OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. (2012)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/oca.2048

Suboptimal robust linear visual servoing for a delayed underactuated system

A. Benitez-Morales, Omar Santos*,†, Hugo Romero and Luis Enrique Ramos-Velasco

Centro de Investigación en Tecnologías de Información y Sistemas, Universidad Autónoma del Estado de Hidalgo, Pachuca, Hidalgo, CP 42184, Mexico

SUMMARY

In this contribution, a suboptimal robust control law for a specific class of underactuated delayed system is synthesized. The control strategy based on very well-known results for delay-dependent stability considers the time delay involved in the dynamical system, which affects to control signal. This contribution illustrates how the theoretical results can be used to improve the real-time performance of the closed-loop system considered. The delay is introduced into the control system by the vision module, due to the time required to perform the image treatment. In order to show the good performance of the control law proposed, real-time experiments are developed by applying a visual servoing technique on the cart-inverted pendulum system. Obtained results also illustrate how the conservativeness of theoretical results affects the performance of the closed-loop system and the negative effects of delays in the control loop. Furthermore, a robust stability analysis is done to establish the robustness of control law with respect to the amount of delay presented in the system. Copyright © 2012 John Wiley & Sons, Ltd.

Received 9 February 2012; Revised 4 June 2012; Accepted 16 July 2012

KEY WORDS: time delay; suboptimal control; visual servoing; underactuated system

1. INTRODUCTION

Many results for stability and stabilization of time-delay systems have been presented in the last years [1]. A common method used is the Lyapunov–Krasovskii approach [2], where the expertise is crucial in order to propose 'convenient' functional and appropriate majorizations that must be used in order to obtain non-conservative results. A great number of theoretical results are presented in this field; see [3, 4] for robust dependent stability conditions, [5] for stability criterion in fuzzy time-delay systems, or [6] for neural networks with distributed time delay. However, there are no experimental validations reported in most of previously published works. We think that illustrating how to use these theoretical results in order to improve a real control process is very important; additionally, we illustrate by experimental results how to affect in control loop the conservativeness of stability results employed. So, in this contribution, we illustrate the use of theoretical results over an underactuated visual servoing system. An underactuated robot has more degrees of freedom than independent control actuators. Then any nonlinear control law proposed to drive an underactuated system should consider that some degrees of freedom or dynamics cannot be directly controlled. For this reason, control of delayed underactuated systems is not a trivial problem.

Visual servoing approaches use a camera as a sensor inside of the closed loop [7]. In the present paper, a visual approach is used to stabilize the cart-inverted pendulum system. Usually, the vision systems work as follows: the image acquired by the camera must be processed by a vision algorithm

*Correspondence to: Omar Santos, Centro de Investigación en Tecnologías de Información y Sistemas, Universidad Autónoma del Estado de Hidalgo, Pachuca, Hidalgo, CP 42184, Mexico.

†E-mail: omarj@uaeh.edu.mx

implemented on a computer or a special device. The main goal of the vision algorithm is to identify significant objects or features inside the image and so deduce the relationship between spatial position of those objects (features) and the camera body. Some related works previously developed are described later. In [8], a control strategy based on visual servoing applied to a class of dynamic underactuated systems is presented, whereas in [9], authors use an optimal control regulator scheme based on a visual servoing approach without considering the intrinsic time delay appearing in a closed loop.

Time delays in a control loop may induce instability or bounded oscillations in the system. Then they must be considered in the stability analysis or design of the controller strategy. However, until now, a few researchers using control schemes based on visual servoing have considered the time delays induced by image processing. Some of them are Xie *et al*. [10], in that in their work, the time delay is assumed known, and it is compensated using a modified Smith predictor. Furthermore, a predictive PID is implemented, and the predictor removes the time delay of control loop. In [11], the authors proposed a visual servoing implementation where the stability is independent of the amount of time delay. It means that they have used an analytical method based on the Nyquist stability criterion to find the upper limit of time delay. Moreover, some practical implementations considering time delays are presented in [12]. According to recent literature, only a few real-time experiments have been presented considering the time delay induced in the closed loop by a vision system. Then considering this fact is very relevant in dealing and illustrating how to apply a real-time optimal control strategy on a delayed dynamic system having a time-domain representation. Stability analysis in time domain allows to consider nonlinear disturbances [4] and increases the robustness of closed-loop system, as is illustrated in this contribution.

By simple simulations *routines* and *real-time* experiments, we realized that the time delay present in the control loop causes instability in the considered dynamic system. Therefore, in this contribution, a real-time optimal control law is synthesized and applied to cart-inverted pendulum system with delayed behavior (Figure 1). The control law is designed by solving the linear quadratic regulator (LQR) problem using two classical results for stability of time-delay linear systems [3, 4]. So, it considers and minimizes the effects of delay, satisfying sufficient conditions to guarantee the robust stability of closed loop and also compensates the negatives effects produced by nonlinear disturbances. Furthermore, it allows to change the matrix of the controller in order to improve the robustness of the closed-loop system. Nevertheless, this approach delivers conservative results compared with approaches based on linear matrix inequalities (LMIs) approach; see, for example, [13–15].

This paper is organized as follows: preliminaries results supporting this contribution are presented in Section 2. Section 3 gives a description of real-time platform used, whereas the vision system and image treatment algorithm applied in this contribution are shown in Section 4. Furthermore,

Figure 1. Real-time experimental platform using visual servoing.

DOI: 10.1002/oca

Section 5 shows the control strategy proposed to improve the performance of closed-loop delayed system. Finally, real-time experiment results and conclusions are presented in Sections 6 and 7, respectively.

2. PRELIMINARIES

The standard linear state-space model for the considered dynamic system is given by

$$\dot{x}(t) = Ax(t) + Bu(t), x(t) \in \mathbb{R}^4, u(t) \in \mathbb{R}, \tag{1}$$

where $x_1 = x_c$ and $x_2 = \dot{x}_c$ define the cart position and cart velocity, respectively, whereas $x_3 = \theta$ is the pendulum angular position, and $x_4 = \dot{\theta}$ is the pendulum angular rate. Furthermore,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{M_p g}{M_c} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{(M_c+M_p)g}{M_c l_p} & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ \frac{1}{M_c} \\ 0 \\ \frac{1}{M_c l_p} \end{bmatrix},$$

with $u = k_f F_c$, where k_f is the motor constant and u is the control signal in volts applied to DC motor. Values of involved constants in the system are shown in Table I.

Now, the following statements are used to develop our results. Theorem 1 and Proposition 1 give a delay-dependent robust stability criterion. Also, the proposition considers time-varying delay and nonlinear disturbances with respect to current state $x(t)$ and delayed state $x(t - r(t))$.

