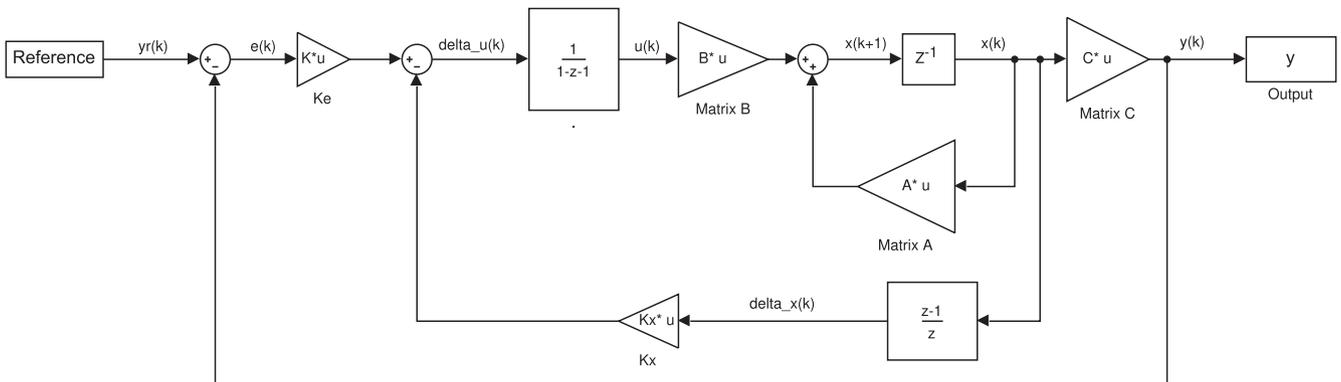


FIGURE 2 Block diagram of LDMPC unconstrained. LDMPC, Laguerre DMPC

The constraints limits can be chosen for each input separately, where the upper limits on the amplitude of the control increment and control, respectively, are given by

$$V_{\Delta U^{\max}} = \begin{bmatrix} \Delta U^{\max}_1 \\ \vdots \\ \Delta U^{\max}_{N_{\text{sum}}} \end{bmatrix}, \quad (26)$$

$$V_{U^{\max}} = \begin{bmatrix} (U^{\max} - U^{k-1})_1 \\ \vdots \\ (U^{\max} - U^{k-1})_{N_{\text{sum}}} \end{bmatrix} \quad (27)$$

and similarly, the lower limits are

$$V_{\Delta U^{\min}} = \begin{bmatrix} \Delta U^{\min}_1 \\ \vdots \\ \Delta U^{\min}_{N_{\text{sum}}} \end{bmatrix}, \quad (28)$$

$$V_{U^{\min}} = \begin{bmatrix} (U^{\min} - U^{k-1})_1 \\ \vdots \\ (U^{\min} - U^{k-1})_{N_{\text{sum}}} \end{bmatrix} \quad (29)$$

and $U^{k-1} = [u_1(k-1) \ u_2(k-1) \ \dots \ u_q(k-1)]^T$ is the vector containing the previous u value for each input, $\Delta U^{\min} = [\Delta u_1^{\min} \ \Delta u_2^{\min} \ \dots \ \Delta u_q^{\min}]^T$, $U^{\min} = [u_1^{\min} \ u_2^{\min} \ \dots \ u_q^{\min}]^T$, and $\Delta U^{\max} = [\Delta u_1^{\max} \ \Delta u_2^{\max} \ \dots \ \Delta u_q^{\max}]^T$, $U^{\max} = [u_1^{\max} \ u_2^{\max} \ \dots \ u_q^{\max}]^T$ are the minimum and maximum values for each input, respectively. $M_{\Delta U}$ and M_U are the matrices reflecting the constraints of the control increment and control signal, respectively. It can be calculated by

$$M_{\Delta U} = \begin{bmatrix} D(1) \\ D(2) \\ \vdots \\ D(N_{\text{sum}}) \end{bmatrix}, \quad (30)$$

$$M_U = \begin{bmatrix} D(1) \\ D(1) + D(2) \\ \vdots \\ D(N_{\text{sum}} - 1) + D(N_{\text{sum}}) \end{bmatrix}, \quad (31)$$

where $D(k)$, $k = 1, 2, \dots, N_{\text{sum}}$, is the block matrix composed of Laguerre functions concerning each input given by:

$$D(k) = \begin{bmatrix} (A_{l_1}^{k-1} L_1(0))^T & \sigma_2^T & \dots & \sigma_i^T \\ \sigma_1^T & (A_{l_2}^{k-1} L_2(0))^T & \dots & \sigma_i^T \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_1^T & \sigma_2^T & \dots & (A_{l_i}^{k-1} L_i(0))^T \end{bmatrix}. \quad (32)$$

As stated before, σ_i^T ($i = 1, \dots, q$ inputs) are all zero row vectors with dimensions N_i . The $N_{\text{sum}} = N_1 + \dots + N_i$ is the sum equal to the Laguerre network order. N_{sum} is also the number of future samples for constraints to be imposed. It is used to settle the prediction of the future control trajectory.

- **Hildreth's quadratic programming:** it is the method used to solve the constrained optimization problem in this article. It is because Hildreth's algorithm is an efficient method for solving large systems of inequalities with sparse matrices,³⁰ and also it does not require any matrix inversion. This quadratic programming is used at each sampling time, having obtained a vector $\lambda \geq 0$ that is called dual variables in the optimization literature. The Hildreth's algorithm, as shown in Reference 25, is repeated in brief, as follows:

$$\lambda_i^{m+1} = \max(0, w_i^{m+1}), \quad (33)$$

where,

$$w_i^{m+1} = -\frac{1}{h_{ii}} \left[k_i + \sum_{j=1}^{i-1} h_{ij} \lambda_j^{m+1} + \sum_{j=i+1}^n h_{ij} \lambda_j^m \right] \quad (34)$$

h_{ij} are the elements of the matrix $H = M\Omega^{-1}M^T$ and k_i are the elements of the vector $K = \gamma + M\Omega^{-1}\Psi x_a(k)$. Then, the optimal solution to the parameter vector η_{restr} that solve the inequality shown in (25), which is the constrained control sequence, is defined as follows:

$$\eta_{\text{restr}} = -\Omega^{-1}(\Psi x_a(k) + M^T \lambda). \quad (35)$$

The receding horizon control law with constraints is given by:

$$\Delta u(k) = \overbrace{\begin{bmatrix} L_1(0)^T & \sigma_2^T & \dots & \sigma_i^T \\ \sigma_1^T & L_2(0)^T & \dots & \sigma_i^T \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_1^T & \sigma_2^T & \dots & L_i(0)^T \end{bmatrix}}^{L_{\text{mat}}} \eta_{\text{restr}}. \quad (36)$$

For a better understanding, a flow chart of LDMPC constrained design using Hildreth's quadratic programming is shown in Figure 3.

