2012 American Control Conference
Fairmont Queen Elizabeth, Montréal, Canada
June 27-June 29, 2012

Decentralized Simple Adaptive Control for Nonsquare Euler-Lagrange Systems

Steve Ulrich, *Student Member, IEEE*, Jurek Z. Sasiadek, *Senior Member, IEEE*, and Itzhak Barkana, *Senior Member*, *IEEE*

***Abstract*—This paper presents a novel direct model reference adaptive control scheme using a decentralized update law for nonsquare Euler-Lagrange systems. The proposed decentralized adaptation law is based on the simple adaptive control (SAC) methodology in which the control gain matrices are updated by the reference model input and state signals, and by the errors between the nonlinear system and a reference model. Numerical simulation results are presented to illustrate the trajectory tracking performance of the proposed approach in comparison with both a decentralized modified simple adaptive control (DMSAC) and a fuzzy logic-based direct adaptive control strategy. Results demonstrate that the designed adaptive control approach achieves improved tracking results compared to the DMSAC and fuzzy logic control methodologies.**

I. Introduction

CONTRARILY to indirect adaptive control methods, the idea behind direct adaptive control techniques consists in merging the identification and control functions into one scheme. With direct methods, the control gains are calculated in a way to force the actual system to behave as closely as possible to a reference model without the use of an explicit identification of system parameters. One obvious advantage of direct over indirect adaptive control is computational efficiency.

Most standard direct adaptation laws assume prior knowledge of the system order and require the use of models of the same order as the system. Since this assumption may be restrictive in actual systems, a direct adaptive control approach, known as simple adaptive control (SAC) was developed by Sobel et al. [1], Barkana et al. [2], and Barkana and Kaufman [3], [4] to mitigate this order-matching requirement.

The SAC technique is similar to the well-known command generator tracker approach [5], but replaces the constant, or fixed, control gains with time-varying ones. The resulting SAC scheme is an output feedback approach that does not require state observers, is applicable to nonminimum phase and MIMO systems, and in which the reference model to be tracked is allowed to be of any order (just sufficiently large to generate the desired command).

However, the greatest difficulty in designing the SAC controller is to determine the various weighting matrix coefficients. In fact, the SAC theory does not provide any means of efficiently selecting the several coefficients of the weighting matrices, although Barkana and Kaufman [6] provide some tuning guidelines. To overcome this design complexity, Ulrich and de Lafontaine [7] recently derived a modified version of the SAC algorithm for SISO systems, referred to as the modified SAC (MSAC) adaptation law. The MSAC algorithm neglects the adaptive control gains that are adapted as a function of the reference model input and state signals. Hence, since the MSAC adaptation mechanism considers only on the control gain being adapted as a function of the error between the reference model outputs and the plant outputs, its computational efficiency is greatly improved over the original SAC methodology, making it suitable for applications where the computational power is limited. The performance comparison of the SAC and MSAC technique for the control of a SISO aircraft system is provided in [8]. More recently, the *decentralized modified simple adaptive control* (DMSAC) methodology was developed for MIMO Euler-Lagrange systems (two-link planar robot manipulators) by adopting a local decentralized approach in which only the diagonal elements of the adaptive control gain matrices are considered [9]. This way, the number of required computation is further decreased. An application of the DMSAC technique for a flexible-joint manipulator can also be found in [10].

Within this context, the original contribution of this paper consists in the design of a *decentralized simple adaptive control* (DSAC) scheme for nonlinear square MIMO Euler-Lagrange systems, with an application to a two-link robot manipulator system. The motivation behind the proposed control strategy is to use more information about the reference model in the adaptation law in order to decrease as much as possible the error between the reference model and the system. The performance of the proposed control approach will be evaluated in a trajectory tracking scenario by the end-effector of a two-link planar robot manipulator, and compared against the DMSAC technique [9] and a decentralized fuzzy logic-based direct adaptive control strategy developed by Ulrich and Sasiadek [11].

This paper is organized as follows. Section II presents the dynamics model of the nonsquare Euler-Lagrange system, and Section III describes the control objective. In Section IV, the proposed control strategy is detailed. Section

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under the Alexander Graham Bell Canada Graduate Scholarship, and Carleton University under the J.Y. and E.W. Wong Research Award in Mechanical and Aerospace Engineering.

S. Ulrich and J. Z. Sasiadek are with the Department of Mechanical and Aerospace Engineering, Carleton University, Ottawa, Ontario, K1S 5B6, Canada (e-mail: sulrich@connect.carleton.ca).