Theorem 1 ([3]*)*
Let the system be of the form

$$\dot{x} = A_0 x(t) + A_1 x(t - h), \tag{2}$$

with initial conditions

$$x(\theta) = \varphi(\theta), \theta \in [-h, 0].$$

Assume that an uncertain time-invariant delay lies in $[0, \bar{h}]$, that is, $h \in [0, \bar{h}]$. If there exist $P > 0$, $Q > 0$, $V > 0$, and W such that

$$\begin{bmatrix} (1,1) & -W^T A_1 & A_0^T A_1^T V & (1,4) \\ -A_1^T W & -Q & A_1^T A_1^T V & 0 \\ V A_1 A_0 & V A_1 A_1 & -V & 0 \\ (1,4)^T & 0 & 0 & -V \end{bmatrix} < 0, \tag{3}$$

then system (2) is asymptotically stable. Where

$$(1,1) \triangleq (A_0 + A_1)^T P + P(A_0 + A_1) + W^T A_1 + A_1^T W + Q \text{ and } (1,4) \triangleq \bar{h}(W^T + P).$$

Table I. Constants of inverted pendulum [16, 17].

Variable	Name	Value	Unit
M_p	Pendulum mass	0.23	kg
g	Gravity constant	9.81	m/s^2
M_c	Mass of the cart	0.52	kg
l_p	Long pendulum length	0.3302	m
k_f	Constant motor	1	V s/kg

DOI: 10.1002/oca

Proposition 1 ([4])
The system described by

$$\dot{x}(t) = A_0 x(t) + A_1 x(t - r(t)) + f(x(t), t) + g(x(t - r(t)), t), \tag{4}$$

with time-varying delay $r(t)$ satisfying

$$0 \leqslant r(t) \leqslant r_M \qquad \dot{r}(t) \leqslant r_d,$$

having initial conditions

$$x(t_0 + \theta) = \varphi(\theta);\ \theta \in [-r_M, 0],$$

with uncertainties described as follows

$$\|f(x(t), t)\| \leqslant \alpha \|x(t)\| \text{ and } \|g(x(t - r(t)), t)\| \leqslant \beta \|x(t - r(t))\|,$$

where $\alpha > 0$ and $\beta > 0$ are given constants, are asymptotically stable, if there exists a real matrix X, symmetric positive definite matrices P, R, Y, and scalars $\varepsilon_1 \geqslant 0$, $\varepsilon_2 \geqslant 0$ such that the following LMI holds:

$$\begin{bmatrix} (1,1) & -X^{\mathrm{T}} A_1 & P & P & A_0^{\mathrm{T}} A_1^{\mathrm{T}} Y & (1,6) \\ -A_1^{\mathrm{T}} X & (2,2) & 0 & 0 & A_1^{\mathrm{T}} A_1^{\mathrm{T}} Y & 0 \\ P & 0 & -\varepsilon_1 I & 0 & A_1^{\mathrm{T}} Y & 0 \\ P & 0 & 0 & -\varepsilon_2 I & A_1^{\mathrm{T}} Y & 0 \\ Y A_1 A_0 & Y A_1 A_1 & Y A_1 & Y A_1 & -Y & 0 \\ (1,6)^{\mathrm{T}} & 0 & 0 & 0 & 0 & -Y \end{bmatrix} < 0, \tag{5}$$

where

$$(1,1) \triangleq (A + B)^{\mathrm{T}} P + P(A + B) + R + X^{\mathrm{T}} B + B^{\mathrm{T}} X + \varepsilon_1 \alpha^2 I,\ (2,2) \triangleq -(1 - r_d) R + \varepsilon_2 \beta^2 I$$

and

$$(1,6) \triangleq r_M (X^{\mathrm{T}} + P).$$

These are useful results in order to synthesize the linear feedback control law, as will be shown later. However, other results based on LMI approach could be used in order to improve the robustness of the closed loop of delayed system.

3. REAL-TIME PLATFORM ARCHITECTURE

Experimental platform is mainly composed by three modules, as follows: the image treatment module, which uses a vision camera together with a digital signal processor (DSP); the underactuated system, which is a Quanser cart-inverted pendulum model IP01 [17]; and the power interface (Figure 2). The cart-inverted pendulum is composed of a cart that moves linearly through a shaft by using a DC motor (± 6 VDC). The cart has coupled on top a pendulum with free movement; this dynamical system is an underactuated system having two degrees of freedom and only one actuator [18].

Algorithms for image acquisition and treatment together with the control strategy are implemented on a DSP platform in order to control the considered underactuated system. This visual servoing platform based on DSP has been chosen, because it is a dedicated image processing device with a robust code instructions set satisfying any application of digital signal processing. Therefore, the image treatment process and control signal estimation are performed faster than a standard personal computer. In addition, platforms based on DSP are easily adaptable to any change of the variables involved in the system, and it has low distortion in the input signal. DSPs also have a high-accuracy control over the behavior of hardware, and usually, they do not need additional hardware to acquire and process an analog signal. Furthermore, they are reprogrammable (does not

DOI: 10.1002/oca

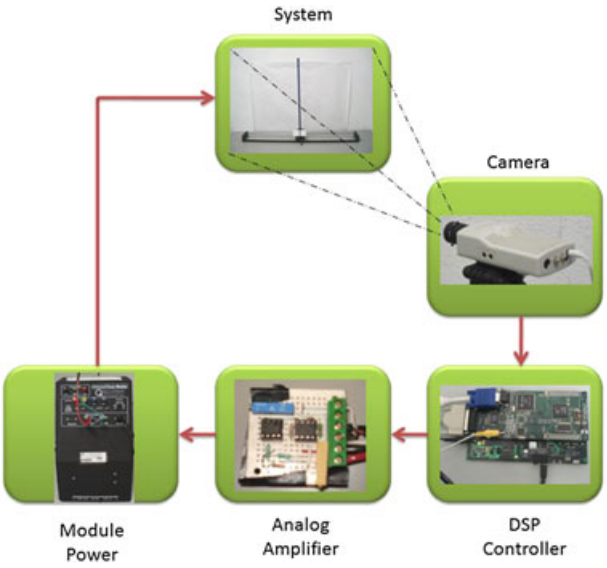


Figure 2. Schematic representation of an underactuated system.

require changing the hardware system), and they have low power consumption because they use a solid-state technology.

In particular, the experimental platform is based on the Texas Instruments TMS320C6000 DSP Imaging Developer's Kit (IDK) [19], which consist of a card containing a DSP together with all necessary interfaces to develop real-time applications. The TMS320C6000 IDK is a kit offering the possibility to acquire images in NTSC or PAL format by using a camera wired through a daughter card to a DSP expansion port. Moreover, it includes software and hardware to acquire and show images captured and processed using libraries and functions developed by the manufacturer. This kit has the following main features: a TMS320C6711 DSP, 150 MHz, capable of drive up to 1200 MIPS, 16 Mbytes SDRAM, 128 kbytes flash ROM, 16 bits of audio codec (TLC320AD535), PC parallel port interface, one port for capturing video in NTSC format, one output port for RGB images with a resolution 640×480 or 800×600 with 16 bits per pixel, and one analog output audio.

However, the DSP kit used has a limitation, because it has no analog or pulse-width modulation output. Then the only one possibility is to use the audio output as control signal. This audio output signal has an amplitude from 0.5 to 1.5 V. Therefore, it is necessary to implement an electronic interface to adjust the voltage level to ± 5 V. Additionally, a DC motor requires an electronic power amplifier to drive it. This power amplifier receives the control signal (± 5 V) and then gives an appropriate signal to be applied to the actuator.

