Trajectory tracking of the robot end effector for the minimally invasive surgeries

José de Jesús Rubio*, Panuncio Cruz,
Enrique Garcia, Cesar Felipe Juárez,
David Ricardo Cruz and Jesús López

Sección de Estudios de Posgrado e Investigación,
Esime Azcapotzalco,
Instituto Politécnico Nacional,
Av. de las Granjas no. 682, Col. Santa Catarina,
México D.F., 02250, México
Email: jrubioa@ipn.mx
Email: rubio.josedejesus@gmail.com
Email: panu_ver_1984@hotmail.com
Email: enriquegarcia@linuxmail.org
Email: cesarjuarezc@hotmail.com
Email: ingdavidrcruz@gmail.com
Email: jesuslopez.asaf@gmail.com
*Corresponding author

Abstract: The surgery technology has been highly investigated, with the purpose to reach an efficient way of working in medicine. Consequently, robots with small tools have been incorporated in many kind of surgeries to reach the following improvements: the patient gets a faster recovery, the surgery is not invasive, and the robot can access to the body occult parts. In this article, an adaptive strategy for the trajectory tracking of the robot end effector is addressed; it consists of a proportional derivative technique plus an adaptive compensation. The proportional derivative technique is employed to reach the trajectory tracking. The adaptive compensation is employed to reach approximation of some unknown dynamics. The robot described in this study is employed in minimally invasive surgeries.

Keywords: trajectory tracking; robot; minimal invasive surgery.

Reference to this paper should be made as follows: Rubio, J.J., Cruz, P., Garcia, E., Juárez, C.F., Cruz, D.R. and López, J. (2020) 'Trajectory tracking of the robot end effector for the minimally invasive surgeries', *Int. J. Business Intelligence and Data Mining*, Vol. 16, No. 1, pp.66–88.

Biographical notes: José de Jesús Rubio is a Full-Time Professor of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published 106 papers in international journals with 1,000 citations.

Panuncio Cruz is a PhD student of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published six papers in international journals.

Enrique Garcia is a PhD student of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published four papers in international journals.

Cesar Felipe Juárez is a PhD student of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published two papers in international journals.

David Ricardo Cruz is a PhD student of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published three papers in international journals.

Jesús López is a PhD student of the Sección de Estudios de Posgrado e Investigación, ESIME Azcapotzalco, Instituto Politécnico Nacional. He has published three papers in international journals.

1 Introduction

The trajectory tracking and robotic are of great utility in medicine; because they can get a high precision to reach the positions, and they provide the necessary information to avoid risks. Consequently; the trajectory tracking of a robot with small tools can be employed to develop a surgery where the doctor moves the robot by a computer (see Figure 1). This technology offers the following three improvements in surgeries:

a it helps the patient to get a faster recovery

b to it is not invasive, i.e., it avoids the damage of other parts of the body

c it can access to occult parts of the body; while this technology has the unique drawback of the high cost.

It motivates more investigations about the robotic and trajectory tracking issues for this kind of applications.

Some investigation is made about intelligent trajectory tracking. In Andonovski et al. (2016) and Angelov et al. (2015), interesting evolving cloud-based strategies are designed. Neural network techniques are introduced in Pan et al. (2014, 2016a, 2016b). In Pratama et al. (2011), an adaptive network-based fuzzy inference system of a cascade method is implemented. Optimal techniques are introduced in Pan et al. (2017) and Precup et al. (2015a, 2015b). In Chandrahaasan and Gnanaprakasaam (2016), Joo (2017), Kumar et al. (2016) and Rajesh et al. (2016), some interesting optimal algorithms are studied. It shows that the intelligent trajectory tracking methods have been applied in some actual applications.

Figure 1 The robot for surgeries (see online version for colours)

Some research is made about robotics. An adaptive PID strategy of robots is addressed in Malekzadeh et al. (2016). In Navarro-Gonzalez et al. (2015), a fuzzy neural network to successfully accomplish the assembly task in robots is proposed. An autonomous template-based detection and tracking method of an unmanned aerial vehicle is introduced in Sadeghi-Tehran and Angelov (2014). In Toubakh and Sayed-Mouchaweh (2015), a hybrid dynamic data-driven approach to achieve fault detection of a wind turbine is discussed. It shows that the robotic techniques have been applied in some actual applications.

Some investigation is made about the trajectory tracking and robotics for surgeries. In Kwoh et al. (1988), the puma robot is utilised in a study of the brain. A robot is employed in an invasive surgery in Davies et al. (1989). In Stoianovici (2000), a robot is utilised to move a camera in a surgery. A robot is used to move a camera and instruments in a surgery in Marescaux and Rubino (2003). In Marohn and Hanly (2004), a more sophisticated arm for surgeries is introduced. A robotic vision plant for the movement of instruments in a surgery is introduced in Krupa et al. (2003). In Climent and Mares (2003), all the surgery is developed by a robotic plant. A robotic plant for a minimum invasive surgery is introduced in Piccigallo et al. (2010). In Kim et al. (2013), a robotic plant which changes between a minimum invasive surgery and an open surgery. A robotic arm with 7 degrees of freedom for a minimum invasive surgery is suggested in Tholey and Desai (2007). In Perry et al. (2007), a robot with 7 degrees of freedom for diagnosis is introduced. A robot to reanimate a patient is addressed in Li and Xu (2007). In Lum et al. (2006), a robot with 6 degrees of freedom of the least generation to make a cleaner surgery is designed. It shows that the robotic and trajectory tracking issues can be combined to reach the purpose of improving the quality of surgeries.

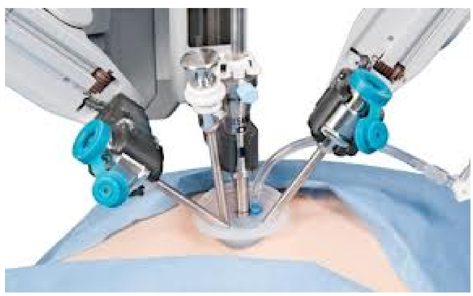
There is a problem when a robot is employed for minimally invasive surgeries: the position of the end effector in a robot must be very precise. In this work, a method is proposed to solve the mentioned problem: an adaptive technique for the trajectory tracking of the robot end effector is suggested. The adaptive strategy of this article consists of a proportional derivative technique plus an adaptive compensation. The proportional derivative technique is employed for the trajectory tracking and it avoids the utilisation of the integral part because it could make the plant unstable. The adaptive compensation is employed to approximate some unknown dynamics to be compensated.

In this article, an adaptive strategy for the trajectory tracking of the robot end effector with 11 degrees of freedom is suggested, the end effector consists of the last two degrees of freedom of the robot. The proposed strategy will be compared with the proportional derivative (PD) technique in the trajectory tracking of a robot employed in minimum invasive surgeries.

2 Laparoscopic surgery and minimum invasive surgery with robots

The laparoscopic surgery is known as the extension of hands and miniaturisation of eyes, it utilises small cameras to make the view of the surgery camp, and it employs a display to see all the elements in the surgery. Robots in medicine have been designed to increase the capacities of the doctors in this kind of surgery.

