J Intell Robot Syst (2011) 62:81–102
DOI 10.1007/s10846-010-9438-3

Adaptive Polar-Space Motion Control for Embedded Omnidirectional Mobile Robots with Parameter Variations and Uncertainties

Hsu-Chih Huang · Ching-Chih Tsai · Shui-Chun Lin

Received: 18 August 2009 / Accepted: 25 May 2010 / Published online: 11 June 2010

Abstract This paper presents an adaptive polar-space motion controller for trajectory tracking and stabilization of a three-wheeled, embedded omnidirectional mobile robot with parameter variations and uncertainties caused by friction, slip and payloads. With the derived dynamic model in polar coordinates, an adaptive motion controller is synthesized via the adaptive backstepping approach. This proposed polar-space robust adaptive motion controller was implemented into an embedded processor using a field-programmable gate array (FPGA) chip. Furthermore, the embedded adaptive motion controller works with a reusable user IP (Intellectual Property) core library and an embedded real-time operating system (RTOS) in the same chip to steer the mobile robot to track the desired trajectory by using hardware/software co-design technique and SoPC (system-on-a-programmable-chip) technology. Simulation results are conducted to show the merit of the proposed polar-space control method in comparison with a conventional proportional-integral (PI) feedback controller and a non-adaptive polar-space kinematic controller. Finally, the effectiveness and performance of the proposed embedded adaptive motion controller are exemplified by conducting several experiments on steering an embedded omnidirectional mobile robot.

H.-C. Huang
Department of Computer Science and Information Engineering, Hungkuang University,
34 Chung-Chie Road, Shalu, Taichung, 433, Taiwan
e-mail: hchuang@sunrise.hk.edu.tw

C.-C. Tsai (✉)
Department of Electrical Engineering, National Chung Hsing University,
250, Kuo-Kuang Road, Taichung, 402, Taiwan
e-mail: cctsai@nchu.edu.tw

S.-C. Lin
Department of Electronic Engineering, National Chin-Yi University of Technology,
35, Lane 215, Sec. 1, Chung-Shan Road, Taiping, Taichung, 411, Taiwan
e-mail: lsc@ncut.edu.tw

Keywords Adaptive control · Embedded · Omnidirectional · Polar coordinates · Stabilization · Trajectory tracking

1 Introduction

Omnidirectional mobile robots have attracted much attention in both academia and industry in the field of robotics. Comparing with several car-like robots [1–5], this type of omnidirectional mobile mechanism has the superior agile capability to move towards any position and to simultaneously attain any desired orientation, namely that such a mobile mechanism does not have so-called nonholonomic constraints. Modeling and control of omnidirectional mobile robot in Cartesian coordinates have been investigated by several researchers. Pin et al. [6] presented the concepts for a family of holonomic wheeled robots. Watanabe et al. [7] proposed a PI feedback control method for an omnidirectional mobile robot which is equipped with three lateral orthogonal-wheel assemblies. Kalmár-Nagy et al. [8] offered the dynamic model and the time-optimal control for an omnidirectional robot. Williams II et al. [9] developed a dynamic model for omnidirectional wheeled mobile robots, considering the occurrence of slip between the wheels and motion surface. Huang and Tsai [10] discussed how to construct a FPGA-based adaptive controller for a kind of omnidirectional mobile robot with parameter variations and uncertainties caused by slip and abruptly changeable payloads. Overall, the aforementioned methods did not directly cope with polar-space motion control problems.

Polar space is especially useful in situations where the relationship between two points is most easily expressed in terms of angle and distance. There are many simple polar-space equations which describe complex curves, for example, the Archimedes' spiral, the rose curves and the Limacon of Pascal. The trajectory tracking problems for nonholonomic mobile robots in polar space have been investigated by several researchers. Park et al. [11] adopted the state-space exact feedback linearization method to achieve point stabilization of mobile robots. Yang and Kim [12] presented the sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots. The polar-space sliding-mode tracking controller to steer a nonholonomic wheeled mobile robot incorporating its dynamic effects and external disturbances was developed by Chwa [13]. To date, there have been few studies related to polar-space omnidirectional mobile robot control. For example, Huang and Tsai [14] proposed a kinematic control approach for both tracking and stabilization of an omnidirectional mobile robot in polar coordinates. Based on [14], an adaptive polar-space kinematic controller for autonomous omnidirectional mobile robot was introduced in [15]. However, the method in [15] cannot be applied to address both polar-space trajectory tracking and regulation problems of an omnidirectional mobile robot incorporating with dynamic effect, parameter variations and uncertainties.

Recently, the new generation of FPGA technology has enabled an embedded processor intellectual property (IP) and custom application IPs to be integrated into an SoPC developing environment. This new SoPC technology has been bringing a major revolution in the design of integrated circuits [16–19]. Since both software and hardware are integrated into a single programmable logic device, the designers can easily combine the flexibility of software unit and high performance of hardware

unit for building a system on a chip. Within hardware/software co-design using SoPC technology, the circuits needed fast processing but simple computations are suitable to be implemented by hardware in FPGA, and the highly sophisticated algorithms with heavy computations can be realized by software in FPGA. The SoPC technology has become increasingly important in implementing motion controllers or navigation controllers for autonomous mobile robots [18, 19]. In comparison with the fixed-processor DSP [20] which was shown to provide a feasible solution for developing a powerful controller in the filed of robotics, the SoPC technology is capable of not only achieving the same software functions running in its embedded processor, but also providing additional hardware IP implementation and an embedded RTOS for further purposes. This kind of SoPC implementation may offer identical functions to hardware-oriented FPGAs [21–23] because the same hardware units can be easily implemented in an SoPC. In addition, this SoPC-based implementation also provides a compromise approach between the special-purpose application specified integrated circuit (ASIC) hardware and general-purpose processors. With the merits of low cost, low power consumption, small circuit size, IP re-usability and reprogrammable hardware/software co-design, rapid prototyping, the SoPC technology has been proven as a powerful means to realize sophisticated but complicated signal processing algorithms, and high-performance but computation-intensive control laws in many kinds of autonomous mobile robots. For example, the SoPC technology has gained wide applications in designing soccer robots, small-scale humanoid robots, entertainment robots, education robots and so on [24–27].

The objectives of this paper are to apply adaptive backstepping method to construct an adaptive polar-space dynamic motion controller to achieve both stabilization and trajectory tracking for an autonomous omnidirectional mobile robot, and to implement such a motion controller in real time by employing the SoPC technology. Overall, the contributions of the paper are threefold.

1. A dynamic model of omnidirectional mobile robot in polar coordinates is derived by incorporating its dynamic effect and uncertainties caused by frictions, slip, and time-varying payloads.
2. A polar-space adaptive motion control law is presented based on backstepping techniques for regulation and trajectory tracking of the autonomous omnidirectional mobile robot. The proposed controller will be proven to be globally asymptotically stable via the Lyapunov stability theory.
3. An adaptive motion controller incorporating with an embedded processor and an embedded operating system is constructed. This embedded adaptive motion controller combining the hardware/software co-design and IP re-use concept takes the advantages of efficient implementation, excellent flexibility and satisfactory performance.

The rest of this paper is organized as follows. In Section 2, the polar-space dynamic model of the omnidirectional three-wheeled mobile robot is briefly described. Section 3 synthesizes an adaptive motion controller via the adaptive backstepping approach in polar coordinates. Section 4 shows how to implement the polar-space adaptive motion controller using the SoPC technology. Section 5 conducts several simulations and experiments that are used for illustration of the merits of the proposed controller. Section 6 concludes the paper.

2 Dynamic Model in Polar Coordinates

This section is devoted to briefly describing the polar-space dynamic model of an omnidirectional mobile robot with three independent omnidirectional wheels equally spaced at 120° from one to another. Figure 1 shows the structure and geometry of the omnidirectional driving configuration where θ denotes the vehicle orientation, and L denotes the distance from any omnidirectional wheel to the centre of the geometry of the robot. Due to structural symmetry of the robot, it is assumed that the centre of geometry coincides with the centre of mass.

Before deriving the polar-space dynamic model, one defines the pose of the robot as $[xy\ \theta]^T$. If no slips occur, the robot's kinematic equation in Cartesian space can be given in [8]:

$$V = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} R\omega_1 \\ R\omega_2 \\ R\omega_3 \end{bmatrix} = P(\theta) \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}, or \begin{bmatrix} \dot{x} \\ \dot{y} \\ \theta \end{bmatrix} = P^{-1}(\theta)\, V \tag{1}$$

where:

$$P(\theta) = \begin{bmatrix} -\sin\theta & \cos\theta & L \\ -\sin\left(\frac{\pi}{3} - \theta\right) & -\cos\left(\frac{\pi}{3} - \theta\right) & L \\ \sin\left(\frac{\pi}{3} + \theta\right) & -\cos\left(\frac{\pi}{3} + \theta\right) & L \end{bmatrix} \text{ and}$$

$$P^{-1}(\theta) = \begin{bmatrix} -\frac{2}{3}\sin\theta & -\frac{2}{3}\sin\left(\frac{\pi}{3} - \theta\right) & \frac{2}{3}\sin\left(\frac{\pi}{3} + \theta\right) \\ \frac{2}{3}\cos\theta & -\frac{2}{3}\cos\left(\frac{\pi}{3} - \theta\right) & -\frac{2}{3}\cos\left(\frac{\pi}{3} + \theta\right) \\ \frac{1}{3L} & \frac{1}{3L} & \frac{1}{3L} \end{bmatrix}$$

R denotes the radius of the omnidirectional wheels; V_i and ω_i, $i = 1,2,3$, respectively represent the linear and angular velocities of each wheel. Based on the kinematic