A single cycle of operation for the real-time platform can be summarized as follows: it start with image capturing process, and then it is transmitted to a DSP in order to be processed with a Sobel filter (edge detection). Once the image is segmented, the variables system estimation is performed, and finally, those variables estimated are used to compute the control law signal applied to DC motor.

4. VISION SYSTEM

The vision system used in this contribution is composed of a DSP card described in Section 3. This device together with an image acquisition module is used to capture and process every image frame provided by the camera. The image treatment algorithm is a Sobel filter, which emphasizes the edges in the image captured. Those edges are used to estimate the variables system x_c, $\dot{x}_c$, θ, and $\dot{\theta}$, which denote the cart's linear position, cart's linear velocity, pendulum angular position, and pendulum angular rate, respectively. Therefore, a whole state of system is estimated using only the

DOI: 10.1002/oca

visual information. Furthermore, in order to increase the accuracy of variable estimation approach, a camera calibration process is performed to recover the geometric and projective characteristics of vision sensor. The image acquisition and treatment processes, the camera calibration method, and the estimation of variables system are described later.

4.1. Real-time image processing

Images captured by the camera are in NTSC format [19]. Image treatment is done in real time; it means that the video frame n is stored in the internal RAM memory of DSP to be segmented by a Sobel filter [20]. Applying this filter, we are able to emphasize the edges of a cart-pendulum image in order to identify the interest points. Once the frame n was processed and variables system have been estimated, it is deleted from the internal RAM memory and the next frame $n+1$ is then stored in the memory to repeat the process. Originally, images of 480×720 pixels are captured in RGB format (3 bytes per pixel), but they must be converted to gray level images (1 byte per pixel) to properly apply the Sobel filter. On the basis of these considerations, the amount of memory required to store each video frame is around 1.4 Mb, which is an acceptable memory amount to be stored in the DSP RAM memory.

As mentioned earlier, the Sobel filter is an edge detector algorithm. Technically, it is a discrete differentiation operator that computes an approximation of the image gradient. In this work, horizontal and vertical 3×3 Sobel masks are used (Figure 3(a)); they are applied to images separately, such that the final result is given by the sum of images convoluted with both Sobel operators. The resulting image from applying the Sobel filter to cart-inverted pendulum is shown in Figure 3(b).

4.2. Camera calibration

Camera calibration is referred to as the process of obtaining the geometric and projective relations between the 3D real world coordinates and the 2D image coordinates. These relations are represented by the intrinsic parameters matrix $\mathbf{K}$, the extrinsic parameters contained in the rotation matrix $\mathbf{R}$, and the translation vector $\mathbf{t}$. The calibration method used in this work is known as *homogeneous transformation*, which assumes a *pinhole* camera model. In order to estimate the camera matrix $\mathbf{K}$, the calibration approach considers n real points defined as $p_i = (x_i,\ y_i,\ z_i,\ 1)^{\mathrm{T}}$ and their corresponding image points $q_i = (\rho_i,\ \gamma_i,\ 1)^{\mathrm{T}}$ with $i = 1, 2, \ldots, n$, both in homogeneous coordinates. Therefore, the real and image points are related as follows:

$$\mathbf{Q} = \mathbb{T}\mathbf{P},$$

where $\mathbf{Q} = [q_1, q_2, \ldots, q_n] \in \mathbb{R}^{3\times n}$ and $\mathbf{P} = [p_1, p_2, \ldots, p_n] \in \mathbb{R}^{4\times n}$, with $\mathbb{T} \in \mathbb{R}^{3\times 3}$ acting as a transformation matrix defined as

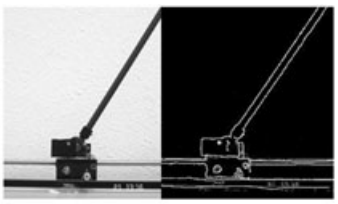
$$\mathbb{T} = [\mathbf{T}_1\ \mathbf{T}_2\ \mathbf{T}_3]^{\mathrm{T}},$$

where $\mathbf{T}_i \in \mathbb{R}^{1\times 4}$ $(i = 1, 2, 3)$. The $\mathbb{T}$ matrix is obtained from the $\mathbf{Q}$ and $\mathbf{P}$ matrices by applying the algorithm described in [21]. This algorithm applies a singular value decomposition on a system of linear equations obtained from $\mathbf{Q} \otimes \mathbb{T}\mathbf{P} = \mathbf{0}$ to estimate the $\mathbb{T}$ matrix, where $\otimes$ denotes the cross product. So, the $\mathbb{T}$ matrix can be rewritten in block matrices as follows:

$$\mathbb{T} = [\Gamma \mid \Lambda] = \Gamma\left[I \mid \Gamma^{-1}\Lambda\right] = \Gamma[I \mid -\Psi],$$

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1

(a)



(b)

Figure 3. (a) Sobel mask operators. (b) Sobel filter application to a real image.

DOI: 10.1002/oca

with $\Gamma \in \mathbb{R}^{3\times3}$ and $\Lambda \in \mathbb{R}^{3\times1}$. Applying the RQ decomposition [21] to Γ block matrix, we have

$$\Gamma = \mathbf{KR},$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ is an upper triangular matrix having the intrinsic parameters, $\mathbf{R}$ represents the rotation matrix, and $\mathbf{t} = -\mathbf{R}\Psi$ is the translation vector; the last two are related to a 3D coordinate system used for calibration process [21,22]. Camera calibration was performed using a Matlab toolbox developed by Caltech Institute [23], with the following results for the camera matrix:

$$\mathbf{K} = \begin{bmatrix} 377.58 & 0 & 359.50 \\ 0 & 377.58 & 239.50 \\ 0 & 0 & 1 \end{bmatrix}, \tag{6}$$

where the focal length $\lambda = 377.58$ pixels and the principal point $q_0 = (359.5,\ 239.5)^{\mathrm{T}}$ pixels.

