LQR hybrid approach control of a robotic arm two degrees of freedom

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 17 (2016) pp 9221-9228

LQR Hybrid Approach Control of a Robotic Arm Two Degrees of Freedom

Jaime Humberto Carvajal Rojas*, Ruthber Rodriguez Serrezuela*
Jair Adrián Quimbayo López*and Karen Lorena Roa Perdomo*

** Faculty of Electrical Engineering and Biomedical, Universidad Antonio Nariño, Republic of Colombia.*

Abstract
The aim of this paper is to propose a system LQR control a robotic arm with two degrees of freedom considering that its operating principle is based on one of the best-known examples that is the robotic arm. First, we consider its mathematical modeling to be able to keep the robotic arm perpendicular in the presence of disturbances by applying appropriate control force. To do this, we consider basic mathematical foundations of classical optimal control and linear quadratic regulator that allows us to develop the dynamic system of equations that will govern the control of the robotic arm. After we will establish the linearized equations system, which gives way to the Simulink simulation process with Mathlab.

INTRODUCTION
Currently techniques designs associated with optimal drivers switched and hybrid dynamic systems are a methodology developed for various types of modern controllers (see, for example, [1], [2], [3]). This is due to the great advances in computer systems that allow you to apply numerical methods for the development and implementation of systems LQ constrained optimization and multivariable systems control (see, eg [3], [4], [5]). In all, real world applications should take into account the constraints present in the system; for example, the inputs or outputs always have values and maximum or minimum states, which are necessary to know in order to operate on these ranges, for optimal performance levels are associated with operating points close to these limits restriction (see [6], [7], [8]]). Failure to consider these constraints implies a direct penalty on system performance. The main objective of our contribution is to apply a consistent computational algorithm for a robotic arm with two degrees of freedom in the presence of piecewise constant input control.
In our work, the application of a numerical method based on a combination of a classical scheme relaxation and a projection approach is proposed. In addition, it should be noted that the proposed algorithm can be effectively used to carry out the synthesis of controllers associated with conventional linear systems and some switched linear systems.

Taking into account that a switched general system where a class of models of dynamic present two types are constituted: continuous and discrete (see for example [9]). To understand how such systems can be operated efficiently, both types of dynamics must be considered during the design process of optimal control.
The preliminary mathematics of optimal control described as hybrid and modeling robotic arm two degrees of freedom which is also done through simulations may be obtained and the results discussed.
The procedure was not based on the solutions of backward differential equations of Riccati, and the optimal control had to be recalculated for each new final state. Calculating the nonlinear gain using the Hamilton-Jacobi-Bellman equation and convex optimization techniques applied in the algorithm has also been made in [10].

FUNDAMENTALS OF MATHEMATICS OPTIMAL CONTROL CLASSIC
We consider a dynamical system represented by the following linear equation:

$$\begin{cases} \dot{x}(t) = f\big(t, x(t), u(t)\big), & t \in [t_0\ ,\ t_f] \\ x(t_0) = x_0 \end{cases} \tag{1}$$

Where $f : [t_0\ , t_f]$x $\boldsymbol{R}^n$x $\boldsymbol{R}^m \rightarrow \boldsymbol{R}^n$. Mapping $u(.) : [t_0\ , t_f] \rightarrow \boldsymbol{R}^m$ month called control where $x_0 \in \boldsymbol{R}^n$is the primary phase and the response of (1) is a function continuous $x(.) : [t_0\ , t_f] \rightarrow \boldsymbol{R}^n$ called path state control $u(.)$.

Now we assume that for any x_0 and any controls $u(.)$there is one answer to the equation (1).Performance control applied to the system as follows is also evaluated:

$$J\big(u(.)\big) = \int_{t_0}^{t_f} g\big(t, x(t), u(t)\big)dt + h(x\big(t_f\big)) \tag{2}$$

Where $g : [t_0\ , t_f]$x $\boldsymbol{R}^n$x $\boldsymbol{R}^m \rightarrow$ $\boldsymbol{Ryh} : \boldsymbol{R}^n \rightarrow \boldsymbol{R}$.The term of the right side of the equation described above is called 'runing cost' and the term on the left is called 'cost terminal'.It is common to find optimal control restrictions in the states and control,they are characterized as follows:

$$x(t) \in S, u(t) \in U, \forall t \in [t_0, t_f].$$

Where $S \subseteq \boldsymbol{R}^n$y $U \subseteq \boldsymbol{R}^m$. $u(.)$ Or control is called admissible control, and the pair $(x(.), u(.))$ Admissible if :

1. $u(.) \in U$.
2. $x(.)$is the only continuous solution (1).
3. the restriction status is satisfied.
4. $t \rightarrow f(t, x(t), u(t)) \in L_1[t_0, t_f]$.