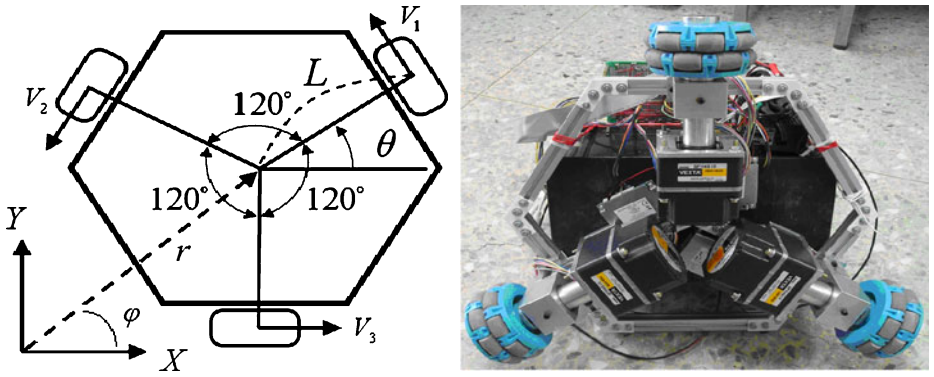

Fig. 1 Structure and geometry of the omnidirectional mobile platform in polar coordinates

model in Cartesian space, the polar-space kinematic model of the omnidirectional mobile robot is presented in [14]:

$$\begin{bmatrix} \dot{r} \\ \dot{\varphi} \\ \dot{\theta} \end{bmatrix} = T^{-1}(r, \theta - \varphi) \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix}, or \quad \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = T(r, \theta - \varphi) \begin{bmatrix} \dot{r} \\ \dot{\varphi} \\ \dot{\theta} \end{bmatrix} \tag{2}$$

where, as Fig. 1 shows, r denotes the polar radius, φ represents the polar angle, $T(r, 0-\varphi)$ and $T^{-1}(r, 0-\varphi)$ are respectively given by:

$$T^{-1}(r, \theta - \varphi) = \begin{bmatrix} -\frac{2}{3}\sin(\theta-\varphi) & -\frac{2}{3}\sin\left(\frac{\pi}{3} - (\theta-\varphi)\right) & \frac{2}{3}\sin\left(\frac{\pi}{3} + \theta - \varphi\right) \\ \frac{2}{3r}\cos(\theta-\varphi) & -\frac{2}{3r}\cos\left(\frac{\pi}{3} - (\theta-\varphi)\right) & -\frac{2}{3r}\cos\left(\frac{\pi}{3} + \theta - \varphi\right) \\ \frac{1}{3L} & \frac{1}{3L} & \frac{1}{3L} \end{bmatrix}$$

and

$$T(r, \theta - \varphi) = \begin{bmatrix} -\sin(\theta-\varphi) & r\cos(\theta-\varphi) & L \\ -\sin\left(\frac{\pi}{3} - (\theta-\varphi)\right) & -r\cos\left(\frac{\pi}{3} - (\theta-\varphi)\right) & L \\ \sin\left(\frac{\pi}{3} + (\theta-\varphi)\right) & -r\cos\left(\frac{\pi}{3} + (\theta-\varphi)\right) & L \end{bmatrix}$$

Note that the model 2 is undefined for zero polar radius; the model is valid for the condition $r > \varepsilon$ where ε is an arbitrarily small and positive real number.

In order to derive the dynamic model of the robot, one assumes that the robot has mass m, the moment of inertia J, and three uncertain but bounded forces exerted on the driving wheels, and neglects the servomotors' dynamics. The uncertain but bounded friction forces may come from several factors, such as static friction between the wheel and the surface, slip phenomena where the force may vary with the surface made by the used materials, and time-varying payloads. Note that the friction force exerted on wheel$i$is divided into two components: the first friction component F_{Wi} in the wheel rolling direction and the second friction component F_{Ti} in the transverse direction (normal to the first one) [9].

The total forces resulting from the force F_i from the$i$th servomotor motor and the friction force F_{Wi} exerted in the rotation direction of wheel i are given by:

$$F_i = \alpha U_i - \beta V_i - F_{Wi}, \quad i = 1, 2, 3. \tag{3}$$

where U_i is the applied voltage of each motor; F_{Wi} satisfies the inequality $-\frac{mg}{3}\mu_{W\max} \le F_{Wi} \le \frac{mg}{3}\mu_{W\max}$ where $\mu_{W\max}$ is the maximum static friction coefficient in the direction of wheel rotation and g is the acceleration of gravity. Note that $\alpha = \frac{k_t}{JR_a r_g}$, $\beta = \frac{k_e k_t + B}{JR_a}$, where R_a is the armature resistance of the servomotors; B is the viscous coefficient; r_g is the gear ratio; k_e denotes the back-emf coefficient; k_t represents the torque coefficient; J denotes the moment of inertia; B is the viscous coefficient. These parameters can be obtained from the motors' data sheets provided by the vender and the two parameters α and β are assumed constant but unknown.

With the force Eq. 3 and the friction force F_{Ti} exerted in the transverse direction of wheel i, one obtains a second-order dynamic model of the omnidirectional

mobile robot from Newton's second law for translation and rotation [9] in Cartesian coordinates:

$$M\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{\theta}\end{bmatrix} = \alpha P^T(\theta)\,U - B\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} - \bar{f} \tag{4}$$

where $M = diag\{m,m,J\}$, $B = diag\{1.5\beta,1.5\beta,3L2\beta\}$, $\bar{f} = [\bar{f}_1\ \bar{f}_2\ \bar{f}_3]^T$, $U = [U_1\ U_2\ U_3]^T$, $\bar{f}_1 = F_{W1}\sin\theta + F_{W2}\sin\left(\frac{\pi}{3}-\theta\right) - F_{W3}\sin\left(\frac{\pi}{3}+\theta\right) + F_{T1}\cos\theta - F_{T2}\cos\left(\frac{\pi}{3}-\theta\right) - F_{T3}\cos\left(\frac{\pi}{3}+\theta\right)$, $\bar{f}_2 = -F_{W1}\cos\theta + F_{W2}\cos\left(\frac{\pi}{3}-\theta\right) + F_{W3}\cos\left(\frac{\pi}{3}+\theta\right) + F_{T1}\sin\theta + F_{T2}\sin\left(\frac{\pi}{3}-\theta\right) - F_{T3}\sin\left(\frac{\pi}{3}+\theta\right)$. F_{Ti} satisfies the inequality $-\frac{mg}{3}\mu_{T\max} \le F_{Ti} \le \frac{mg}{3}\mu_{T\max}$, where $\mu_{T\max}$ is the maximum static friction coefficient for the transverse wheel direction; $\bar{f}_3 = L\sum_{i=1}^{3} F_{Wi}$, and the uncertain friction force vector satisfies the inequality, i.e., $\|\bar{f}\|_\infty \le k_{\max}$, where $\|\bar{f}\|_\infty$ denotes the infinity-norm of the vector $\bar{f}$ and $k_{\max}$ is the least upper bound of $\|\bar{f}\|_\infty$.

Next, Eq. 4 will be expressed in polar coordinates. With both relations $x = r\cos\varphi$ and $y = r\sin\varphi$, and the model 2, it is easy to obtain:

$$\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{\theta}\end{bmatrix} = \Phi\begin{bmatrix}\ddot{r}\\ \ddot{\varphi}\\ \ddot{\theta}\end{bmatrix} + \Gamma,\ \Phi = \begin{bmatrix}\cos\varphi & -r\sin\varphi & 0\\ \sin\varphi & r\cos\varphi & 0\\ 0 & 0 & 1\end{bmatrix},\ \Gamma = \begin{bmatrix}\Gamma_1\\ \Gamma_2\\ 0\end{bmatrix}, \tag{5}$$

where:

$$\begin{aligned}\Gamma_1 &= -2\dot{r}\dot{\varphi}\sin\varphi - r\dot{\varphi}^2\cos\varphi\\ &= -2\sin\varphi\left(-\frac{2}{3}V_1\sin(\theta-\varphi) - \frac{2}{3}V_2\sin\left(\frac{\pi}{3}-(\theta-\varphi)\right) + \frac{2}{3}V_3\sin\left(\frac{\pi}{3}+\theta-\varphi\right)\right.\\ &\qquad\left.\times\left(\frac{1}{r}\left[\frac{2}{3}V_1\cos(\theta-\varphi) - \frac{2}{3}V_2\cos\left(\frac{\pi}{3}-(\theta-\varphi)\right) - \frac{2}{3}V_3\cos\left(\frac{\pi}{3}+\theta-\varphi\right)\right]\right)\right.\\ &\quad\left.-\frac{1}{r}\left[\frac{2}{3}V_1\cos(\theta-\varphi) - \frac{2}{3}V_2\cos\left(\frac{\pi}{3}-(\theta-\varphi)\right) - \frac{2}{3}V_3\cos\left(\frac{\pi}{3}+\theta-\varphi\right)\right]^2\cos\varphi\end{aligned}$$

and

$$\begin{aligned}\Gamma_2 &= 2\dot{r}\dot{\varphi}\cos\varphi - r\dot{\varphi}^2\sin\varphi\\ &= 2\cos\varphi\left(-\frac{2}{3}V_1\sin(\theta-\varphi) - \frac{2}{3}V_2\sin\left(\frac{\pi}{3}-(\theta-\varphi)\right) + \frac{2}{3}V_3\sin\left(\frac{\pi}{3}+\theta-\varphi\right)\right.\\ &\qquad\left.\times\left(\frac{1}{r}\left[\frac{2}{3}V_1\cos(\theta-\varphi) - \frac{2}{3}V_2\cos\left(\frac{\pi}{3}-(\theta-\varphi)\right) - \frac{2}{3}V_3\cos\left(\frac{\pi}{3}+\theta-\varphi\right)\right]\right)\right.\\ &\quad\left.-\frac{1}{r}\left[\frac{2}{3}V_1\cos(\theta-\varphi) - \frac{2}{3}V_2\cos\left(\frac{\pi}{3}-(\theta-\varphi)\right) - \frac{2}{3}V_3\cos\left(\frac{\pi}{3}+\theta-\varphi\right)\right]^2\sin\varphi\end{aligned}$$

