

# Robust Backstepping Tracking Control Using Hybrid Sliding-Mode Neural Network for a Nonholonomic Mobile Manipulator with Dual Arms

Meng-Bi Cheng, *Student member, IEEE*, and Ching-Chih Tsai, *Senior member, IEEE*

**Abstract**— This paper presents a methodology for trajectory tracking control of a wheeled dual-arm mobile manipulator with parameter uncertainties and external load variations. Based on backstepping technique, the proposed control laws comprise two levels: kinematic and dynamic. First, the auxiliary kinematic velocity control laws for the mobile robot and the two onboard arms are separately established. Second, a robust backstepping tracking control based on hybrid sliding-mode neural networks (HSMNN) is presented to ensure the velocity tracking ability in spite of the uncertainties. The proposed robust backstepping tracking controller is actually composed of a neural network controller, a robust controller, and a proportional controller. To achieve the overall trajectory tracking goal, a neural network controller is developed to imitate an equivalent control law in the sliding-mode control, a robust controller is designed to incorporate the system dynamics into the sliding surface for guaranteeing the asymptotical stability, and the proportional controller is designed to improve the transient performance for randomly initializing neural network. All the adaptive learning algorithms for the proposed controller are derived from the Lyapunov stability theory so that the close-loop asymptotical tracking ability can be guaranteed no matter the uncertainties taken place or not. Simulation results demonstrate the feasibility as well as usefulness of the proposed control strategy in comparison with other conventional control methods.

## I. INTRODUCTION

Recently, tracking control of nonholonomic mobile manipulators has been investigated by several researchers [1]-[10]. Being composed of a mobile robot and the onboard manipulator(s), mobile manipulators extend the manipulator workspace and its ability to work efficiently. Due to the mobile base, such a manipulator is capable of configuring itself to perform missions at any operational points. Hence, mobile manipulator(s) have been effectively implemented in a variety of tasks (such as opening door, handing object, nursing care, etc. [1]). In general, the mobile manipulators can be categorized into three classes: a) mobile manipulator with one arm, b) multiple mobile manipulators, and c) mobile manipulator with dual arms. Much work has been reported on the first kind in dealing with the problems of system modeling [2]-[3], trajectory planning, motion control

[4]-[6], etc. In addition, few studies have paid attention to the second class to cope with the key issues including cooperation [7] and coordination. In the other hand, the studies of the third one have become important in designing service robots and humanoid robots for their dual-arm configurations [8]-[10]. However, as is always the case, there is a cost to pay for advantages: more difficulties in system modeling, control and synthesis.

Dynamic modeling and control of mobile manipulators with dual arms have been addressed by few researchers. For example, Yamamoto and Yun [8] first presented the general form of the motion equation, and then analyzed its system mobility and manipulability. Cheng and Tsai [9] utilized *Mathematica* package to derived the system dynamics motion via Lagrange multipliers methodology, and proposed a nonlinear backstepping tracking controller to deal with the trajectory tracking problem under the assumption of given system's parameters.

In contrary to the previous robot systems, the control problems for mobile manipulator with one arm have been investigated using many alternatives such as feedback linearization [3], computed torque method, robust adaptive control [4], neural network [5]-[6], and etc. The conventional methods as the feedback linearization and computed-torque method require the precise system modeling in the torque controller to cancel out the nonlinear dynamic effects; such methods are impractical in reality for the reason that the system's parameters are not accurately acquired in advance and some external disturbances would be exerted on the robots. However, it is well-known that entire uncertainties have been proven to degrade the control performance dramatically; therefore, a novel robust tracking controller is required to count for the dynamic uncertainties and to release the linearly in parameters (LIP) condition in adaptive robust control [4].

In the past decades, there have been a variety of successful applications of neural networks (NNs) in the field of robots for their on-line learning and function approximations [11]. For instance, Lin and Goldenberg [5] performed real experiment of the radial basis function (RBF) NN tracking controllers for the mobile platform and one arm on the IRIS facility, called *RoboTwin*. Lee *et al.* [6] presented a neuron-adaptive tracking control for a mobile manipulator, employing the RBF learning and adaptive compensation of approximation error based on a priori knowledge on the system dynamics.

In this paper, a HSMNN tracking controller, consisting of sliding-mode control, neural control, and proportional control, will be proposed to achieve the tracking control of the nonholonomic mobile manipulator with dual arms under

This work was supported by National Science Council, Taiwan, R.O.C. under grant NSC93-2213-E-005-037.

M. B. Cheng is with the Electrical Engineering Department, National Chung Hsing University, Taiwan, R.O.C. (e-mail: [chenmb.tw@msa.hinet.net](mailto:chenmb.tw@msa.hinet.net)).

C. C. Tsai is with the Electrical Engineering Department, National Chung Hsing University, Taiwan, R.O.C. (phone: 886-4-24859351, e-mail: [cctsay@dragon.nchu.edu.tw](mailto:cctsay@dragon.nchu.edu.tw)).

some uncertainties including sudden mass variations, friction effects, and external disturbances. The proposed method is novel in releasing the limitations of the LIP, reducing the heavy computation of the regression matrix, and requiring no system dynamics.

The rest of this paper is organized as follows. Section 2 presents the dynamic model of the nonholonomic mobile manipulator with dual arms. In Section 3, a robust tracking controller is synthesized to eliminate the detrimental effects of the uncertainties. In Section 4, numerical results are given to illustrate the usefulness of the proposed tracking controller. Section 5 concludes this paper.