4 | LAGUERRE DMPC OF THE ARM-MANIPULATOR

This section describes the characteristics of the robotic arm used in this article and also the experimental and simulation results obtained using the LDMPC design presented in Section 2 compared with another CDMPC with receding horizon control. Furthermore in this Section, it is done the evaluation of the LDMPC control-loop robustness and stability analysis.

4.1 | Robotic arm-manipulator description

The robotic arm used is a Mentor desktop robot, made by Feedback Instruments Limited and depicted in Figure 4. This robot has 5-DOF, but within our tests, only 3-DOF were used. It has revolute joints, where each joint is driven by DC servo motors with potentiometric angular sensors for precise positional control. The analog-to-digital and digital-to-analog interface, however, is not the original from Feedback, but an Arduino-based interface hardware with a DAQ library for Scilab/Xcos and Matlab/Simulink, known as *LACOS DaqDuino v2*, found at Mathworks FileExchange¹.

¹<http://www.mathworks.com/matlabcentral/fileexchange/50784-daquino>

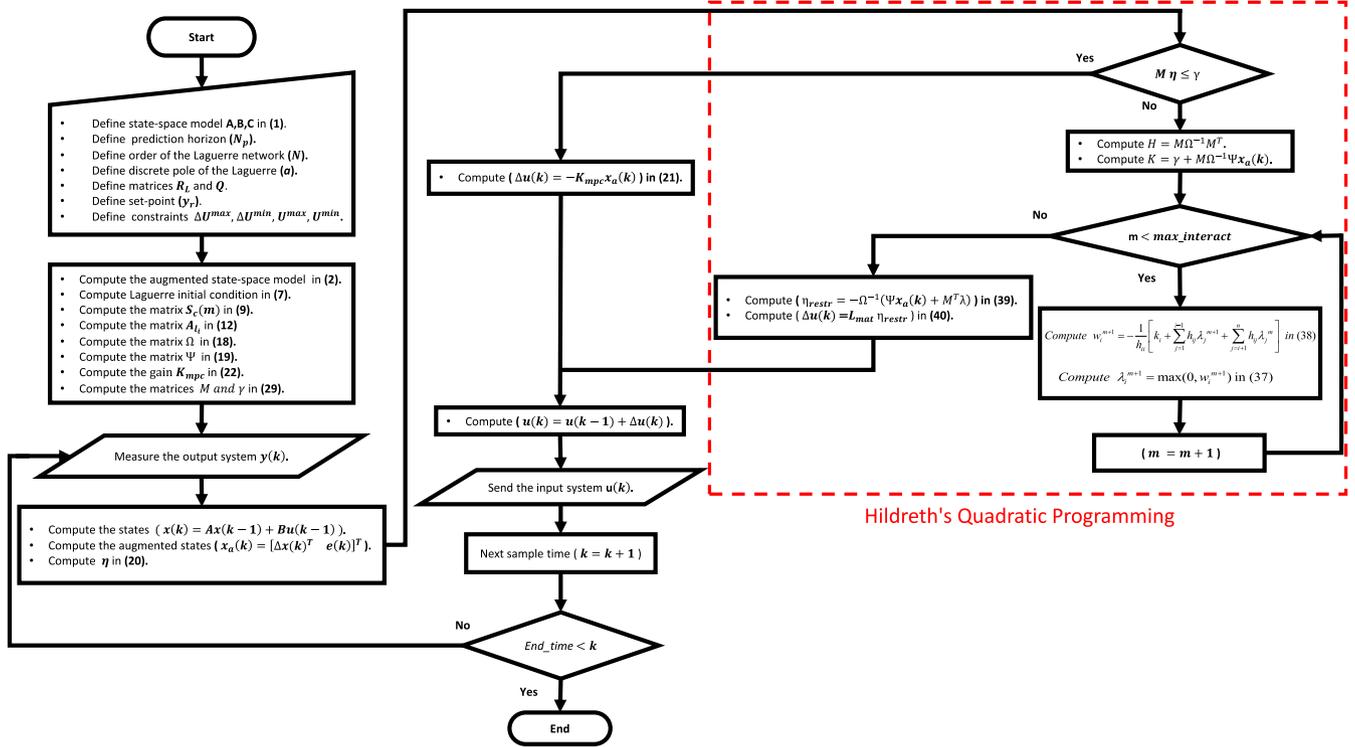


FIGURE 3 Flow chart of constrained LDMPC. LDMPC, Laguerre DMPC [Colour figure can be viewed at wileyonlinelibrary.com]

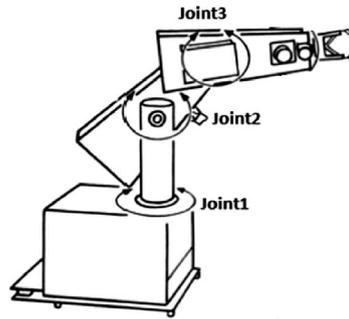


FIGURE 4 Photo and diagram of the arm-manipulator Mentor, from Feedback [Colour figure can be viewed at wileyonlinelibrary.com]

4.2 | Experimental and simulation results

This experimental case study is to demonstrate that the Laguerre function can improve the trade-off between control energy and closed-loop performance, and the future control trajectory optimization under the same control horizon window, with respect to CDMPC. This case study brings significant results for both LDMPC cases, constrained and unconstrained.