4.3. Variables estimation

After applying the Sobel filter, three different image rows γ_1, γ_2, and γ_3 are arbitrarily chosen; obviously, they define three vertical image coordinates. This arbitrary rows selection is based on prior knowledge about interval of rows where cart-inverted pendulum system is projected into an image plane (it has an almost constant image size). Through these rows, we have to find their respective horizontal image coordinates (columns) ρ_1, ρ_2, and ρ_3 of pixels belonging to edges detected by the Sobel filter. Therefore, with these horizontal and vertical coordinates, we are able to set the characteristic image points $q_1 = (\rho_1,\ \gamma_1)$, $q_2 = (\rho_2,\ \gamma_2)$, and $q_3 = (\rho_3,\ \gamma_3)$. The points q_1 and q_2 are used to estimate the pendulum angle θ whereas q_3 is used to determine the cart position x_c. From Figure 4(a) by simple trigonometric relations, the angle θ is defined as

$$\theta = \tan^{-1}\left(\frac{\rho_1 - \rho_2}{\gamma_1 - \gamma_2}\right).$$

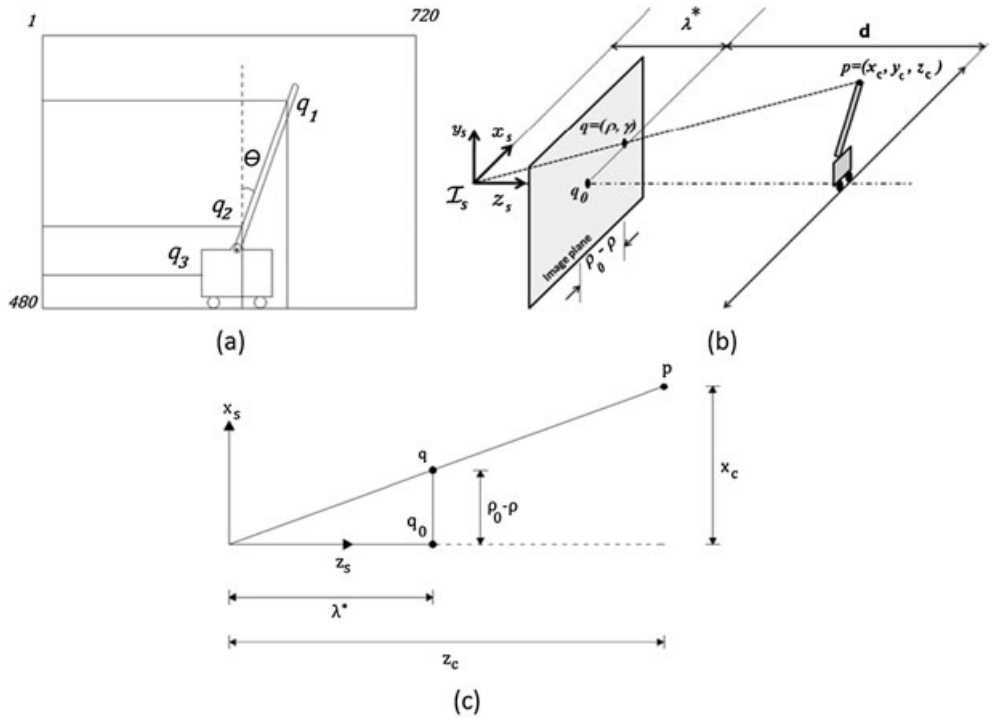


Figure 4. (a) Coordinates detection method scheme. (b) Basic model of image formation. (c) Plane scheme $x_s - z_s$.

DOI: 10.1002/oca

The cart position is measured with respect to $\mathcal{I}_s = \{\{x_s\}\ \{y_s\}\ \{z_s\}\}$, a fixed coordinate system located in the camera focus (Figure 4(b)). Suppose that the $x_s - y_s$ plane is parallel to an image plane and z_s-axis coincides with the principal vector of camera (optical axis). The camera optical axis is the straight line passing through the camera focus, and it is perpendicular to an image plane. Therefore, an image plane is intersected by z_s-axis in the principal image point q_0. Under these considerations, it is clear that the cart-inverted pendulum system moves on the x_s-axis. Thus, conveniently, the origin of an image plane is now considered in q_0; it means that the origin of x_s-axis is the intersection point between x_s and z_s axes. In order to perform a 3D visual servoing technique based on real-world coordinates, we have to compute the focal distance in metric units λ^* and the distance $\mathbf{d}$ from the image plane to a cart-inverted pendulum system. Focal distance λ^* is computed using the procedure presented in [24]. Besides, in practice, an acceptable estimation of distance $\mathbf{d}$ is the distance from the frontal lens of a camera to the object projected. Geometric relations described earlier are shown in Figure 4(b). Analyzing the $x_s - z_s$ plane (Figure 4(c)) by similar triangles, we obtain

$$\frac{\rho_0 - \rho}{\lambda} = \frac{x_c}{z_c},$$

where we assume without loss generality that $z_c = \mathbf{d} + \lambda^*$. Therefore, the cart position in real-world coordinates (metric units) is given by

$$x_c = \frac{(\rho_0 - \rho)\, z_c}{\lambda}.$$

Estimated values for these variables are $\lambda^* = 0.0079$ m and $\mathbf{d} = 0.8$ m; hence, $z_c = 0.8079$ m.

5. CONTROL STRATEGY

In this section, we use results presented in the preliminaries to synthesize a control strategy to improve the performance of cart-inverted pendulum. The mathematical model of the cart-inverted pendulum is used to simulate and synthesize the optimal control law and also to analyze the robust stability of a closed-loop system by considering time delays. The synthesis of optimal control is performed using Matlab software with the command 'ric-schr' where Q_c and R_c are chosen as

$$Q_c = \begin{bmatrix} 100 & 0 & 0 & 0 \\ * & 4.5 & 0 & 0 \\ * & * & 19 & 0 \\ * & * & * & 2 \end{bmatrix},\ R_c = 10. \tag{7}$$

Solving the Riccati equation [25] with Q_c and R_c, we obtain the matrix P_1 given by

$$P_1 = \begin{bmatrix} 104.7225 & 52.584 & -124.2328 & -22.8079 \\ * & 42.4225 & -107.2918 & -19.7070 \\ * & * & 448.2064 & 76.1690 \\ * & * & * & 13.2616 \end{bmatrix},$$

where the associated optimal feedback gain vector is defined by $K_1^* = R_c^{-1} B^T P_1$. So, it is given as follows:

$$K_1^* = \begin{bmatrix} 3.1623 & 3.3116 & -23.6981 & -3.9286 \end{bmatrix}. \tag{8}$$

Introducing the optimal control $u = -K_1^* x(t)$ into system (1), we get

$$\dot{x}(t) = Ax(t) + \left(-BK_1^*\right) x(t). \tag{9}$$

Nevertheless, as mentioned earlier, the visual servoing approach applied in this contribution induces an intrinsic time delay in the closed-loop system. The real amount of delay is unknown, but

DOI: 10.1002/oca

an upper bound can be set. As the image acquisition rate is 30 frames per second, then in practice, the time delay can be established at $h = 0.033$ s. Considering this delay, system (9) becomes

$$\dot{x}(t) = Ax(t) + \left(-BK_1^*\right) x(t-h). \tag{10}$$

It is clear that system (10) has a characteristic equation defined by a delayed quasipolynomial having an infinite number of roots. The roots of large magnitude for this kind of quasipolynomial are inside a region bounded by a exponential envelope [26]. Then in order to deduce the stability of a system, only the roots closest to an imaginary axis must be computed. These roots of small magnitude are computed with no computational errors by using the methodology proposed in [27], which is based on the argument principle and the Mikhailov diagram. Values for the first four roots closest to the imaginary axis are shown in following table:

Roots	Precision
$r_1 = -1.4817 \pm 1.251j$	4.0×10^{-5}
$r_2 = -19.1498 \pm 14.6945j$	5.0×10^{-5}
$r_3 = -81.592 \pm 226.59j$	4.0×10^{-3}
$r_4 = -99.106 \pm 420.85j$	3.2×10^{-3}

