Free model control of robot manipulator end-effector tracking by an equivalent system using an adaptive Kalman filter

Josué Gómez, * América Morales, Chidentree Treesatayapun

* *Robotics and Advanced Manufacturing Programm CINVESTAV-Saltillo (e-mail:josuegomezcasas@gmail.com).*

Abstract: This paper proposes a free-model task space control for a robot's end-effector. The robot is considered a non-linear discrete time system, using the data driven control and model free control approach. An adaptive Kalman filter provides an equivalent system by means of the robotic Jacobian matrix, which only needs the input-output data to control the robot's position. Besides, we design the adaptive gains for sliding mode controller, using a neuro-fuzzy network structure. In the simulation was tested the performance of the equivalent Jacobian matrix and, slidng mode controller with adaptive gains for tracking control. We also provided a Lyapunov analysis of the equivalent model to guarantee convergence based on the adaptive Kalman filter.

Keywords: robot's end-effector, equivalent system, adaptive Kalman filter, adaptive gains, sliding mode control

1. INTRODUCTION

The methods of nonlinear system control intend to solve the main problematic for control design as: nonlinearities, parametric uncertainties, and superposition properties losses. The robot's motion control is a common issue for nonlinear control. Furthermore, the robotic application field has matured, and the manipulator robots control design needs to enhance the task execution accuracy. The classic control techniques allow to deal with main robot tasks, but the inaccuracy in the mathematical model can cause low performance in the robot. In contrast, currently the use of free model control has gained attention, in parallel with the current research tendency to use data control. The Data Driven Control (DDC) allows working with an equivalent system by the dynamic linearization model, Hou and Jin (2011). The Model Free Adaptive Control (MFAC) based on DDC has been implemented in robotic systems: Zen et al. (2018) designed an equivalent system by the pseudo Jacobian matrix, and Li et al. (2018) applied the Jacobian estimate by an adaptive Kalman filter to control a continuum robot. In previous work was presented a free-model robot control with a proportional controller, Gómez et al. (2018). The equivalent model (estimated Jacobian matrix) only needs the on-line input-output information, in comparison with classical robot model: the proposed control omits the knowledge of the class and the physical parameters. The objective of this paper is to test the end-effector tracking control with an equivalent system provided by an adaptive Kalman filter. The robotic plant is an omnidirectional mobile manipulator; whit 5 revolute dof in the robotic arm and 3 prismatic dof in the omnidirectional mobile platform. The Strong Tracking Kalman Filter (STKF) is a class of adaptive algorithm which allows to obtain the Jacobian matrix as an equivalent model, regarding the robot input-output data. The main characteristics of the equivalent model provided by the STKF are: the robustness against uncertainties, and instantaneous changes in the movement.

⋆ CONACyT support acknowledgment.

As well, we designed the controller based on the equivalent system. The control proposal is a Sliding Mode Controller (SMC) with on-line adaptive gains using the Fuzzy Rules Emulated Network structure (FREN) in the work presented by Treesatayapun and Uatrongjit (2005). The basic principle of SMC allows to an equivalent system deals with nonlinearities, external disturbance and uncertainties. Aditional, FREN adapts the SMC gains in terms of control error change. The simulation results show the equivalent model control performance, and we introduce the system model stability analysis. The structure of this paper is: section 2 describes of the unknown nonlinear discrete time system, section 3 presents the simulation results, and section 4 gives the conclusions.

2. DISCRETE TIME SYSTEM

The position of the robot's end-effector $\chi(k) = f(q_n)$ is:

$$\chi(k) = [p(q)] \ \in \mathbb{R}^m \subset SE(3), \tag{1}$$

where $p(q)$ denotes the end-effector position; m is the robot dof and n is the end-effector dof. The end-effector velocity approximates within a discrete time derivative:

$$\frac{\chi(k+1)-\chi(k)}{T_s} = J_A(k)\left[\frac{q(k)-q(k-1)}{T_s}\right] \ \in \mathbb{R}^m, \tag{2}$$