The MIMO system that represents the 3-DOF robotic arm was modeled using experimental data and Recursive Least Squares estimation. The identified MIMO transfer function model is given by:

$$\begin{bmatrix} Y_1(z) \\ Y_2(z) \\ Y_3(z) \end{bmatrix} = \begin{bmatrix} G_{11}(z) & G_{12}(z) & G_{13}(z) \\ G_{21}(z) & G_{22}(z) & G_{23}(z) \\ G_{31}(z) & G_{32}(z) & G_{33}(z) \end{bmatrix} \begin{bmatrix} U_1(z) \\ U_2(z) \\ U_3(z) \end{bmatrix}, \quad (37)$$

where,

$$\begin{aligned}
 G_{11}(z) &= \frac{0.0007272z + 0.003917}{z^2 - 1.604z + 0.6041}; & G_{12}(z) &= \frac{5.2074e - 9z + 3.4674e - 11}{z^2 - 0.28848z - 0.70734}; \\
 G_{13}(z) &= \frac{5.6358e - 09z + 5.5471e - 09}{z^2 - 0.5z - 0.5}; & G_{21}(z) &= \frac{1.1382e - 9z + 7.6199e - 12}{z^2 - 0.49889z - 0.498254}; \\
 G_{22}(z) &= \frac{0.0002373z + 0.005518}{z^2 - 1.5340z + 0.5340}; & G_{33}(z) &= \frac{0.002278z + 0.01048}{z^2 - 1.5393z + 0.5393}; \\
 G_{13}(z) &= G_{23}(z); & G_{31}(z) &= G_{21}(z); G_{12}(z) = G_{32}(z);
 \end{aligned} \tag{38}$$

. The sampling period of this system is 0.05 second, because the data acquisition device used with the arm-manipulator Mentor process has an USB interface, which limits the maximum sample-and-hold frequency to 100 Hz. Since the maximum singular value, of the system shown in (37), has its cut-off frequency around 0.125 Hz, a sampling frequency of 20 Hz was selected, obeying the Shannon-Nyquist sampling theorem and coping with the sampling frequency used with the robotic system shown in a similar work of Wang.²⁰

4.2.1 | Simulation results

In this first test, the joints of the Robot are actuated by a set-point for each motor. The set-point positions are 180° for joint-1 (J_1), 90° for joint-2 (J_2), and 60° for joint-3 (J_3). The prediction horizon and control horizon for CDMPC are, respectively, $N_p = 100$, $N_c = 2$. For the LDMPC, $N_p = 100$ and the parameters for the Laguerre functions are $N_1 = N_2 = N_3 = 2$ and $a_1 = a_2 = a_3 = 0.3$. The weighting matrices are $R_1 = R_2 = R_3 = 0.008$ and $Q = C_a^T C_a$. This experiment is not in the constrained case and these controllers tuning parameters were selected by trial and error, whilst forcing the equivalence of prediction horizons of CDMPC and LDMPC, granting a fair comparison between these two MPCs. This initial condition values of tuning parameters were chosen as follows: N_p should be incremented until no significant increase in the output response performance is achieved. N_c and N need to be as small as possible, but both are increased until promotes an internally stable controller. A small control horizon means fewer variables to compute in the quadratic problem solved at each control interval, which promotes faster computations. Finally, a is the performance fine-tuning, where if a is close to zero, a faster closed-loop response is produced, but if a is close to one, a slower closed-loop response is achieved.

Moreover, DLQR that is based on an infinite horizon, was benchmarked with CDMPC and LDMPC, using the same R and Q matrices. It is shown to justify the effectiveness of the closed-loop performance of both MPC controllers in the unconstrained simulation case.

Figures 5-7, are used to show the output signals, the control signals and the incremental control signals, respectively, for the unconstrained simulation case. These results are discussed later in the text, along with performance indices data.

4.2.2 | Experimental constrained results

In this second test, the LDMPC and CDMPC controllers are applied on the real Robot. The controllers parameters and setpoints are the same as in the first simulation test, but in this case, the control signals are constrained. The constraints used on these control signals are $\Delta U^{\max} = 2.5$, $\Delta U^{\min} = -2.5$ and $U^{\max} = 5$, $U^{\min} = -5$ for all inputs. For the CDMPC design it was adopted a simple saturation control, and for the LDMPC design, it was adopted the Hildreth's quadratic programming, given in Section 3. The results for the joint positions, control signals and incremental signals, are respectively, shown in Figures 8-10. This experimental results, along with the first simulated case results, are discussed next.

4.2.3 | Simulation and experimental results analysis

The performance of the control-loops can be evaluated by a set of discrete versions of integral performance indices, based on power or energy signals. The lower the value of indices, the better of performance of the closed-loop system. With n_i being the *number of iterations* of the tests, the control signals matrix, $\mathbf{u}_{n_i \times 3}$, power, will be related to the index $ISU = \sum_{k=1}^m \mathbf{u}(k)^2$ associated to power control signal consumption; the reference tracking error matrix, $\mathbf{e}_{n_i \times 3}$, power, will be