## II. DYNAMIC MODELS

As shown in Fig.1, consider a typical wheeled mobile manipulator with dual arms subject to  $m$  nonholonomic constraints  $A(q_v)\dot{q}_v=0$  whose dynamics behavior can be modeled via Lagrange's equation [5, 9] in the following form  $\mathbf{M}(q)\ddot{q} + \mathbf{C}_m(q, \dot{q})\dot{q} + \mathbf{F}(q) + \mathbf{G}(q) + \tau_d = \mathbf{B}(q)\tau - \mathbf{A}^T(q)\lambda_l$  (1) where  $q = (q_1, \dots, q_n)^T \in \mathbb{R}^{n \times 1}$  denotes  $n$  generalized coordinates;  $\mathbf{M}(q) \in \mathbb{R}^{n \times n}$  a symmetric, positive definite inertia matrix;  $\mathbf{C}_m(q, \dot{q}) \in \mathbb{R}^{n \times n}$  the centripetal and Coriolis matrix;  $\mathbf{F}(q) \in \mathbb{R}^{n \times 1}$  the friction force vector;  $\mathbf{G}(q) \in \mathbb{R}^{n \times 1}$  the gravitational vector;  $\tau_d$  bounded unknown disturbances including unmodeled dynamics;  $\mathbf{B}(q) \in \mathbb{R}^{n \times (n-m)}$  the input transformation matrix;  $\tau(t) \in \mathbb{R}^{(n-m) \times 1}$  the input torque vector;  $\mathbf{A}(q) \in \mathbb{R}^{m \times n}$  the matrix associated with the constraints; and  $\lambda_l \in \mathbb{R}^{m \times 1}$  the vector of constraint force.

In order to eliminate the constraint force  $\lambda_l$ , let  $\mathbf{S}_i(q)$  be a full rank matrix  $(n-m)$  formed by a set of smooth and linearly independent vector fields spanning the null space of  $\mathbf{A}(q)$ , i.e.,  $\mathbf{S}_i^T(q)\mathbf{A}^T(q) = 0$ ; hence, the original dynamic model (1) can be represented as two levels: kinematic part and dynamic part for the purpose of controller design.

$$\dot{q} = \mathbf{S}_i(q)\eta(t) \quad (2)$$

where  $\eta(t) \in \mathbb{R}^{(n-m) \times 1}$  is an auxiliary velocity vector. The dynamic model (1) can be reduced as follows;

$$\bar{\mathbf{M}}\dot{\eta} + \bar{\mathbf{C}}_m\eta + \bar{\mathbf{F}}(q_v) + \bar{\mathbf{G}} + \bar{\tau}_d = \bar{\mathbf{B}}\tau \quad (3)$$

where  $\bar{\mathbf{M}} = \mathbf{S}_i^T \mathbf{M} \mathbf{S}_i$ ,  $\bar{\mathbf{C}}_m = \mathbf{S}_i^T [\dot{\mathbf{M}} \mathbf{S}_i + \mathbf{C}_m \mathbf{S}_i]$ ,  $\bar{\mathbf{G}} = \mathbf{S}_i^T \mathbf{G}$ ,  $\bar{\mathbf{F}} = \mathbf{S}_i^T \mathbf{F}$ ,  $\bar{\tau}_d = \mathbf{S}_i^T \tau_d$ , and  $\bar{\mathbf{B}} = \mathbf{S}_i^T \mathbf{B}$ .

Hence, the overall dynamics of this system can also be separated into three subsystems: one mobile platform and two onboard manipulators. Let the  $n$  generalized coordinates be divided into three sets  $q = (q_v^T, q_r^T, q_l^T)^T$ , where  $q_v \in \mathbb{R}^{n_v \times 1}$  describes the generalized coordinates of the mobile platform, and  $q_r \in \mathbb{R}^{n_r \times 1}$  as well as  $q_l \in \mathbb{R}^{n_l \times 1}$  denote the joint angle vectors of the each link in the clamped right and left manipulators, respectively;  $n = n_v + n_r + n_l$ . In the following, the subscript 'r' denotes the right arm, and 'l' represents the

left arm. Then, the mobile manipulator's dynamics (1) can be rewritten as the following three subsystems' dynamics [5],[9]:

$$\begin{aligned} \mathbf{M}_v(q_v)\ddot{q}_v + \mathbf{C}_{mv}(q, \dot{q})\dot{q}_v + \mathbf{F}_v(\dot{q}_v) + \mathbf{G}_v(q_v) + \tau_{dv} \\ = \mathbf{B}_v(q_v)\tau_v - \mathbf{A}_v^T(q_v)\lambda_v - f_{vr} - f_{vl} \end{aligned} \quad (4)$$

$$\mathbf{M}_r(q_r)\ddot{q}_r + \mathbf{C}_{mr}(q, \dot{q})\dot{q}_r + \mathbf{F}_r(\dot{q}_r) + \mathbf{G}_r(q_r) + \tau_{dr} = \tau_r - f_{rv} \quad (5)$$

$$\mathbf{M}_l(q_l)\ddot{q}_l + \mathbf{C}_{ml}(q, \dot{q})\dot{q}_l + \mathbf{F}_l(\dot{q}_l) + \mathbf{G}_l(q_l) + \tau_{dl} = \tau_l - f_{lv} \quad (6)$$

where  $f_{vj} \triangleq \mathbf{M}_{vj}\ddot{q}_j + \mathbf{C}_{mvj}\dot{q}_j$ , and  $f_{jv} \triangleq \mathbf{M}_{jv}\ddot{q}_v + \mathbf{C}_{mjv}\dot{q}_v$ ,  $j = r, l$ , represent the dynamic interaction caused by the two onboard manipulators and mobile platform, respectively. Moreover, the dynamics of the mobile platform (4) can be represented in a more compact form as

$$\bar{\mathbf{M}}_v\dot{\eta}_v + \bar{\mathbf{C}}_m\eta_v + \bar{\mathbf{F}}_v + \bar{\mathbf{G}}_v + \bar{\tau}_{dv} = \bar{\mathbf{B}}_v\tau_v - \bar{f}_{vr} - \bar{f}_{vl} \quad (7)$$

where  $\eta_v = (v, \omega)^T$ ,  $\bar{\mathbf{G}}_v = \mathbf{S}_v^T \mathbf{G}_v$ ,  $\bar{\mathbf{M}}_v = \mathbf{S}_v^T \mathbf{M}_v \mathbf{S}_v$ ,  $\bar{\mathbf{C}}_{mv} = \mathbf{S}_v^T [\dot{\mathbf{M}}_v \mathbf{S}_v + \mathbf{C}_{mv} \mathbf{S}_v]$ ,  $\bar{\mathbf{F}}_v = \mathbf{S}_v^T \mathbf{F}_v$ ,  $\bar{\tau}_{dv} = \mathbf{S}_v^T \tau_{dv}$ ,  $\bar{\mathbf{B}}_v = \mathbf{S}_v^T \mathbf{B}_v$ ,  $\bar{f}_{vr} = \mathbf{S}_v^T \cdot f_{vr}$ , and  $\bar{f}_{vl} = \mathbf{S}_v^T \cdot f_{vl}$ .