I. Barkana is with Barkana Consulting, 47209 Ramat-Hasharon, Israel (e-mail: i.barkana@ieee.org).

V provides a discussion on the stability. In Section VI, the simulation results are presented, and in Section VII, a conclusion is provided.

II. DYNAMIC MODEL

Neglecting gravity and disturbances, the class of systems considered in this study is described by the following second-order nonlinear Euler-Lagrange formulation:

$$M(q)\,\ddot{q}+C(q,\dot{q})\,\dot{q}=\tau \tag{1}$$

where $\tau\in\mathbb{R}^2$ denotes the control torque vector, $M(q)\in\mathbb{R}^{2\times2}$ is the symmetric and positive definite inertia matrix, $C(q,\dot{q})\in\mathbb{R}^{2\times2}$ denotes the matrix comprising the Coriolis and centripetal effects, and $q\in\mathbb{R}^2$ denotes the generalized coordinate vector. In the following, it is assumed that both q and $\dot{q}$ are measurable, and that the system matrices and Jacobian matrix, $J(q)\in\mathbb{R}^{2\times2}$, are known.

For a two-link manipulator, the inertia matrix $M(q)$ is given by [12]:

$$M(q)=\begin{bmatrix} M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix} \tag{2}$$

with

$$M_{11}=m_1l_{c1}^2+m_2(l_1^2+l_{c2}^2+2l_1l_{c2}\cos q_2)+I_{z1}+I_{z2} \tag{3}$$

$$M_{12}=M_{21}=m_2(l_{c2}^2+l_1l_{c2}\cos q_2)+I_{z2} \tag{4}$$

$$M_{22}=m_2l_{c2}^2+I_{z2} \tag{5}$$

and $C(q,\dot{q})$ is:

$$C(q,\dot{q})=-m_2l_1l_{c2}\sin q_2\begin{bmatrix}\dot{q}_2 & \dot{q}_1+\dot{q}_2\\ -\dot{q}_1 & 0\end{bmatrix} \tag{6}$$

where $m_i\in\mathbb{R}\ \forall i=1,2$ denotes the mass of the links, $q_i\in\mathbb{R}\ \forall i=1,2$ denotes the angular displacement of the revolute joints, $l_i\in\mathbb{R}\ \forall i=1,2$ denotes the length of the links, $l_{ci}\in\mathbb{R}\ \forall i=1,2$ denotes the distance from the previous joint to center of gravity of the link. Modeling the links as uniformly distributed masses, the moment of inertia of each link about the axis perpendicular to the *xy* plane (*z* axis) passing through the center of gravity of the links is denoted by $I_{zi}\in\mathbb{R}\ \forall i=1,2$ and is given by:

$$I_{zi}=\frac{m_il_i^2}{12} \tag{7}$$

The output vector $y\in\mathbb{R}^4$ is defined as the Cartesian position and velocity coordinates of the manipulator end-effector with respect to the robot reference frame, that is:

$$y=\begin{bmatrix}x_r & \dot{x}_r\end{bmatrix}^T \tag{8}$$

where the end-effector Cartesian position coordinate vector along the x and y axis is denoted by $x_r\in\mathbb{R}^2$, and is obtained as follows:

$$x_r=\Omega(q) \tag{9}$$

where $\Omega(q)\in\mathbb{R}^2$ is the forward kinematic transformation taking joint angular positions into Cartesian position coordinates. The Jacobian matrix is used to transform the joint velocity vector, $\dot{q}\in\mathbb{R}^2$, into Cartesian velocity coordinates, $\dot{x}_r\in\mathbb{R}^2$, as follows:

$$\dot{x}_r=J(q)\dot{q} \tag{10}$$

III. CONTROL OBJECTIVE

The control objective consists in designing a decentralized adaptive control law which ensures that the output vector y of the nonlinear system described in (1)-(10) tracks the output vector y_m of the following reference model:

$$\dot{x}_m=A_mx_m+B_mu_m \tag{11}$$

$$y_m=C_mx_m \tag{12}$$

where $A_m,B_m,C_m\in\mathbb{R}^{4\times4}$, and $x_m,y_m\in\mathbb{R}^4$ are defined as:

$$A_m=\begin{bmatrix}0&0&0&0\\0&0&0&0\\-\omega_n^2&0&-2\zeta\omega_n&0\\0&-\omega_n^2&0&-2\zeta\omega_n\end{bmatrix} \tag{13}$$