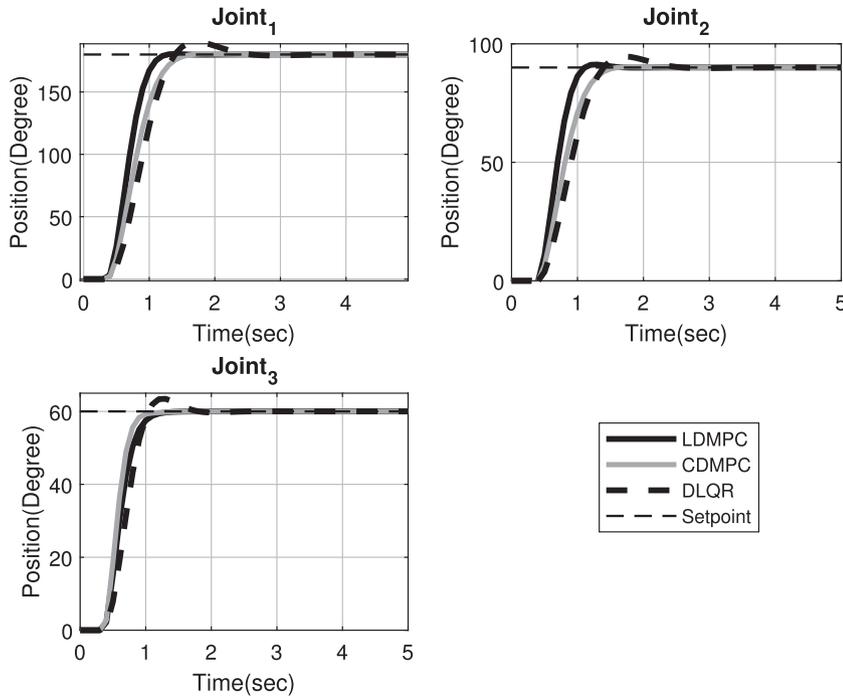


FIGURE 5 Unconstrained simulation test: Joints positions

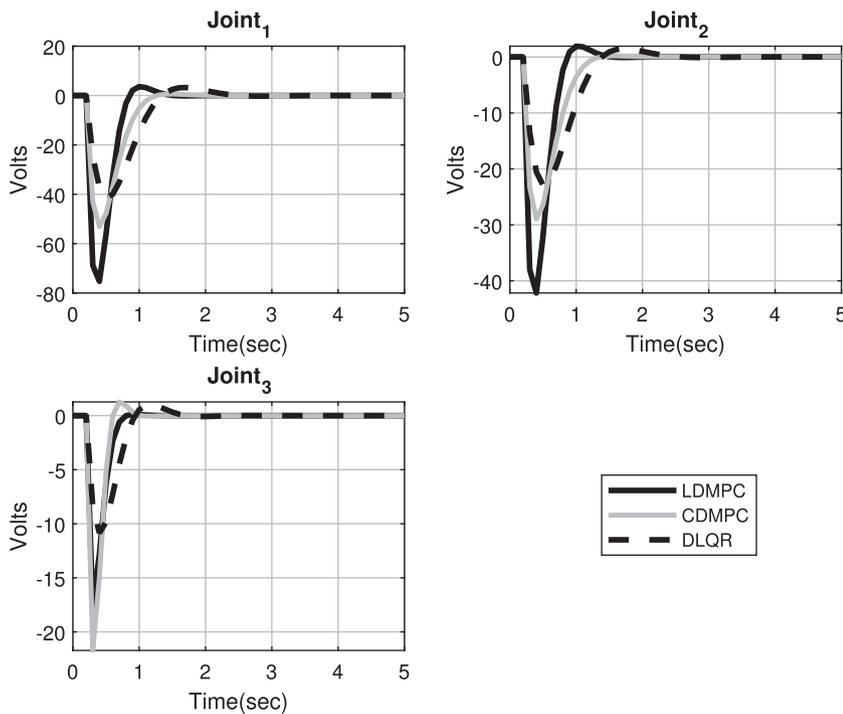


FIGURE 6 Unconstrained simulation test: Control signals

related to the integral of the squared error, $ISE = \sum_{k=1}^m e(k)^2$; the control difference matrix, $\Delta \mathbf{u}_{ni \times 3}$, power, namely, total variation control, will be $TVC = \sum_{k=1}^m |\Delta u(k)|$; and the percent overshoot will be referred to PO. The performance results of CDMPC and LDMPC control-loops, with and without constraints, are presented in Tables 1 and 2, respectively.

Analyzing the results between the controllers performance, for the unconstrained simulation case, CDMPC and LDMPC presented similar results under the same output and control prediction horizons (recall that in LDMPC $N_c = N$). Observe Figures 5 to 7, to see that all three joints had a fast response with asymptotic convergence and with similar control effort for the LDMPC case, which can be confirmed by the performance indices shown in Table 1, where LDMPC outperformed CDMPC slightly for the majority of the indices, but losing in the TVC. The major contribution of the LDMPC appears when dealing with the nonlinear system in the experimental test with constraints. Observe Figures 8 to 10, where

FIGURE 7 Unconstrained simulation test: Control increment signals

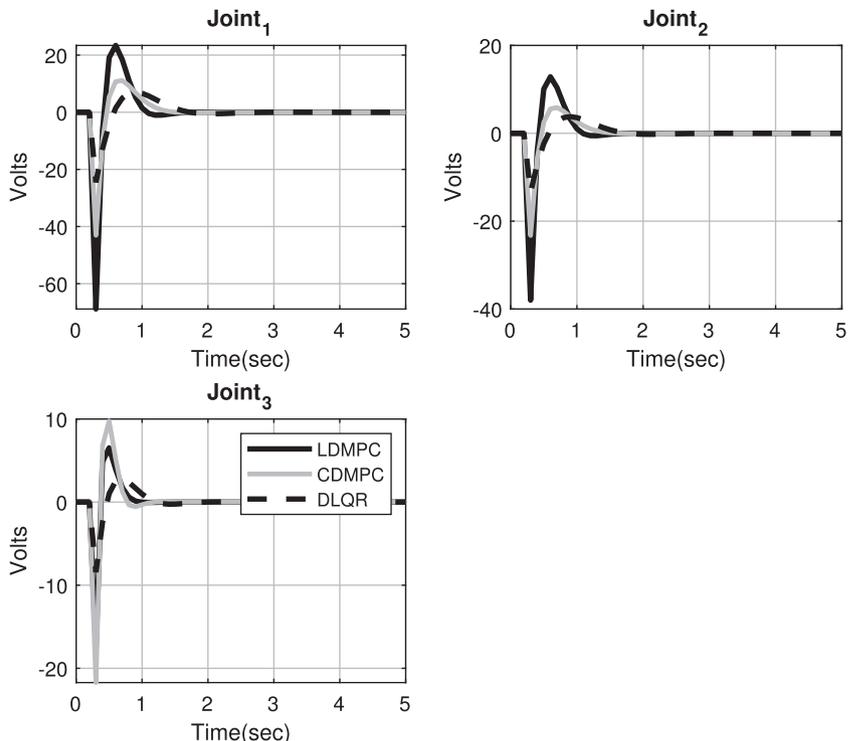
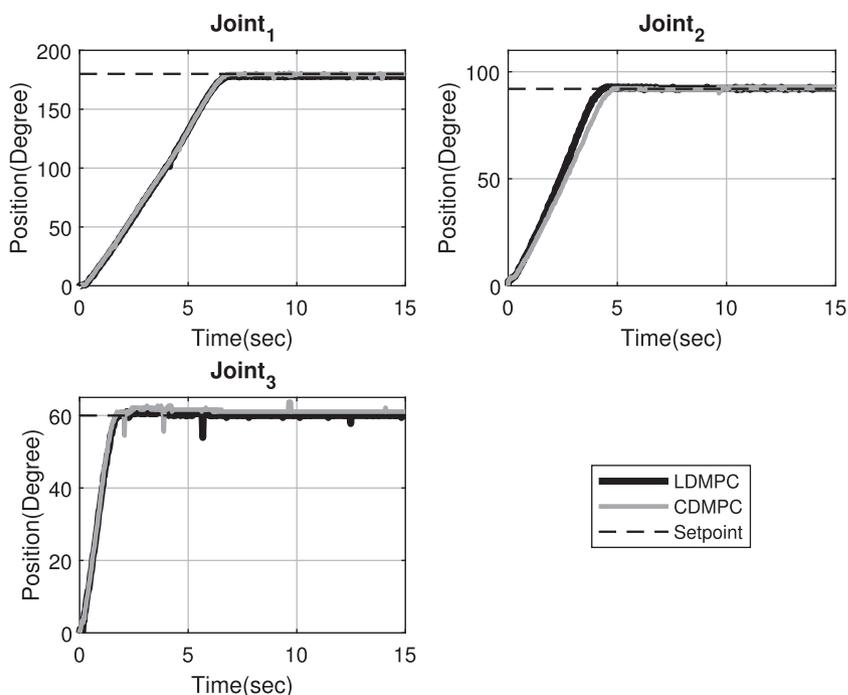