The group of admissible controls is framed by $U_{adm}[t_0, t_f]$.

PONTRYAGIN'S PRINCIPLE OF MAXIMUM

The principle of maximum is very important in the optimal control theory. It says that any optimal control along with state trajectory must satisfy the condition called Hamiltonian system. In addition to the mathematics of the maximum principle it is easy to maximize the Hamiltonian system as the main topic.

Maximum principle: let $x^*(.), u^*(.))$. An optimal pair of classical optimal control, then there is a continuous function $p(.): [t_0, t_f] \rightarrow \boldsymbol{R}$

$$\begin{cases} \dot{p}(t) = f_t(x^*(t), u^*(t))^T p(t) + g_x(t, x^*(t), u^*(t)), & t \in [t_0, t_f] \\ p(t_f) = -h_f(x^*(t_f)) \end{cases}$$

Where x is the partial derivative, o be;

$$f_x = \frac{\partial f}{\partial x}, \quad g_x = \frac{\partial g}{\partial x}, h_x = \frac{\partial h}{\partial x}$$

$$H(t, x^*(t), u^*(t), p(t)) = \frac{max}{u \in U} H(t, x^*(t), u, p(t)),$$
$$t \in [t_0, t_f].$$

Where U represents the set of admissible controls plus:

$$H(t, x, u, p) = \langle p, f(t, x, u) \rangle - g(t, x, u), \quad (t, x, u, p) \in [t_0, t_f] X \boldsymbol{R}^n X \boldsymbol{U} X \boldsymbol{R}^n.$$

With $\langle ., . \rangle$Represents the standard domestic product in $\boldsymbol{R}^n$.

THE HAMILTON-JACOBI-BELLMAN EQUATION AND DYNAMIC PROGRAMMING SYSTEM

We consider a dynamic system as follows:

$$\begin{cases} \dot{x}(t) = f(t, x(t), u(t)), & t \in [t_0, t_f] \\ x(t_0) = x_0 \end{cases}$$

It seeks to minimize performance control applied to the given system:

$$J(u(.)) = \int_{t_0}^{t_f} g(t, x(t), u(t))dt + h(x(t_f))$$

It should be noted that the initial time $t_0$and the initial state $x(t_0) = x_0$which are represented in the problem remain static.

REGULATOR LINEAR QUADRATIC

The following linear system we consider as follows:

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) + b(t), \quad x(t_0) = x_0$$

where:
$A(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{n\,x\,n}], B(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{n\,x\,m}]$, $b(.) \in \boldsymbol{L}_2[t_0, t_f; \boldsymbol{R}^n]$

$\boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{nxn}]yB(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{nxm}]$ representing the classical Lebesgue spaces associated with matrix functions defined by $[t_0, t_f]$. That is:

$$\boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{nxn}] = \{F: [t_0, t_f] \rightarrow \boldsymbol{R}^{nxn} | \|F\|_\infty = ess \sup_{t \in [t_0, t_f]} |F(t)| < \infty\}$$

Similarly $\boldsymbol{L}_2[t_0, t_f; \boldsymbol{R}^n]$represent the classic spaces of vector functions associated with Lebesgue square integrable also defined by $[t_0, t_f]$].

$$\boldsymbol{L}_2[t_0, t_f; \boldsymbol{R}^n] = \{u(.): [t_0, t_f] \rightarrow \boldsymbol{R}^n | \int_{t_0}^{t_f} u(t)^T u(t)dt < \infty\}$$

Given the above quadratic regulator control, performance as applied to the system takes the following form:

$$J(u(.)) = \frac{1}{2}\int_{t_0}^{t_f} \langle Q(t)x(t), x(t) \rangle + 2\langle S(t)x(t), u(t) \rangle + \langle R(t)u(t), u(t) \rangle dt + \frac{1}{2}\langle G_x(t_f), x(t_f) \rangle \quad (3)$$