Figure 2 The laparoscopic surgery (see online version for colours)



The minimum invasive surgery (see Figure 2) is a plant in which instruments and vision devices are inserted in the patient by small incisions. The laparoscopic is a kind of minimum invasive surgery which is utilised in the abdominal zone and it lets the surgeon to obtain a high access to the inside part of the patient and to manipulates the instruments in an easy way (Schwartz and Hunter, 1999).

2.1 Design of the robotic arm

A robot is designed for the application of a laparoscopic surgery. It has to satisfy the following characteristics:

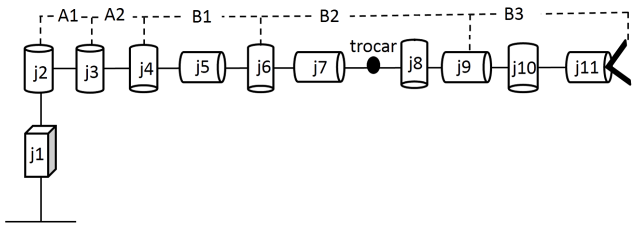
1 it has to obtain six degrees of freedom to get a good positioning

2 two degrees of freedom of the robot have to be set in abdominal zone of the patient

3 two degrees of freedom of the robot have to be used as the centre of rotation to maintain a reference.

Considering the before characteristics, this robot has nine degrees of freedom, In Figure 3, it can be seen the kinematic structure in the robot design; distances of the robot are labelled as A and B, only the last two links of the robot enter inside of the abdominal par of the patient.

Figure 3 Kinematics of the robot



The first three links are used for the positioning of the arm in a three dimensional space, the four, five, and six links are employed to assure that the robot is upper to the patient, the next three links are passive, the last two links are the end effector.

2.2 Mathematical model of the robot

Analysing the structure of the robot the parameters of Table 1 are found (Khalil and Dombre, 2002).

Table 1 Geometric parameters of the robot

j	η_j	ρ_j	φ_j	A_j	θ_j	B_j
1	1	1	0	0	0	B_1
2	1	0	0	0	ϕ_2	0
3	1	0	0	A_1	ϕ_3	0
4	0	0	$90°$	A_2	ϕ_4	0
5	0	0	$90°$	0	ϕ_5	B_1
6	0	0	$-90°$	0	ϕ_6	0
7	1	0	$90°$	0	ϕ_7	B_2
8	1	0	$-90°$	0	ϕ_8	0
9	1	0	$90°$	0	ϕ_9	0
10	1	0	$-90°$	0	ϕ_{10}	0
11	1	0	$90°$	0	ϕ_{11}	0
end	0	1	0	0	0	B_3

where

j it represents the number of the joint

η_j it indicates whether the joint is active (with motor) or passive (without motor)

ρ_j	it shows if the joint is shown prismatic (1) or rotational (0)
φ_j, θ_j	they are angles that depend on joint axes or rotation of a toroide joint
A_j, B_j	they are distances between the joints axes in rotational joints or displacements of prismatic joints.

Furthermore, the robot is modelled as a set of rigid bodies connected in series with one end attached to ground and the other free end. The robot dynamic equation is given by Sadeghi-Tehran and Angelov (2014) and Toubakh and Sayed-Mouchaweh (2015):

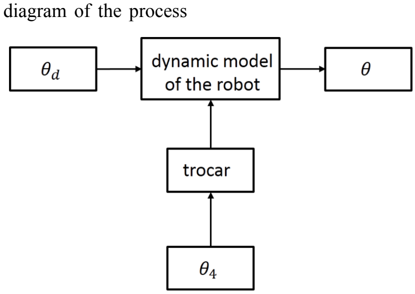
$$D(\theta)\ddot{\theta} + B\left(\theta,\dot{\theta}\right)\dot{\theta} + F\left(\theta,\dot{\theta}\right) + G(\theta) = U - u_d \tag{1}$$

where

U	it is the vector of joint tracking pairs provided by the actuators $(n \times 1)$
$D(\theta)$	it is the matrix of inertia $(n \times n)$
$B(\theta,\dot{\theta})$	it is the vector representing the effects of Coriolis and centrifugal $(n \times 1)$
$G(\theta)$	it is the vector representing gravity $(n \times 1)$
$F(\theta,\dot{\theta})$	it is the vector of the friction forces $(n \times 1)$
u_d	it is the vector describing the unmodelled dynamics and external disturbances $(n \times 1)$
θ	it is the vector of the links positions $(n \times 1)$
$\dot{\theta}$	it is the vector of the links speeds $(n \times 1)$
$\ddot{\theta}$	it is the vector of the links accelerations $(n \times 1)$.

The robot model (1) lets to obtain values of φ_j and ϕ_j associated with joints, and can be analysed in specific desired trajectory tracking of the robot end effector in Cartesian space. A general way of understanding the proposed plant for the robotic surgery is seen in Figure 4.

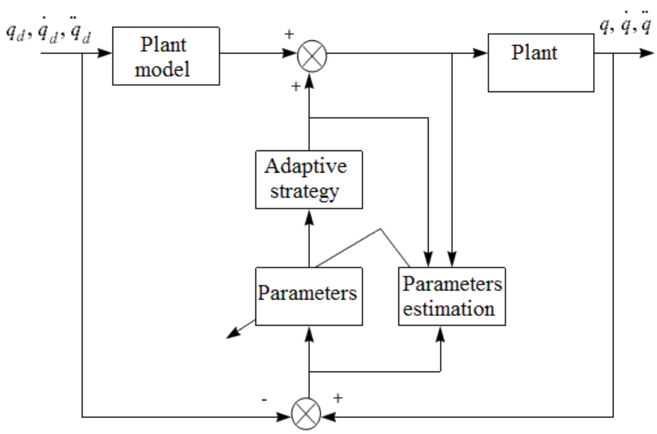
Figure 4 Schematic diagram of the process



3 Trajectory tracking of the robot

Methods where pairs are calculated are the basis for all robots (Fu et al., 1987; Spong and Vidyasagar, 1989). The robotic model, a transformation model linearisation, and the decoupling through feedback state and therefore PD techniques are designed for each robot joint. Robust tracking terms can be added to the calculated tracking signal to compensate the uncertainties of modelling or external shocks. Otherwise, when the robot model is unknown, the calculated tracking method can be implemented within an intelligent scheme such as adaptive algorithms learn the unknown dynamics. The architecture of the strategy is seen in Figure 5.