FIGURE 8 Constrained experimental test: Joint positions



the LDMPC case response speed is still faster than the classical case, but now, its control signals are also more conservative, as can be seen at Figures 9 and 10, that despite the maximum control amplitudes had achieved similar values, the time spent at those values were shorter for the LDMPC as can be confirmed by the indices shown in Table 2, where LDMPC has lost only in the PO. These same results were also observed in the constrained simulation case, but not shown in this article in order to present the experimental test and give the proof of concept instead, confirming that the algorithm works on a real experiment.

In addition, the elapsed time to calculate the computation time in the unconstrained simulation environment was measured with the *tic-toc* function from MATLAB[®] R2017a and for the computation of simulation using a desktop

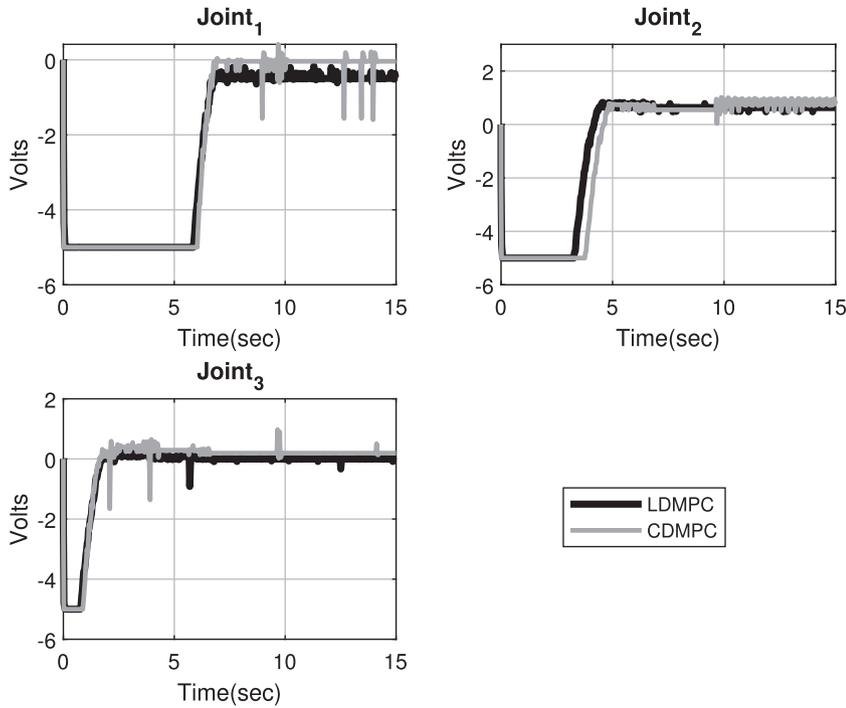


FIGURE 9 Constrained experimental test: Control signals

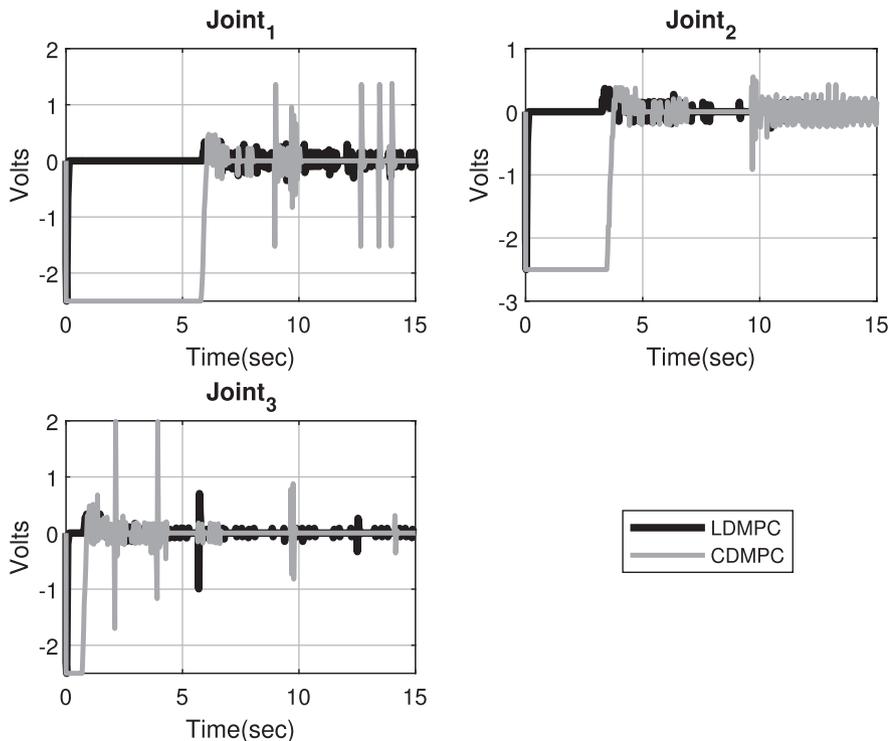


FIGURE 10 Constrained experimental test: Control increment signals

PC (with Microsoft Windows 10, 2.2 GHz Intel Core i7-3632QM, 8 GB RAM). The elapsed time, with period of 400 samples for LDMPC was ≈ 0.02 second, for CDMPC was ≈ 0.03 second, and for DLQR was ≈ 0.05 second. Thus, this result can confirm that the Laguerre function can reduce the computational load of the algorithm. The Laguerre polynomial-based approach can be very useful for adaptive MPC design, where the future optimal control trajectory needs to be calculated for every sampling time. Similar results about controller computation time reduce can be found in Reference 31.