Note that the matrix Φ in Eq. 5 is invertible for any nonzero polar radius, $r > \varepsilon > 0$. Substituting Eqs. 1 and 4 into Eq. 5 gives:

$$M \begin{bmatrix} \ddot{r} \\ \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} = \alpha \Phi^{-1} P^T(\theta) U - \Phi^{-1} M \Gamma - \Phi^{-1} B P^1(\theta) V - \bar{g} \tag{6}$$

where $\bar{g} = \Phi^{-1} \bar{f}$ and $\|\bar{g}\|_\infty \leq c < \infty$, namely that $\bar{g}$ is bounded by the constant $c$for $r > \varepsilon > 0$. By defining the following two vectors $Z_1 = \begin{bmatrix} r & \varphi & \theta \end{bmatrix}^T$ and $Z_2 = \begin{bmatrix} \dot{r} & \dot{\varphi} & \dot{\theta} \end{bmatrix}^T$, one rewrites the dynamic model 6 in the following standard state space form:

$$\dot{Z}_1 = Z_2 \tag{7a}$$

$$M \dot{Z}_2 = \alpha \Phi^{-1} P^T(\theta) U - \Phi^{-1} M \Gamma - \Phi^{-1} B P^1(\theta) V - \bar{g} \tag{7b}$$

3 Adaptive Dynamic Motion Controller Design

This section synthesizes a polar-space adaptive motion controller for the robot's dynamic models 7a and 7b with two unknown but constant parameters, m and J, and three uncertain but bounded friction forces exerted on the driving wheels. This controller aims at steering the robot to reach the destination pose or exactly follow desired differentiable trajectory described by $Z_d = \begin{bmatrix} r_d & \varphi_d & \theta_d \end{bmatrix}^T$. In what follows, the well-known adaptive backstepping technique [28] is employed to synthesize an adaptive controller for regulation and trajectory tracking of the robot's dynamic models 7a and 7b with two unknown but constant parameters, m and J, and with the uncertainty $\bar{g}$. In doing so, one defines the tracking error vector by $Z_e = Z_1 - Z_d$. Differentiating Z_e and Z_e yields:

$$\dot{Z}_e = \dot{Z}_1 - \dot{Z}_d = Z_2 - \dot{Z}_d \tag{8}$$

Considering Z_2 as a virtual control which is designed as $Z_2 = \phi(Z_e) = -K_1 Z_e + \dot{Z}_d$, one obtains $\dot{Z}_e = -K_1 Z_e + \dot{Z}_d - \dot{Z}_d = -K_1 Z_e$ where the matrix K_1 is diagonal and positive definite. The asymptotic stability of the Z_e dynamics can be shown via selecting the quadratic Lyapunov function $V_1 = \frac{1}{2} Z_e^T M K_1^2 Z_e$, thus resulting in:

$$\dot{V}_1 = Z_e^T - M K_1^2 \dot{Z}_e = \dot{Z}_e^T M K_1^2 (-K_1 Z_e) = -Z_e^T M K_1^3 Z_e < 0 \tag{9}$$

To achieve the controller design, the following backstepping error vector is defined by:

$$\eta = Z_2 - \phi(Z_e) = Z_2 + K_1 Z_e - \dot{Z}_d \tag{10}$$

From the definition 8, it follows that:

$$\dot{Z}_e = Z_2 - \dot{Z}_d = \left(Z_2 + K_1 Z_e - \dot{Z}_d\right) - K_1 Z_e = \eta - K_1 Z_e \tag{11}$$

Taking the time derivative of the error η and premultiplying η with the matrix M give:

$$\begin{aligned} M\dot{\eta} &= M\dot{Z}_2 + MK_1\dot{Z}_e - M\ddot{Z}_d = \alpha\Phi^{-1}P^T(\theta)U - \Phi^{-1}M\Gamma \\ &\quad -\Phi^{-1}BP^{-1}(\theta)V - \bar{g} + MK_1\dot{Z}_e - M\ddot{Z}_d \\ &= \alpha\Phi^{-1}P^T(\theta)U - \Phi^{-1}M\Gamma - \Phi^{-1}BP^{-1}(\theta)V - \bar{g} + MK_1\eta - MK_1^2 Z_e - M\ddot{Z}_d \end{aligned} \tag{12}$$

Clearly, the control goal is to design an adaptive control law for U such that $Z_e \to 0$ and $\eta \to 0$ as $t \to \infty$. This can be easily done by choosing the subsequent adaptive control law:

$$U = \left(P^T(\theta)\right)^{-1}\alpha^{-1}\left(\hat{M}\Gamma + BP^{-1}(\theta)V + \Phi\left[\hat{M}\ddot{Z}_d - \left(K_2 + \hat{M}K_1\right)\eta - \hat{c}\,\mathrm{sgn}(\eta)\right]\right) \tag{13}$$

where $\hat{M} = diag\left\{\hat{m}, \hat{m}, \hat{J}\right\}$; $K_1 = diag\{k_{11}, k_{12}, k_{13}\}$; the matrix K_2 is symmetric and positive-definite and the control gain $\hat{c}$ is real and positive. The parameter adjustment rules for $\hat{m}$, $\hat{J}$ and $\hat{c}$ are to be determined in the following. Substituting Eq. 13 into Eq. 12 gives:

$$\begin{aligned} M\dot{\eta} &= \alpha P^T(\theta)\left(P^T(\theta)\right)^{-1}\alpha^{-1}\left(\hat{M}\Gamma + BP^{-1}(\theta)V + \Phi[\hat{M}\ddot{Z}_d - (K_2 + \hat{M}K_1)\eta - \hat{c}\,\mathrm{sgn}(\eta)]\right) \\ &\quad -\Phi^{-1}M\Gamma - \Phi^{-1}P^{-1}(\theta)V - \bar{g} + MK_1\eta - MK_e^2 Z_e - M\ddot{Z}_d \\ &= -\Phi^{-1}\tilde{M}\Gamma - \tilde{M}\ddot{Z}_d + \tilde{M}K_1\eta - K_2\eta - MK_1^2 Z_e - \hat{c}\,\mathrm{sgn}(\eta) - \bar{g} \end{aligned} \tag{14}$$

where $\tilde{M} = diag\left\{m - \hat{m}, m - \hat{m}, J - \hat{J}\right\}$. The closed-loop stability of the feedback error system and the parameter adjustment rules for $\hat{m}$, $\hat{J}$ and $\hat{c}$ can be simultaneously accomplished by the Lyapunov stability theory. For the goal, one chooses the radial and unbounded Lyapunov function:

$$V_2 = \frac{1}{2}Z_e^T MK_1^2 Z_e + \frac{1}{2}\eta^T M\eta + \frac{1}{2\lambda_m}\tilde{m}^2 + \frac{1}{2\lambda_J}\tilde{J}^2\frac{1}{2\lambda_c}\tilde{c}^2, \lambda_m > 0, \lambda_J > 0, \lambda_c > 0 \tag{15}$$

where $\tilde{c} = c - \hat{c}$, $\tilde{J} = J - \hat{J}$ and $\tilde{m} = m - \hat{m}$. Taking the time derivative of V_2 using the inequality $\eta^T \bar{g} \leq \|\eta\|_1 \|\bar{g}\|_\infty \leq \|\eta\|_1 c$ yields:

$$\begin{aligned}
\dot{V}_2 &= Z_e^T \left(MK_1^2\right) \dot{Z}_e + \eta^T M \dot{\eta} + \frac{1}{\lambda_m}\left(-\tilde{m}\dot{\hat{m}}\right) + \frac{1}{\lambda_J}\left(-\tilde{J}\dot{\hat{J}}\right) + \frac{1}{\lambda_c}\left(-\tilde{c}\dot{\hat{c}}\right) \\
&= -Z_e^T MK_1^3 Z_e + \eta^T \left(-\Phi^{-1}\tilde{M}\Gamma - \tilde{M}\ddot{Z}_d + \tilde{M}K_1\eta - K_2\eta - MK_1^2 Z_e - \hat{c}\,\mathrm{sgn}(\eta) - \bar{g}\right) \\
&\quad + \frac{1}{\lambda_m}\left(-\tilde{m}\dot{\hat{m}}\right) + \frac{1}{\lambda_J}\left(-\tilde{J}\dot{\hat{J}}\right) + \frac{1}{\lambda_c}\left(-\tilde{c}\dot{\hat{c}}\right) \\
&= -Z_e^T MK_1^3 Z_e - \eta^T \Phi^{-1}\tilde{M}\Gamma - \eta^T \tilde{M}\ddot{Z}_d - \eta^T \tilde{M}K_1\eta - \eta^T K_2\eta - \eta^T MK_1^2 Z_e \\
&\quad - \eta^T \hat{c}\,\mathrm{sgn}(\eta) - \eta^T \bar{g} + \frac{1}{\lambda_m}\left(-\tilde{m}\dot{\hat{m}}\right) + \frac{1}{\lambda_J}\left(-\tilde{J}\dot{\hat{J}}\right) + \frac{1}{\lambda_c}\left(-\tilde{c}\dot{\hat{c}}\right) \\
&\leq -Z_e^T MK_1^3 Z_e - \eta^T K_2\eta + \tilde{m}\left(-\frac{1}{\lambda_m}\dot{\hat{m}} - (\eta_1\ddot{r}_d + \eta_2\ddot{\varphi}_d) + \left(k_{11}\eta_1^2 + k_{12}\eta_2^2\right) - W\right) \\
&\quad + \tilde{J}\left(-\frac{1}{\lambda_J}\dot{\hat{J}} - \eta_3\ddot{\theta}_d + k_{13}\eta_3^2\right) + \tilde{c}\left(\|\eta\|_1 - \frac{1}{\lambda_c}\dot{\hat{c}}\right) \qquad (16)
\end{aligned}$$

where $W = \Gamma_1(\eta_1 \cos\varphi + \eta_2 \sin\varphi) + \Gamma_2(\eta_1 \sin\varphi + \eta_2 r \cos\varphi)$.