So, the stability margin called α_1-stability is defined by r_1, the root closest to the imaginary axis (Figure 5). Theorem 1 is applied with $A_0 = A$ and $A_1 = -BK_1^*$ in order to analyze the stability of the system considering a time-invariant delay. To solve the LMI described by (3), the LMIedit interface from Matlab is used, with $h = 0.033$ s. Results obtained are as follows:

$$P = \begin{bmatrix} 2691.3 & 615.9 & -3161 & -325 \\ * & 735.5 & -1387 & -451 \\ * & * & 7554.8 & 1037.4 \\ * & * & * & 370.8 \end{bmatrix}, \quad Q = \begin{bmatrix} 2411 & 1194 & -10\,230 & -1430 \\ * & 2218 & -9330 & -2450 \\ * & * & 68\,996 & 11\,095 \\ * & * & * & 2748 \end{bmatrix}$$

$$V = \begin{bmatrix} 13\,420 & 0 & -60 & 0 \\ * & 11\,960 & 0 & -3950 \\ * & * & 13\,770 & 0 \\ * & * & * & 1300 \end{bmatrix}, \quad W = \begin{bmatrix} -2763 & -530 & 2886.8 & 175.4 \\ * & -488 & 443.9 & 157.2 \\ * & * & -4931 & 46.7 \\ * & * & * & -22 \end{bmatrix}$$

where P, Q, and V are positive definite matrices; then in accordance with Theorem 1, system (10) is asymptotically stable.

However, the dynamic model (10) is a linear approximation of the real system. So, it makes sense to consider a dynamic model including nonlinear disturbances and time-varying delay in order to have a more representative model of our experimental platform. The nonlinear disturbances include nonmodeled dynamics and nonstructured disturbances, such as dead zone, saturation, hysteresis,

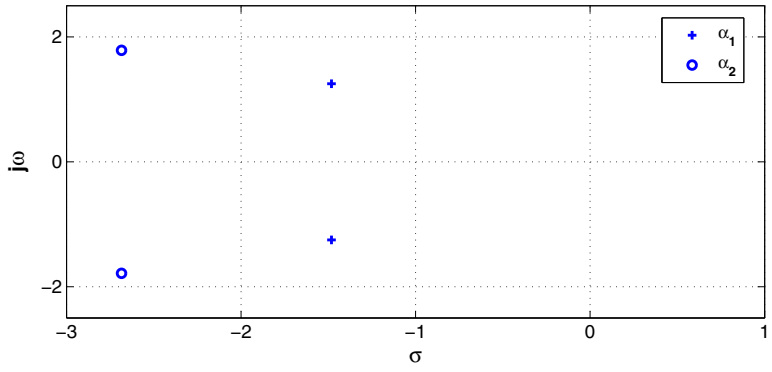


Figure 5. Stability margins α_1-stability and α_2-stability.

DOI: 10.1002/oca

backlash, and Coulomb friction [28], whereas the time-varying delay is due to the image acquisition and treatment processes, which do not have a constant sampling period. Disturbances and uncertainties described earlier are included in system (4). Then applying Proposition 1 with Q_c defined in (7) and R_c, α, β, r_d, and r_M given as follows,

$$R_c = 1,\ \alpha = 0.2,\ \beta = 0.06,\ r_d = 0.593 \text{ and } r_M = 1.047$$

we get

$$P_2 = \begin{bmatrix} 79.788 & 29.5806 & -63.6087 & -11.4920 \\ * & 17.1275 & -39.2601 & -7.0301 \\ * & * & 115.7936 & 19.2303 \\ * & * & * & 3.4158 \end{bmatrix},$$

where matrix P_2 satisfies the Riccati equation with associated optimal feedback gain vector defined by

$$K_2^* = \begin{bmatrix} 10 & 7.9788 & -36.4234 & -6.3609 \end{bmatrix}. \tag{11}$$

Then the first four roots closest to the imaginary axis produced with this controller and computed as was mentioned earlier are as follows:

Roots	Presicion
$r_5 = -2.6847 \pm 1.7847j$	3.0×10^{-5}
$r_6 = -4.5178$	3.0×10^{-5}
$r_7 = -13.3140 \pm 25.5311j$	3.0×10^{-5}
$r_8 = -73.047 \pm 227.531j$	3.1×10^{-3}

so, the α_2-stability defined by r_5 with $A_0 = A$ and $A_1 = -BK_2^*$ is greater that the α_1-stability (Figure 5). It means that the system is more robust using K_2^* than K_1^*.

Using the LMI toolbox of Matlab, LMI (5) is solved, so matrices X, P, R, and Y and scalars ε_1 and ε_2 are defined by

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -843 & -85 & 1412 & 134 \\ 0 & 0 & 0 & 0 \\ -277 & -259 & 4274 & 405 \end{bmatrix},\ P = \begin{bmatrix} 1435 & 618 & -1643 & -240 \\ * & 604 & -1292 & -242 \\ * & * & 4585 & 563 \\ * & * & * & 111 \end{bmatrix} 1 \times 10^3,$$

$$R = \begin{bmatrix} 1065 & 721.6 & -2120 & -557 \\ * & 998.7 & -1675 & -665 \\ * & * & 7006 & 1340 \\ * & * & * & 572 \end{bmatrix} 1 \times 10^3,\ Y = \begin{bmatrix} 2853 & -10 & 9 & -3 \\ * & 2589 & 7 & -855 \\ * & * & 2909 & -2 \\ * & * & * & 282 \end{bmatrix} 1 \times 10^3,$$

$\varepsilon_1 = 3.25 \times 10^6$ and $\varepsilon_2 = 5.07 \times 10^6$.

Matrices Y, P, and R are positive definite; X is a real matrix; and scalars ε_1 and ε_2 are positive definite; then system (10) is robustly asymptotically stable.

Remark 1

The α, β, and r_d constants have been chosen heuristically to satisfy Proposition 1, and r_M is an upper bound of a delay.

Remark 2

The controller with gain vector K_1^* satisfies Theorem 1 but not Proposition 1. It stabilizes the platform in simulations (see the next section); nevertheless, the dynamic system is not stabilized in real-time experiments. On the other hand, the controller with gain vector K_2^* satisfies both

DOI: 10.1002/oca

Theorem 1 and Proposition 1. With this controller, the cart-inverted pendulum is properly stabilized in its unstable equilibrium point in real-time experiments and simulation. From Figure 5, it is clear that the α_2-stability is more robust than α_1-stability; this is a consequence of the α_2-stability margin, which is defined by an optimal controller with gain vector K_2^*. This fact illustrates how the real-time performance of the delayed systems could be affected by the conservativeness of the stability conditions.