Figure 5 Tracking of the robot, including online estimation of the unknown dynamics



If the dynamic model is accurate, dynamics of the disturbances are cancelled. The total trajectory tracking that drives the robot is:

$$U = \widehat{D}(\theta)\, u + \widehat{B}\left(\theta, \dot{\theta}\right)\dot{\theta} + \widehat{G}(\theta) + u_d \tag{2}$$

where u is defined as:

$$u = \ddot{\theta}_d + K_v\left(\dot{\theta}_d - \dot{\theta}\right) + K_w\left(\theta_d - \theta\right) \tag{3}$$

Substituting the equation (3) in (1) and considering that $\widehat{D} = D$, $\widehat{B} = B$, $\widehat{G} = G$, it gives:

$$D(\theta)\left[\ddot{\theta}_d + K_v\left(\dot{\theta}_d - \dot{\theta}\right) + K_w\left(\theta_d - \theta\right)\right] = 0 \tag{4}$$

The following error vectors can be defined as: $e = (\theta_d - \theta)$, $\dot{e} = (\dot{\theta}_d - \dot{\theta})$ and $\ddot{e} = (\ddot{\theta}_d - \ddot{\theta})$. Considering that $D(\theta)$ is positive definite, the following equation is gotten:

$$\ddot{e} + K_v\dot{e} + K_w e = 0 \tag{5}$$

The trajectory tracking of the inverse dynamics transform the nonlinear plant of equation (1) in a lineal; consequently, lineal strategies can be applied (Lewis et al., 1999; Wang and Gao, 2004). Considering that two robot end joints are taken to obtain the calculated torque and analyse whether it is appropriate for its application. From equation (1), it is satisfied:

$$M_{11}\ddot{\theta} + M_{12}\ddot{v} + F_1\left(z, \dot{z}\right) = U\left(t\right) \tag{6}$$

$$M_{21}\ddot{\theta} + M_{22}\ddot{v} + F_2\left(z, \dot{z}\right) + D\dot{v} + Kv = 0 \tag{7}$$

The equation (7) is solved with respect to $\ddot{v}$:

$$\ddot{v} = -M_{22}^{-1}M_{21}\ddot{\theta} - M_{22}^{-1}F_2\left(z, \dot{z}\right) - M_{22}^{-1}D\dot{v} - M_{22}^{-1}Kv \tag{8}$$

The equation (8) is substituted in equation (6) producing that:

$$\begin{aligned}&\left(M_{11} - M_{12}M_{22}^{-1}M_{21}\right)\ddot{\theta} - M_{12}M_{22}^{-1}F_2\left(z, \dot{z}\right)\\&-M_{12}M_{22}^{-1}D\dot{v} - M_{12}M_{22}^{-1}Kv + F_1\left(z, \dot{z}\right) = U\left(t\right)\end{aligned} \tag{9}$$

3.1 Adaptive strategy of the robot

The adaptive strategy of robots is considered. The equation (9) represents the robot. It is the nonlinear transformation for inputs to outputs. The nonlinear model is written as follows:

$$\ddot{\theta} = D\left(\theta, \dot{\theta}, v, \dot{v}, U\right) \tag{10}$$

Accordingly, the inverse dynamics of the robot flexures is a relationship that provides the torque generated by the joints engine and joints angles, angular velocity and acceleration to take certain values. The inverse model of the equation (10) is given by:

$$U\left(t\right) = D^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}, v, \dot{v}\right) \tag{11}$$

The inverse model of the equation (11) can be calculated in an explicit way. In this article, an inverse adaptive algorithm is utilised to approximate the inverse dynamics of the equation (11). Variables θ, $\dot{\theta}$, $\ddot{\theta}$, are measurable and variables v, $\dot{v}$ are not measurable. Consequently, the inverse dynamics of the robot can be decomposed in n sub-models in the following way:

$$U\left(t\right) = D^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) = \begin{bmatrix} d_1^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) \\ d_2^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) \\ \cdots \\ d_n^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) \end{bmatrix} \tag{12}$$

where each d_i^{-1}, $(i = 1, 2, ..., n)$, it defines the inverse dynamics of the corresponding board, while n is the number of robot joints. An adaptive algorithm can be used to approximate each sub-model d_i^{-1} of the robot inverse dynamics. Consequently, the inverse dynamics of the whole plant can be represented by an adaptive algorithm $N(\theta, \dot{\theta}, \ddot{\theta}, \omega)$

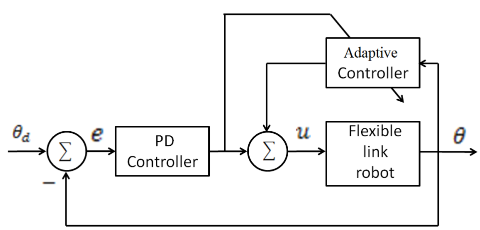
$$U(t) \backsimeq N\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) = \begin{bmatrix} N_1\left(\theta, \dot{\theta}, \ddot{\theta}, \omega_1\right) \\ N_2\left(\theta, \dot{\theta}, \ddot{\theta}, \omega_2\right) \\ \cdots \\ N_n\left(\theta, \dot{\theta}, \ddot{\theta}, \omega_n\right) \end{bmatrix} \tag{13}$$

where $N_i(\theta, \dot{\theta}, \ddot{\theta}, \omega_i)$ $i = 1, 2, ..., n$, is the i^{th} adaptive algorithm that approximates the i^{th} sub-model of the robot and ω_i is the vector of the associated parameters. The adaptive algorithm to model the dynamics of the flexible link robot provides a mapping between the joints angles and the motors pairs. The torque in the output of the adaptive algorithm can be combined with a feedback PD technique to generate the global tracking signal that finally drives the motors. Therefore, the trajectory tracking scheme can be:

$$U(t) = N\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) + k_p e + k_d \dot{e} \tag{14}$$

where $N(\theta, \dot{\theta}, \ddot{\theta}, \omega)$ is the approximation of the adaptive algorithm of the current inverse robot dynamics, while k_p, $k_d \in \Re^{n \times m}$ are diagonal gain matrices. In the equation (14) $e = \theta_d - \theta$, $\dot{e} = \dot{\theta}_d - \dot{\theta}$, are the position and speed of the robot joints, respectively. The architecture of the adaptive algorithm based on the proposed trajectory tracking is seen in Figure 6.

Figure 6 Adaptive control of the robot



Using the equations (12) and (13), the adaptive strategy that has been described in the equation (14) becomes to:

$$N\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) + k_p e + k_d \dot{e} = D^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) \tag{15}$$

Equivalently

$$k_p e + k_d \dot{e} = D^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) - N\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) = \widetilde{N}\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \tag{16}$$

The equation (16) represents a decoupled linear plant, impulse by the nonlinear $\widetilde{N}(\theta, \dot{\theta}, \ddot{\theta}, \omega) \in \Re^n$. This function represents the error between the current dynamic inverse $D^{-1}(\theta, \dot{\theta}, \ddot{\theta})$ and its estimated model $N(\theta, \dot{\theta}, \ddot{\theta}, \omega)$ which can be rewritten as:

$$\widetilde{N}\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) = \begin{bmatrix} \widetilde{N}_1\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \\ \cdots \\ \widetilde{N}_n\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \end{bmatrix} = \begin{bmatrix} d_1^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) - N_1\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \\ \cdots \\ d_n^{-1}\left(\theta, \dot{\theta}, \ddot{\theta}\right) - N_n\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \end{bmatrix} \quad (17)$$