Where:
$G \in \boldsymbol{R}^{n\,x\,n}, Q(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{n\,x\,n}]$
$S(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{n\,x\,m}], R(.) \in \boldsymbol{L}_\infty[t_0, t_f; \boldsymbol{R}^{m\,x\,m}]$

In addition $t \in [t_0, t_f]$:

$G \geq 0, Q(t) \geq 0, R(t) \geq \delta I \;\; con: \delta > 0$
It is assumed that the allowable square integrable Control $[t_0, t_f]$, and there is no restriction, namely:

$U_{adm}[t_0, t_f] = \{u(.) | u(.) \in \boldsymbol{L}_2 [t_0, t_f; \boldsymbol{R}^m]\}$

The obstacle presented by the linear quadratic regulator is seeking control signal $u^{opt} \in U_{adm}[t_0, t_f]$which performs performance evaluation given by (3) that seeks to reduce as possible, O be:

$$J(u^{opt}(.)) = \inf_{u \in U_{adm}[t_0, t_f]} \{J(u(.))\}$$

DYNAMIC SYSTEM OF THE ROBOTIC ARM.

Then a reference system is presented as shown in Figure 1, where $l_1$is the length of one link; $l_{c1}$and l_{c2} are the distances to the center of the masses of the links; q_1 and q_2 angular displacements are made by the links.

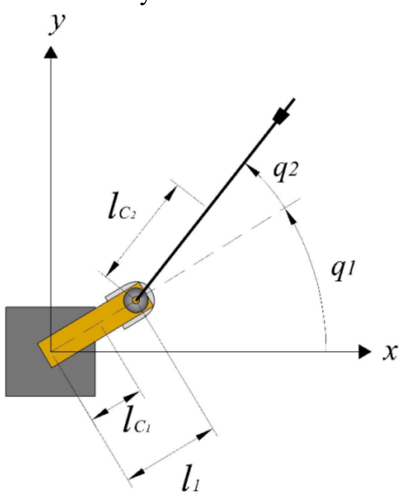

Figure 1: Modeling robotic arm

By Euler-Langrange methodology are the equations describing the system behavior.

$M((q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau$

Where $q, \dot{q}, \ddot{q} \in R^{nx1}$are the vector position, velocity and acceleration of the arm, respectively, M (q) $\in R^{nx1}$is the inertia matrix, C $(q, \dot{q}) \in R^{nx1}$is the matrix of Coriolis and centripetal force, g $(q) \in R^{nx1}$is the gravitational torque, $\tau \in R^{nx1}$is the torque applied and $\tau_d \in R^{nx1}$represents external shocks.

Simplification of operations described above equation is performed.

$\theta_1 = m_1 l_{c1}^2 + m_2 l_1^2 + l_{c1}$

$\theta_2 = m_2 l_{c2}^2 + l_{c2}$

$\theta_3 = m_2 l_1 l_{c2}$

$\theta_4 = m_1 l_{c1} + m_2 l_1$

- m_1= total mass of the first link [kg]
- l_1= length of the first link [m]
- l_{c1}= distance to the center of mass of the first link [m]
- I_{c1}= moment of inertia of the first link relative to the axis passing through its center of mass [k gm^2]
- m_2= total mass of the second link [kg]
- l_{c2}= distance to the center of mass of the second link [m]
- I_{c2}= moment of inertia of the second link relative to the axis passing through its center of mass link [m]
- g= acceleration of gravity [m/s^2]

Thus we can write the matrices M (q), C $(q, \dot{q})$ y g (q) as follows:

$m_{11} = \theta_1 + \theta_2 + 2\theta_3 \cos(q_2)$

$m_{12} = m_{21} = \theta_2 + \theta_3 \cos(q_2)$

$m_{22} = \theta_2$

$c_{11} = -\theta_3 \sin(q_2) q_2$

$c_{11} = -\theta_3 \sin(q_2) q_1 - \theta_3 \sin(q_2) q_2$

$c_{21} = \theta_3 \sin(q_2) q_1$

$c_{22} = 0$

$g_1 = \theta_4 g \cos(q_1) + \theta_5 g \cos(q_1 + q_2)$

$g_2 = \theta_5 g \cos(q_1 + q_2)$

$\det M = m_{11} m_{22} - m_{21} m_{12}$

$T_1 = \tau - c_{11}\dot{q}_1 - c_{12}\dot{q}_2 - g_1$

$T_2 = 0 - c_{21}\dot{q}_1 - c_{22}\dot{q}_2 - g_2$

The system matrices would be simplified as follows:

$$M(q) = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

$$g(q) = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

Solving equation:

$\ddot{q} = M(q)^{-1} [\tau - C(q, \dot{q})\dot{q} - g(q)]$

We will state variables

$x_1 = q_1, x_2 = \dot{q}_1, x_3 = q_2, x_4 = \dot{q}_2, \dot{x}_1 = x_2$

$\dot{x}_2 = (T_1 m_{22} - T_2 m_{12})/detM$

$\dot{x}_3 = x_4$

$\dot{x}_4 = (T_2 m_{11} - T_1 m_{21})/detM$

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(I+ml^2)b}{I(M+m)+(Mml^2)} & \frac{m^2gl^2}{I(M+m)+(Mml^2)} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-mlb}{I(M+m)+(Mml^2)} & \frac{mgl(M+m)}{I(M+m)+(Mml^2)} & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{I+ml^2}{I(M+m)+(Mml^2)} \\ 0 \\ \frac{ml}{I(M+m)+(Mml^2)} \end{bmatrix} \quad (4)$$

$$Y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

LOCAL CONTROL

For the process of linearized equations of the robotic arm, Taylor series are used to design a state feedback controller.

$$f_a = (x,u) = f_a(x_r,u_r) + \frac{\partial f}{\partial x}|x_r,u_r\,(x-x_r) + \frac{\partial f}{\partial u}|x_r,u_r\,(u-u_r)$$

Being x∈ $R^{nx1}$the state vector, u is the input signal to the robotic arm; x_r andx_r are the equilibrium points, then to find and evaluate the partial derivative f (x_r ,u_r) at breakeven, it is f (0,0):

$$\frac{\partial f}{\partial x}|(0,0) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f_2}{\partial x_1} & 0 & \frac{\partial f_2}{\partial x_3} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f_4}{\partial x_1} & 0 & \frac{\partial f_4}{\partial x_3} & 0 \end{bmatrix}$$

$$\frac{\partial f}{\partial u}|(0,0) = \begin{bmatrix} 0 \\ \frac{\partial f_2}{\partial u} \\ 0 \\ \frac{\partial f_4}{\partial u} \end{bmatrix}$$

To analyze the robotic arm propose two equilibrium points: the first when the arm is located in the upward position, Figure 2, obtained as equilibria points $x_{d1}= \pi/2$, $x_{d3} = 0$ y $u_d = 0$

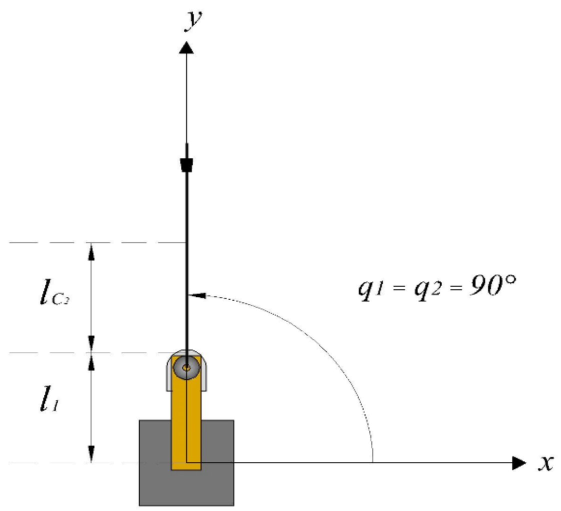

Figure 2: The robotic arm in the up position

The second point is located when the second link is up, this position is known as mean, Figure 3, giving as points of equilibrium $x_{d1}= -\pi/2$, $x_{d3} = \pi$ y $u_d = 0$