In summary, in this contribution, we are proposing the following algorithm:

1. Consider the free delay system (1) and synthesize an optimal control K^* for some matrices Q_c and R_c.
2. Introduce the optimal control defined by K^* into the delayed system (10) with $h \neq 0$ and obtain the characteristic quasipolynomial of closed loop. So, compute the set of roots closest to the imaginary axis to determine the α-stability.
3. Finally, for systems (2) and (4) with $A_1 = -BK^*$, check if the LMIs (3) and (5) are respectively satisfied. If the sufficient conditions for robust stability are reached, we conclude the analysis. If not, go back to Step 1 and select another Q_c and R_c to find a new α-stability more robust than the previous one.

6. SIMULATION AND REAL-TIME EXPERIMENTAL RESULTS

This section shows the simulation and real-time experimental results obtained when the suboptimal robust control law is applied to a delayed underactuated system considering nonlinear disturbances. Time delay considered in both cases is 33 ms. Simulation results are shown in Figures 6 and 7. State variables behavior is presented in Figure 6 when two independent control laws K_1^* and K_2^* are applied from Equations (8) and (11), respectively. Initial conditions have been chosen relatively far from unstable equilibrium point. One can realize that the delayed system response is very satisfactory because in both cases the system is properly stabilized. Both control signals are plotted in Figure 7. Simulations have been performed on Matlab-Simulink, which uses the numerical solver ode45 based on the Dormand–Prince method.

On the other hand, the real-time experiment considers that the initial condition of the underactuated delayed system is near to the unstable equilibrium point. The control law based on K_2^* vector is chosen to perform the experiment, because this control law shows an appropriate performance in a simulation where the dynamic system is properly stabilized with an initial condition of 0.5 rad of angular position. It is easy to realize from Figure 8 that the state variables have a nice behavior, and the car-inverted pendulum is stabilized in the unstable equilibrium point. Furthermore, Figure 9

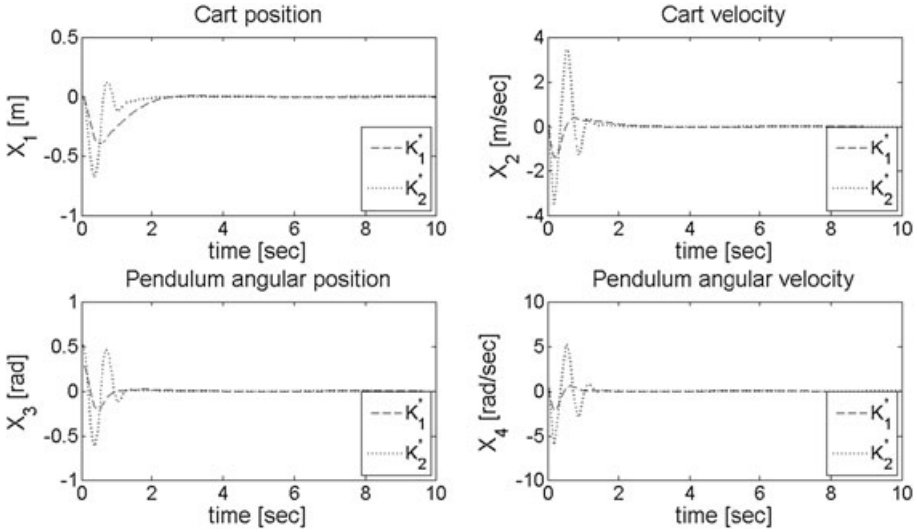


Figure 6. State variables behavior (simulation).

DOI: 10.1002/oca

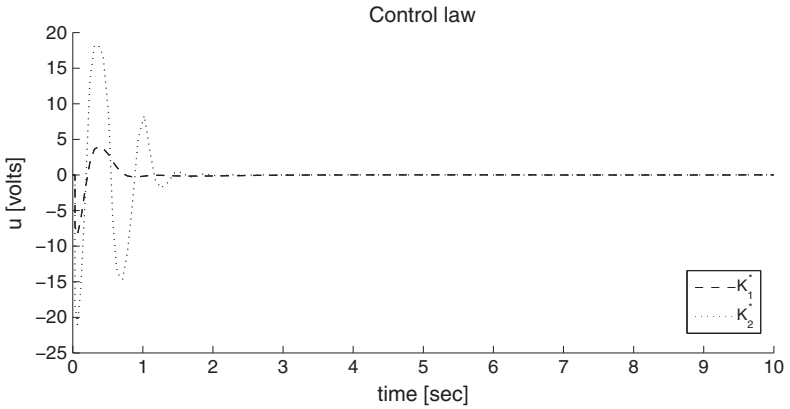


Figure 7. Control law (simulation).

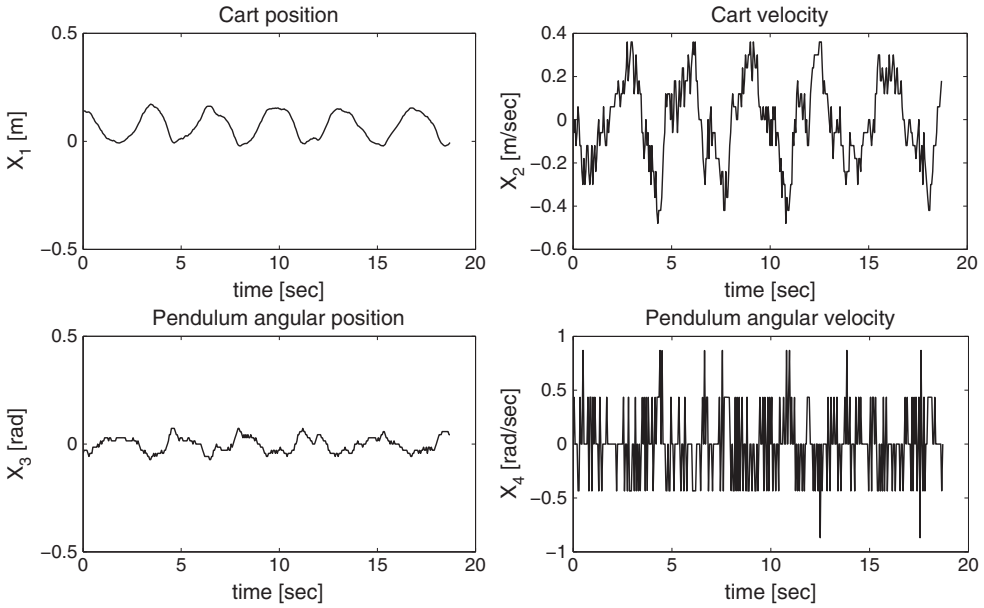


Figure 8. Real-time experiment results using a control law based on K_2^* vector (state variables behavior).

