Robotica: (2019) volume 37, pp. 1937–1955. © Cambridge University Press 2019
doi:10.1017/S0263574719000365

Adaptive Neural Feedback Linearizing Control of Type (*m*,*s*) Mobile Manipulators with a Guaranteed Prescribed Performance

Khoshnam Shojaei†‡* and Ali Kazemy§

†*Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran*
‡*Digital Processing and Machine Vision Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran*
§*Department of Electrical Engineering, Tafresh University, Tafresh 39518-79611, Iran.*
E-mail: kazemy@tafreshu.ac.ir

(Accepted March 3, 2019. First published online: April 10, 2019)

SUMMARY
In this paper, a neural network (NN)-based tracking controller is proposed for a general class of type (*m*,*s*) wheeled mobile manipulators (WMMs) subjected to model uncertainties with prescribed transient and steady-state performance specifications. First, an input–output model of WMMs is derived by introducing proper output equations. Then, the prescribed performance technique is employed to propose a proportional integral derivative trajectory tracking controller for WMMs to ensure that the tracking errors converge to a smaller, arbitrary ultimate bound with a predefined maximum overshoot/undershoot and convergence speed. The learning capabilities of multilayer NNs are incorporated into the controller to approximate the uncertain nonlinear dynamics of the robot. An adaptive saturation-type controller is utilized to compensate NN estimation errors and external disturbances. A Lyapunov-based stability analysis is used to demonstrate that the tracking errors are uniformly ultimately bounded and converge to a small neighborhood of zero with a guaranteed prescribed performance. Numerical computer simulations are presented to show the effectiveness of the proposed controller.

KEYWORDS: Adaptive robust control; Model uncertainty; Multilayer neural networks; Prescribed performance function; Trajectory tracking; Wheeled mobile manipulators.

1. Introduction

Recent decades have witnessed a growing interest in the motion control of wheeled mobile manipulators (WMMs) because of the existence of high nonlinearities, coupling, and interaction between the robot manipulator and the mobile platform. This challenging nature of WMMs demands very high performance requirements from the control engineering viewpoint. A type (*m*,*s*) mobile manipulator refers to a robotic arm mounted on a type (*m*,*s*) mobile platform where *m* and *s* denote the mobility and steerability of the mobile platform. The WMM mobility lets the robot manipulate objects in an infinite workspace compared with a fixed robotic manipulator. In recent years, many control strategies have been proposed for the motion control of WMMs.[1–20] The modeling, coordinated control, and stability of mobile manipulators have been addressed previously[1] using the feedback linearization technique. The internal and zero dynamics stability of WMMs has been investigated previously.[1] Since an exact modeling of WMMs is a difficult task among many robotic systems, they suffer from

* Corresponding author. E-mail: khoshnam.shojaee@gmail.com

a high degree of uncertainties. For this purpose, many researchers often employ robust and adaptive control approaches for their motion control.[2–9] However, the classic adaptive and robust control methods always require the linear-in-parameter assumption of the dynamic model. Toward this end, the artificial intelligence methods, including fuzzy and neural networks, have been utilized repeatedly in the literature[10–16] for the motion control of WMMs. For example, Lin and Goldenberg[10] have addressed the neural network (NN) control of mobile manipulators originally. An adaptive neural fuzzy controller was proposed for multiple uncertain constrained nonholonomic WMMs.[11] Xu *et al.*[12] proposed a robust NN-based sliding mode controller for omnidirectional WMMs. A decentralized adaptive fuzzy tracking controller has been proposed in ref. [13] for unicycle-type WMMs. A hybrid sliding mode fuzzy NN controller for mobile manipulators is presented in ref. [14]. A constrained model predictive control technique was employed in the design of tracking controllers for WMMs in ref. [15]. More recently, the cooperative and teleoperation control problems of networked mobile manipulators have been studied in refs. [17–20]. However, one main drawback of all the aforementioned control approaches is the unpredictable transient behavior of system response due to the online adaptation in adaptive and NN control.[1–20] Therefore, the design of an approximation-based tracking controller with a guaranteed prescribed transient and steady-state performance is of a great importance. To address this problem, one of the best available strategies is the prescribed performance function (PPF), which has been originally introduced in ref. [21] and its applications are reported in refs. [22–25].

To the best of our knowledge, there exist limited works addressing the design of tracking controllers for WMMs with a guaranteed prescribed performance. Moreover, almost all the previous controllers are only applicable to a special type of mobile manipulators with a unicycle-type or omnidirectional mobile platform. Compared with the current state of the art, the main contributions of this paper are listed as follows:

(i) A neural adaptive robust input–output feedback linearizing proportional integral derivative (PID) controller is proposed for the general representation of type (m,s) mobile manipulators by introducing a proper set of output equations, while the previous works, including refs. [1–20], have been only devoted to a special type of WMMs.
(ii) In contrast to previous works,[1–20] a prescribed maximum overshoot/undershoot, convergence rate, and final tracking accuracy in the transient and steady-state behavior of the WMM control system are taken into account.
(iii) Since the precise modeling and derivation of a regression matrix are impossible for the general formulation of type (m,s) WMMs, a multilayer NN (MLNN) is utilized to effectively compensate the uncertain nonlinear dynamics of the robot. Then, the NN approximation error and external disturbances are efficiently compensated by an adaptive saturation-type robust controller.