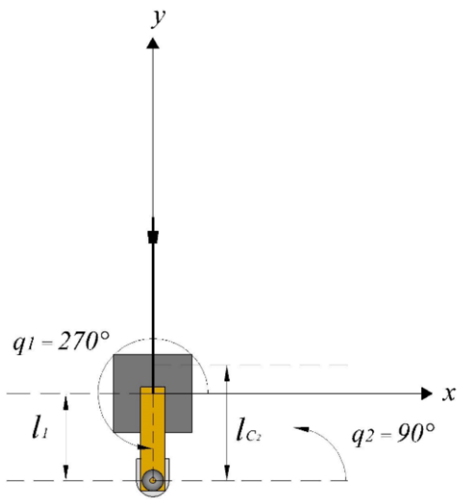

Figure 3: The robotic armmidposition

The following suggested parameters from the design done in Solidwork:

Table 1: Design parameters done in Solidwork

θ_1	0.062402
θ_2	0.054284
θ_3	0.025912
θ_4	0.80278
θ_5	0.31406

The following suggested parameters from the design done in Solidwork:

Taken into account the equilibrium points and parameters of the manufacturer, the linear model formula as follows:

Above

$\dot{x}_1 = AX + BU$

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 17 (2016) pp 9221-9228
© Research India Publications. http://www.ripublication.com

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 108.69545 & -24.959145 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -112.387 & 85.0663 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 13.930735 \\ 0 \\ -20.58139 \end{bmatrix} u \qquad (5)$$

Half

$\dot{x}_1 = AX + BU$

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -108.69545 & 24.959145 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 105.039 & 35.148095 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\Phi} \\ \ddot{\Phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 13.9307 \\ 0 \\ -7.281015 \end{bmatrix} u \qquad (6)$$

Arrays are described above in the continuous space, to need credit in our system: $x_{k+1} = G\, x_k + K\, u_k 0.002$ seconds.

Our goal is to find a permissible control law, which minimizes the value of the second functional level costs

$$I(u(.)) = \frac{1}{2}\int_{t_0}^{t_f} [X^T(t) + QX(t) + u^T(t) + Ru(t)]dt \rightarrow \underset{u(.)}{min} \qquad (7)$$

The resulting Linear Quadratic Regulator has the following form

$$U^{opt}(t) = -R^{-1}(t)B^t(t)P(t)X^{opt}(t) \qquad (8)$$

where P (t) is a solution of the Riccati equation

$$P(t) = -(A^T P(t) + P(\dot{t})A(t) + P(t)B(t)R^{-1}(t)B^T(t)P(t) - Q(t) \qquad (9)$$

With the final condition

$$P(t_f) = 0 \qquad (10)$$

ANALYSIS OF RESULTS

In Figure No 4, we can observe the behavior of the two links of the robot which were modeled in Solidwork (See figure No. 5) and simulated in Matlab, when we move the robotic arm to $x = 1$position and $y = 1$. In control systems it is undesirable that there is a response with overshoot or much oscillating. It usually turns out that the controlled system has an overshoot between 0% and 20% with the least destabilizing time possible. Here we can see that the system has a lot of momentum, approximately 24%. Likewise, a rise time of 0.35 seconds, a peak time of 0.45 seconds and a settling time of 2.5 seconds. The steady state value is 0.019, while the desired value is 0.02 respectively.

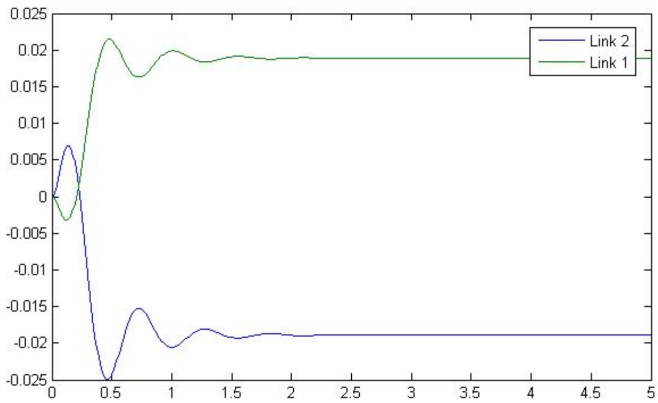

Figure 4: No displacement of the two links of the robot

In Figure No 5, we can observe the behavior of the two links modeling robot handled with continuous LQR controller. In the, we can see how the over-momentum in the system disappears, setup times are improved from 2.5 to 1.5 seconds and the same steady state value of 0.019, respectively is obtained.