## III. ROBUST BACKSTEPPING TRACKING CONTROL DESIGN

The trajectory tracking problem of a wheeled mobile manipulator with dual arms is formulated as follows. Let the mobile platform track a prescribed model cart with the reference trajectory  $q_{vr} = [x_r, y_r, \theta_r]^T$  along with the velocities  $V_r = [v_r, \omega_r]^T$ ,  $v_r > 0$ , for all  $t \geq 0$ , i.e.,  $\dot{x}_r = v_r \cos \theta_r$ ,  $\dot{y}_r = v_r \sin \theta_r$ ,  $\dot{\theta}_r = \omega_r$ . The pair  $(x_r, y_r)$  represents the Cartesian coordinates of the cart,  $\theta_r$  is the orientation,  $v_r$  and  $\omega_r$  are the linear and angular velocities. Moreover, the two onboard arms aim to independently track along with desired position  $q_{jd}(t)$  and velocity trajectories  $\dot{q}_{jd}(t)$  simultaneously, where  $j = r, l$ . The goal of tracking controller design is to find a smooth torque control  $\tau$  such that the position and velocity tracking errors of the system under parameter variations and disturbances can also converge to zeros, i.e.,  $\lim_{t \rightarrow \infty} (q_{vr} - q_v) = \lim_{t \rightarrow \infty} (\dot{q}_{vr} - \dot{q}_v) = 0$ , and

$$\lim_{t \rightarrow \infty} (q_j - q_{jd}) = \lim_{t \rightarrow \infty} (\dot{q}_j - \dot{q}_{jd}) = 0, \text{ where } j = r, l.$$

Based on backstepping technique, the design procedure for the proposed controller is divided into subsequent three steps. The first step is to employ the two auxiliary velocity controllers to steer the mobile vehicle and the onboard arms, and the second step is to propose the sliding-mode equivalent torque controller in dynamic level. Finally, the NN controller is designed to compensate for the dynamic uncertainties in the proposed equivalent control law.

**Step 1: kinematic level control laws design** [9].

The tracking error vector is expressed in the basis of a frame attached to the mobile base as

$$e_p = [e_1, e_2, e_3]^T \equiv T_e \cdot (q_{vr} - q_v) \quad (8)$$

where  $e_1$ ,  $e_2$ , and  $e_3$  denote the tangential, normal and orientation tracking errors, respectively.  $T_e$  represents the

coordinate transformation matrix. Among many existing ways in the literature [9, 14] to select the auxiliary velocity control input,  $\alpha_c(t)$ , to realize the velocity tracking goal, we adopt our previous result in [9].

$$\alpha_c = \begin{bmatrix} v_c \\ w_c \end{bmatrix} = \begin{bmatrix} v_r \cos e_3 + k_1 e_1 \\ \frac{1}{1 + \alpha e_1} (w_r + \alpha v_r \sin(e_3) + k_2 \delta \cdot \bar{e}_3 + \delta \frac{v_r \sin(e_1)}{\bar{e}_3} e_2) \end{bmatrix} \quad (9)$$

where  $\bar{e}_3 = e_3 + \alpha e_2$ ,  $k_1 > 0$ ,  $k_2 > 0$ ,  $k_3 > 0$ ,  $\delta > 0$  and  $0 \leq \alpha \leq \frac{1}{\|e_1\|_{\max}}$ . Moreover, the auxiliary velocity command for the onboard arms is employed to ensure the position tracking ability in [12].

$$\alpha_i(t) = -K_i \cdot z_{i1}(t) + \dot{q}_{id}(t); i = r, l \quad (10)$$

where  $z_{i1} = q_i(t) - q_{id}(t) = e_i(t) \in \mathbb{R}^{n_i}$ ,  $i = r, j$  are the two tracking error vectors. Then, if the system parameters are known in advance, the dynamic interactions between the three subsystems can be directly measured, and without any uncertainties including unmodeled dynamics and disturbances, then the following three tracking control laws can be designed for the mobile platform and the onboard dual arms to drive to the desired trajectories.

$$\tau_v = \bar{B}_v^{-1} (K_4 \cdot e_s + \bar{M}_v \dot{V}_{cd} + C_{mv} V_{cd} + \bar{F}_v + \bar{\tau}_{dv}) \quad (11)$$

$$\tau_i = f_{iv} + \tau_{di} + C_{mi}(q, \dot{q}) \alpha_i + F_i(\dot{q}_i) + G_i(q_i) + \bar{M}_i[\ddot{q}_{id} - \dot{z}_{i1}] - z_{i1} - K_{bi} z_{i2}, i = r, j \quad (12)$$

where  $e_v = \alpha_c - \eta_v$ , and  $z_{i2} = \dot{q}_i(t) - \alpha_i(t)$ ,  $i = r, l$  are the velocity tracking errors for the mobile vehicle and the two arms, respectively;  $K_4$  and  $K_{bi}$  are positive-definite diagonal matrices.