 Copyright©AMCA. Todos los Derechos Reservados www.amca.mx

where the Jacobian matrix is $J_A(k) = \dfrac{\partial f(\chi(k+1))}{\partial f(q_n(k))} \in \mathbb{R}^{m\times n}$ nad T_s is the sampling time

$$\frac{\chi(k+1)-\chi(k)}{T_s} = J_A(k)\omega(k) \ \in \mathbb{R}^m, \tag{3}$$

and $\omega(k) \in \mathbb{R}^n$ represents the discrete joint velocities

2.1 *Equivalent model system*

The Jacobian vector $J_v(k)$ contains the dynamics system from the Jacobian matrix $J_A(k)$ in the next way

$$J_v(k) = \left[\frac{\partial f_x}{\partial q_n}\ \frac{\partial f_y}{\partial q_n}\ \frac{\partial f_z}{\partial q_n}\right]^T \ \in \mathbb{R}^a, \tag{4}$$

where x, y and z are the end-effector axes, and $a = 24$ is the multiplication of the Jacobian matrix subspaces $m = 3$ and $n = 8$. The measurement matrix $H(k)$ is a diagonal matrix with the control signals $\omega(k)$:

$$H(k) = \begin{bmatrix} [\omega_1(k)\ \ldots\ \omega_n(k)] & & 0 \\ & \ddots & \\ 0 & & [\omega_1(k)\ \ldots\ \omega_n(k)] \end{bmatrix} \in \mathbb{R}^{m\times a} \tag{5}$$

The Strong Tracking Kalman Filter (STKF) is an adaptive algorithm with estimation error tracking, Li et al. (2018). Thus, fading factor λ_k enhances the robustness against uncertainties, and the updated covariance matrix Q_k captures the instantaneous change of the Jacobian. The STKF estimates the Jacobian vector $J_v(k)$ in equation (6)

$$\begin{aligned} K_F(k) =& P(k)H^T(k)\left[H(k)P(k)H^T(k)\right]^{-1} \\ P(k+1) =& \left[I - K(k)H(k)P(k)\right]\lambda_k + Q_k \\ \hat{J}_v(k+1) =& \hat{J}_v(k) + K(k)\left[\frac{\Delta\chi}{Ts} - H(k)\hat{J}_v(k)\right], \end{aligned} \tag{6}$$

where $K_F(k) \in \mathbb{R}^{a\times n}$ is the Kalman gain, $P(k+1) \in \mathbb{R}^{a\times a}$ is the error covariance matrix, $\hat{J}_v(k+1) \in \mathbb{R}^a$ is the updated state of Jacobian vector. The next expression defines the error estimation

$$v_n(k) = \frac{\Delta\chi}{Ts} - H(k)\hat{J}_v(k) \ \in \mathbb{R}^m, \tag{7}$$

the updated covariance matrix Q_k is

$$Q_k = K_F(k)\hat{C}(k)K_F(k), \tag{8}$$

the matrix $\hat{C}(k)$ is the result of weighting quadratic error estimation

$$\hat{C}(k) = \frac{1}{N}\sum_{n=k-N+1}^{k} v_n v_n^T, \tag{9}$$

the value of b_k is in terms of the trace value of matrices M_k and N_k

$$b_k = \frac{tr\,[N_k]}{tr\,[M_k]}, \tag{10}$$

M_k and N_k are computed in equation (11)

$$\begin{aligned} N_k &= V_k - H(k)Q_kH(k)^T \\ M_k &= H(k)P(k)H(k)^T, \end{aligned} \tag{11}$$

the matrix V_k is

$$V_k = \begin{cases} v_0v_0^T & k = 0 \\ \frac{0.95V_{k-1}+v_kv_k^T}{1.95} & k \geq 1 \end{cases} \tag{12}$$

the value of b_k restricts the value of λ_k in the equation (13)

$$\lambda_k = \begin{cases} b_k, & \text{when } b_k \geq 1 \\ 1, & \text{when } b_k < 1 \end{cases} \tag{13}$$

2.2 *System stability analysis*

This section presents the stabilty analysis of the equivalent system $\hat{J}_v(k+1)$ provided by the STKF, we consider the next assumptions:

Assumption 1: The output is observable, i.e $H(k)\hat{J}_v(k) = \dfrac{\Delta\chi}{Ts}$ $\forall k > 0$.