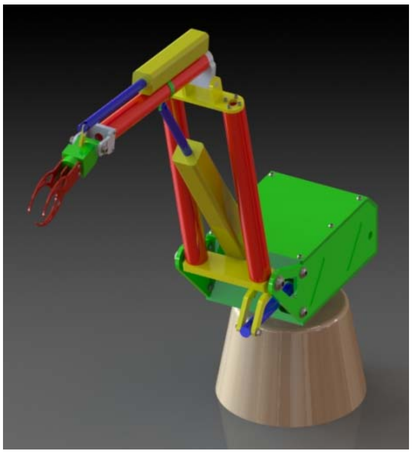

Figure 5: Mount Solidwork

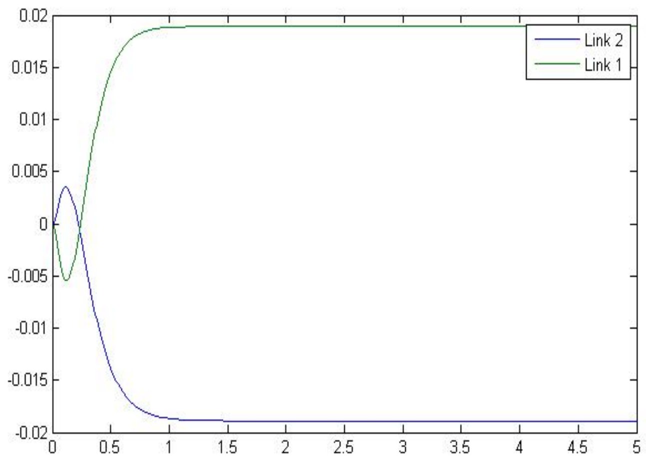

Figure 6: Move the two links of the robot with continuous monitoring LQR

We will now present a conceptual algorithm for a specific calculation of the optimal pair $(U^{opt}, X^{opt}(.))$ in this mechanical example. We refer to the general mathematical tool related to the LQR Hybrids.

Algorithm

1. Select a $t_{swi} \in [0, t_f]$, put an index $j = 0$
2. Solve the Riccati equation (9) (10) in the time intervals $[0, t_f] \cup [t_{swi}, t_f]$
3. Solve the initial problem (4) for (8)
4. Calculate $X_4(t_{swi}) + 10, if |X_4(t_{swi}) + 10| \cong \in$ for a prescribed accuracy $\in > 0$ then stops. plus $j = j + 1$improve $t_{swi} = t_{swi} + \Delta t$and return 1
5. Finally, resolver (4) and obtained with initial conditions (the final conditions for the $X(t_{swi})$calculated from it.

In Figure No 6, we can observe the system response discreetly which corresponds to the red line, in the representation, we can observe the presence of an overshoot of up to 15% in the system. Similarly, we can analyze the signal has a settling time of about 20 seconds. The time to reach the peak is 6.5 seconds. Solving our algorithm, we can see that using the hybrid control LQR is possible to improve the response of the system by eliminating the overshoot present (the blue line), similarly, we can see that the signal is stabilized and reaches its value steady state much more fast, about 5 seconds.

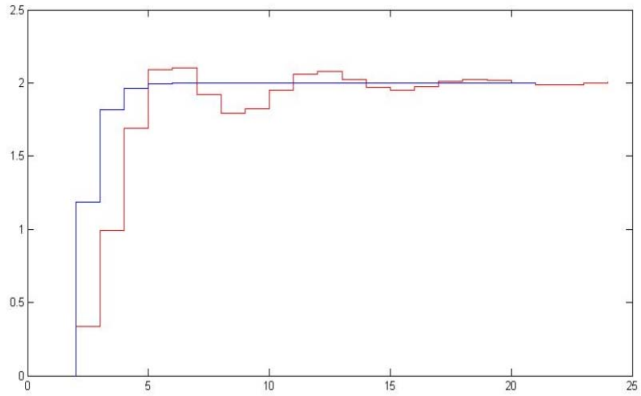

Figure 6: Response robot in discrete time

CONCLUSIONS

This article proposes a new theoretical and computational approach to a specific class of OCP hybrid moved by a robotic arm system. We use the variational structure of nonlinear mechanical systems described by the Euler-Lagrange type hybrid equations or Hamilton equations; one can formulate a multi-objective optimization problem to help address this issue. This problem and theoretical and numerical techniques corresponding multi-objective optimization can be applied effectively to the numerical solution of the initial hybrid OCP and especially so for this particular problem.