### Step 2: Torque controller design via sliding surface

However, if the parameters of this system are perturbed or unknown, then the ideal tracking backstepping control laws for the mobile robot (11) and the onboard arms (12) cannot guarantee the tracking performance and the closed-loop stability. Thus, the goal of this step is to find a torque controller  $\tau \in \mathbb{R}^{(n-m) \times 1} = \mathbb{R}^{n_s \times 1}$  for the overall system to deal with the problems of parameters uncertainties and external bounded disturbances such that  $\lim_{t \rightarrow \infty} e_s(t) = 0$ , where  $e_s = V_{cd} - \eta$  be an auxiliary velocity tracking error and  $V_{cd}$  is composed of the two stabilization functions (9) and (10), i.e.,

$$V_{cd} = [\alpha_c^T, \alpha_r^T, \alpha_l^T]^T \in \mathbb{R}^{(n-m) \times 1} \quad (13)$$

Let a proportional-integration (PI) type sliding surface be defined as

$$S(t) = \left( \frac{d}{dt} + \lambda \right) \int_0^t e_s(\tau) dz = e_s(t) + \lambda \int_0^t e_s(z) dz \quad (14)$$

where  $\lambda$  is a positive constant. Note that, since the function  $S(t) = 0$  when  $t = 0$ , there is no reaching phase as in the traditional sliding-mode control. Differentiating  $S(t)$  and using (3), one obtains

$$\begin{aligned} \dot{S}(t) &= \dot{e}_s(t) + \lambda e_s(t) \\ &= \dot{V}_{cd}(t) - U(t) + \bar{M}^{-1} (\bar{C}_m \eta + \bar{F} + \bar{G} + \bar{\tau}_d) + \lambda e_s(t) \end{aligned} \quad (15)$$

where  $U(t) = \bar{M}^{-1} B \tau$  can be regarded as a new control vector. The aforementioned tracking problem is thought of as finding

a control law  $U(t)$  to make the state remain in the sliding surface  $S(t) = 0$  for all  $t > 0$ . To design this sliding-mode controller, we derive first the equivalent control law  $U_{eq}(t)$ , which will decide the system behavior on the sliding surface, by setting

$$\dot{S}(t)|_{U=U_{eq}} = 0 \quad (16)$$

which leads to

$$U_{eq} = \dot{V}_{cd} + \bar{M}^{-1} (\bar{C}_m \eta + \bar{F} + \bar{G} + \bar{\tau}_d) + \lambda e_s. \quad (17)$$

Therefore, the error dynamics of the system in the sliding surface for  $t \geq 0$  becomes

$$\dot{e}_s + \lambda e_s = 0 \quad (18)$$

Clearly, the system performance can be easily assured by a suitable  $\lambda$ .

### Step 3: Robust tracking control (RTC) using HSMNN

As mentioned before, if the system parameters are unknown or perturbed, neither the performance specified by (18) nor the system stability can be guaranteed by the equivalent control design. To ensure the system performance (18) in despite of existence of the uncertain system dynamics, a sliding-mode neural network (SMNN) is proposed which consists of a radial basis function (RBF) controller and a robust controller. The general function of the used three-layer neural network (NN) with  $n_i$  input neurons,  $k$  hidden neurons and  $n_o$  output neurons can be represented in following form [11]

$$y = U_{NN}(e_s, V, W, m, c) = W \sigma(V e_s) \quad (19)$$

where the velocity tracking error  $e_s$  is the input vector of the NN;  $V \in \mathbb{R}^{k \times n_i}$  is the input-to-hidden layer weighting matrix;  $W \in \mathbb{R}^{n_o \times k}$  is the hidden-to output layer weighting matrix; the activation function used in the hidden layer is chosen as RBF,

i.e.,  $\sigma(V \cdot e_s) = \exp \left[ -\frac{(V \cdot e_s - m)^2}{c^2} \right] \in \mathbb{R}^{k \times 1}$ , in which there are two

adjustable vector variables: the centroid vector  $m \in \mathbb{R}^{k \times 1}$  and the covariance vector  $c \in \mathbb{R}^{k \times 1}$ . Thus, an ideal NN controller  $U_{NN}^*$  will be designated to achieve the equivalent control law shown in (17) such that

$$U_{eq}(t) = U_{NN}^*(e_s, V^*, W^*, m^*, c^*) + \varepsilon \equiv W^* \sigma^*(V^* e_s) + \varepsilon \quad (20)$$

where  $\varepsilon$  is a minimum approximation error;  $W^*$ ,  $V^*$ ,  $m^*$ , and  $c^*$  are the optimal parameter vectors of  $W$ ,  $V$ ,  $m$  and  $c$  in the NN, respectively. The SMNN control law is assumed to take the following form:

$$U(t) = \hat{U}_{NN}(e_s, \hat{V}, \hat{W}, \hat{m}, \hat{c}) + U_s \equiv \hat{W} \hat{\sigma}(\hat{V} e_s) + U_s \quad (21)$$

where  $\hat{U}_{NN}$  is the NN controller;  $U_s$  is the robust controller;  $\hat{V}$ ,  $\hat{W}$ ,  $\hat{m}$  and  $\hat{c}$  are the estimates of the ideal parameter vectors. In order to cancel out the initial learning errors from the neural networks and to well improve the transient response, one can add a proportional controller to the SMNN control law (21) such that

$$U_{hc}(t) = k_v e_s + \hat{W} \hat{\sigma}(\hat{V} e_s) + U_s \quad (22)$$

This form of controller (22) is called hybrid sliding-mode neural network controller (HSMNN) whose block diagram is illustrated in Fig.2. With (22), the error dynamics of the closed-loop system in the sliding surface becomes

$$\dot{e}_s + (\lambda + k_v)e_s = 0, \quad t \geq 0. \quad (23)$$

Obviously, (23) has a faster transient response than (18). Substituting (22) into (15), the error equation governing the closed-loop system can be obtained from

$$\dot{e}_s + (\lambda + k_v)e_s = U_{eq} - U_{hc} = \dot{S} \quad (24)$$

Moreover,  $\tilde{U}_{hc}$  is defined as

$$\begin{aligned} \tilde{U}_{hc} &= U_{eq} - U_{hc} = \mathbf{W}^* \sigma^* + \varepsilon - \hat{\mathbf{W}} \hat{\sigma} - U_s \\ &= \tilde{\mathbf{W}} \sigma^* + \tilde{\mathbf{W}} \tilde{\sigma} + \varepsilon - U_s \end{aligned} \quad (25)$$

where  $\tilde{\mathbf{W}} = \mathbf{W}^* - \hat{\mathbf{W}}$  and  $\tilde{\sigma} = \sigma^* - \hat{\sigma}$ . In this study, the hidden-layer output error  $\tilde{\sigma}$  can be linearized in terms of Taylor series [13],