Assumption 2: The initialization of the Jacobian vector $J_v(0) \in \mathbb{R}^a$, and covariance matrix are in a normal distribution $P(0) \in \mathbb{R}^{a\times a} \sim N(J_{v_0}, P_0)$.

Assumption 3: The covariance matrix $P(k) \in \mathbb{R}^{a\times a}$ is a positive definite matrix, and $P^{-1}(0) > 0$.

Using the estimation error in the equation (7) the cost function proposed is:

$$\xi(k) = \frac{1}{2}v_n(k)v_n^T(k), \tag{14}$$

at each time step updates the value of the $\hat{J}_v(k+1)$

$$\hat{J}_v(k+1) = \hat{J}_v(k) - K_F(k)\frac{\partial\xi}{\partial\hat{J}_v}, \tag{15}$$

the chain rule method calculates the term $\dfrac{\partial\xi}{\partial\hat{J}_v}$ by

$$\frac{\partial\xi}{\partial\hat{J}_v} = \frac{\partial\xi}{\partial v_n}\frac{\partial v_n}{\partial H\hat{J}_v}\frac{\partial H\hat{J}_v}{\partial\hat{J}_v} = v_n(k)\left[-1\right]H(k), \tag{16}$$

Copyright©AMCA. Todos los Derechos Reservados www.amca.mx

the equation (6) determines the updated vector $\hat{J}_v(k+1)$. It is considering the next Lyapunov's function

$$V(k+1) = \frac{1}{2} v_n(k+1) v_n^T(k+1), \tag{17}$$

the rate of change in the Lyapunov function is

$$\Delta V(k+1) = V(k+1) - V(k), \tag{18}$$

the change in the error estimation is

$$\Delta v_n(k+1) = v_n(k+1) - v_n(k), \tag{19}$$

the equation (18) becomes:

$$\Delta V(k+1) = \Delta v_n(k) \left[v_n(k) + \frac{1}{2} \Delta v_n(k) \right], \tag{20}$$

$\Delta v_n(k)$ is approximated in the next way

$$\Delta v_n(k) \approx \frac{\partial v_n}{\partial \hat{J}_v} \Delta \hat{J}_v, \tag{21}$$

where

$$\frac{\partial v_n}{\partial \hat{J}_v} = \frac{\partial v_n}{\partial H \hat{J}_v} \frac{\partial H \hat{J}_v}{\partial \hat{J}_v} = -H(k), \tag{22}$$

the equation (21) and equation (22) produces

$$\Delta v_n(k) \approx \frac{\partial v_n}{\partial \hat{J}_v} \Delta \hat{J} = -H(k) K_F(k) v_n(k), \tag{23}$$

the change in the Lyapunov's function is

$$\Delta V(k) = -H(k) K_F(k) v_n^2(k) \left[1 - \frac{1}{2} H(k) K_F(k) \right], \tag{24}$$

according with the stability condition $\Delta V < 0$

$$0 < K_F(k) < 2H^{-1}(k), \tag{25}$$

the Kalman filter gain $K_F(k)$ should lie in the range indicated in equation (25) to guarantee system stability.

2.3 Controller design

This section introduces the SMC with adaptive gains, and the control error defines

$$e(k) = \chi(k) - \chi_d(k) \in \mathbb{R}^m, \tag{26}$$

where $\chi(k)$ is the current end-effector position and $\chi_d(k)$ is the desired end-effector position. The sliding mode surface is

$$s(k) = \Delta e(k) + G(k) e(k) \in \mathbb{R}^m, \tag{27}$$

$G(k) \in \mathbb{R}^{m \times m}$ is a diagonal matrix that contains the adaptive gains for SMC: $G_x(k), G_y(k)$, and $G_z(k)$

the equation (28) uses a continuous smooth function for SMC

$$\nu(k) = -\tanh\left(\frac{s(k)}{\epsilon}\right) \in \mathbb{R}^m, \tag{28}$$

where $\epsilon > 0$, it is possible to calculate the signals of the controller using the pseudoinverse of $\hat{J}_A(k)$