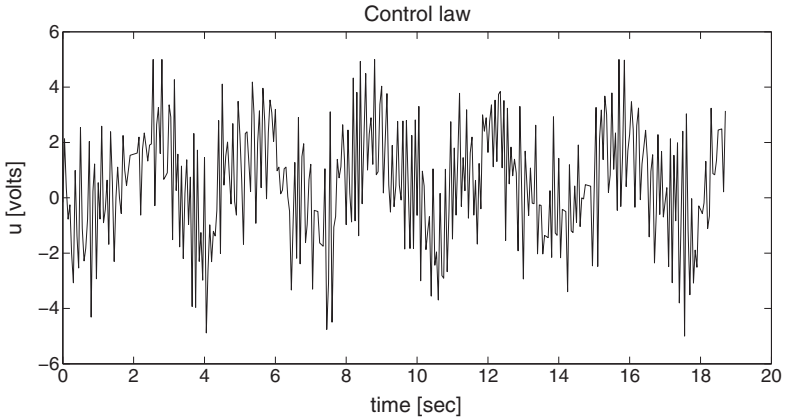


Figure 9. Real-time control law using K_2^* (experimental result).

DOI: 10.1002/oca

shows the control law applied to a DC motor. This control law saturates the actuator by only a few moments.

6.1. *Effect to delays in the performance of the platform*

In order to illustrate how the delay affects the performance of the dynamic system, the matrices Q_c and R_c are now proposed as follows:

$$Q_c = \begin{bmatrix} 10 & 1 & 3 & 4 \\ * & 7 & 5 & 2 \\ * & * & 20 & 6 \\ * & * & * & 42 \end{bmatrix}, \; R_c = 2,$$

then the matrix P_1, which satisfies the Riccati equation [25] for these matrices, is given by

$$P_1 = \begin{bmatrix} 16.0914 & 11.1057 & -35.1314 & -4.4379 \\ * & 15.3924 & -50.7699 & -6.3992 \\ * & * & 261.4712 & 27.4582 \\ * & * & * & 4.5065 \end{bmatrix},$$

with optimal gain vector for system (1) defined by

$$K_3^* = \begin{bmatrix} 2.236 & 3.821 & -31.088 & -6.961 \end{bmatrix}. \tag{12}$$

Nevertheless, system (10) in closed loop with the controller (12) does not satisfy Theorem 1 nor Proposition 1; it implies that the controller based on gain vector K_3^* assures the stability of the disturbed delayed system (10), but it does not assure its robust stability. In order to illustrate this fact, the gain vector (12) is introduced in (10), and the resulting characteristic equation in closed loop is stable with roots located in

Roots	Accuracy
$r_9 = -1.4351 \pm 0.8672j$	5×10^{-5}
$r_{10} = -6.4637 \pm 38.77j$	3×10^{-5}
$r_{11} = -59.818 \pm 229.63j$	1.8×10^{-3}
$r_{12} = -77.69 \pm 422.52j$	1.35×10^{-3}

where r_9 defines the α_3-stability margin.

Figure 10 shows graphically the α-stability margins for the control laws based on K_2^* and K_3^*, respectively; the control law based on K_2^* is more robust than the control law based on K_3^*. In a simulation, the stability margins for both cases are sufficiently large to stabilize the delayed system.

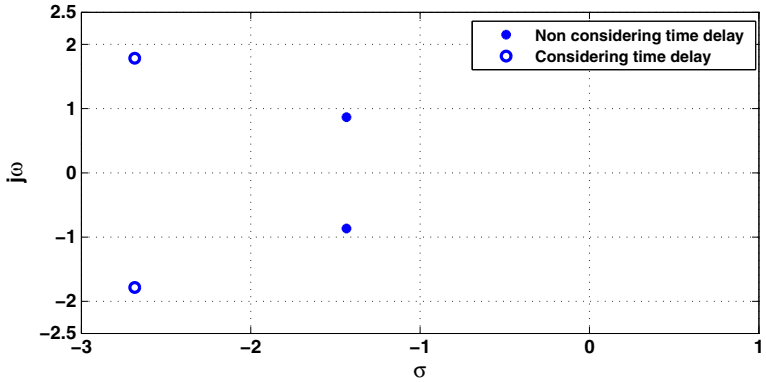


Figure 10. Graphic representation of α_2-stability and α_3-stability.

DOI: 10.1002/oca

Then, in order to verify if it is feasible to neglect the time delay in the control law design, the control law defined by K_3^* is implemented in real time to be compared with another real-time experiment using the control law defined by K_2^*. Figure 11 contrasts the state variables behavior when these control laws are applied to a dynamic system. Evidently, the control law using K_3^* does not perform the real-time stabilization of the delayed system, because the inverted pendulum has oscillations and finally falls about 11 s of the experiment.

Moreover, the control law signals are shown in Figure 12. So, the control law based on K_3^* is more abrupt than the other law based on K_2^*, because the actuator remains saturated by longer periods. Therefore, it is necessary to consider the time delay to guarantee the real-time stability of the system.

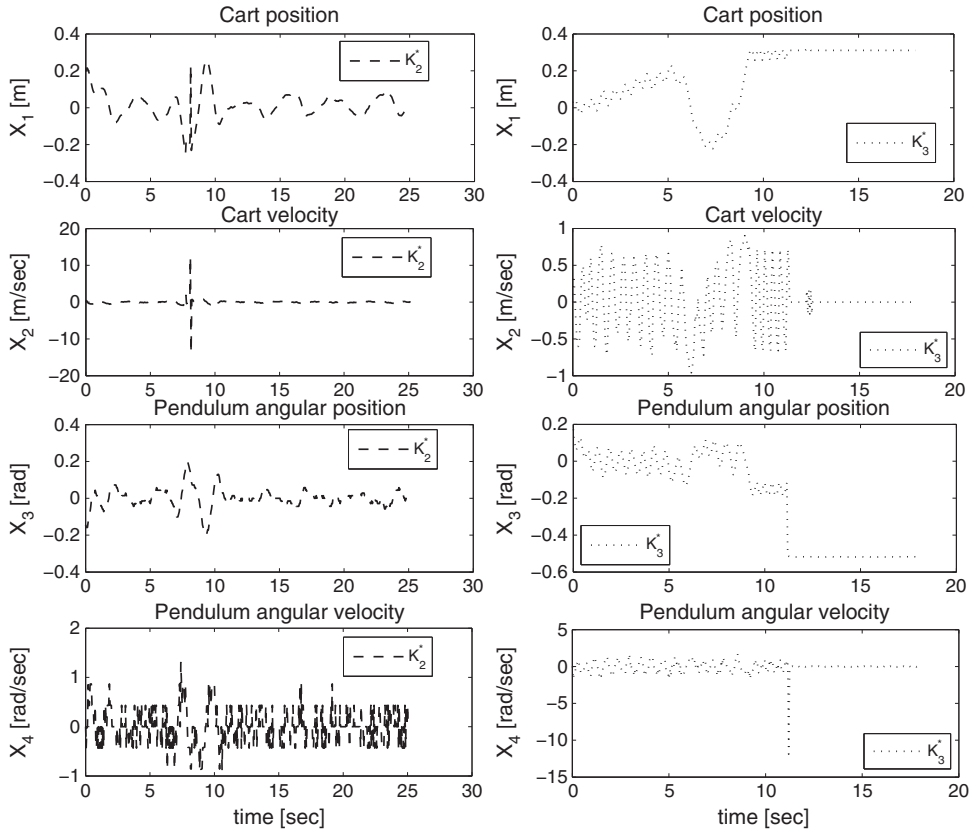


Figure 11. State variables behavior. Left column, controller based on K_2^*. Right column, controller based on K_3^*.