Instead of using an adaptive algorithm to approximate the robot inverse dynamics, a separate network is employed for each robot joint. Then, employing the equation (17), the error equation for the articulation of the i^{th} robot becomes to:

$$k_p^i e_i + k_d^i \dot{e}_i = \widetilde{N}_i\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right), i = 1, 2, ..., n \quad (18)$$

The objective is to eliminate the approximation error, i.e., the $\lim_{t \longrightarrow \infty} \widetilde{N}(\theta, \dot{\theta}, \ddot{\theta}, \omega) = 0$. In this case, by selecting the appropriate gains from k_p^i, k_d^i, it results:

$$k_p^i e_i + k_d^i \dot{e}_i = 0 \Longrightarrow \lim_{t \longrightarrow \infty} e_i(t) = 0, \lim_{t \longrightarrow \infty} \dot{e}_i(t) = 0 \quad (19)$$

The equation (19) explains that the convergence condition for the closed-loop plant yields that the surface error $\varepsilon_i = k_p^i e_i + k_d^i \dot{e}_i$ reaches zero. Then, the proper selection of gains results in an asymptotic convergence of $e(t)$ to zero:

$$k_p^i e_i + k_d^i \dot{e}_i = 0 \Longrightarrow \dot{e}_i(t) = -\frac{k_p^i}{k_d^i} e_i(t) \Longrightarrow e_i(t) = e_i(0)\, e^{-\frac{k_p^i}{k_d^i} t} \quad (20)$$

Therefore a measure of the output error can be considered:

$$\varepsilon_i = \widetilde{N}_i\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) \quad (21)$$

Which reflects the discrepancy between the real inverse dynamics of the robot and its approximation by an adaptive algorithm. The parameters updating of the adaptive algorithm has to be carried out in such a way as to maintain the stability of the plant in closed loop. To this end, the following cost function is defined for each set:

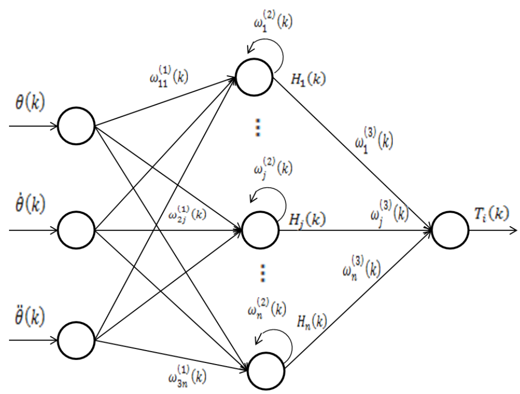
$$E_i(t) = \frac{1}{2}\varepsilon_i^2 \quad (22)$$

The cost function gives the squared distance of the error function $\varepsilon_i = k_p^i e_i + k_d^i \dot{e}_i$. The parameters updating is derived from the minimisation of the cost function $E_i(t)$.

3.2 Approximation of the flexible-links dynamics

A recurrent diagonal adaptive algorithm was used for the first time to approximate the inverse dynamics of each link of the robot. The structure of the i^{th} recurrent diagonal adaptive algorithm is represented in Figure 7.

Figure 7 Structure of the adaptive algorithm



The input vector of the adaptive algorithm is $X = [\theta, \dot{\theta}, \ddot{\theta}]^T$, i.e., that consists of the position, angular velocity, and acceleration of the robot links. The self-feedback connections in the hidden nodes of the adaptive algorithm allow the approximation of the plant dynamics. Otherwise, the gradient algorithm with no connected delays provide only static input and output mapping (Er et al., 2016; Pratama et al., 2013; Zain et al., 2017). Then, the adaptive algorithm is perceived as the most appropriate for the modelling of the inverse dynamics of the joints robot.

The parameters update of the adaptive algorithm consists of forward and backward computing. The forward computing is written as:

1 Input layer:

$$I_m(k) = x_m(k), m = 1, 2, 3, ... \tag{23}$$

where $[x_1, x_2, x_3] = [\theta_i, \dot{\theta}_i, \ddot{\theta}_i]$.

2 Hidden layer:

$$\begin{aligned} H_j(k) &= F_j(u_j(k)) \\ u_j(k) &= \omega_j^{(2)} H_j(k-1) + \textstyle\sum_m \omega_{mj}^{(1)} x_m(k) \end{aligned} \tag{24}$$

3 Output layer:

$$O(k) = N_i\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) = \sum_j \omega_j^{(3)} H_j(k) \tag{25}$$

where $u_j(k)$ and $H_j(k)$ are the input and output of the hidden layer of the j^{th} unit, respectively, $O(k)$ is the output of the adaptive algorithm (pair of the i^{th} joint), $F_j(.)$ is a sigmoid function, written by $F_j(u_j) = \frac{1}{1+e^{-u_j}}$, $\omega_{mj}^{(1)}$ is the parameter that connects the m^{th} neuron of the input layer with the j^{th} neuron of the hidden layer, $\omega_j^{(2)}$ is the self-feeding of the j^{th} neuron of the hidden layer and $\omega_j^{(3)}$ is the parameter that connects the j^{th} node of the hidden layer to the node of the output layer.

The parameters updating in the adaptive algorithm is based on the descending gradient:

$$\frac{\partial E_i}{\partial \omega_j^{(3)}} = -\varepsilon_i(k)\frac{\partial O(k)}{\partial \omega_j^{(3)}} \tag{26}$$

$$\frac{\partial E_i}{\partial \omega_j^{(2)}} = -\varepsilon_i(k)\frac{\partial O(k)}{\partial H_j(k)}\frac{\partial H_j(k)}{\partial \omega_j^{(2)}} = -\varepsilon_i(k)\,\omega_j^{(3)}\alpha_j^{(i)}(k) \tag{27}$$

$$\frac{\partial E_i}{\partial \omega_j^{(1)}} = -\sum_j\left[\varepsilon_i(k)\frac{\partial O(k)}{\partial H_j(k)}\right]\frac{\partial H_j(k)}{\partial \omega_{mj}^{(1)}} = \sum_j \varepsilon_i(k)\,\omega_j^{(3)}\beta_{mj}^{(i)}(k) \tag{28}$$

where the index i denotes the adaptive algorithm that models the inverse dynamics of the i^{th} flexible joint:

$$\alpha_j^{(i)}(k) = \frac{\partial H_j(k)}{\partial \omega_j^{(2)}} = F'(u_j(k))\left[H_j(k-1) + \omega_j^{(2)}\alpha_j^{(i)}(k-1)\right] \tag{29}$$

$$\beta_{mj}^{(i)}(k) = \frac{\partial H_j(k)}{\partial \omega_{mj}^{(1)}} = F'(u_j(k))\left[x_m(k) + \omega_j^{(2)}\beta_{mj}^{(i)}(k-1)\right] \tag{30}$$

with $F'(u_j(k)) = F_j(u_j)(1 - F_j(u_j))$. The law for parameters updating is:

$$\omega(k+1) = \omega(k) - \delta\nabla E_i(\omega) \tag{31}$$

and the gradient $\nabla E_i(\omega)$ is given by:

$$\nabla E_i(\omega) = \left[\frac{\partial E_i}{\partial \omega_{mj}^{(1)}}, \frac{\partial E_i}{\partial \omega_j^{(2)}}, \frac{\partial E_i}{\omega_j^{(3)}}\right]^T \tag{32}$$

where $\omega = [\partial\omega_{mj}^{(1)}, \partial\omega_j^{(2)}, \omega_j^{(3)}]^T$ and δ is the rate of learning. In equations (21), (22) and (31), it can be seen that the parameters involved in the parameters adapting in the networks are: the output of the robotic plant θ, $\dot{\theta}$, $\ddot{\theta}$, the output of the PD methods, and the learning rate δ.