The evidence of our results and consideration of the main numerical concepts are performed under some conditions of differentiability and convexity assumptions. These assumptions restrictive softness are motivated by the "classic" structure mechanical hybrid systems under consideration. On the other hand, modern variational analysis product without previous restrictive assumptions softness. Obviously, no smooth variational analysis and corresponding optimization techniques can be considered as a possible mathematical tool for the analysis of discontinuous (eg, variable structure) and impulsive (not smooth) hybrid mechanical systems. Numerical calculations are performed on a Solid work prototype developed and implemented the Matlab logarithm, demonstrating its effectiveness.

Finally, note that the theoretical focus and numerical conceptual aspects presented herein may be extended to some restricted OCP with state and / or mixed limitations. In this case we must choose a suitable discretization procedure for initial use sophisticated OCP and use the corresponding conditions for necessary optimization. It also seems possible to apply the theoretical and computational schemes for some practically motivated nonlinear hybrid and start the OCP in mechanical, for example, optimization problems in the robot dynamics for our case.

REFRENCIAS

[1] Azhmyakov, V., & Raisch, J. (2006). A gradient-based approach to a class of hybrid optimal control problems. IFAC Proceedings Volumes, 39(5), 89-94.

[2] Azhmyakov, V., Attia, S. A., & Raisch, J. (2008, April). On the maximum principle for impulsive hybrid systems. In International Workshop on Hybrid Systems: Computation and Control (pp. 30-42). Springer Berlin Heidelberg.

[3] Rojas, J. H. C. (2007). Modelamiento y diseño de robots industriales. Universidad de la Salle.

[4] Azhmyakov, V., Serrezuela, R. R., & Trujillo, L. G. (2014, October). Approximations based optimal control design for a class of switched dynamic systems. In IECON 2014-40th Annual Conference of the IEEE Industrial Electronics Society (pp. 90-95). IEEE.

[5] Azhmyakov, V., Shamsi, M., Trujillo, L. A. G., & Rojas, J. H. C. (2015). A Robust Pseudo-Spectral Method for a Class of Optimal Control Problems. IFAC-PapersOnLine, 48(14), 253-258.

[6] Falk, R. S. (1973). Approximation of a class of optimal control problems with order of convergence estimates. Journal of Mathematical Analysis and Applications, 44(1), 28-47.

[7] Shaikh, M. S., &Caines, P. E. (2007). On the hybrid optimal control problem: theory and algorithms. IEEE Transactions on Automatic Control, 52(9), 1587-1603.

[8] [Cassandras, C. G., Pepyne, D. L., &Wardi, Y. (2001). Optimal control of a class of hybrid systems. IEEE Transactions on Automatic Control, 46(3), 398-415.

[9] Tröltzsch, F. (2010). Optimal control of partial differential equations. Graduate studies in mathematics, 112.

[10] Azhmyakov, V., Rodriguez Serrezuela, R., Rios Gallardo, A. M., & Gerardo Vargas, W. (2014). An Approximations Based Approach to Optimal Control of Switched Dynamic Systems. Mathematical Problems in Engineering, 2014.

[11] Azhmyakov, V., Martinez, J. C., Poznyak, A., & Serrezuela, R. R. (2015, July). Optimization of a class of nonlinear switched systems with fixed-levels control inputs. In 2015 American Control Conference (ACC) (pp. 1770-1775). IEEE.

[12] Christos G.. Cassandras, & Lafortune, S. (2008). Introduction to discrete event systems. Springer.

[13] Mejía, S. G., Scarpetta, J. M. R., & Rodríguez, E. J. A. (2015). Técnicas de control para el balance de un robot bípedo: un estado del arte. Tecnura, 19(43), 139-162.