$$B_m=\begin{bmatrix}0&0&1&0\\0&0&0&1\\\omega_n^2&0&0&0\\0&\omega_n^2&0&0\end{bmatrix}\qquad C_m=I_4 \tag{14}$$

$$x_m=y_m\triangleq\begin{bmatrix}x_{rm}\\ \dot{x}_{rm}\end{bmatrix} \tag{15}$$

where, $\omega_n,\zeta\in\mathbb{R}$ represents the constant ideal damping ratio, and undamped natural frequency, I_4 denotes the 4×4 identity matrix, $x_{rm}\in\mathbb{R}^2$ denotes the ideal end-effector Cartesian position coordinates and where $\dot{x}_{rm}\in\mathbb{R}^2$ represents the ideal end-effector Cartesian velocity coordinates, as calculated by the reference model. The outputs of the reference model represent the response that exhibits satisfactory closed-loop performance when tracking the desired trajectory denoted by $u_m \in \mathbb{R}^4$. Following common practice, the desired (commanded) trajectory is defined in terms of Cartesian positions and velocity, as follows:

$$u_m \triangleq \begin{bmatrix} x_{rd} \\ \dot{x}_{rd} \end{bmatrix} \tag{16}$$

where $x_{rd} \in \mathbb{R}$ denotes the desired end-effector Cartesian position coordinates and where $\dot{x}_{rd} \in \mathbb{R}$ represents the desired end-effector Cartesian velocity coordinates. In this study, a large square trajectory was selected as the desired trajectory. The outputs of the reference model are compared to the outputs of the system to form the tracking error vector between them, $e_y \in \mathbb{R}^4$, that is:

$$e_y \equiv \begin{bmatrix} e_{y1} \\ e_{y2} \\ e_{y3} \\ e_{y4} \end{bmatrix} \triangleq y_m - y = \begin{bmatrix} x_{rm} - x_r \\ \dot{x}_{rm} - \dot{x}_r \end{bmatrix} \tag{17}$$

This output tracking error vector will be used in the adaptive controller to update the control gains according to the decentralized SAC approach, as it will be shown in the next section.

IV. Decentralized Simple Adaptive Control

Adopting the SAC control approach [5], the following transpose Jacobian control law applicable to Euler-Lagrange systems is herein proposed:

$$\tau \triangleq J^T(q)\left[K_e(t)e_y + K_x(t)X_m + K_u(t)U_m\right] \tag{18}$$

where $K_e(t) \in \mathbb{R}^{2\times4}$ is an adaptive stabilizing control gain matrix, and $K_x(t) \in \mathbb{R}^{2\times4}$ and $K_u(t) \in \mathbb{R}^{2\times4}$ are adaptive feedforward control gain matrices that maintain stability of the controlled system and bring the output tracking error to zero. Each control gain matrix is calculated as the summation of a proportional and an integral component, as follows [5]:

$$K_e(t) = K_{Pe}(t) + K_{Ie}(t) \tag{19}$$

$$K_x(t) = K_{Px}(t) + K_{Ix}(t) \tag{20}$$

$$K_u(t) = K_{Pu}(t) + K_{Iu}(t) \tag{21}$$

It must be noted that only the integral adaptive control gains are absolutely necessary to guarantee the stability of the proposed direct adaptive control system. However, it is common practice to include proportional adaptive control gains as well, in order to increase the rate of convergence of the adaptive system toward perfect tracking.

Compared with previous work on SAC-based control for nonsquare robot manipulators [9], [10], the main difference with the proposed control law (18) is that more information about the reference model is used in the controller structure. In other words, as shown in (18), the calculation of τ depends on $K_x(t)$ and $K_u(t)$, the feedforward gain matrices multiplying x_m and u_m, respectively, whereas in [9], [10], τ is only obtained as a function of $K_e(t)$.