$$\tilde{\sigma} = [\tilde{\sigma}_1 \quad \tilde{\sigma}_2 \quad \dots \quad \tilde{\sigma}_k]^T = \sigma_v \tilde{V} e_v + \sigma_m \tilde{m} + \sigma_c \tilde{c} + O_n \quad (26)$$

where

$$\sigma_v = \left[ \frac{\partial \sigma_1}{\partial (V e_v)} \quad \dots \quad \frac{\partial \sigma_k}{\partial (V e_v)} \right]_{V e_v = \hat{V} e_v} \in \mathbb{R}^{k \times k}, \quad \sigma_m = \left[ \frac{\partial \sigma_1}{\partial m} \quad \dots \quad \frac{\partial \sigma_k}{\partial m} \right]_{m = \hat{m}} \in \mathbb{R}^{k \times k},$$

$$\tilde{V} = V^* - \hat{V}, \quad \tilde{m} = m^* - \hat{m}, \quad \sigma_c = \left[ \frac{\partial \sigma_1}{\partial c} \quad \frac{\partial \sigma_2}{\partial c} \quad \dots \quad \frac{\partial \sigma_k}{\partial c} \right]_{c = \hat{c}} \in \mathbb{R}^{k \times k},$$

$\tilde{c} = c^* - \hat{c}$ .  $O_n \in \mathbb{R}^{k \times 1}$  denotes the vector of containing higher-order terms. Then, equation (26) gives

$$\sigma^* = \hat{\sigma} + \sigma_v \tilde{V} e_s + \sigma_m \tilde{m} + \sigma_c \tilde{c} + O_n \quad (27)$$

$$\tilde{U}_{hc} = \mathbf{W}^* (\hat{\sigma} + \hat{\sigma}_v \tilde{V} e_s + \hat{\sigma}_m \tilde{m} + \hat{\sigma}_c \tilde{c} + O_n) + \varepsilon - \hat{\mathbf{W}} \hat{\sigma} - U_s \quad (28)$$

$$= \tilde{\mathbf{W}} \hat{\sigma} + \hat{\mathbf{W}} \sigma_v \tilde{V} e_s + \hat{\mathbf{W}} \sigma_m \tilde{m} + \hat{\mathbf{W}} \sigma_c \tilde{c} - U_s + E$$

where the uncertain term  $E = \hat{\mathbf{W}} \sigma_v \tilde{V} e_v + \hat{\mathbf{W}} \sigma_m \tilde{m} + \hat{\mathbf{W}} \sigma_c \tilde{c} + \mathbf{W}^* O_n + \varepsilon$  is assumed bounded by  $\|E\| < \psi(t)$ .

### Theorem 3.1

Consider the dynamic model (1) and the kinematic model (2) of the dual-arm wheeled mobile manipulator. Given the auxiliary velocity command  $V_{cd}(t)$  governed by (13), the hybrid sliding-mode neural network tracking control law is designed as (22), in which the adaptation laws of the NN controllers are updated along with (29-30) and the robust controller (31) is proposed as follows:

$$\dot{\hat{\mathbf{W}}} = \eta_1 (\hat{\sigma} S^T)^T; \quad \dot{\hat{V}} = \eta_2 (e_v S^T \hat{\mathbf{W}} \sigma_v)^T \quad (29)$$

$$\dot{\hat{m}} = \eta_3 (S^T \hat{\mathbf{W}} \sigma_m)^T; \quad \dot{\hat{c}} = \eta_4 (S^T \hat{\mathbf{W}} \sigma_c)^T \quad (30)$$

$$U_s = \psi(t) \text{sgn}(S); \quad \dot{\hat{\psi}}(t) = \eta_5 S^T \text{sgn}(S) \quad (31)$$

where  $\eta_1, \eta_2, \eta_3, \eta_4$ , and  $\eta_5$  are positive constants;  $\text{sgn}(\cdot)$  is the signum function;  $\hat{\psi}(t)$  is the estimated value of the uncertain term bound  $\psi(t)$ . Then the closed-loop system dynamics can be always stayed within the sliding surface such that asymptotically stability can be guaranteed if the uncertainty satisfies  $\|E\| < \psi(t)$ .

**Proof:**

Choose the following Lyapunov function candidate:

$$\begin{aligned} V_a &= V_b + \frac{1}{2} S^T S + \frac{1}{2\eta_1} \text{Tr}(\tilde{\mathbf{W}} \tilde{\mathbf{W}}^T) + \frac{1}{2\eta_2} \text{Tr}(\tilde{V}^T \tilde{V}) + \frac{1}{2\eta_3} \tilde{m}^T \tilde{m} \\ &\quad + \frac{1}{2\eta_4} \tilde{c}^T \tilde{c} + \frac{1}{2\eta_5} \tilde{\psi}^2(t) \end{aligned} \quad (32)$$

where  $V_b = \frac{1}{2}(e_1 + e_2)^2 + \frac{1}{2\delta} \bar{e}_3^2 + \sum_{j=r,l} (\frac{1}{2} z_{j1}^T z_{j1} + \frac{1}{2} z_{j2}^T M_j z_{j2})$

$+ \frac{1}{2} e_v^T \bar{M}_v e_v$ ,  $z_{j2} = \dot{q}_j(t) - \alpha_j(t)$ ,  $j = r, l$ ,  $\text{Tr}(\cdot)$  is the trace operator, and  $\tilde{\psi} = \psi - \hat{\psi}(t)$  is defined as the estimation error.