$$\omega(k) = \hat{J}_A^+(k) \nu(k) \in \mathbb{R}^n, \tag{29}$$

and the updated joint position

$$q(k+1) = q(k) + \omega(k) \cdot Ts \in \mathbb{R}^n \tag{30}$$

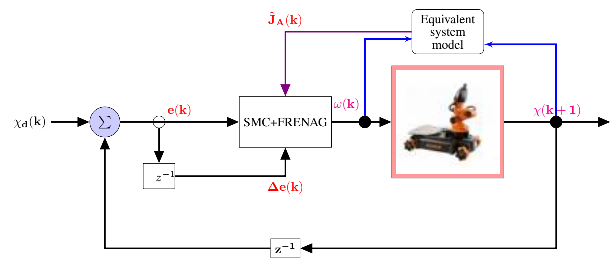

Fig. 1. Control block diagram of an equivalent system control.

The Fig. 1 depicts the control scheme proposed for this work. The Fuzzy Rules Emulated Network (FREN) adapts the gains matrix $G(k)$. The architecture of artificial network has 4 distinct layers in terms of: (1) input error $e(k)$; (2) linguistic variables $\mu_i(k)$; (3) linear consequence parameters β_i; and (4) adaptive gains $G(k)$. Fig. 2 shows the architecture of Fuzzy Rules Emulated Network Adaptive Gains (FRENAG), and the output gives the adaptive gains as:

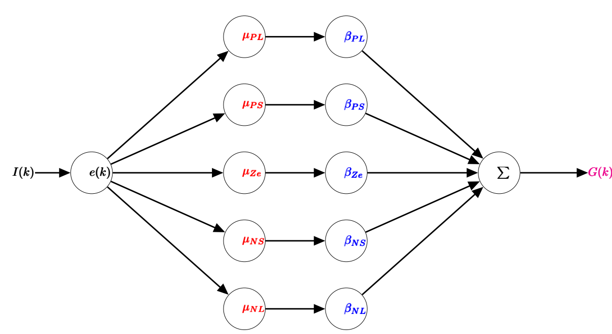

Fig. 2. Fuzzy rules emulad network architecture for adaptive gains.

$$G(k) = \sum_{i=1} \beta_i \mu_i(k) \tag{31}$$

The membership functions designs for adaptive gains are in the Fig. 3 and the linear consequence parameters are in table

Copyright©AMCA. Todos los Derechos Reservados www.amca.mx

1. Where the linguistic variables: PL is positive large, PS is positive small, Ze is zero, NS is negative small, and NL is negative large. The membership functions μ_i are designed in terms of the end-effector axes, and the linear consequence parameters βi (constant values) are tuned intuitively.

3. SIMULATION RESULTS

3.1 Robotic system

The base of study is the Kuka youBot mobile manipulator, which has 3 dof for omnidirectional mobile platform and 5 dof for manipulator arm ($n = 8$). The joint configuration space is the cartesian product $SE(2)\times\mathbb{T}^5$. The equivalent model $\hat{J}_A(k)$ includes the 8 dof of the robot, the omnidirectional platform and the robotic arm is considered the plant. The kinematic model of Kuka youBot generates the input and output data for simulations, but it is not included in the robot controller. In the simulations the estimated Jacobian is considered the real model for the system. The simulations takes the estimated Jacobian coming from the STKF to be considered in the control law in the equation (29). The Fig. 1 shows the diagram of the robot and controller approach.

3.2 Tracking control

The robot home position is $\chi(0) = [0.1430, 0, 0.6480, 0, 0, 0]$, and the desired trajectory is a circle giving for the next functions:

$$
\begin{aligned}
\chi_{xd}(k) =&0.5\sin(\frac{4\pi k}{kmax})\\
\chi_{yd}(k) =&0.5\cos(\frac{4\pi k}{kmax})\\
\chi_{zd}(k) =&0.55
\end{aligned}
\tag{32}
$$

The end-effector follows a circle with x and y and z remains in constant position, in the equation (32) $kmax$ is the maximum time index of the simulation. Fig. 4 shows the simulation results of end-effector circular trajectory, the evolution of the end-effector's axes, the convergence of the control errors, the gains adaptation, and the joint velocity as a control signal (mobile platform and robot arm), while the equivalent Jacobian system completes the control. The equivalent system fulfills the demands in the end-effector trajectory control by the proposed controller based on SMC and FRENAG.