Proposing a local decentralized SAC-based adaptation law, the proportional and the integral components of the stabilizing control gain in (19), $K_{Pe}(t) \in \mathbb{R}^{2\times4}$ and $K_{Ie}(t) \in \mathbb{R}^{2\times4}$, are both driven by the tracking error, thus resulting in the following update law:

$$K_{Pe}(t) = W^T \operatorname{diag}\left\{e_y e_y^T\right\}\Gamma_{Pe} \tag{22}$$

$$\dot{K}_{Ie}(t) = W^T\left[\operatorname{diag}\left\{e_y e_y^T\right\}\Gamma_{Ie} - \operatorname{diag}\left\{\sigma_e W K_{Ie}\right\}\right] \tag{23}$$

where $W \in \mathbb{R}^{2\times4}$ is a scaling matrix defined by

$$W \triangleq \begin{bmatrix} I_2 & I_2 \end{bmatrix}$$

where I_2 represents the 2×2 identity matrix and where $\operatorname{diag}\{A\}$ denotes the diagonal of a square matrix $A \in \mathbb{R}^{n\times n}$ whose elements are denoted $a_{i,j}$, and is given by:

$$\operatorname{diag}\{A\} = \begin{bmatrix} a_{1,1} & 0 & \cdots & 0 \\ 0 & a_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{n,n} \end{bmatrix} \tag{24}$$

Similarly, the components of the feedforward control gain matrices, $K_{Px}(t), K_{Ix}(t), K_{Pu}(t), K_{Iu}(t) \in \mathbb{R}^{2\times4}$, are updated as follows:

$$K_{Px}(t) = W^T \operatorname{diag}\left\{e_y x_m^T\right\}\Gamma_{Px} \tag{25}$$

$$\dot{K}_{Ix}(t) = W^T\left[\operatorname{diag}\left\{e_y x_m^T\right\}\Gamma_{Ix} - \operatorname{diag}\left\{\sigma_x W K_{Ix}\right\}\right] \tag{26}$$

$$K_{Pu}(t) = \begin{bmatrix} I_2 & I_2 \end{bmatrix} \operatorname{diag}(e_y u_m^T)\,\Gamma_{Pu} \tag{27}$$

$$\dot{K}_{Iu}(t) = W^T\left[\operatorname{diag}\left\{e_y u_m^T\right\}\Gamma_{Iu} - \operatorname{diag}\left\{\sigma_u W K_{Iu}\right\}\right] \tag{28}$$

In (22), (23) and (25)–(28), $\Gamma_{Pe}, \Gamma_{Ie}, \Gamma_{Px}, \Gamma_{Ix}, \Gamma_{Pu}, \Gamma_{Iu} \in \mathbb{R}^{4\times4}$ are constant diagonal parameter matrices that control the rate of adaptation. Based on Ioannou and Kokotovik's work [13], the diagonal coefficient matrices $\sigma_e, \sigma_x, \sigma_u \in \mathbb{R}^{4\times4}$ have been introduced in the adaptation law algorithm to avoid divergences of the integral adaptive control gains in cases where the tracking error would not reach zero. With this adjustment, the integral control gains are obtained as a first-order filtering of the tracking error.

V. Comments on Stability

It was recently shown by Barkana [14] that the basic almost strictly passivity conditions required in order to guarantee stability with nonlinear and nonstationary controllers, such as the controller proposed in this paper, are equivalent to requiring that the nonlinear plant $\{A_p(x,t), B_p(x,t), C_p(x,t)\}$ be uniformly strictly minimum-phase and with the product $C_p(x,t)B_p(x,t)$ being positive definite symmetric (PDS). Recall that a nonlinear square plant is uniformly strictly minimum-phase if its zero-dynamics is uniformly stable. However, such conditions are only applicable to square systems. Even in the linear time-invariant (LTI) case, equivalent conditions for nonsquare systems do not yet exist, although inroads to solve this problem for LTI systems are provided by Fradkov [15] and Rasvan and Stefan [16]. Nevertheless, the application of the new adaptive control law (18) to a nonsquare robot manipulator system demonstrates convergence of the trajectory tracking errors (see next section).

VI. Simulation Results

This section describes the numerical simulations that were conducted in order to analyze the performance of the adaptive controller developed in this paper. For completeness, the proposed controller is also compared to two other decentralized direct adaptive methodologies that were recently developed [9], [11]. To assess the performance of all controllers, a 12.6 m × 12.6 m square trajectory is required to be tracked in 60 seconds with constant velocity in a counter-clockwise direction starting at the lower-right-hand corner by the end-effector of a two-link planar manipulator described by (1), where the shoulder joint coincides with the fixed base of the robot located at the center of the square trajectory. Compared to other types of Cartesian trajectories, such as straight lines or circles, a square trajectory represents greater control challenges. Indeed, each corner of the square represents an abrupt change in directions. Hence, it is required that the endpoint reaches each corner and then redirects itself along an orthogonal direction with minimum overshoot and settling time. In the literature, such square trajectories have been widely used as benchmarks for controller performance assessment, for example see [17]-[20].