Differentiating (32), one obtains

$$\begin{aligned} \dot{V}_a &= \dot{V}_b + S^T \dot{S} + \frac{1}{\eta_1} \text{Tr}(\tilde{\mathbf{W}} \dot{\tilde{\mathbf{W}}}^T) + \frac{1}{\eta_2} \text{Tr}(\dot{\tilde{V}}^T \tilde{V}) + \frac{1}{\eta_3} \dot{\tilde{m}}^T \tilde{m} \\ &\quad + \frac{1}{\eta_4} \dot{\tilde{c}}^T \tilde{c} + \frac{1}{\eta_5} \dot{\tilde{\psi}} \tilde{\psi} \end{aligned} \quad (33)$$

Substituting (15) and (28) into (33) yields

$$\begin{aligned} \dot{V}_a &= \dot{V}_b + \text{Tr} \left\{ \tilde{\mathbf{W}} \left[ \hat{\sigma} S^T - \frac{1}{\eta_2} \hat{\mathbf{W}}^T \right] \right\} + \text{Tr} \left\{ \left[ e_v S^T \hat{\mathbf{W}} \sigma_v - \frac{1}{\eta_2} \dot{\hat{V}}^T \right] \tilde{V} \right\} \\ &\quad + \left[ S^T \hat{\mathbf{W}} \sigma_m - \frac{1}{\eta_3} \dot{\tilde{m}}^T \right] \tilde{m} + \left[ S^T \hat{\mathbf{W}} \sigma_c - \frac{1}{\eta_4} \dot{\tilde{c}}^T \right] \tilde{c} + S^T (E - U_s) + \dot{V}_b \end{aligned}$$

If the adaptation laws for the neural network controller are chosen as (29)-(30), the robust controller is designated as (31), and the result in [12] is employed, then  $\dot{V}_a$  becomes

$$\begin{aligned} \dot{V}_a &= \dot{V}_b + S^T E - S^T \hat{\psi} \text{sgn}(S) - \frac{1}{\eta_5} \dot{\hat{\psi}} \cdot \eta_5 S^T \text{sgn}(S) \\ &\leq \dot{V}_b + \|S^T\| (\|E\| - \psi) \\ &\leq -k_1 e_1^2 - k_\delta \bar{e}_3^2 - e_v^T \mathbf{K}_4 e_v - z_{r1}^T \mathbf{K}_r z_{r1} - z_{r2}^T \mathbf{K}_{\sigma r} z_{r2} \\ &\quad - z_{l1}^T \mathbf{K}_l z_{l1} - z_{l2}^T \mathbf{K}_{\sigma l} z_{l2} - \beta \|S^T\| \leq 0 \end{aligned} \quad (34)$$

where  $\mathbf{K}_j, \mathbf{K}_{\sigma j}, j = r, l$  are positive-definite diagonal matrices, and the constant  $\beta = \psi - \|E\| > 0$  is positive. Since

$\dot{V}_a(S(t), \tilde{\psi}(t), \tilde{\mathbf{W}}, \tilde{V}, \tilde{m}, \tilde{c}) \leq 0$ ,  $\dot{V}_a$  is negative semidefinite, that is,  $V_a(S(t), \tilde{\psi}(t), \tilde{\mathbf{W}}, \tilde{V}, \tilde{m}, \tilde{c}) \leq V_a(S(0), \tilde{\psi}(0), \tilde{V}, \tilde{\mathbf{W}}, \tilde{m}, \tilde{c})$  implies that  $S(t), \tilde{\mathbf{W}}, \tilde{V}, \tilde{m}, \tilde{c}$  are uniformly bounded. Let  $\Theta(t) = \beta \|S^T\| \leq -\dot{V}_a$  and integrate function  $\Theta(t)$  over the time interval  $[0, t]$  to have

$$\int_0^t \Theta(z) dz \leq V_a(S(0), \tilde{\psi}(0), \tilde{V}, \tilde{\mathbf{W}}, \tilde{m}, \tilde{c}) - V_a(S(t), \tilde{\psi}(t), \tilde{V}, \tilde{\mathbf{W}}, \tilde{m}, \tilde{c}) \quad (35)$$

Since the fact that  $V_a(S(0), \tilde{\psi}(0), \tilde{V}, \tilde{\mathbf{W}}, \tilde{m}, \tilde{c})$  is bounded and  $V_a$  is non-increasing and bounded, the following result is obtained as

$$\lim_{t \rightarrow \infty} \int_0^t \Theta(z) dz < \infty \quad (36)$$

Moreover,  $\dot{\Theta}(t)$  is also bounded. With Barbalat's lemma, it can be shown that  $\lim_{t \rightarrow \infty} \Theta(t) = 0$ , i.e.,  $S(t) \rightarrow 0$  as  $t \rightarrow \infty$ . As a result, the error dynamics is maintained in the sliding surface so that the robust tracking control system is asymptotically stable. Furthermore, the velocity tracking error,  $e_s(t)$ , will converge to the zero due to  $s(t) \rightarrow 0$ . Accordingly, both the position and velocity tracking performance are guaranteed by

the auxiliary velocity control input  $V_{cd}(t)$  in (13).

#### IV. SIMULATION RESULTS AND DISCUSSION

For simplicity, a mobile manipulator composed of the two driving wheels mobile platform and two-link onboard dual-arm with parameters in [3] is adopted in this study to verify the efficacy of the proposed robust backstepping tracking controller (RBST) and the backstepping controller (BSC) [5, 9,14].

##### A. Uncertain Case in Tracking a Circular Path with Larger Initial Errors

The reference trajectory for the mobile robot is designed as a circular trajectory with a radius of 2 m at the center of (2 m, 2 m) along with the reference velocity  $[v_r, \omega_r] = [2\text{ m/s}, 1.0\text{ rad/s}]$ . Meanwhile, the reference position trajectories for the onboard dual manipulators are assigned as  $q_{rd}(t) = q_{ld}(t) = (\sin(t), \cos(t))^T$ . The mobile manipulator was driven from the initial values  $[x, y, \phi, \theta_{r1}, \theta_{r2}, \theta_{l1}, \theta_{l2}, v, w, \dot{\theta}_{r1}, \dot{\theta}_{r2}, \dot{\theta}_{l1}, \dot{\theta}_{l2}] = [-5.5\text{ m}, -4.5\text{ m}, 20^\circ, -1\text{ rad}, -1\text{ rad}, -1\text{ rad}, -1\text{ rad}, 0, 0, 0, 0, 0, 0]$ . The system parameters suffered from sudden changes of the mass and inertia, and the friction and bounded external disturbances added after 0.5 seconds as follows;

$$\begin{aligned} \Delta m_c &= 50.0\text{ kg}, \Delta I_c = 15.0\text{ kg} - \text{m}^2, \Delta m_2 = \Delta m_{b2} = 20\text{ kg}, \\ \Delta I_{m2} &= \Delta I_{mb2} = 10\text{ kg} - \text{m}^2, \bar{\tau}_d = [-5, 10, 10, 10, 10, 10]^T, \\ \bar{F} &= [4v, 4w, 4\dot{\theta}_{r1}, 4\dot{\theta}_{r2}, 4\dot{\theta}_{l1}, 4\dot{\theta}_{l2}]. \end{aligned}$$