Furthermore, for constrained case in real-time application, the elapsed time to each sampling period for LDMPC using Hidreth's quadratic programming was ≈ 0.0008 second, whereas the elapsed time for CDMPC using simple saturation

TABLE 1 Unconstrained performance results

			LDMPC	DLQR
ISU	Joint 1	244.05	263.88	280.11
	Joint 2	137.94	149.91	154.63
	Joint 3	44.52	39.98	48.19
ISE	Joint 1	32.512	26.841	37.944
	Joint 2	10.069	8.297	11.667
	Joint 3	3.070	3.062	4.184
PO (%)	Joint 1	0.2347	0.2567	5.5889
	Joint 2	0.3026	1.3865	5.1515
	Joint 3	0	0	5.7711
TVC	Joint 1	107.41	158.03	89.99
	Joint 2	58.34	88.37	48.90
	Joint 3	46.08	35.51	23.87

Abbreviations: DLQR, discrete linear quadratic regulator; LDMPC, Laguerre DMPC.

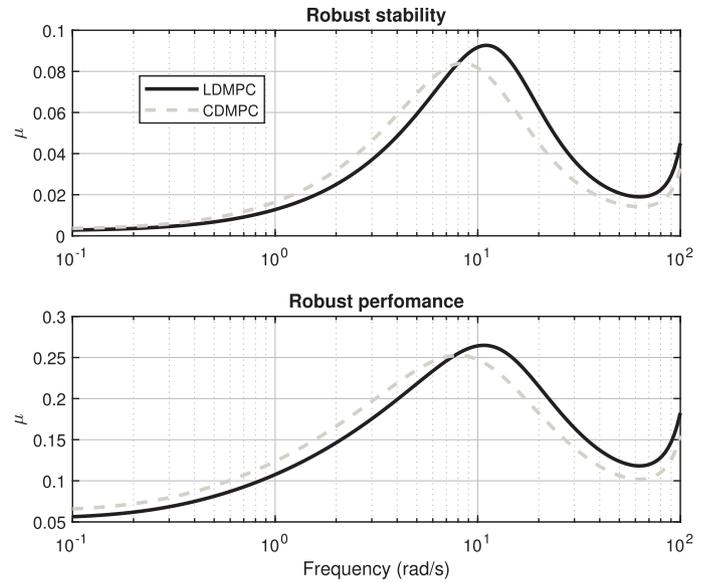
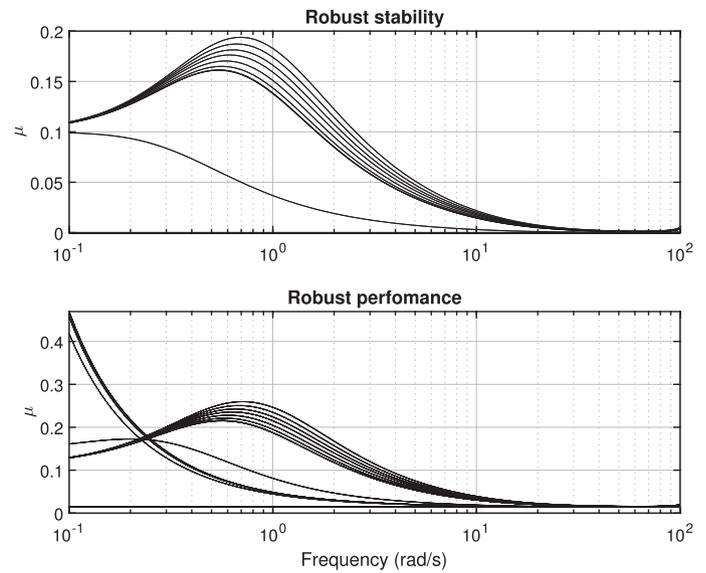
TABLE 2 Constrained performance results

			CDMPC	LDMPC
ISU	Joint 1	4982.28	4906.13	
	Joint 2	3362.26	2906.77	
	Joint 3	8753.17	7684.17	
ISE	Joint 1	2.57e+06	2.55e+06	
	Joint 2	4.28e+05	3.79e+05	
	Joint 3	6.79e+04	6.46e+04	
PO (%)	Joint 1	0.71	0.72	
	Joint 2	0.63	0.64	
	Joint 3	4.50	2.75	
TVC	Joint 1	505.79	27.95	
	Joint 2	325.21	23.27	
	Joint 3	94.62	24.30	

Abbreviations: CDMPC, classical discrete model-based predictive control; LDMPC, Laguerre DMPC.

was ≈ 0.0007 second. In spite of CDMPC be a bit faster than LDMPC, due to the linear matrix inequality in Hildreth's algorithm that spend more time to provide a solution, the LDMPC increased the performance of closed-loop system reasonably, justifying its use. It is important to emphasize that if LDMPC not been used to enhance the time solution of Hildreth's Quadratic, the elapsed time be much longer, especially regarding hardwares with low processing-capacity. This feature is very important, especially in the real-time applications that the controller must calculate the control signal solution within each sampling time.

These experimental results can confirm that the LDMPC, in the constrained case, enhance the control horizon effect and can capture the system dynamics better under the same horizon window used in the classic approach, hence, it improves the control-loop performance using less power. Moreover, the Laguerre function can reach a satisfactory performance without long control horizon. By contrast to CDMPC where is necessary to increase N_c and N_p to achieve the same LDMPC's performance. Consequently, this can increase the implied computational burden dramatically.