In all cases, the parameters of the two-link manipulator are selected as $m_1 = m_2 = 1.5075$ kg, $l_1 = l_2 = 4.50$ m, $l_{c1} = l_{c2} = 2.25$ m. The integral structure of the integral control gains in (23), (26), and (28) is computed online via a standard Tustin algorithm. In addition, all integral control gains were initialized to zero. The reference model parameters were selected as $\omega_n = 10$ rad/s and $\zeta = 0.9$.

Case 1: In the first case, the decentralized direct adaptive controller in (18) was implemented by considering only the adaptive stabilizing control gain matrix $K_e(t)$ defined in (19). The adaptation law given in (22) and (23) was used to adapt in real-time this control gain. The resulting DMSAC-based controller takes the form [9]:

$$\tau = J^T(q)K_e(t)e_y \tag{29}$$

The parameter and coefficient matrices for this controller that yielded the best performance were determined as follows:

$$\Gamma_{Pe} = \mathrm{diag}\{120, \quad 120, \quad 15, \quad 15\}$$

$$\Gamma_{Ie} = \mathrm{diag}\{180, \quad 180, \quad 30, \quad 30\}$$

$$\sigma_e = \mathrm{diag}\{0.018, \quad 0.018, \quad 0.018, \quad 0.018\}$$

Case 2: In the second case, the decentralized direct adaptive controller of the following form was implemented:

$$\tau = J^T(q)\begin{bmatrix} h_1\lambda_1(t) & 0 & h_3\lambda_3(t) & 0 \\ 0 & h_2\lambda_2(t) & 0 & h_4\lambda_4(t) \end{bmatrix} e_y \tag{30}$$

where $\lambda_i(t) \in \mathbb{R}$, $\forall i = 1, 2, 3, 4$ are the adaptive outputs of four normalized fuzzy logic systems (FLS). Each FLS is a Mamdani type and their single input variable, which corresponds to one element of the tracking error vector e_y through input scaling gains, is designed with nine Gaussian membership functions. Their single output variable, which corresponds to an adaptive control gain $\lambda_i(t)$, is designed with five Gaussian membership functions. Universes of discourse range from −2 to 2 m for e_{y1} and e_{y2}, from -10 to 10 m/s for e_{y3} and e_{y4}, and from 0 to 1 for the control gains $\lambda_i(t)$. The input scaling gains $g_i \in \mathbb{R}$, $\forall i = 1, 2, 3, 4$ are chosen so that the left-most membership function saturates (peaks) at -1 and the right-most one at +1 for both the input and output universe of discourse, hence resulting in a normalized FLS. Output scaling gains $h_i \in \mathbb{R}$, $\forall i = 1, 2, 3, 4$ multiply their corresponding adaptive control gain $\lambda_i(t)$ to modify their base widths and provide greater tracking accuracy as the scaling gain increases. More details about this decentralized direct adaptive control scheme using fuzzy logic can be found in [11].

Case 3: In the third case, the control input vector given in (18) was used. The DSAC-based adaptation law defined in (22)-(30) was used to update the control gains defined in (19)-(21). The following adaptation parameters and coefficients were selected to provide satisfactory tracking performance:

$$\Gamma_{Pe} = \Gamma_{Ie} = \mathrm{diag}\{1500, \quad 1500, \quad 125, \quad 125\}$$

$$\Gamma_{Px} = \Gamma_{Ix} = \Gamma_{Pu} = \Gamma_{Iu} = \mathrm{diag}\{100, \quad 100, \quad 100, \quad 100\}$$