The simulation results, shown in Fig.6, indicate that the both sum of square errors (SSEs) in the position and velocity of the proposed hybrid controller are much smaller than those of the backstepping approach, and also the proposed controller has less control efforts. It is worthy to mention that there is no prior knowledge of the mobile manipulator dynamics, and the controller indeed compensates for the system uncertainties through the on-line adaptation algorithm. Despite of the sudden uncertainties occurred after 0.5 seconds, the system trajectories quickly converge to the steady values due to the effect of the sliding-mode control.

##### B. Uncertain Case for the Mobile Base in Tracking a Straight Line with Small Initial Errors

Moreover, in order to verify the efficacy of proposed method by comparing with other conventional methods like NN controllers, the system models can be simplified as a mobile robot by neglecting the effects of the two onboard manipulators. There are four controllers considered in the subsequent simulations: (a) computed torque method (CTC), (b) NN controller [14], (c) SMNN (21), and (d) proposed hybrid tracking controller (22). The parameters of the mobile base are given as  $m = 10\text{ kg}, I = 5.0\text{ kg} - \text{m}^2, R = 0.5\text{ m}$  and  $r = 0.05\text{ m}$  [14], but the system parameters suffer from sudden change and some uncertainties caused by the friction and the bounded external disturbance added after 1 second as  $\Delta m = 30.0\text{ kg}, \Delta I = 30.0\text{ kg} - \text{m}^2, \bar{\tau}_d = [-25, 50]^T, \bar{F} = [10*v, 20*w]$ .

The reference trajectory is designed as a straight line with the initial position (1m, 2m) and the slope of  $45^\circ$ , and the reference velocity is assigned as  $[v_r, \omega_r] = [0.5\text{ m/s}, 0.0\text{ rad/s}]$ . The initial position of the mobile robot is  $[x_0, y_0, \theta_0] = [2\text{ m}, 1\text{ m}, 26.56^\circ]$ . The design parameters of the controller are selected as:  $k_1 = 10, k_2 = 16, k_3 = 8$  and the parameters of the adaptive learning algorithm are selected as  $\eta_1 = 50, \eta_2 = 50, \eta_3 = 10, \eta_4 = 10, \eta_5 = 40$ . Meanwhile, the sliding-model neural network controller is constructed by means of two input neurons, six hidden neurons, and two output neurons. The simulation results are presented in Fig.4. The performance comparisons of the four controllers show that the proposed controller is glossier than others, and its SSE is the best. As can be seen in Fig.4 (b), the tracking position and velocity errors sharply drop to zero and the control torque quickly converges to its steady state. From the previous two cases, the simulation results have well illustrated the effectiveness of the proposed robust backstepping tracking controller.

#### V. CONCLUSIONS

This paper has developed a robust backstepping tracking controller using hybrid sliding-mode neural network for a nonholonomic mobile manipulator with dual arms to achieve precise velocity and position tracking under uncertainties including parameter variations, frictions and disturbances. The robust backstepping tracking controller comprises the sliding mode neural network controller, the robust controller, and the proportional controller; it has been shown to guarantee closed-loop asymptotically stability and also to restrain the system behavior in the sliding surface under system uncertainties. All parameter adaptation laws in the HSMNN control are derived via the Lyapunov stability theory. The simulation results have shown the feasibility and effectiveness of the proposed control methods by comparing with the well-known backstepping methods and neural network controllers. An interesting topic for future research might be to extend the proposed method to the force and position tracking problem.

#### REFERENCES

- [1] G. T. Herbert, K. J. Kyriakopoulos, and N. J. Krikelis, "Modeling of multiple mobile manipulators handling a common deformable object," *Journal of Robotic Systems*, vol.15, no.11, pp.599-623, Nov. 1998.
- [2] Q. Yu and I-M. Chen, "A General Approach to the Dynamics of Nonholonomic Mobile Manipulator Systems," *Journal of Dynamic Systems Measurement and Control*, vol.124, no.4, pp.512-611, Dec. 2002.
- [3] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," *IEEE Trans. on Automatic Control*, vol. 39, no. 6, pp.1326 - 1332, June 1994.
- [4] W. Dong, "On trajectory and force tracking control of constrained mobile manipulators with parameter uncertainty," *Automatica*, vol.38, no.9, pp.1475-1484, Sept. 2002.
- [5] S. Lin and A. A. Goldenberg, "Neural-network control of mobile manipulators," *IEEE Trans. on Neural Networks*, vol.12, no.5, pp.1121-1133, Sept. 2001.