If the following parameter update laws 17, 18 and 19 for $\hat{m}$, $\hat{J}$ and $\hat{c}$ are chosen by:

$$\dot{\hat{m}} = -\lambda_m(\eta_1\ddot{r}_d + \eta_2\ddot{\varphi}_d) + \lambda_m\left(k_{11}\eta_1^2 + k_{12}\eta_2^2\right) - \lambda_m W \qquad (17)$$

$$\dot{\hat{J}} = \lambda_J\left(-\eta_3\ddot{\theta}_d + k_{13}\eta_3^2\right) \qquad (18)$$

$$\dot{\hat{c}} = \lambda_c\|\eta\|_1 \qquad (19)$$

then one obtains:

$$\dot{V}_2 \leq -Z_e^T MK_1^3 Z_e - \eta^T K_2\eta \leq 0 \qquad (20)$$

which shows that $\dot{V}_2$ is negative semidefinite. The use of La Salle invariance principle indicates that $Z_e \to 0$ and $\eta \to 0$ as time tends to infinity, and the estimates $\hat{m}$, $\hat{J}$ and $\hat{c}$ are globally uniformly bounded. Hence, the globally asymptotical stability of the closed-loop error system is ensured. This main result is summarized as below.

Theorem 1 *Consider the robot's dynamic models 7a and 7b with the desired differentiable trajectory* $Z_d = \left[r_d\ \varphi_d\ \theta_d\right]^T \in C^2, r_d \neq 0$, *two unknown but constant parameters m and J, and three uncertain but bounded forces exerted on the driving wheels. If the adaptive control 13 along with the parameter adjustment rules 17, 18, and 19 is applied, then the robot can be steered to achieve trajectory tracking in the sense of globally asymptotical stability, i.e.,* $Z_1 \to Z_d$ *and* $Z_2 \to \dot{Z}_d$ *as* $t \to \infty$.

Remark 1 The adaptive dynamic control 13 will be reduced to the set-point stabilization law if the reference trajectory is independent of time, i.e., a point. This fact indicates that the control 13 can be though of as a unified control law to achieve both set-point stabilization and trajectory tracking for the dynamic models 7a and 7b.

Remark 2 The proposed polar-space adaptive controller 13 can be straightforward extended to address the path following problem of this kind of omnidirectional mobile robot [15].

Remark 3 Given the positive time constant τ, if the parameter adjustment rules 17, 18, and 19 for $\hat{m}$, $\hat{J}$ and $\hat{c}$ are modified based on the e-modification given by:

$$\dot{\hat{m}} = -\lambda_m \left(\eta_1 \ddot{r}_d + \eta_2 \ddot{\varphi}_d\right) + \lambda_m \left(k_{11}\eta_1^2 + k_{12}\eta_2^2\right) - \lambda_m W - \lambda_m \tau \left\|\eta\right\|_2 \hat{m} \tag{21}$$

$$\dot{\hat{J}} = \lambda_J \left(-\eta_3 \ddot{\theta}_d + k_{13}\eta_3^2\right) - \lambda_J \tau \left\|\eta\right\|_2 \hat{J} \tag{22}$$

$$\dot{\hat{c}} = \lambda_c \left\|\eta\right\|_1 - \lambda_c \tau \left\|\eta\right\|_2 \hat{c} \tag{23}$$

then $\dot{V}_3$ becomes:

$$\begin{aligned}\dot{V}_3 &\leq -Z_e^T M K_P^3 Z_e - \eta^T K \eta + \tau \left\|\eta\right\|_2 \left(\tilde{m}\hat{m} + \tilde{J}\hat{J} + \tilde{c}\hat{c}\right) \\ &\leq -Z_e^T M K_P^3 Z_e - \bar{k}_{\min} \left\|\eta\right\|_2^2 + \tau \left\|\eta\right\|_2 \left((m - \tilde{m})\,\tilde{m} + \left(J - \tilde{J}\right)\tilde{J} + (c - \tilde{c})\,\tilde{c}\right)\end{aligned}$$

where $\bar{k}_{\min}$ denotes the minimum positive eigenvalue of the diagonal matrix K. Hence, $\dot{V}_3$ is negative semidefinite outside the compact set $\left\{\eta \middle| \left\|\eta\right\|_2 < \tau(m^2 + J^2 + c^2)/4\bar{k}_{\min}\right\}$; this reveals that the tracking errors Z_e and η are uniformly ultimate bounded (UUB) and the estimates $\hat{m}$, $\hat{J}$ and $\hat{c}$ are also uniformly ultimate bounded (UUB). These results indicate that the proposed adaptive control law 13 with the parameter adaptation rules 21, 22, and 23 is capable of steering the platform with the dynamic model 6 to reach any destination pose or follow any differentiable and time-varying trajectory in the sense of uniformly ultimately bounded (UUB) stability.

4 Embedded Realization

This subsection is devoted to using the SoPC technology to implement the previous adaptive polar-space motion control law 13 for omnidirectional mobile robots. Figure 2 depicts the architecture of the Altera FPGA implementation for the proposed polar-space adaptive mobile robot controller, including hardware circuits (IPs) and software code in one chip. This hardware/software co-design strategy is more powerful and efficient than conventional hardware-only or software-only designs do. The Avalon memory-mapped (Avalon-MM) interface is an address-based read/write interface typical of reading/writing master–slave connections and is used for reading/writing interfaces on master and slave components in a memory-mapped system. As shown in Fig. 2, these components include microprocessors (Nios II), memories, UARTs, custom logic (user IP) and timers. Note that these components are connected by a system interconnect fabric.

The polar-space adaptive control law 13 for the mobile robot has been implemented into the 32-bit Nios II processor whose numerical precision and computation

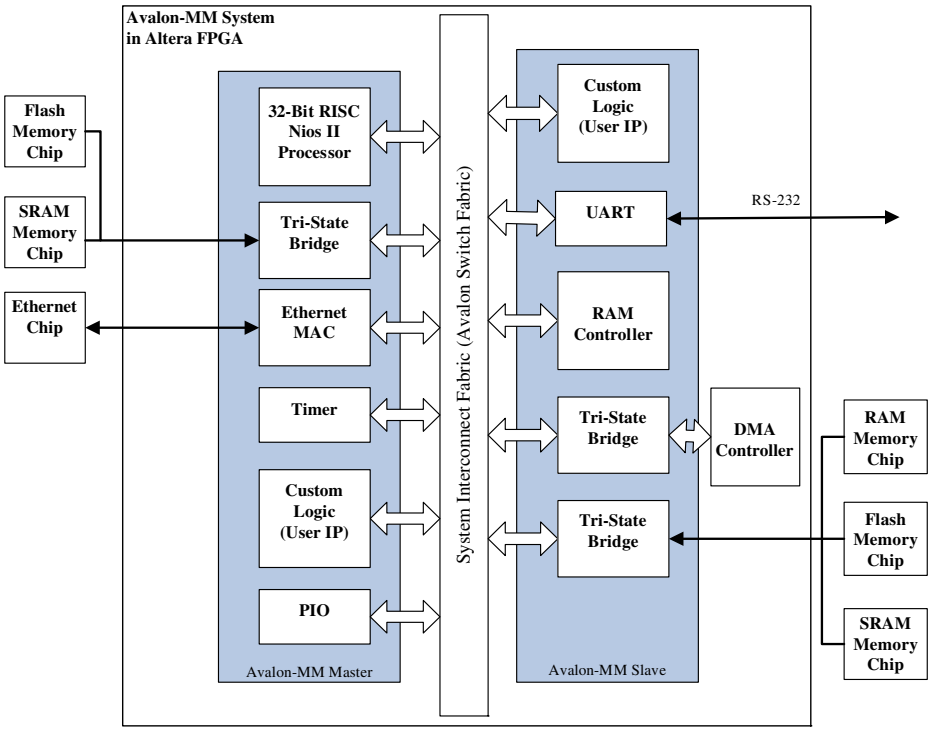

Fig. 2 FPGA implementation of the polar-space adaptive mobile platform controller

speed are high enough to realize the adaptive control law. The user IP cores (hardware circuits or custom logic) for this robotics application have been developed by VHDL (VHSIC Hardware Description Language). The software-based adaptive controller and hardware-based custom logic are connected to the system interconnect fabric via Avalon-MM for achieving the adaptive motion controller in one FPGA chip.