$$\sigma_e = \mathrm{diag}\{0.018, \quad 0.018, \quad 0.018, \quad 0.018\}$$

$$\sigma_x = \sigma_u = \mathrm{diag}\{0.5, \quad 0.5, \quad 0.5, \quad 0.5\}$$

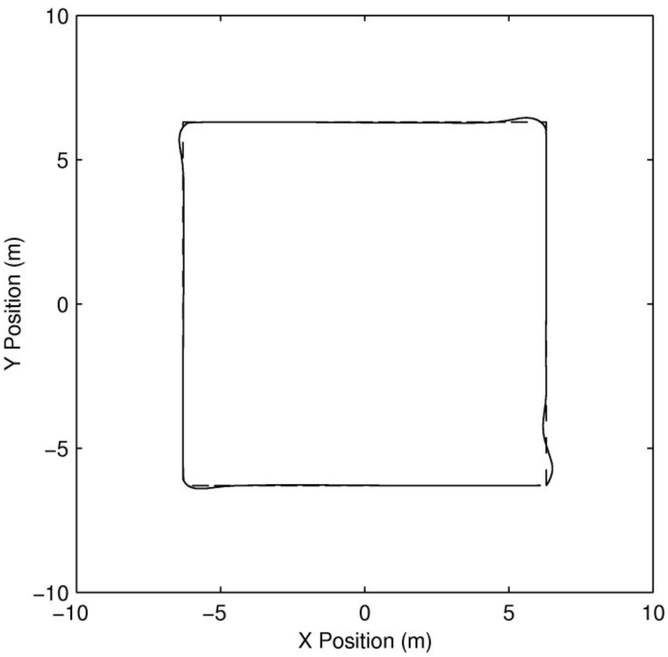

Fig. 1. DMSAC control tracking results [9].

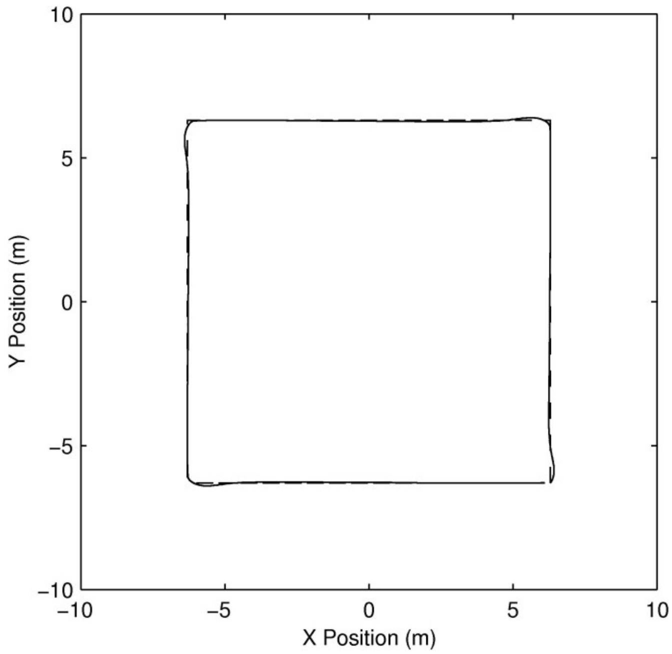

Fig. 3. Decentralized fuzzy logic-based control tracking results [11].

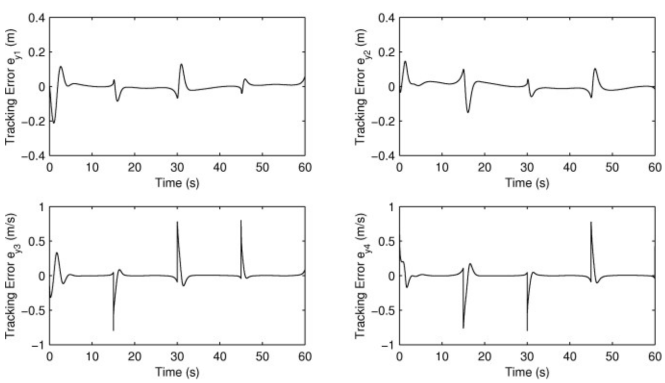

Fig. 2. DMSAC trajectory tracking errors.

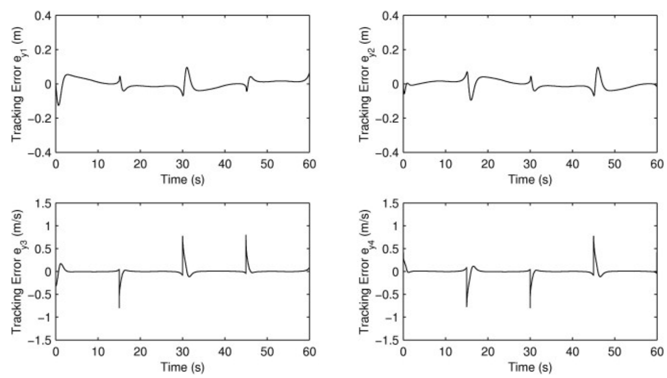

Fig. 4. Decentralized fuzzy logic-based trajectory tracking errors.