3.3 Local stability properties of the adaptive strategy

The parameters updating law given in equation (31), can be properly tuned to ensure the stability of the closed loop plant. Thus, following discrete-time Lyapunov function is defined:

$$V(k) = \frac{1}{2}\varepsilon_i(k) \tag{33}$$

Before, the change of the Lyapunov function, due to the parameters updating is given by:

$$\triangle V(k) = V(k+1) - V(K) = \frac{1}{2}\left[\varepsilon_i^2(k+1) - \varepsilon_i^2(k)\right] \tag{34}$$

The evolution of the modelling error ε_i in discrete time is given by:

$$\varepsilon_i(k+1) = \varepsilon_i(k) + \triangle\varepsilon_i(k) = \varepsilon_i(k) + \left[\frac{\partial E_i}{\partial \omega}\right]^T \triangle\omega \tag{35}$$

where $\triangle\omega$ represents a change in a vector of arbitrary parameter. From the laws of the parameters updating, in the equation (26) to (28), the parameters of the output layer are obtained as:

$$\triangle\omega = -\delta\nabla E_i(\omega) = \delta\varepsilon_i(k)\frac{\partial O(k)}{\partial \omega} \tag{36}$$

Based on the above, we have the following general convergence lemma.

Lemma 1: Be δ the learning rate of the parameters in the adaptive algorithm and $z_{\max}$ is defined as $z_{\max} = \max\|z(k)\|^2$, where $z(k) = \frac{\partial O(k)}{\partial \omega}$ and $\|\cdot\|$ is the Euclidean norm in $\Re^n$.

Convergence is guaranteed if δ is chosen as:

$$0 < \delta < \frac{2}{z_{\max}^2} \tag{37}$$

Using the lemma, the learning rate δ that ensures the convergence of the parameters updating in the adaptive algorithm, acceptable ranges of variation of learning rates δ are given in the following theorem.

Theorem 1: Be δ_3 The learning rate for the parameters in the adaptive algorithm $\omega^{(3)}$. The dynamics of the gradient algorithm converges if $0 < \left|\omega_j^{(3)}\right| < 1,\ (j = 1, 2, ..., h)$ and the learning rate δ_3 is selected as:

$$0 < \delta_3 < \frac{2}{h} \tag{38}$$

where h is the number of recurrent neurons in the hidden layer.

Proof: From equation (26), it is satisfied that:

$$z(k) = \frac{\partial O(k)}{\partial \omega^{(3)}} = H(k) \tag{39}$$

where $H(k) = [h_1(k), h_2(k), ..., h_h(k)]^T$ and $h_j(k)$ is the value of the output of the j^{th} neuron in the hidden layer. Since $0 < h_j(k) < 1$ (due to the sigmoid function $F(x) = 1/(1+e^{-x})$) and using the definition of the Euclidean norm in $\Re^h$, it is satisfied that $\|z(k)\| < \sqrt{h}$ and $z_{\max}(k) = h$. From the previous lemma, the equation (38) can be obtained. In the same way, learning rates δ_2 and δ_1 for parameters $\omega^{(2)}$ and $\omega^{(1)}$ can be gotten as:

$$0 < \delta_2 < \frac{1}{h}\left[\frac{1}{\omega_{\max}^{(2)}}\right]^2 \tag{40}$$

$$0 < \delta_1 < \frac{1}{h+3}\left[\frac{1}{\omega_{\max}^{(1)} x_{\max}}\right]^2 \tag{41}$$

where $\omega_{\max}^{(2)} = \max\left\|\omega^{(2)}(k)\right\|$ ■

4 Simulations

In the simulations, the adaptive strategy is compared with the PD for trajectory tracking of the robot end effector described by the links 10 and 11. The part of the robot that is introduced into the abdominal cavity must reach a maximum distance between 0.24 and 0.30 m from the point of incision (Madhani, 1998; Mosso et al., 2001).

4.1 PD strategy

For the PD strategy, the following gains were employed for the simulations:

$$K_d = \begin{bmatrix} 10 & 0 \\ 0 & 12 \end{bmatrix}, K_p = \begin{bmatrix} 90 & 0 \\ 0 & 85 \end{bmatrix}, U(t) = k_p e + k_d \dot{e}.$$

The graphs are shown in Figures 8, 9, 10 and 11, with their corresponding errors in Figures 12, 13, 14 and 15.

4.2 Adaptive strategy

The adaptive strategy is utilised for the simulations with the following parameters

$$K_d = \begin{bmatrix} 8 & 0 \\ 0 & 7 \end{bmatrix},\ K_p = \begin{bmatrix} 120 & 0 \\ 0 & 179 \end{bmatrix},\ U(t) = N\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) + k_p e + k_d \dot{e},$$

$$O(k) = N_i\left(\theta, \dot{\theta}, \ddot{\theta}, \omega\right) = \sum_j \omega_j^{(3)} H_j(k),\ \omega(k+1) = \omega(k) - \delta \nabla E_i(\omega),$$

$$\nabla E_i(\omega) = \left[\frac{\partial E_i}{\partial \omega_{mj}^{(1)}}, \frac{\partial E_i}{\partial \omega_j^{(2)}}, \frac{\partial E_i}{\omega_j^{(3)}}\right]^T,\ F_j(u_j) = \frac{1}{1+e^{-u_j}}.$$

Figure 8 Tracking of link 10 with PD strategy and square reference (see online version for colours)

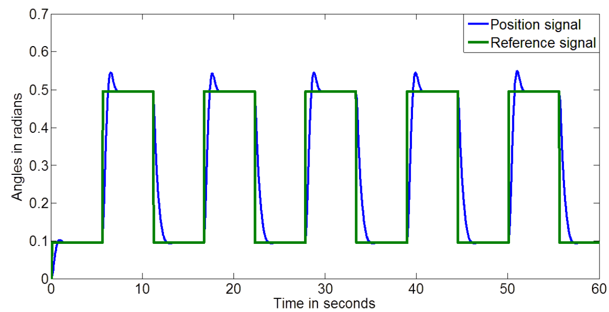


Figure 9 Tracking of link 10 with PD strategy and saw tooth reference (see online version for colours)

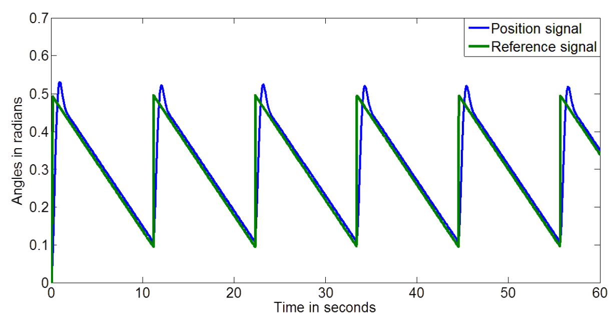


Figure 10 Tracking of link 11 with PD strategy and square reference (see online version for colours)