FIGURE 12 Robust analysis for unconstrained case**FIGURE 13** Robustness analysis for LDMPC constrained case. LDMPC, Laguerre DMPC

In Figure 12 it is shown the μ -plots of LDMPC and CDMPC for unconstrained case. Observe that for both cases, RS and RP meet the robustness requirements. Therefore, the Laguerre controller can guarantee the robust stability and robust performance in a similar way as the classic method can. In this example, using the unrestricted LDMPC, this controller has achieved a μ -plot that is laying on top of the CDMPC plot, since LDMPC was exploring a less conservative tuning (despite more conservative in terms of power consumption) as depicted by the performance indices shown in Table 1. In spite of being on top, LDMPC still has guaranteed robustness in the sense of μ -robustness analysis, showing that the Laguerre function do not degrade, considerably, the robustness properties of the DMPC control-loop.

In the constrained case using Hildreth's quadratic programming, the robust analysis was done for each gain $K_{\text{mpc}} = \frac{-(L_{\text{Mat}}\eta_{\text{restr}})}{x_a(k)}$, which was calculated through η_{restr} from Equation (35) for every $k = 1, \dots, \text{End}_{\text{time}}$ iteration of the simulation, as shown in the flow chart in Figure 3. The results can be analyzed in Figure 13, where it is shown the μ -curves of LDMPC that has the peak values less than one for the whole set of gains, confirming that the constrained closed-loop system achieves robust performance and robust stability to modeled uncertainties. In addition, in Figure 14 it is shown the closed-loop eigenvalues for LDMPC constrained case, where all poles are inside the unit circle, ensuring the stability of system when the control signal is constrained by Hildreth's quadratic programming.

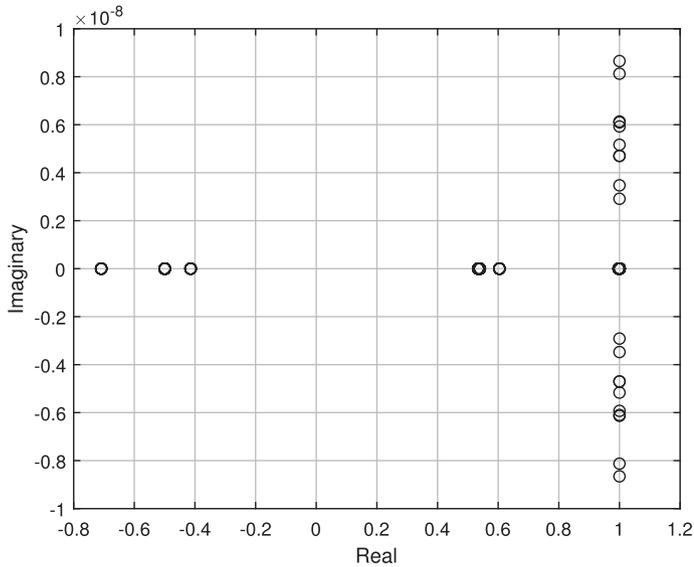


FIGURE 14 Closed-loop eigenvalues for LDMPC constrained case. LDMPC, Laguerre DMPC

5 | CONCLUSIONS

This article reviews a discrete-time predictive controller using Laguerre function applied to the problem of angular position control of a MIMO robot arm-manipulator. The work contends for the potential benefits of the Laguerre's function within DMPC applied to MIMO systems control in the constrained and unconstrained cases, regarding input constraints. In short, the disadvantage of the investigated control method, with respect to CDMPC, is the time spent to tune the system, because it uses a set of Laguerre's function tuning parameters for each input of the system.

The closed-loop performance of both MPC controllers were benchmarked with the DLQR, justifying the effectiveness of the closed-loop performance of LDMPC. Furthermore, for unconstrained and constrained cases, the LDMPC still keeping robustness and stability margins as is confirmed by μ -curves and poles location, respectively.

This work also showed that the LDMPC algorithm can reduce the computational load and runs faster than CDMPC, due to its orthogonal property, which results in a parsimonious future control trajectory with few parameters. Thus, LDMPC does not require a large control horizon to reach a satisfactory control-loop performance. In essence, large N_c can be obtained with a small number of N .

Despite the great number of tuning parameters in the LDMPC design, it is important to remark that it brings more flexibility to the designer and opens up new solutions for the model-based predictive control area, with smaller control and controlled variable variances and consequently less power consumption. Observe, for instance, in Tables 1 and 2, that the power consumption, associated to the *ISU* index, was smaller for the LDMPC, whereas the tracking performance, associated to the *ISE* index, was also smaller for the LDMPC. This means that LDMPC is solving the same problem more efficiently and using less power to do so. Furthermore, the Hildreth's quadratic programming can reduce the variance incremental power control and ensure that constraint not is broken when the incremental control is constrained.

The increasing number of tuning parameters within new complex control algorithms such as LDMPC is a reality that the academic and industrial fields must handle in some way in the near future. For example, CDMPC is already a product available in the market and its three term tuning configuration, respective to the output horizon, control horizon and energy weighting factor, for SISO loops, are becoming a new standard among technicians and engineers. It is also remarkable that it has started in the decade of 1970 and is still undergoing a transition stage. As for the LDMPC, which is just an augmentation of the classic case, our intention is now to investigate ways to correlate what is known by technicians and engineers, for example, respective to PID tuning, to be encompassed into the Laguerre's function procedure, in order to make it hidden to the end-user but keeping its advantages in terms of smaller variances and increased performance. This would benefit the automatic control marked with state-of-the-art predictive control algorithms, whilst keeping a well-known front-end interface to the end-user.

At last, since the Laguerre network, associated with DMPC, has brought reduced variance results, a stochastic Laguerre function version can be developed to solve stochastic problems, such as the approach within the Generalized Predictive Control (GPC) area, based on stochastic process models. The study presented in this article gives substantial

proof of concept that not only the computational load could be reduced, but also that a minimum variance cost is possible against non Laguerre-based minimum variance algorithms such as the GPC.

ACKNOWLEDGEMENT

The authors thankfully acknowledge the financial support of the Brazilian National Council for Scientific and Technological Development (CNPq) under grant 142414/2018-2 and the project 408559/2016-0.