4. CONCLUSION

We found that an equivalent system based on data driven control (STKF) can control the end-effector trajectory only with the knowledge between the input/output relationship. As well, we demonstrated the stability analysis of the Jacobian matrix estimation (equivalent system) by the STKF, and we proposed adaptive gains for a SMC based on neuro-fuzzy artificial network architecture (FREN). The control advantages of the equivalent system $\hat{J}_A(k)$ are: less robot's parameters knowledge, on-line data driven controller, and classical model independency. The simulations for tracking control shows the effectiveness of the Jacobian matrix estimation and control. As a future plan we extend the work to cover the proposed controller stability analysis, the robot's orientation control, and the validation in a real robot for some experimental setup.

ACKNOWLEDGEMENTS

The authors would like to thank CONACyT (Project number 257253) for the financial support through this work and Science Basic project Number: 285599) which is called Toma de decisiones multiobjetivo para sistemas altamente complejos. The first author thanks CONACyT for his PhD scholarship.

REFERENCES

Chiang, C.J. and Chen, Y.C. (2017). Neural network fuzzy sliding mode control of pneumatic muscle actuators. *Engineering Applications of Artificial Intelligence*, 65, 68–86.

Facundo, L., Gómez, J., Treesatayapun, C., Morales, A., and Baltazar, A. (2018). Adaptive control with sliding mode on a double fuzzy rule emulated network structure. *IFAC-PapersOnLine*, 51(13), 609–614.

Gómez, J., Treesatayapun, C., and Morales, A. (2018). Robot manipulation control using jacobian estimation. *Congreso Nacional de Control Automatico 2018*.

Hou, Z. and Jin, S. (2011). Data-driven model-free adaptive control for a class of mimo nonlinear discrete-time systems. *IEEE Transactions on Neural Networks*, 22(12), 2173–2188.

Li, M., Kang, R., Branson, D.T., and Dai, J.S. (2018). Model-free control for continuum robots based on an adaptive kalman filter. *IEEE/ASME Transactions on Mechatronics*, 23(1), 286–297.

Sharma, S., Kraetzschmar, G.K., Scheurer, C., and Bischoff, R. (2012). Unified closed form inverse kinematics for the kuka youbot. In *ROBOTIK 2012; 7th German Conference on Robotics*, 1–6. VDE.

Treesatayapun, C. and Uatrongjit, S. (2005). Adaptive controller with fuzzy rules emulated structure and its applications. *Engineering Applications of Artificial Intelligence*, 18(5), 603–615.

Treesatayapun, C. (2015). Data input-output adaptive controller based on if-then rules for a class of non-affine discrete-time systems: The robotic plant. *Journal of Intelligent & Fuzzy Systems*, 28(2), 661–668.

Zen, Z., Cao, R., and Hou, Z. (2018). Mimo model free adaptive control of two degree of freedom manipulator. In *2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*, 693–697. IEEE.