The trajectory tracking results obtained with the DMSAC-based controller (case 1), adaptive fuzzy logic-based controller (case 2) and the DSAC-based controller developed in this paper (case 3) are depicted in Figs. 1 to 6. In these figures, the dashed line corresponds to the desired end-effector position $x_{rd}(t)$ (i.e. the input of the reference model), and the solid line corresponds to the actual end-effector position $x_r(t)$.

As illustrated in Fig. 1, the trajectory for the DMSAC control approach exhibits position overshoots in the task space of 0.151 m, 0.129 m and 0.104 m at the first, second and third corner, respectively. The successive increase in tracking performance along each side of the trajectory can be explained by analyzing the adaptation history of the control gains, which are increasing at each direction change, and thus providing better tracking results, as explained and demonstrated in [9]. The resulting trajectory tracking errors are illustrated in Fig. 2

As shown in Figs. 3 and 4, the decentralized fuzzy logic-based direct adaptive control scheme provides better tracking accuracy when compared to the DMSAC strategy, especially at the corner of the trajectory where the tracking task is more challenging. In fact, the fuzzy logic-based scheme exhibits an overshoot of 0.100 m at each direction switch. Thus, the response obtained with the adaptive fuzzy control strategy is closer to a perfect square than that of DMSAC design.

Finally, the adaptive controller based on the DSAC methodology proposed in this paper performed better than the previous strategies, as illustrated in Fig. 5, where a square trajectory with no overshoots is obtained. Furthermore, a significant reduction in position tracking errors is achieved, as shown in Fig. 6.

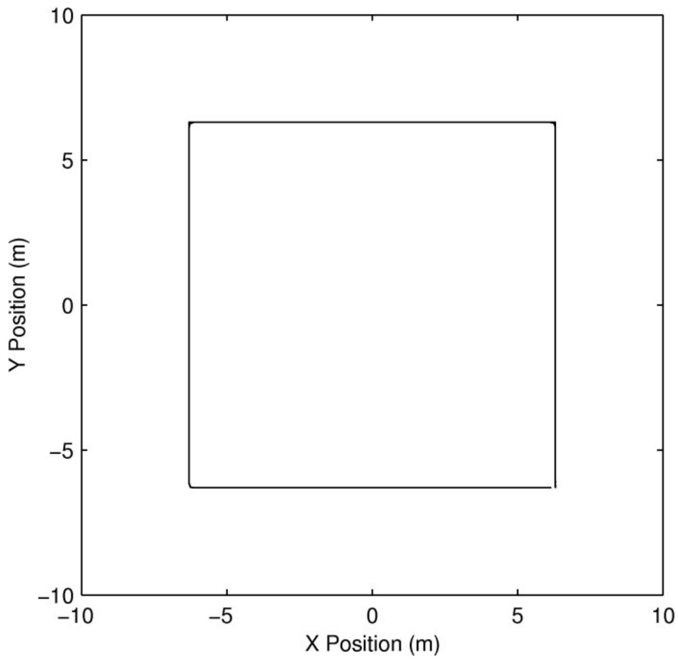

Fig. 5. DSAC control tracking results.

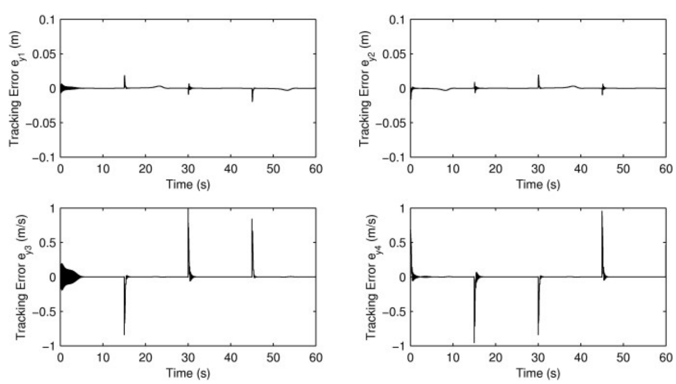

Fig. 6. DSAC trajectory tracking errors.

VII. CONCLUSION

A novel direct adaptive control technique was proposed for a class of second-order nonsquare Euler-Lagrange systems, with an application to two-link robot manipulators. The developed controller is based on a direct adaptive control approach in which the control gains are adapted using a decentralized SAC adaptation law to ensure that the system tracks a reference model. With the proposed approach, the control gains are adapted as a function of the tracking errors between the system and the reference model, and as a function of the reference model inputs and states with the motivation of injecting more information in the adaptation law.