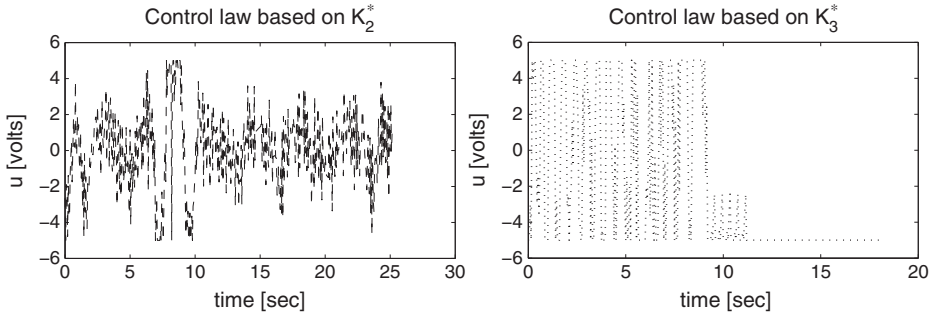


Figure 12. Control law signal: left based on K_2^*; right based on K_3^*.

DOI: 10.1002/oca

7. CONCLUSIONS

In this contribution, a suboptimal robust control law considering the time delay has been synthesized and applied to an underactuated system under visual servoing control scheme. We have illustrated how to use theoretical results in order to improve the performance of time-delayed system. In fact, using an experimental platform, we verify how the conservativeness of certain results of stability analysis affect the performance of delayed plant in closed loop. Nonlinear disturbance is also considered to affect the closed-loop system. Firstly, an LQR approach has been applied to obtain a linear control law. Then applying the Lyapunov–Krasovskii technique, a nonlinear suboptimal robust control law is synthesized, such that the closed-loop system is stable and robust. According to simulation and experimental results, this suboptimal robust control law shows a better performance than the control laws not considering the nonlinear disturbances and delay.

Additionally, the camera has been calibrated in order to improve the variable estimation. Future works should test less conservative stability conditions and nonlinear control law implementation considering delays and disturbances.

REFERENCES

1. Richard J. Time delay systems: an overview of some recent advances and open problems. *Automatica* 2003; **39**(10):1667–1694.
2. Krasovskii NN. *Stability of Motion*, (1st edn). Standford University Press: Stanford, California, USA, 1963.
3. Park P. A delay-dependent stability for a class criterion for systems with uncertain time-invariant delays. *IEEE Transactions on Automatic Control* 1999; **44**(4):876–877.
4. Han L, Yu L. Robust stability of linear neutral systems with nonlinear parameter perturbations. *IEE Proceedings—Control Theory and Applications* 2004; **151**(5):539–546.
5. Li H, Gao H, Shi P. Reliable fuzzy control for active suspension systems with actuator delay and fault. *IEEE Transactions Neural Networks* 2011; **20**(2):342–357. DOI: 10.1109/TFZZ.2011.2174244.
6. Li H, Gao H, Shi P. Passivity analysis for neural networks with discrete and distributed delays. *IEEE Transactions Neural Networks* 2010; **22**(11):1842–1847.
7. Corke PI, Hager GD, Hutchinson S. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation* 1996; **12**(5):651–670.
8. Hamel T, Mahony T. Visual servoing of a class of underactuated dynamic rigid-body system. *39th IEEE Conference on Decision and Control*, Sydney, NSW, December 2000; 3933–3938.
9. Espinoza-Quesada ES, Ramos-Velasco LE. Visual servoing for an inverted pendulum using a digital signal processor. *The 6th IEEE International Symposium on Signal Processing and Information Technology*, Vancouver, Canada, August 2006; 76–80.
10. Xie H, Sun L, Rong W, Yuan X, Visual servoing with modified Smith predictor for micromanipulation tasks. *IEEE International Conference on Mechatronics and Automation*, Niagara, Canada, 2005; 71–76.
11. Zhang J, Lumia R, Wood J, Starr G. Delay dependent stability limits in high performance real-time visual servoing systems. *International Conference on Intelligent Robots and Systems*, Nevada, USA, October 2003; 485–491.
12. Chiasson J, Loiseau JJ (eds). *Applications of Time Delay Systems*. Springer: Berlin, Germany, 2007. 352.
13. Zhang J, Xia Y, Shi P, Mahmoud MS. New results on stability and stabilisation of systems with interval time-varying delay. *IET Control Theory and Applications*; **5**(3):429–436.
14. Xu D, Tian J. New delay-dependent stability criteria for systems with interval time-varying delay. *Applied Mechanics and Materials* 2011; **48–49**:734–739.
15. Zhu XL, Wang Y, Improved Jensen integral inequality approach to stability analysis of continuous-time systems with interval time-varying delay. *IEEE American Control Conference*, USA, 2010; 6573–6578.
16. Innovate Educate Quanser. IP01-02 User Manual-501, 2000. (Available from: http://www.quanser.com) [Accessed on December 2011].
17. Innovate Educate Quanser. IP01-02 SESIP User Manual-514, 2002. (Available from: http://www.quanser.com) [Accessed on December 2011].
18. Fantoni I, Lozano R. *Non-linear Control for Underactuated Mechanical Systems*, (1st edn). Springer: London, Great Britain, 2002.
19. Instruments Incorporated Texas. TMS320C6000, Imaging Developer's Kit (IDK) Programmers guide. (Available from: http://www.ti.com,2003) [Accessed on December 2011].
20. Castleman K. *Digital Image Processing*, (1st edn). Pretice Hall: Upper Saddle River, New Jersey, USA, 1996.
21. Hartley R, Zisserman A. *Multiple View Geometry in Computer Vision*, (2nd edn). Cambridge University Press: Cambridgeshire, England, 2004.
22. Corke PI. *Visual Control of Robots: High-Performance Servoing*, (1st edn). John Wiley and Sons Inc.: Taunton, Somerset, England, 1997.

DOI: 10.1002/oca

23. Bouguet J-Y. Camera Calibration toolbox for Matlab, 1997. (Available from: http://www.vision.caltech.edu/bouguetj/calibdoc) [Accessed on December 2011].
24. CCT Rainbow: Model Rainbow 1/3" L3.3-8mm, 2000. (Available from: http://www.rainbowcctv.com/tech/formula.html) [Accessed on December 2011].
25. Kalman RE. Contributions to the theory of optimal control. *Boletín de la Sociedad Matemática Mexicana*; **5**(32):102–119.
26. Bellman R, Cooke KL. *Differential-Difference Equations*. Academic Press: New York, USA, 1963.
27. Santos O, Mondié S. Control laws involving distributed delays: robustness of the implementation. *in Proc. of IEEE American Control Conference*, Vol. 4, 2000; 2479–2480.
28. Vukic Z, Kuljaca L. *Nonlinear Control Systems*, (1st edn). Marcel Dekker Inc.: New York, USA, 2003.

DOI: 10.1002/oca