Copyright©AMCA. Todos los Derechos Reservados www.amca.mx

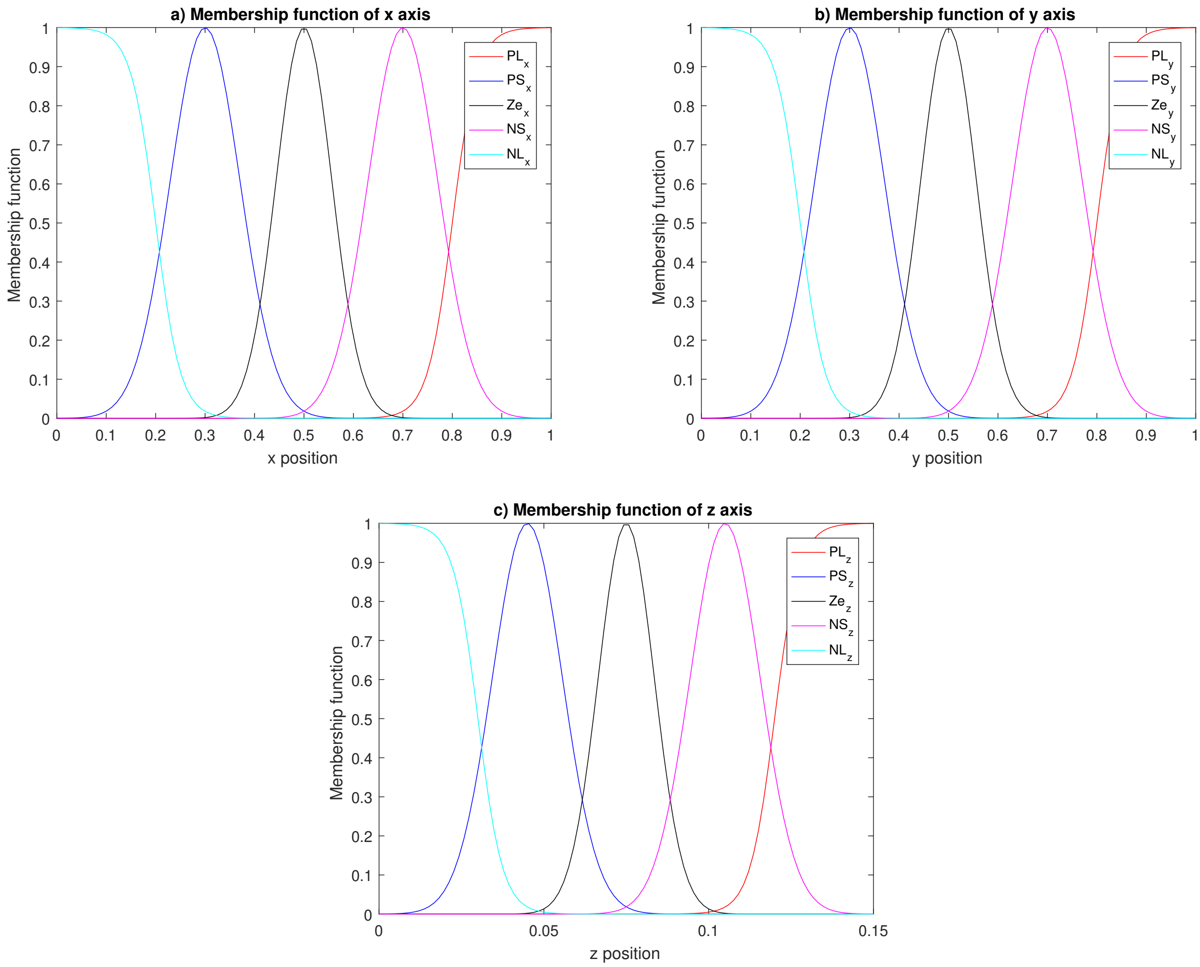

Fig. 3. Membership function for the end-effector tracking $\chi(k) = [x,\ y,\ z]$: a) membership functions of x axis, b) membership functions of y axis, and c) membership functions of z axis.

Table 1. Linear consequence parameters of adaptive gains parameters for $G_x(k)$, $G_y(k)$ and $G_z(k)$.