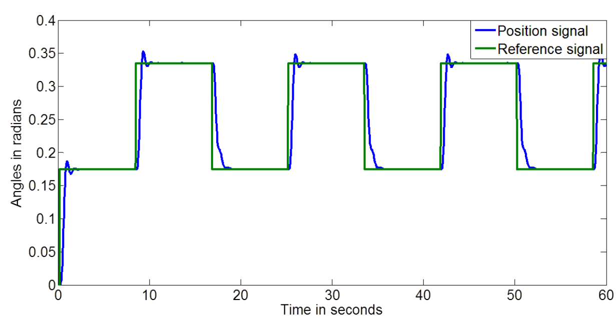


Figure 11 Tracking of link 11 with PD strategy and saw tooth reference (see online version for colours)

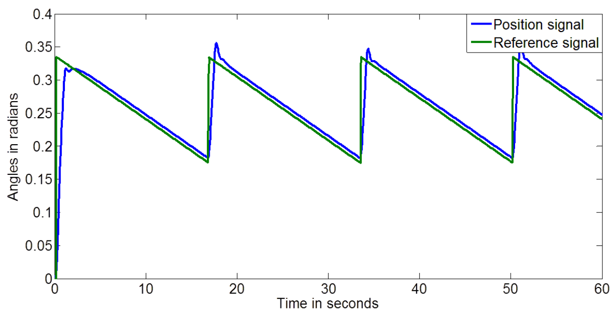


Figure 12 Error of link 10 with PD strategy and square reference (see online version for colours)

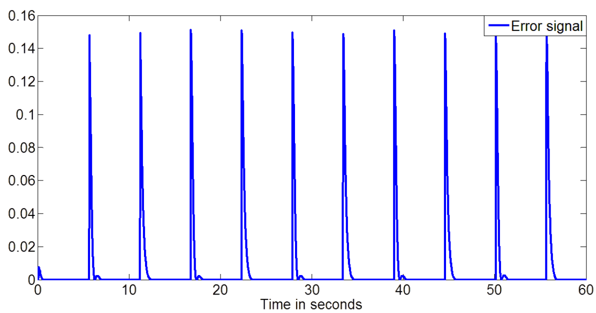


Figure 13 Error of link 10 with PD strategy and tooth saw reference (see online version for colours)

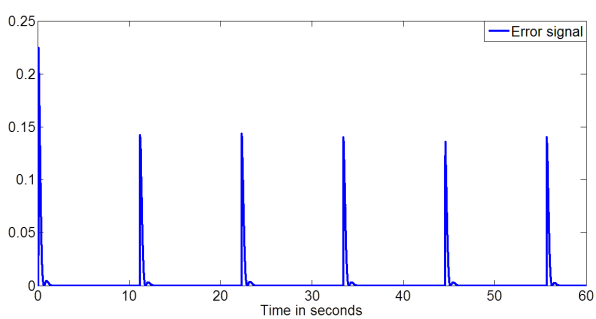


Figure 14 Error of link 11 with PD strategy and square reference (see online version for colours)

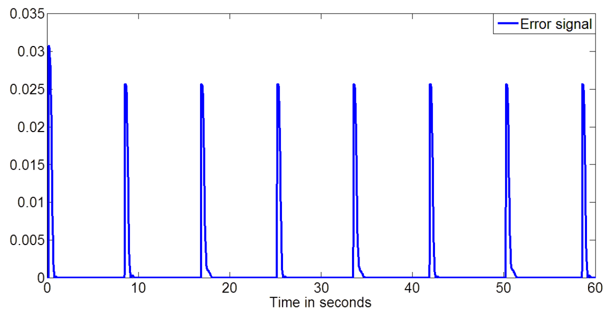


Figure 15 Error of link 11 with PD strategy reference tooth saw (see online version for colours)

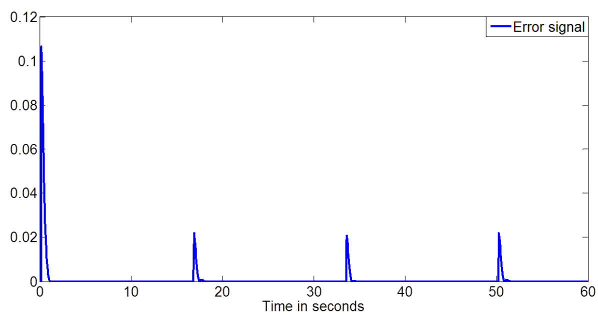


Figure 16 Tracking of link 10 with adaptive strategy and square reference (see online version for colours)

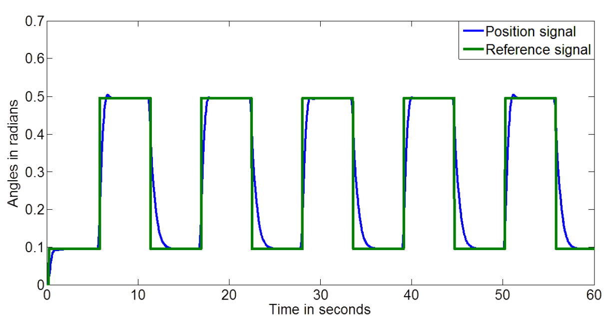


Figure 17 Tracking of link 10 with adaptive strategy and saw tooth reference (see online version for colours)

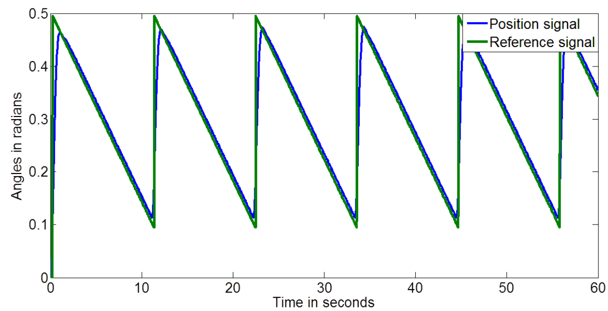


Figure 18 Tracking of link 11 with adaptive strategy and square reference (see online version for colours)

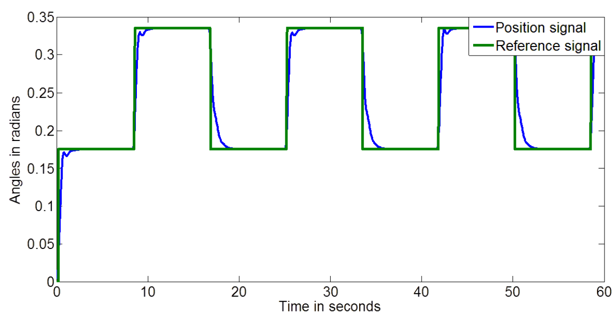


Figure 19 Tracking of link 11 with adaptive strategy and saw tooth reference (see online version for colours)