Figure 3 presents the embedded system design of the mobile robot in FPGA, including hardware circuits (user IPs) and system IPs. As shown in Fig. 3, the polar-space adaptive motion control law 13 has been efficiently implemented by software executed by a 32-bit Nios II processor in FPGA chip. The 12-bit D/A converter, AD7541, is employed to convert the output commands into analog voltage signals for driving three DC brushless motors mounted on the three omnidirectional wheels. The three quadrature-encoder-pulse (QEP) signals generated from the three motors are fed back to the embedded motion controller. With the QEP signals, the real position and orientation of the mobile robot can be dead-reckoned by the embedded soft-core CPU Nios II. The QEP decoder hardware circuit was implemented by VHDL, and the real-time OS MicroC/OS-II was ported into the FPGA chip to handle the data communication with PC via TCP/IP protocol. Moreover, the embedded soft-core Nios II processor works with the lwIP (lightweight IP) for the Ethernet connectivity, thereby significantly reducing resource usage. The FPGA chip is Altera Stratix EP1S10F780C6 with 10,570 LEs (Logic Element), 426 user I/O pins, 6 DSP

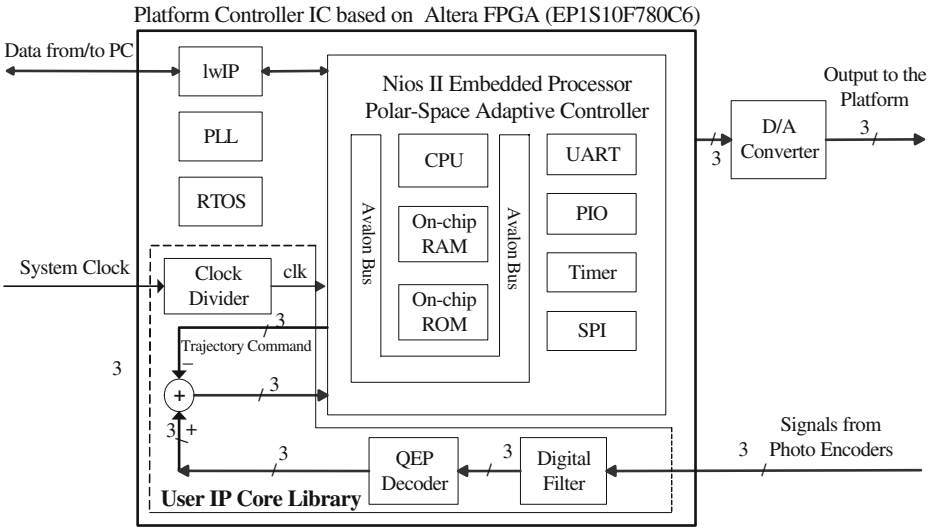

Fig. 3 Embedded polar-space adaptive controller of the omnidirectional mobile platform in FPGA

blocks, 920,448 RAM bits memory, 6 PLLs (Phase-Lock Loop) and an embedded Nios II 32-bit RISC (Reduced Instruction Set Computer) processor. The resource usage of the proposed adaptive motion controller IC is 10,042 LEs (95% of total LEs), 736,358 memory bits (80% of total RAM bits). With the hardware/software co-design and SoPC technology, the adaptive motion controller takes the advantage of software flexibility for complicated algorithm with low sampling frequency in motion control (<1 kHz), and high sampling frequency required in hardware IP (>1 MHz) [29].

5 Simulations, Experimental Results and Discussion

In this section, four simulations and three experiments are conducted to illustrate the feasibility, performance and merit of the embedded adaptive polar-space dynamic motion controller. The second and fourth simulations are respectively performed to compare tracking performance with the non-adaptive controller in [14] and feedback controller in [7]. All experiments are used for illustration of effectiveness of the proposed adaptive polar-space controller for an embedded omnidirectional mobile robot.

5.1 Simulation Results and Discussion

The aims of the simulations are to examine the effectiveness and performance of the proposed adaptive motion control law 13 to the omnidirectional mobile robot incorporated with the dynamic effect and friction. These simulations are done with the following parameters: $L = 0.23$ m, $R = 0.0508$ m, $m = 35$ kg, $J = 0.93$ kg m^2, $\alpha = 0.757$, $\beta = 25.74$, $c = 1$, $K_2 = diag\{30,30,30\}$ and $K_1 = diag\{1,1,1\}$. All the friction forces, F_{Wi} and F_{Ti}, $i = 1,2,3$, are assumed to be 5 N.

5.1.1 Adaptive Limacon of Pascal Trajectory Tracking

This subsection aims to conduct two simulations to investigate the tracking performance of the adaptive motion controller 13. In the first simulation, the robot gets started at $\left[r_0 \ \varphi_0 \ \theta_0 \right] = \left[1\text{m} \ 0\text{rad} \ 0\text{rad} \right]$, and then is steered to move along the special trajectory, called Limacon of Pascal described by $r_d(t) = 2 + 1.5\cos\varphi_d(t)$(unit: m),

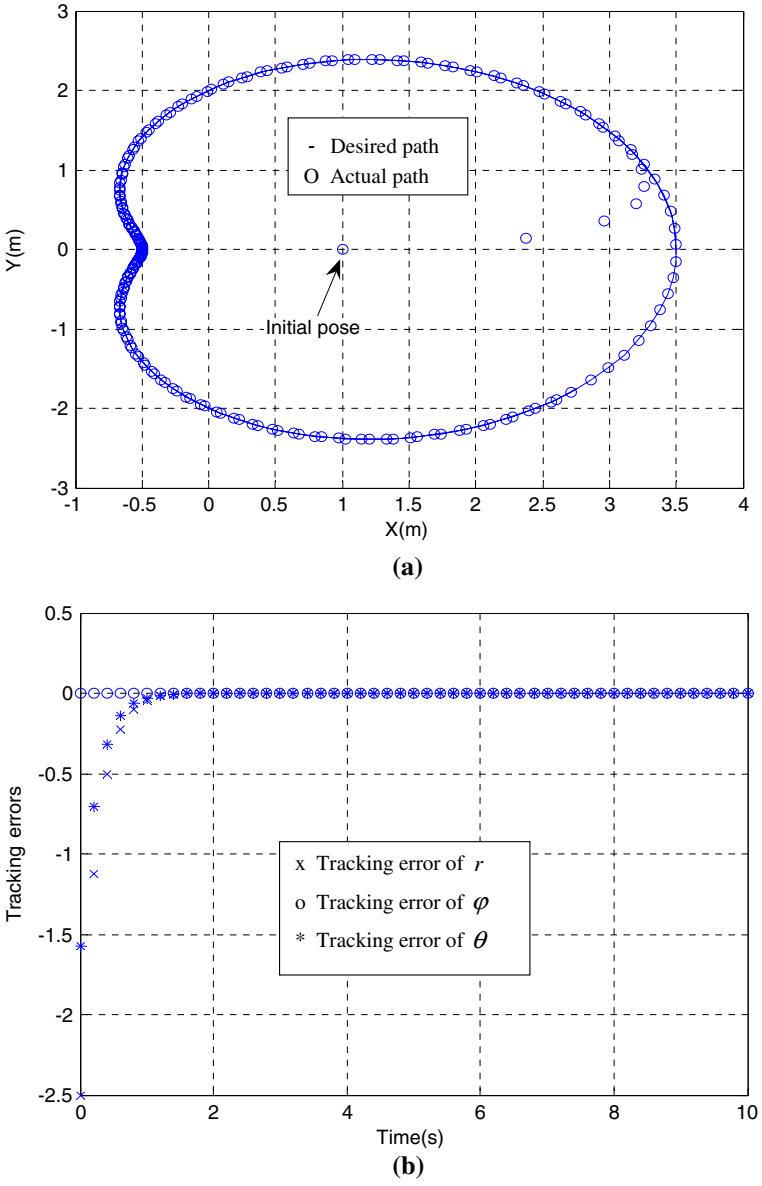

Fig. 4 **a** Simulation result of the Licamon of Pascal trajectory tracking. **b** Tracking errors in respect to time

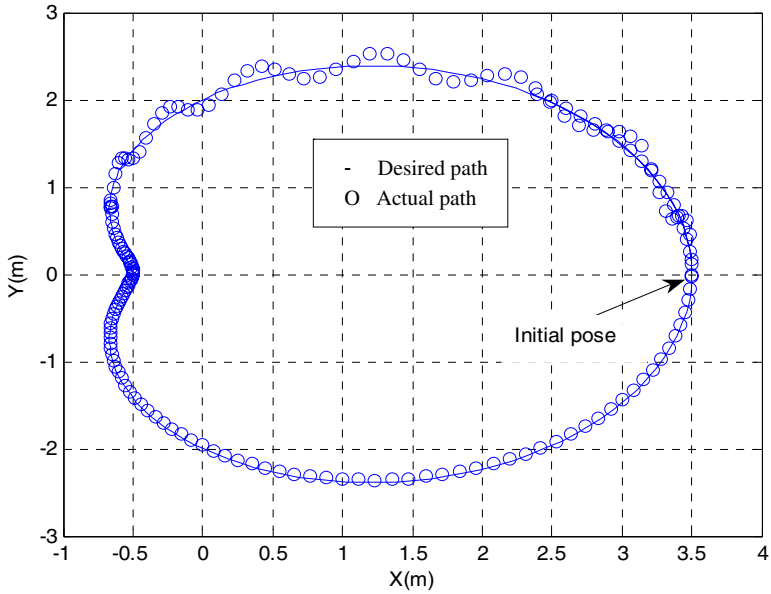

Fig. 5 Simulation result of the Licamon of Pascal trajectory tracking using the non-adaptive controller in [14]: the parameter m is altered to $m = 41$ kg in the time interval $0 \leq t \leq 15$ s

$\varphi_d(t) = 0.2t$(unit: rad), and $\theta_d(t) = \pi/2$ (unit: rad). To make the motion controller exhibit its adaptive performance, the mass parameter m is changed to $m = 41$ kg during the time duration $0 \leq t \leq 15$ s. Figure 4 presents the simulated trajectory of the controlled robot to track the Limacon of Pascal trajectory in the whole time interval $0 \leq t \leq 40$ s. As can be seen in Fig. 4, the simulation result indicates that the proposed motion controller is capable of steering the mobile robot to track the desired trajectories with the described uncertainties.

The second simulation was conducted to compare the proposed adaptive motion controller 13 with the non-adaptive polar-space kinematic controller in [14]. Figure 5 shows the tracking performance of the non-adaptive controller for the parameter variations identical to Fig. 4. Comparing Fig. 5 with Fig. 4, the adaptive motion controller has a more smooth response to track the Licamon of Pascal trajectory than the non-adaptive one does, especially in the time interval $0 \leq t \leq 15$ s in Fig. 5. The results clearly indicate that the proposed adaptive motion controller 13 outperforms the non-adaptive one [14].