REFERENCES

[1] K. Sobel, H. Kaufman and L. Mabius, "Implicit adaptive control for a class of MIMO systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 18, no. 5, pp. 576–589, 1982.

[2] I. Barkana, H Kaufman and M. Balas, "Model reference adaptive control of large structural systems," *J. Guidance, Control, and Dynamics*, vol. 6, no. 2, pp. 112–118, 1983.

[3] I. Barkana and H. Kaufman, "Some applications of direct adaptive control to large structural systems," *J. Guidance, Control, and Dynamics*, vol. 7, no. 6, pp. 717–724, 1984.

[4] I. Barkana and H. Kaufman, "Global stability and performance of an adaptive control algorithm," *Int. J. Control*, vol. 42, no. 6, pp. 1419–1505, 1985.

[5] M. O'Brien and R. Broussard, "Feedforward control to track the output of a forced model," *Proc. IEEE Conf. on Decision and Control*, Piscataway, NJ, pp. 1149–1155, 1979.

[6] H. Kaufman, I. Barkana and K. Sobel, *Direct Adaptive Control Algorithms: Theory and Applications*, 2nd Ed., Communications and Control Engineering Series, Springler, 1997.

[7] S. Ulrich and J. de Lafontaine, "Autonomous atmospheric entry on Mars: Performance improvement using a novel adaptive control algorithm," *The J. Astronautical Sci.*, Vol. 55, No. 4, pp. 431–449, 2007.

[8] S. Ulrich and J. de Lafontaine, "Development of a novel adaptive control algorithm for a fighter aircraft," *Proc. AIAA Guidance, Navigation and Control Conf. and Exhibit,* Hilton Head, South Carolina, 2007.

[9] S. Ulrich and J. Z. Sasiadek, "Modified simple adaptive control for a two-link space robot," *Proc. American Control Conf.*, Baltimore, MD, 30 June - 2 July, 2010.

[10] S. Ulrich and J. Z. Sasiadek, "Modeling and direct adaptive control of a flexible-joint manipulator," *J. Guidance, Control, and Dynamics*, vol. 35, no.1, pp. 25–39, 2012.

[11] S. Ulrich and J. Z. Sasiadek, "Direct adaptive fuzzy control for a two-link robot," *16th IEEE/IFAC Int. Conf. Methods and Models in Automation and Robotics*, Miedzyzdroje, Poland, 22-25 August, 2011.

[12] H. Asada, and J.-J. E. Slotine, *Robot Analysis and Control*, Wiley, New York, 1986, pp. 104–118.

[13] P. A. Ioannou and P. V. Kokotovic, "Singular perturbation and robust redesign of adaptive control," *Proc. 21st IEEE Conf. Decision and Control*, Orlando, Florida, Dec, 1982.

[14] I. Barkana, "Output feedback stabilizability and passivity in nonstationary and nonlinear systems," *Int. J. Adaptive Control and Signal Process.*, vol. 24, no. 7, pp. 568–591, 2010.

[15] A. L. Fradkov, "Passication of non-square linear systems and feedback Yakubovich-Kalman-Popov lemma," *European J. Control*, vol. 9, no. 6, pp. 573-582, 2003.

[16] V. Rasvan and R. Stefan, "Discussion on: Passification of non-square linear systems and feedback Yakubovich-Kalman-Popov lemma," *European J. Control*, vol. 9, no. 6, 2003.

[17] A. Green and J. Z.Sasiadek, "Heuristic design of a fuzzy controller for a flexible robot," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 2, pp. 293–300, 2006.

[18] A. K. Banerjee and W. Singhose, "Command shaping in tracking control of a two-Link flexible robot," *J. Guidance, Control, and Dynamics*, vol. 21, no. 6, pp. 1012–1015, 1998.

[19] M. Romano, B. N. Agrawal, and F. Bernelli-Zazerra, "Experiments on command shaping control of a manipulator with flexible links," *J. of Guidance, Control, and Dynamics*, vol. 25, no. 2, pp. 232–239, 2002.

[20] J. Carusone, K. S. Buchan, and G. M. T. D'Eleuterio, "Experiments in end-effector tracking control for structurally flexible space manipulators," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 553–560, 1993.