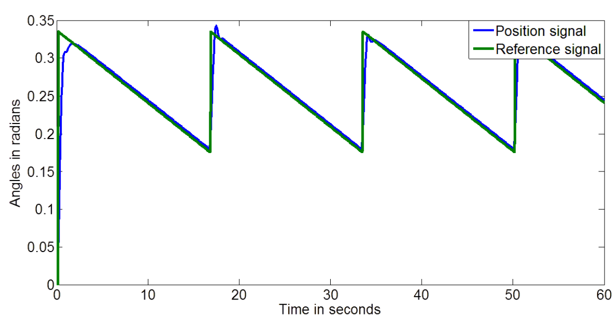


Figure 20 Error of link 10 with adaptive strategy and square reference (see online version for colours)

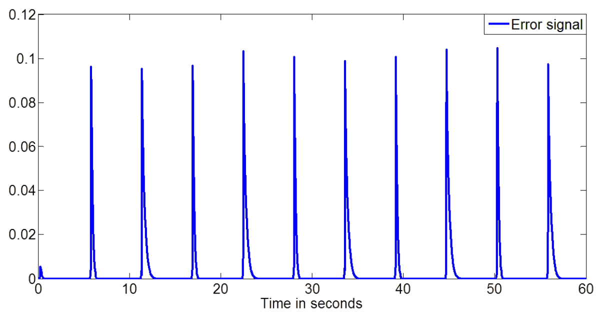


Figure 21 Error of link 10 with adaptive strategy and saw tooth reference (see online version for colours)

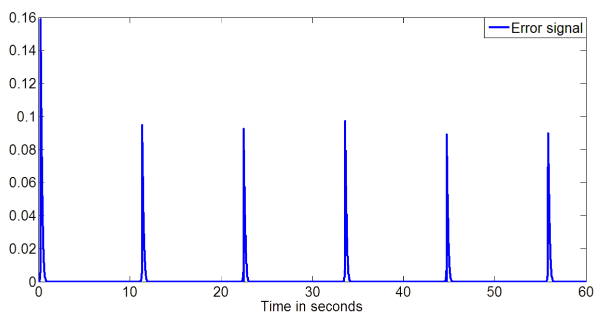


Figure 22 Error of link 11 with adaptive strategy and square reference (see online version for colours)

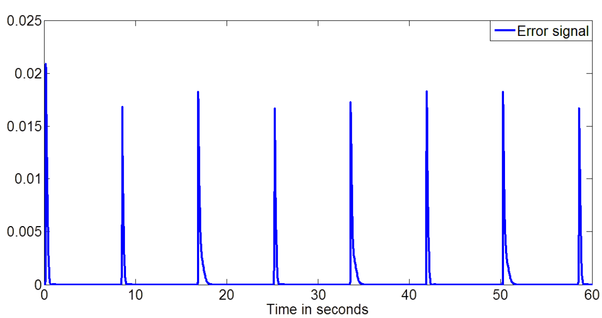
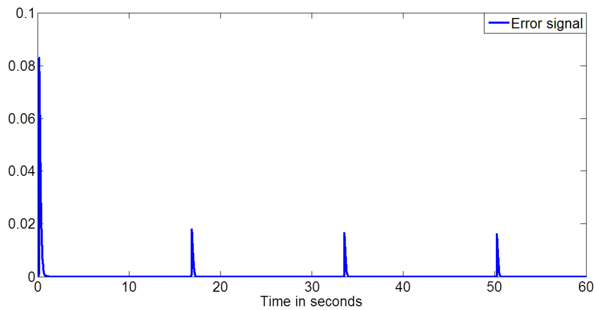


Figure 23 Error of link 11 with adaptive strategy and saw tooth reference (see online version for colours)



The results of the simulations are shown in Figures 16, 17, 18 and 19, with the errors in Figures 20, 21, 22 and 23, respectively.

4.3 Comparisons

In this work, an adaptive strategy for a robot for minimum invasive surgery, it is more efficient than the PD strategy, both strategies are compared. In addition, one way to see this comparison is by the utilisation of the root mean square error (EMC), Table 2 shown the comparison results.

Table 2 EMC comparison results

Signal type	*EMC – saw tooth reference*	*EMC – square reference*
PD strategy	0.2458	0.3349
Adaptive strategy	0.002101	0.00291

5 Conclusions

In this work, an adaptive strategy was proposed for the trajectory tracking of the end effector in a robot utilised in minimum invasive surgeries, it was compared in simulations with a PD method. The stability of the introduced strategy is studied based in the Lyapunov method. As a future work, it will be applied for a real minimum invasive surgery.

Acknowledgements

The authors are grateful with the editors and reviewers for their valuable comments and insightful suggestions, which helped to improve this research significantly. The authors thank the Instituto Politécnico Nacional, the Secretaría de Investigación y Posgrado, the Comisión de Operación y Fomento de Actividades Academicas, and the Consejo Nacional de Ciencia y Tecnología for their help in this research.

References

Andonovski, G., Angelov, P., Blazic, S. and Skrjanc I. (2016) 'A practical implementation of robust evolving cloud-based controller with normalized data space for heat-exchanger plant', *Applied Soft Computing*, November, Vol. 48, pp.29–38.

Angelov, P., Skrjanc, I. and Blazic, S. (2015) 'A robust evolving cloud-based controller', *Springer Handbook of Computational Intelligence*, pp.1435–1449, Springer Berlin Heidelberg.

Chandrahaasan, K. and Gnanaprakasaam, A. (2016) 'Optimal feature subset selection in high dimensional data clustering', *Int. J. Business Intelligence and Data Mining*, Vol. 11, No. 3, pp.242–263.

Climent, J. and Mares, P. (2003) 'Sistema de seguimiento en tiempo real para intervenciones quirúrgicas asistidas', *IEEE Latin America Transactions*, Vol. 1, No. 1, pp.8–14.

Davies, B., Hibberd, R., Coptcoat, M. and Wickham J. (1989) 'A surgeon robot prostatectomy – a laboratory evaluation', *Journal of Medical Engineering Technology*, Vol. 13, No. 6, pp.273–277.

Er, M.J., Zhang, Y., Wang, N. and Pratama, M. (2016) 'Attention pooling-based convolutional neural network for sentence modelling', *Information Sciences*, December, Vol. 373, pp.388–403.

Fu, K.S., Gonzalez, R.C. and Lee, G.S.G. (1987) *Robotics: Control, Sensing, Vision and Intelligence*, McGraw-Hill, New York.

Lewis, F.L., Jagannathan, S. and Yesildirek, A. (1999) *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor & Francis, Abington.

Joo, Y. (2017) 'Research on systematisation shipbuilding production management platform for flexible and agile controls on very complex production environments', *Int. J. Business Intelligence and Data Mining*, Vol. 12, No. 1, pp.25–43.

Khalil, W. and Dombre, E. (2002) *Modeling, Identification and Control of Robots*, Kogan Page Science, London, UK.