5.1.2 Adaptive Rose Curve Trajectory Tracking

This simulation is devoted to investigating the performance of the adaptive rose curve trajectory tracking. The rose curve trajectory is expressed by $r_d(t) = 5\cos\left(\frac{1}{3}\varphi_d(t)\right) + 6$(unit: m), $\varphi_d(t) = 0.2t$(unit: rad), and $\theta_d(t) = \pi/2$ (unit: rad). In the simulation, the initial starting pose is given by $\left[r_0\ \varphi_0\ \theta_0\right] = \left[11\text{m}\ 0\text{rad}\ 0\text{rad}\right]$ in the whole time interval $0 \leq t \leq 100\,s$. Figure 6 shows the simulation result of the trajectory tracking using the proposed polar-space adaptive motion controller. Note that the parameter m is changed to $m = 41$ kg during the time duration

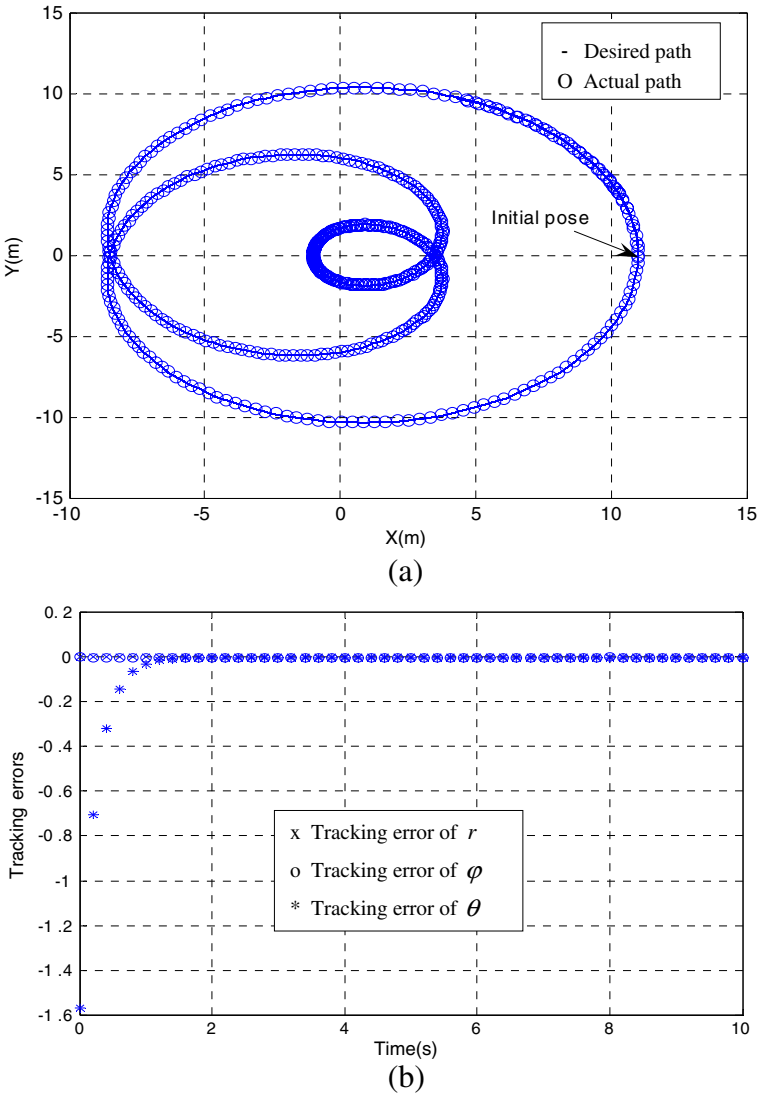

Fig. 6 **a** Trajectory tracking result of the rose curve using the polar-space control law 13. **b** Tracking errors in respect to time

$0 \leq t \leq 15\,s$. The result shows that the proposed motion controller steers the robot to achieve trajectory tracking for the rose curve. The fourth simulation was conducted to compare the proposed adaptive motion controller 13 with the Cartesian-space PI feedback controller in [7]. Figure 7 depicts the tracking performance of the feedback controller for the parameter variations identical to Fig. 6. In comparison with Fig. 6, the adaptive polar-space motion controller has a more smooth response to track the rose curve. These results also show that the proposed polar-space adaptive controller outperforms the Cartesian-space PI controller in tracking this special trajectory.

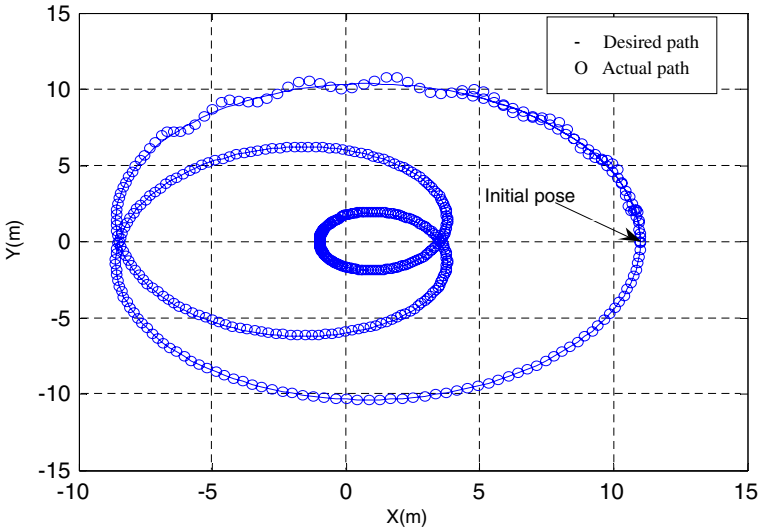

Fig. 7 Trajectory tracking result of the rose curve using Cartesian-space PI feedback controller in [7]

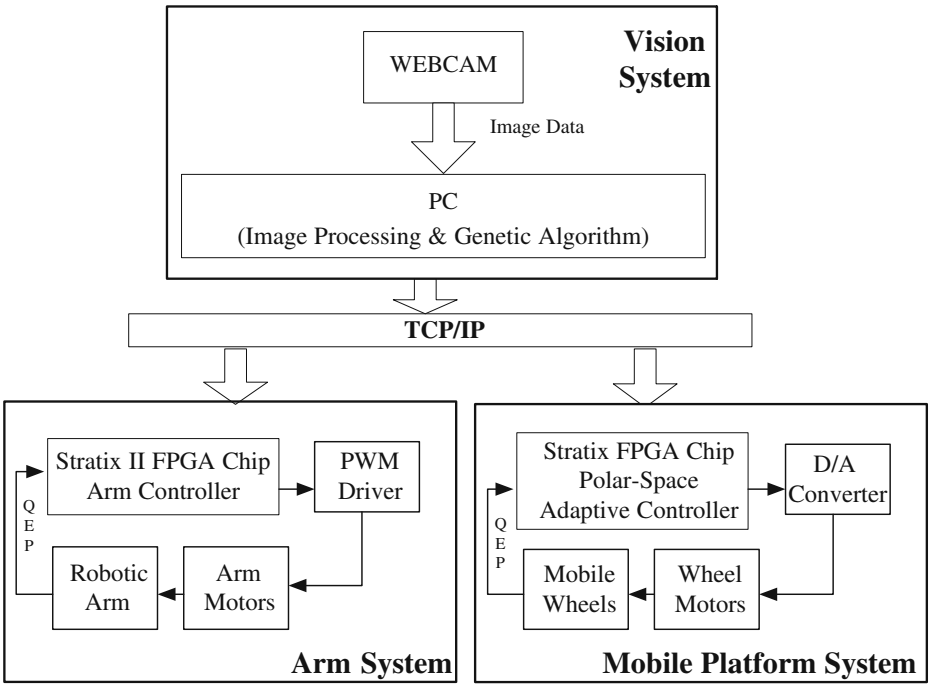

Fig. 8 Block diagram of the experimental omnidirectional mobile service robot

5.2 Experimental Results and Discussion

5.2.1 System Architecture of the Experimental Omnidirectional Mobile Service Robot

The aim of the following three experiments is to examine the effectiveness and performance of the proposed embedded polar-space adaptive motion control method by constructing an experimental mobile service robot incorporating with the autonomous mobile robot and an on-board manipulator. Figure 8 shows the block diagram of the experimental omnidirectional mobile service robot which has three subsystems: PC-based vision system, SoPC-based embedded one-arm control system and SoPC-based embedded mobile robot control system. To connect these three subsystems, the experimental setup constructed the networked embedded system using the embedded processor, RTOS and lwIP.

The vision system is composed of CCD cameras and one compact PC to execute image processing algorithm to determine the true position of the desired object. The position information is delivered into genetic algorithm (GA) for resolving redundant problem to find the optimal configuration of the mobile robot and the manipulator. The optimal configuration information for the whole service robot is sent to the two embedded systems to control the mobile robot and the manipulator via standard TCP/IP protocol. The two embedded systems aim to perform motion control of the servo motors. All the hardware and software design of the omnidirectional mobile robot are integrated into the FPGA chips.