[14] Wei, S., Uthaichana, K., Žefran, M., & DeCarlo, R. (2013). Hybrid model predictive control for the stabilization of wheeled mobile robots subject to wheel slippage. IEEE transactions on control systems technology, 21(6), 2181-2193.

[15] Sharma, K. D., Chatterjee, A., &Rakshit, A. (2012). A PSO–Lyapunov hybrid stable adaptive fuzzy tracking control approach for vision-based robot navigation. IEEE Transactions on Instrumentation and Measurement, 61(7), 1908-1914.

[16] Alexis, K., Huerzeler, C., & Siegwart, R. (2014). Hybrid predictive control of a coaxial aerial robot for physical interaction through contact. Control Engineering Practice, 32, 96-112.

[17] Taherkhorsandi, M., Mahmoodabadi, M. J., Talebipour, M., & Castillo-Villar, K. K. (2015). Pareto design of an adaptive robust hybrid of PID and sliding control for a biped robot via genetic algorithm optimization. Nonlinear dynamics, 79(1), 251-263.

[18] Ding, J., Abate, A., & Tomlin, C. (2013, June). Optimal control of partially observable discrete time stochastic hybrid systems for safety specifications. In 2013 American Control Conference (pp. 6231-6236). IEEE.

[19] Duan, H., Luo, Q., Shi, Y., &Ma, G. (2013). ? Hybrid Particle Swarm Optimization and Genetic Algorithm for Multi-UAV Formation Reconfiguration. IEEE Computational Intelligence Magazine, 8(3), 16-27.

[20] Raibert, M. H., & Craig, J. J. (1981). Hybrid position/force control of manipulators. Journal of Dynamic Systems, Measurement, and Control, 103(2), 126-133.

[21] Melin, P., & Castillo, O. (2007). An intelligent hybrid approach for industrial quality control combining neural networks, fuzzy logic and fractal theory. Information Sciences, 177(7), 1543-1557.

[22] Borrelli, F., Bemporad, A., Fodor, M., & Hrovat, D. (2001, March). A hybrid approach to traction control. In International Workshop on Hybrid Systems: Computation and Control (pp. 162-174). Springer Berlin Heidelberg.

[23] Melin, P., & Castillo, O. (2003). Adaptive intelligent control of aircraft systems with a hybrid approach combining neural networks, fuzzy logic and fractal theory. Applied Soft Computing, 3(4), 353-362.

[24] Moura, S. J., Fathy, H. K., Callaway, D. S., & Stein, J. L. (2011). A stochastic optimal control approach for power management in plug-in hybrid electric vehicles. IEEE Transactions on control systems technology, 19(3), 545-555.

[25] Rostalski, P., Besselmann, T., Barić, M., Belzen, F. V., & Morari, M. (2007). A hybrid approach to modelling, control and state estimation of mechanical systems with backlash. International Journal of Control, 80(11), 1729-1740.

[26] Tabuada, P. (2009). Verification and control of hybrid systems: a symbolic approach. Springer Science & Business Media.

[27] Uzam, M., & Wonham, W. M. (2006). A hybrid approach to supervisory control of discrete event

systems coupling RW supervisors to Petri nets. The International Journal of Advanced Manufacturing Technology, 28(7-8), 747-760.

[28] Borrelli, F., Bemporad, A., Fodor, M., & Hrovat, D. (2006). An MPC/hybrid system approach to traction control. IEEE Transactions on Control Systems Technology, 14(3), 541-552.

[29] Toninelli, A., Bradshaw, J. M., Kagal, L., & Montanari, R. (2005, September). Rule-based and ontology-based policies: Toward a hybrid approach to control agents in pervasive environments. In Semantic Web and Policy Workshop (pp. 42-54).

[30] Corke, P. I., & Hutchinson, S. A. (2001). A new partitioned approach to image-based visual servo control. IEEE Transactions on Robotics and Automation, 17(4), 507-515.

[31] Daafouz, J., Riedinger, P., & Iung, C. (2002). Stability analysis and control synthesis for switched systems: a switched Lyapunov function approach. IEEE transactions on automatic control, 47(11), 1883-1887.

[32] Antsaklis, P. J., & Nerode, A. (1998). Hybrid control systems: An introductory discussion to the special issue. IEEE Transactions on Automatic Control, 43(4), 457-460.