- [6] C.Y. Lee, T. D. Eom, and J. J. Lee, "Neuron-adaptive control of mobile manipulators based on compensation of approximation error," *Electronics Letters*, vol.38, no.16, pp.935-936, Aug. 2002.
- [7] Y. Yamamoto and X. Yun, "Unified Analysis on Mobility and Manipulability of Mobile Manipulators," in *Proc. IEEE Internal. Conference on Robotics and Automation*, Vol. 2, 1999, pp.1200-1206.
- [8] T. G. Sugar and V. Kumar, "Control of cooperating mobile manipulator," *IEEE Trans. on Robotic and Automation*, vol. 18, no. 1, pp.94-103, Feb. 2002.
- [9] M. P. Cheng and C. C. Tsai, "Dynamic modeling and tracking control of a nonholonomic wheeled mobile manipulator with two robotic arms," in *Proc. 42nd IEEE Conference on Decision and Control*, Vol. 3, 2003, pp.2932-2937.
- [10] Y. Hirata, and T. Sawada, "Leader-follower type motion control algorithm of multiple mobile robots with dual manipulators for handing a single object in coordination," in *Proc. 2004 IEEE Conference on Intelligent Mechatronics and Automation*, 2004, pp.362-367.
- [11] R.J. Wai, "Tracking control based on neural network strategy for robot manipulator," *Neurocomputing*, vol.51, pp.425-445, 2003.
- [12] M. B. Cheng, and C. C. Tasi, "Hybrid robust tracking control using sliding-mode neural network for a nonholonomic mobile manipulator," in *Proc. IEEE International Conference on Mechatronics*, 2005, pp.537-542.
- [13] F. L. Lewis, "Nonlinear network structure for feedback control," *Asian Journal of Control*, vol.4, pp.205-228,1999.
- [14] R. Fierro, and F. L. Lewis, "Control of a nonholonomic mobile robot using neural network," *IEEE Trans. on Neural Network*, vol. 9, no. 4, pp.589-600, July. 1998.

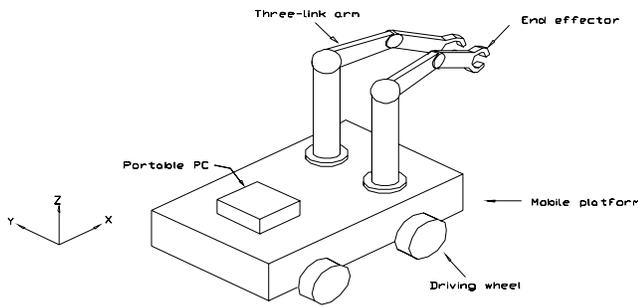


Fig. 1. The typical model of a dual-arm nonholonomic mobile manipulator

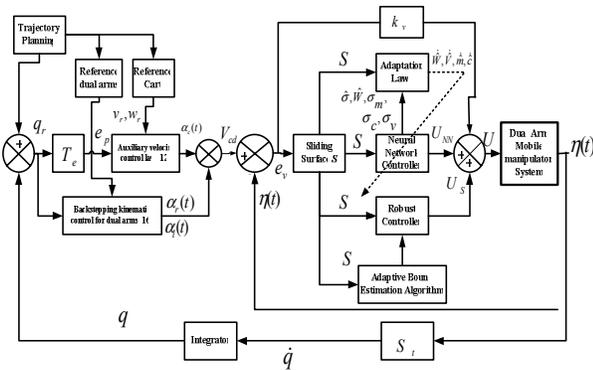


Fig. 2. Structure of the proposed robust backstepping tracking controller.

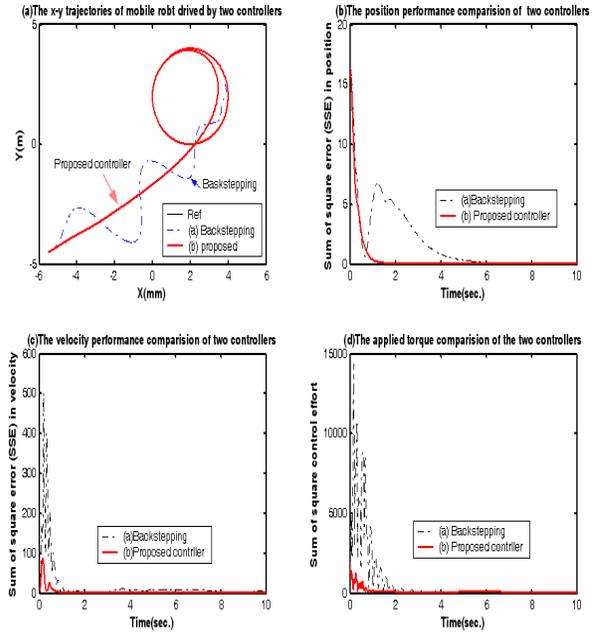


Fig. 3. Simulation results of *Case A* under two controllers: the proposed robust backstepping tracking controller and backstepping controller. (a) The comparison of x-y trajectories of the mobile platform. (b) SSEs in overall position vector. (c) SSEs in overall velocity vector. (d) The absolute magnitude sum of applied torques of the overall system.

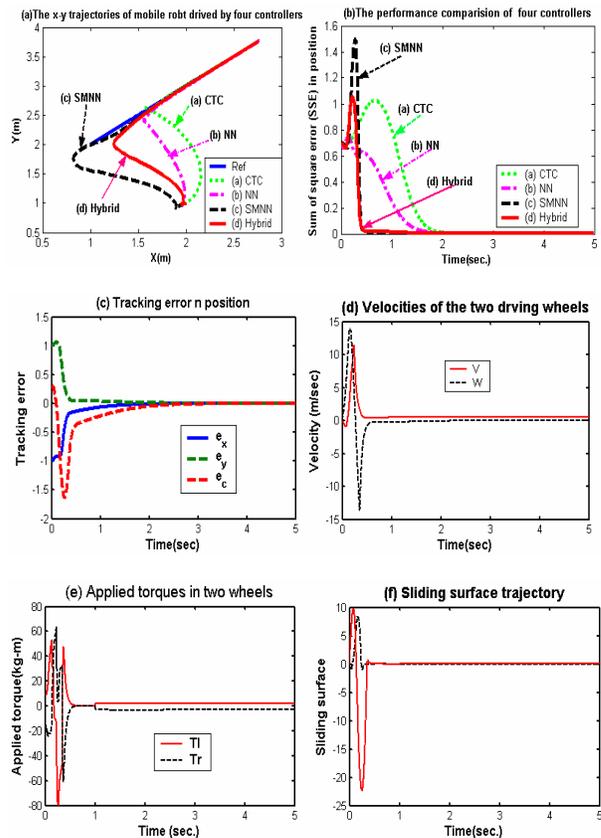


Fig.4. Simulation results of the four controller in *Case B*. (a) The x-y trajectories of the mobile robot. (b) The performance comparisons. (c) Tracking position error trajectories. (d) The velocities of the mobile robot. (e) The applied torques. (f) The sliding surface trajectories.