The experimental SoPC-based omnidirectional mobile service robot is shown in Fig. 9. The proposed unified motion control law 13 was implemented using

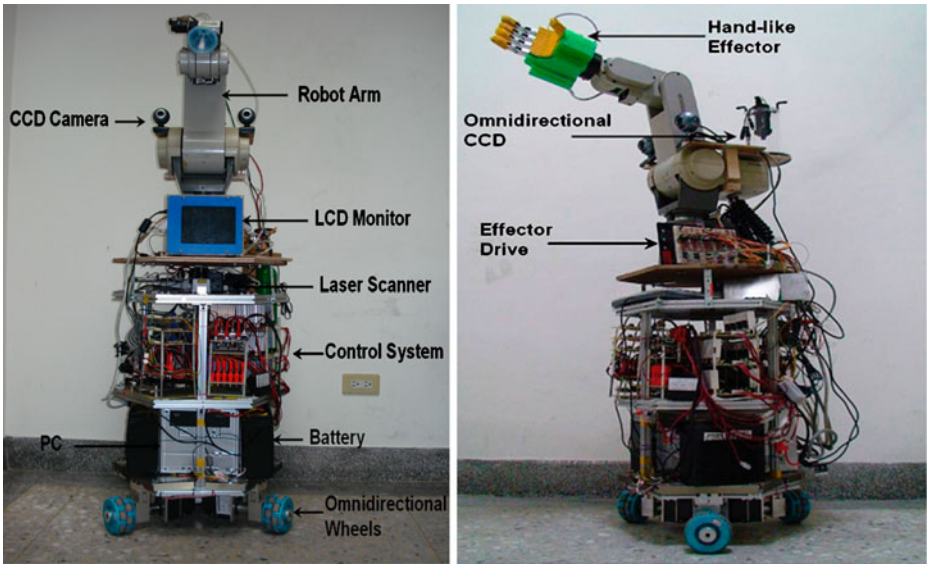

Fig. 9 Picture of the experimental omnidirectional mobile service robot

C/C++ code and standard programming techniques in the Altera Nios II embedded processor. The FPGA chip integrated the embedded processor, RTOS, lwIP, and VHDL-based IP circuits to perform the adaptive motion control law of the mobile robot. All the experiments were conducted with the system parameters: $L = 23$ cm

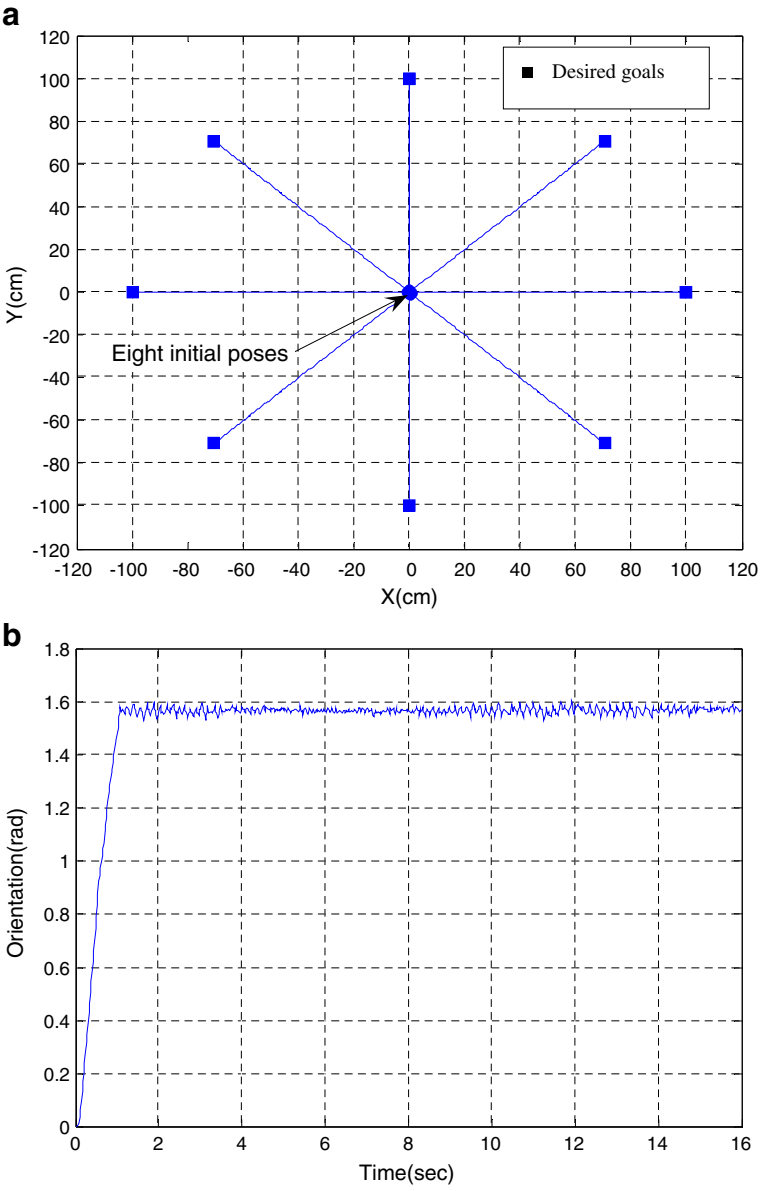

Fig. 10 **a** Experimental results of the proposed controller for point stabilization. **b** Time response of the vehicle orientation in the case of $\alpha = \pi/4$

and $R = 5.08$ cm. Moreover, in the experiment, the three encoders were employed to measure the angular velocities of the three DC brushless motors in order to achieve the dead-reckoning of the robot. The purpose of the dead-reckoning of the robot is, given a correct initial pose, to continuously keep track of its correct poses with respect to the reference frame. This dead-reckoning pose estimation can be improved by fusing the laser scanning data.

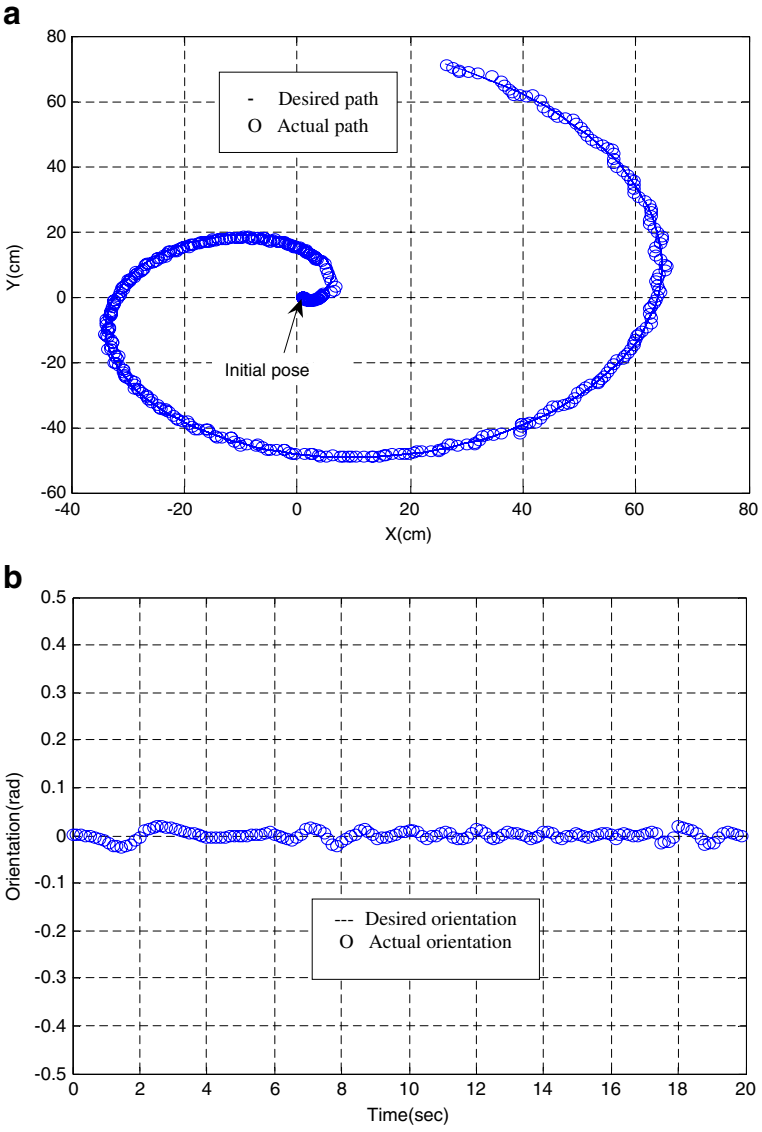

Fig. 11 **a** Experimental result of the Archimedes' spiral trajectory tracking. **b** Time response of the orientation $\theta(t)$

5.2.2 Point Stabilization

The first experiment was conducted to investigate the regulation performance of the proposed polar-space adaptive control law 13. The initial poses of the omnidirectional mobile robot were assumed at $(x_0,\ y_0,\ \theta_0,) = (1\text{ cm},\ \alpha\text{ cm},\ 0\text{rad})$, and the desired final eight goal poses were located on a circle with the radius of 100cm and given by $(x_d,\ y_d,\ \theta_d) = (100\cos(\alpha)\text{ cm},\ 100\sin(\alpha)\text{ cm},\ \pi/2\text{ rad})$, where