Kim, K.Y., Song, H.S., Suh, J.W. and Lee, J.J. (2013) 'A novel surgical manipulator with workspace-conversion ability for telesurgery', *IEEE/ASME Transactions on Mechatronics*, Vol. 18, No. 1, pp.200–211.

Krupa, A., Gangloff, J., Doignon, C., de Mathelin, M.F., Morel, G., Leroy, J., Soler, L. and Marescaux, J. (2003) 'Autonomous 3-D positioning of surgical instruments in robotized laparoscopic surgery using visual servoing', *IEEE Transactions on Robotics and Automation*, Vol. 19, No. 5, pp.842–853.

Kumar, P.D., Praveen, R. and Shoba, C. (2016) 'Optimal blowfish algorithm-based technique for data security in cloud', *Int. J. Business Intelligence and Data Mining*, Vol. 11, No. 2, pp.171–189.

Kwoh, Y., Hou, J., Jonckheere, E. and Hayall, S. (1988) 'A robot with improved absolute positioning accuracy for CT guide, stereoscopic brain surgery', *IEEE Transactions on Biomedical Engineering*, Vol. 35, No. 2, pp.153–161.

Li, Y. and Xu, Q. (2007) 'Design and development of a medical parallel robot for cardiopulmonary resuscitation', *IEEE/ASME Transactions on Mechatronics*, Vol. 12, No. 3, pp.265–273.

Lum, M.J.H., Rosen, J., Sinanan, M.N. and Hannaford, B. (2006) 'Optimization of a spherical mechanism for a minimally invasive surgical robot: theoretical and experimental approaches', *IEEE Transactions on Biomedical Engineering*, Vol. 53, No. 7, pp.1440–1445.

Madhani, A. (1998) *Design of Teleoperated Surgical Instruments for Minimally Invasive Surgery*, PhD thesis, Massachusetts Institute of Technology (MIT).

Malekzadeh, M., Sadati, J. and Alizadeh, M. (2016) 'Adaptive PID controller design for wing rock suppression using self-recurrent wavelet neural network identifier', *Evolving Systems*, Vol. 7, No. 4, pp.267–275.

Marescaux, J. and Rubino, F. (2003) 'The Zeus robotic system: experimental and clinical applications', *Surgical Clinics of North America*, Vol. 83, No. 6, pp.1305–1315.

Marohn, M. and Hanly, E. (2004) 'Twenty-first century surgery using twenty-first century technology: surgical robotics', *Current Surgery*, Vol. 61, No. 5, pp.466–473.

Mosso, J., Minor, A., Lara, V. and Maya, E. (2001) 'Brazo robótico para sujetar y posicionar laparoscopios', *Revista Cirugía y Cirujanos*, Vol. 69, No. 6, pp.295–299.

Navarro-Gonzalez, J.L., Lopez-Juarez, I., Ordaz-Hernandez, K. and Rios-Cabrera, R. (2015) 'On-line incremental learning for unknown conditions during assembly operations with industrial robots', *Evolving Systems*, Vol. 6, No. 2, pp.101–114.

Pan, Y., Er, M.J., Sun, T., Xu, B. and Yu, H. (2017) 'Adaptive fuzzy PD control with stable H∞ tracking guarantee', *Neurocomputing*, May, Vol. 237, pp.71–78.

Pan, Y., Liu, Y., Xu, B. and Yu, H. (2016a) 'Hybrid feedback feedforward: an efficient design of adaptive neural network control', *Neural Networks*, April, Vol. 76, pp.122–134.

Pan, Y., Zhang, J. and Yu, H. (2016b) 'Model reference composite learning control without persistency of excitation', *IET Control Theory and Applications*, Vol. 10, No. 16, pp.1963–1971, DOI: 10.1049/iet-cta.2016.0032.

Pan, Y., Yu, H. and Er, M.J. (2014) 'Adaptive neural PD control with semiglobal asymptotic stabilization guarantee', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 25, No. 12, pp.2264–2274.

Perry, J.C., Rosen, J. and Burns, S. (2007) 'Upper-limb powered exoskeleton design', *IEEE/ASME Transactions on Mechatronics*, Vol. 12, No. 4, pp.408–417.

Piccigallo, M., Scarfogliero, U., Quaglia, C., Petroni, G., Valdastri, P., Menciassi, A. and Dario, P. (2010) 'Design of a novel bimanual robotic system for single-port laparoscopy', *IEEE/ASME Transactions on Mechatronics*, Vol. 15, No. 6, pp.871–878.

Pratama, M., Rajab, S. and Er, M.J. (2011) 'Extended approach of ANFIS in cascade control', *International Journal of Computer and Electrical Engineering*, Vol. 3, No. 4, pp.572–576.

Pratama, M., Er, M.J., Li, X., Oentaryo, R.J., Lughofer, E. and Arifin, I. (2013) 'Data driven modeling based on dynamic parsimonious fuzzy neural network', *Neurocomputing*, June, Vol. 110, pp.18–28.

Precup, R.E., Angelov, P., Jales Costa, B.S. and Sayed-Mouchaweh M. (2015a) 'An overview on fault diagnosis and nature-inspired optimal control of industrial process applications', *Computers in Industry*, December, Vol. 74, pp.75–94.

Precup, R.E., Hellendoorn, H. and Angelov, P. (2015b) 'Synergy of computers, cognition, communication and control with industrial applications', *Computers in Industry*, December, Vol. 74, pp.71–74.

Rajesh, A., Sasikala, R. and Kalpana, A.M. (2016) 'An optimal record retrieval technique in text mining using GSO-based prefix span algorithm and improved K-means', *Int. J. Business Intelligence and Data Mining*, Vol. 11, No. 3, pp.264–281.

Sadeghi-Tehran, P. and Angelov, P. (2014) 'ATDT: autonomous template-based detection and tracking of objects from airborne camera', *Intelligent Systems*, pp.555–565, Springer International Publishing, New York; Dordrecht; London.

Schwartz, S. and Hunter J. (1999) *Principios de Cirugía*, 7th ed., McGraw-Hill Interamericana, México DF, Capítulo 44, México.

Spong, M.W. and Vidyasagar, M. (1989) *Robot Dynamics and Control*, J. Wiley, Chichester.

Stoianovici, D. (2000) 'Robotic surgery', *World Journal Urology*, Vol. 18, No. 4, pp.289–295.

Tholey, G. and Desai, J.P. (2007) 'A general-purpose 7 dof haptic device: applications toward robot-assisted surgery', *IEEE/ASME Transactions on Mechatronics*, Vol. 12, No. 6, pp.662–669.

Toubakh, H. and Sayed-Mouchaweh, M. (2015) 'Hybrid dynamic data-driven approach for drift-like fault detection in wind turbines', *Evolving Systems*, Vol. 6, No. 2, pp.115–129.

Wang, F.Y. and Gao, Y. (2004) *Advanced Studies of Flexible Robotic Manipulators*, World Scientific, Singapore.

Zain, C., Pratama, M., Lughofer E. and Anavatti S.G. (2017) 'Evolving type-2 web news mining', *Applied Soft Computing*, May, Vol. 54, pp.200–220.