Name	Parameters	$G_x(k)$ values	$G_y(k)$ values	$G_z(k)$ values
Positive Large	β_{PL}	1	1.25	0.25
Positive Small	β_{NL}	0.85	1	0.35
Zero	β_{Ze}	0.85	0.5	0.75
Negative Small	β_{NS}	0.25	0.5	0.85
Negative Large	β_{NL}	0.15	0.25	1

 Copyright©AMCA. Todos los Derechos Reservados www.amca.mx

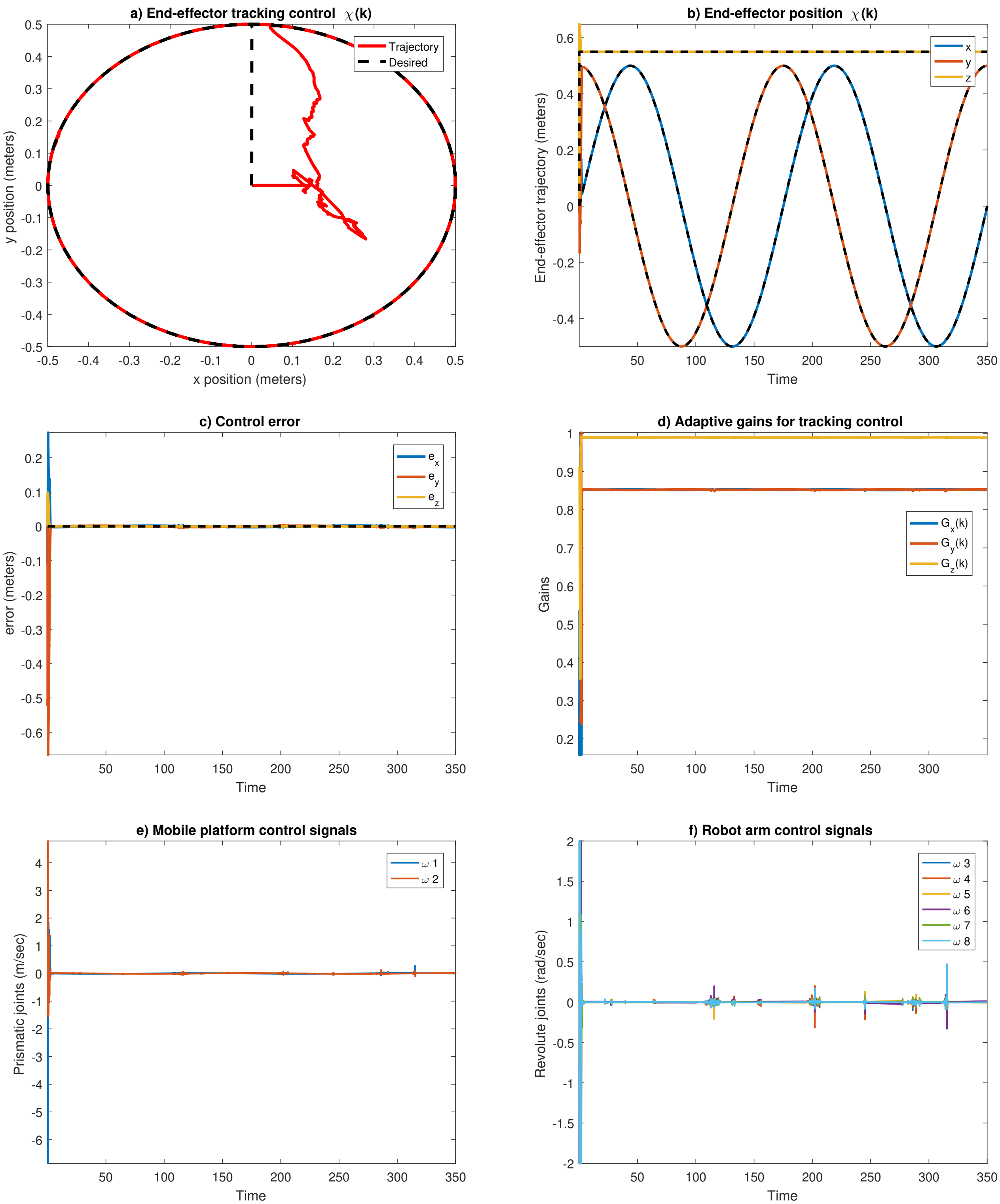

Fig. 4. Performance of end-effector tracking control: a) end-effector desired trajectory (cricle), b) end-effector desired functions, c) end-effector tracking error, d) mobile platform control signals, and e) robot arm control signals.

Copyright©AMCA. Todos los Derechos Reservados www.amca.mx