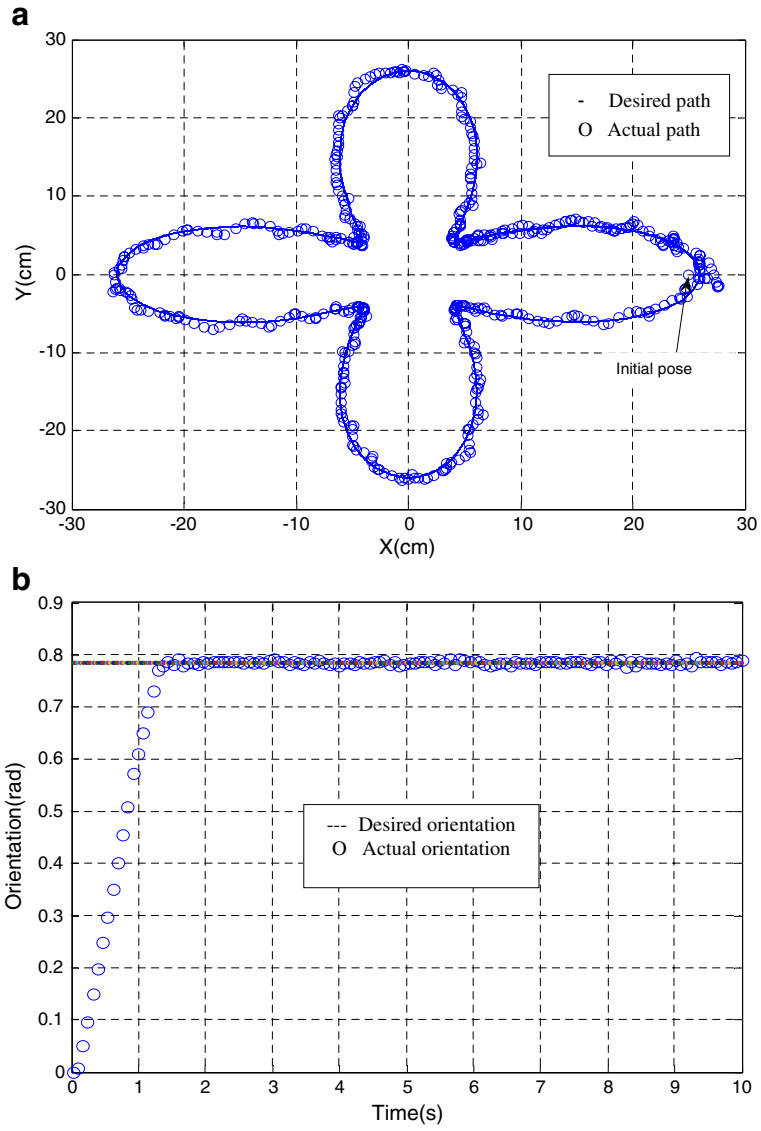

Fig. 12 **a** Experimental result of the rose curve trajectory tracking. **b** Time response of the orientation $\theta(t)$

$\alpha = 0,\ \pi/4,\ \pi/2,\ 3\pi/4,\ \pi,\ 5\pi/4,\ 6\pi/4,\ 7\pi/4$, respectively. Figure 10a depicts all the trajectories of the omnidirectional mobile robot from the initial pose to the eight goal poses and Fig. 10b presents the orientation response in the case of $\alpha = \pi/4$.

5.2.3 Polar-Space Adaptive Archimedes' Spiral and Rose Curve Trajectory Tracking

This subsection aims at constructing two experimental results to explore how the proposed embedded polar-space motion controller 13 steers the mobile robot to track the Archimedes' spiral trajectory described by $r_d(t) = 10\varphi_d(t) + 1$ (Unit: cm) with $\theta_d(t) = 0$. (unit: rad) and the rose curve trajectory expressed by $r_d(t) = 16 + 10\cos(4\varphi_d(t))$(Unit: cm) with $\theta_d(t) = \pi/4$ (unit: rad). Figures 11 and 12 respectively demonstrate the experimental results of Archimedes' spiral and rose curve trajectory tracking for the mobile robot. These results indicate that the proposed adaptive motion controller 13 is capable of successfully steering the omnidirectional mobile robot to track these special trajectories.

6 Conclusions

This paper has presented an embedded polar-space adaptive motion controller for an autonomous omnidirectional mobile robot with three independent driving wheels equally spaced at 120° from one to another. With the derived polar-space dynamic model, the adaptive motion controller has been synthesized via adaptive backstepping to achieve trajectory tracking and stabilization simultaneously. The proposed adaptive motion controller has been efficiently implemented into an FPGA chip using the hardware/software co-design and SoPC techniques. The reusable user IP core library has been rapidly developed in the same FPGA chip by incorporating with the embedded processor and the RTOS in the same chip, in order to steer the mobile robot to follow the desired trajectory. From the simulation results, the proposed control method has been shown to outperform the conventional Cartesian-space PI feedback control approach in [7] and non-adaptive kinematic polar-space controller in [14]. Through experimental results, the proposed polar-space adaptive motion control scheme has been successfully shown to be capable of giving satisfactory dynamic tracking performance for the embedded three-wheeled omnidirectional mobile robot. An interesting topic for future research would be the extension of the aforementioned technique and method to four-wheeled omnidirectional mobile robots.

Acknowledgement The authors gratefully acknowledge the financial support from the National Science Council, Taiwan, Republic of China, under grant NSC98-2218-E-241-003.

References

1. Wu, C.J., Tsai, C.C.: Localization of an autonomous mobile robot based on ultrasonic sensory information. J. Intell. Robot. Syst. **30**, 267–277 (2001)
2. Jiang, Z.P., Nijmeijer, H.: A recursive technique for tracking control of nonholonomic systems in chained form. IEEE Trans. Automat. Contr. **44**, 265–279 (1999)
3. Dixon, W.E., Dawson, D.M., Zergeroglu, E., Zhang, F.: Robust tracking and regulation control for mobile robots. Int. J. Robust Nonlinear Control **10**, 199–216 (2000)

4. Lee, T.C., Song, K.T., Lee, C.H., Teng, C.C.: Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. IEEE Trans. Control Syst. Technol. **9**, 305–318 (2001)
5. Li, T.H., Chang, S.J.: Autonomous fuzzy parking control of a car-like mobile robot. IEEE Trans. Syst. Man Cybern. Syst. Hum. **33**(4), 451–465 (2003)
6. Pin, F.G., Killough, S.M.: A new family of omnidirectional and holonomic wheeled platforms for mobile robots. IEEE Trans. Automat. Contr. **10**(4), 480–489 (1994)
7. Watanabe, K., Shiraishi, Y., Tzafestas, S., Tang, J., Fukuda, T.: Feedback control of an omnidirectional autonomous platform for mobile service robots. J. Intell. Robot. Syst. **22**, 315–330 (1998)
8. Kalmár-Nagy, T., D'Andrea, R., Ganguly, P.: Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle. Robot. Auton. Syst. **46**, 47–64 (2004)
9. Williams, R.L. II, Carter, B.E., Gallina, P., Rosati, G.: Dynamic model with slip for wheeled omnidirectional robots. IEEE Trans. Automat. Contr. **18**(3), 285–293 (2002)
10. Huang, H.C., Tsai, C.C.: FPGA implementation of an embedded robust adaptive controller for autonomous omnidirectional mobile platform. IEEE Trans. Ind. Electron. **56**(5), 1604–1616 (2009)
11. Park, K., Chung, H., Lee, J.G.: Point stabilization of mobile robots via state-space exact feedback linearization. Robot. Comput.-Integr. Manuf. **16**(5), 353–363 (2000)
12. Yang, J.M., Kim, J.H.: Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots. IEEE Trans. Automat. Contr. **15**(3), 578–587 (1999)
13. Chwa, D.K.: Sliding mode tracking control of nonholonomic wheeled mobile robots in polar coordinates. IEEE Trans. Control Syst. Technol. **12**(4), 637–644 (2004)
14. Huang, H.C., Tsai, C.C.: Simultaneous tracking and stabilization of an omnidirectional mobile robot in polar coordinates: a unified control approach. Robotica **27**(3), 447–458 (2009)
15. Huang, H.C., Tsai, C.C.: Adaptive robust control of an omnidirectional mobile platform for autonomous service robots in polar coordinates. J. Intell. Robot. Syst. **51**(4), 439–460 (2008)
16. Campo, I., Echanobe, J., Bosque, G., Tarela, J.M.: Efficient hardware/software implementation of an adaptive neuro-fuzzy system. IEEE Trans. Fuzzy Syst. **16**(3), 761–778 (2008)
17. Kung, Y.S., Tsai, M.H.: FPGA-based speed control IC for PMSM drive with adaptive fuzzy control. IEEE Trans. Power Electron. **22**(6), 2476–2486 (2007)
18. Solano, S.S., Cabrera, A.J., Baturone, I., Moreno-Velo, F.J., Brox, M.: FPGA implementation of embedded fuzzy controllers for robotic applications. IEEE Trans. Ind. Electron. **54**(4), 1937–1945 (2007)
19. Huang, S.J., Wu, S.S.: Vision-based robotic motion control for non-autonomous environment. J. Intell. Robot. Syst. **54**(5), 733–754 (2009)
20. Han, S.H., Lee, M.H., Mohler, R.R.: Real-time implementation of a robust adaptive controller for a robotic manipulator based on digital signal processors. IEEE Trans. Syst. Man Cybern. Syst. Hum. **29**(2), 194–204 (1999)
21. Li, T.H., Chang, S.J., Chen, Y.X.: Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot. IEEE Trans. Ind. Electron. **50**(5), 867–880 (2003)
22. Deliparaschos, K.M., Nenedakis, F.I., Tzafestas, S.G.: Design and implementation of a fast digital fuzzy logic controller using FPGA technology. J. Intell. Robot. Syst. **45**(1), 77–96 (2006)
23. Juang, C.F., Chen, J.S.: Water bath temperature control by a recurrent fuzzy controller and its FPGA implementation. IEEE Trans. Ind. Electron. **53**(3), 941–949 (2006)
24. Newman, K.E., Hamblen, J.O., Hall, T.S.: An introductory digital design course using a low-cost autonomous robot. IEEE Trans. Ed. **45**(3), 289–296 (2002)
25. Fife, W.S., Archibald, J.K.: Reconfigurable on-board vision processing for small autonomous vehicles. EURASIP Journal on Embedded Systems **2007**, 1–14 (2007)
26. Esmaeilzadeh, H., Farzan, F., Shahidi, N., Fakhraie, S.M., Lucas, C., Tehranipoor, M.: NnSP: embedded neural networks stream processor. In: 48th Midwest Symposium on Circuits and Systems 1, 223–226 (2005)
27. Kuo, C.H., Chen, C.Y.: Development of vision guided small size humanoid soccer robot using chip based controllers. In: Proceedings of the International Automatic Control Conference, pp. 854–859 (2007)
28. Krstic, M., Kanellakopoulos, L., Kokotovic, P.: Nonlinear and Adaptive Control Design. Wiley, New York (1995)
29. Kung, Y.S., Shu, G.S.: Design and implementation of a control IC for vertical articulated robot arm using SOPC technology. In: Proceedings of the IEEE International Conference on Mechatronics, pp. 532–536 (2005)