ORCID

Tarcisio Carlos F. Pinheiro  <https://orcid.org/0000-0002-2753-0963>

REFERENCES

1. Siciliano B, Sciavicco L, Villan L, Oriolo G. *Robotics: Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing series. London, UK. Springer-Verlag; 2009.
2. Cubero S. . *Industrial Robotics: Theory, Modelling and Control*. Advanced Robotic Systems International. Mammendorf, Germany. Pro Literatur Verlag; 2007.
3. Rossiter JA, Haber R. The effect of coincidence Horizon on predictive functional control. *Processes*. 2015;3(1):25-45.
4. Dovžan D, Škrjanc I. Predictive functional control based on an adaptive fuzzy model of a hybrid semi-batch reactor. *Control Eng Pract*. 2010;18(8):979-989.
5. Causa J, Karer G, Núñez A, Sáez D, Škrjanc I, Zupančič B. Hybrid fuzzy predictive control based on genetic algorithms for the temperature control of a batch reactor. *Comput Chem Eng*. 2008;32(12):3254-3263.
6. Potočnik B, Mušič G, Škrjanc I, Zupančič B. Model-based predictive control of hybrid systems: a probabilistic neural-network approach to real-time control. *J Intell Robot Syst*. 2008;51(1):45-63.
7. Quevedo DE, Aguilera RP, Geyer T. Model Predictive Control for Power Electronics Applications. In: Raković S., Levine W. (eds) *Handbook of Model Predictive Control*. Control Engineering, Birkhäuser, Cham, 2019. https://doi.org/10.1007/978-3-319-77489-3_23.
8. Eren U, Prach A, Koçer BB, Raković SV, Kayacan E, Açıkmeşe B. Model predictive control in aerospace systems: current state and opportunities. *J Guid Control Dyn*. 2017;40(7):1541-1566.
9. Wang L. Discrete model predictive controller design using Laguerre functions. *J Process Control*. 2004;14(2):131-142.
10. Schwickart T, Voos H, Darouach M, Bezzaoucha S. A flexible move blocking strategy to speed up model-predictive control while retaining a high tracking performance. Paper presented at: Proceedings of the 2016 European Control Conference. Aalborg, Denmark: IEEE; 2016:764-769.
11. Cagienard R, Grieder P, Kerrigan EC, Morari M. Move blocking strategies in receding horizon control. *J Process Control*. 2007;17(6):563-570.
12. Shahzad A, Kerrigan EC, Constantinides George A. A warm-start interior-point method for predictive control; Coventry, England; 2010:949-954.
13. Otta P, Santin O, Havlena V. *Measured-State Driven Warm-Start Strategy for Linear MPC*. Linz, Austria: IEEE; 2015:3132-3136.
14. Ferreau HJ, Bock HG, Diehl M. An online active set strategy to overcome the limitations of explicit MPC. *Int J Robust Nonlinear Control IFAC-Affiliat J*. 2008;18(8):816-830.
15. Herceg M, Jones CN, Morari M. Dominant speed factors of active set methods for fast MPC. *Opt Control Appl Methods*. 2015;36(5):608-627.
16. Richter S, Mariéthoz S, Morari M. High-speed online MPC based on a fast gradient method applied to power converter control. Paper presented at: Proceedings of the 2010 American Control Conference. Marriott Waterfront, Baltimore, MD, USA: IEEE; 2010:4737-4743; IEEE.
17. Abdullah M, Rossiter JA. Utilising Laguerre function in predictive functional control to ensure prediction consistency. Paper presented at: Proceedings of the 2016 UKACC 11th International Conference on Control (CONTROL). Belfast, UK: IEEE; 2016:1-6.
18. Rossiter J, Anthony WL, Valencia-Palomo G. Efficient algorithms for trading off feasibility and performance in predictive control. *Int J Control*. 2010;83(4):789-797.
19. Wang L. Continuous time model predictive control design using orthonormal functions. *Int J Control*. 2001;74(16):1588-1600.
20. Wang L, Freeman CT, Chai S, Rogers E. Multivariable repetitive-predictive control of a robot arm with experimental results. *IFAC Proc Vol*. 2011;44(1):7672-7677.
21. Wahlberg B. System identification using Laguerre models. *IEEE Trans Automat Control*. 1991;36(5):551-562.
22. Akçay H, Ninness B. Orthonormal basis functions for modelling continuous-time systems. *Signal Process*. 1999;77(3):261-274.
23. Olivier PD. System identification using Laguerre functions: simple examples. Paper presented at: Proceedings of the 29th Southeastern Symposium on System Theory; Cookeville, TN: IEEE, 1997:457-459.
24. Tuma M, Jura P. Dynamical system identification with the generalized Laguerre functions. Paper presented at: Proceedings of the 2015 7th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT); 2015:220-225; IEEE.
25. Wang L. *Model Predictive Control System Design and Implementation Using MATLAB®*. Advances in Industrial Control, Second edition. London, England: Springer-Verlag; 2009.
26. Yakub F, Mori Y. Model predictive control based on Kautz function for autonomous ground vehicle path following control application. Paper presented at: Proceedings of the 2014 Proceedings of the SICE Annual Conference (SICE); 2014:1035-1040; IEEE.

27. Mingzhu X, Yiping J, Cunzhi P, Zhanzhong W. An incremental predictive functional control based on Kautz model. Paper presented at: Proceedings of the 2014 SICE Annual Conference (SICE); Nanjing, China: IEEE; 2009:1-5.
28. Dubravić A, Šehić Z, Burgić M. Orthonormal functions based model predictive control of pH neutralization process. *Tehnički vjesnik. Osijek, Croatia*, 2014;21(6):1249-1253.
29. Yin G, Wang Z, Jin XJ. Active steering of autonomous vehicle using model predictive control with Legendre function. Paper presented at: Proceedings of the 2016 Chinese Control and Decision Conference (CCDC). Yinchuan, China: IEEE; 2016:3277-3281.
30. Jamil N, Chen X, Cloninger A. Hildreths algorithm with applications to soft constraints for user interface layout. *Journal of Computational and Applied Mathematics*. 2015;288:193-202.
31. Abdullah M, Idres M. Fuel cell starvation control using model predictive technique with Laguerre and exponential weight functions. *J Mech Sci Technol*. 2014;28(5):1995-2002.
32. Gu D-W, Petkov P, Konstantinov MM. *Robust Control Design with MATLAB®*. Advanced Textbooks in Control and Signal Processing London, England: Springer-Verlag; 2013.

How to cite this article: Pinheiro TCF, Silveira AS. Constrained discrete model predictive control of an arm-manipulator using Laguerre function. *Optim Control Appl Meth*. 2020;1–20. <https://doi.org/10.1002/oca.2667>