The remainder of this article is presented in the following order. Section 2 explains the WMM control problem formulation and some mathematical preliminaries. Then, a neural adaptive input–output feedback linearization controller is proposed in Section 3. Afterward, numerical simulations are illustrated in Section 4 to evaluate the performance of tracking controller. In the last section, concluding remarks are given.

2. Problem Formulation

2.1. Kinematic and dynamic models of WMMs

Consider a general class of n-link WMMs with mobility m and steerability s, which are called type (m,s) WMMs.[26–35] According to ref. [33], mobile platforms are classified based on their mobility and steerability, which are also reviewed in ref. [32] and completely defined in refs. [33, 34]. The kinematic mobility of a robot chassis is its ability to directly move in the environment, which quantifies the degrees of controllable freedom based on changes to the wheel velocity and is limited to the range $1 \leq m \leq 3$. A basic constraint limiting the mobility is the rule that every wheel must satisfy its sliding constraint. Steerability is the number of conventional centered orientable wheels that can be

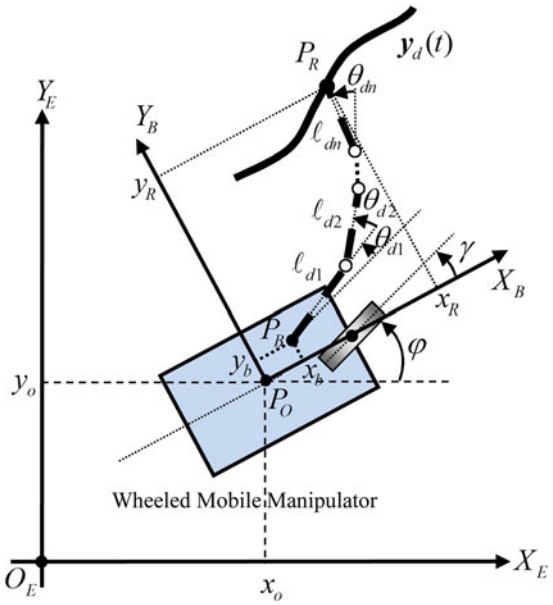

Fig. 1. Planar configuration of a general type (*m*,*s*) WMM.

oriented independently to steer the mobile platform and is limited to $0 \le s \le 2$. The sum of mobility m and steerability s is called *maneuverability*, which defines the overall degrees of freedom that a mobile robot can manipulate.[34] The motion equations of a type (*m*,*s*) WMM are given by:

$$\dot{\boldsymbol{q}}_v = \boldsymbol{G}(\varphi, \boldsymbol{\gamma})\boldsymbol{\mu}(t), \tag{1}$$

$$\boldsymbol{\mu}(t) = \boldsymbol{X}(\boldsymbol{q}_v)\boldsymbol{v}(t), \tag{2}$$

$$\boldsymbol{M}_a(\boldsymbol{q})\dot{\boldsymbol{v}}(t) + \boldsymbol{C}_a(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{v}(t) + \boldsymbol{\tau}_{da}(t, \boldsymbol{v}) = \boldsymbol{B}_a(\boldsymbol{q})\boldsymbol{\tau}_a(t), \tag{3}$$

$$\boldsymbol{\tau}_{da}(t, \boldsymbol{v}) = \boldsymbol{D}_a\boldsymbol{v}(t) + \boldsymbol{D}_c\mathrm{sgn}(\boldsymbol{v}) + \boldsymbol{\tau}_d(t), \tag{4}$$

where $\boldsymbol{M}_a$, $\boldsymbol{C}_a$, $\boldsymbol{D}_a$, and $\boldsymbol{B}_a$ are given by

$$\boldsymbol{M}_a(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{M}_{vv} & \boldsymbol{M}_{vr} \\ \boldsymbol{M}_{rv} & \boldsymbol{M}_{rr} \end{bmatrix}, \ \boldsymbol{C}_a(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} \boldsymbol{C}_{vv} & \boldsymbol{C}_{vr} \\ \boldsymbol{C}_{rv} & \boldsymbol{C}_{rr} \end{bmatrix}, \boldsymbol{D}_a = \begin{bmatrix} \boldsymbol{D}_v & 0 \\ 0 & \boldsymbol{D}_r \end{bmatrix}, \ \boldsymbol{B}_a(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{B}_v(\boldsymbol{q}_v) & 0 \\ 0 & \boldsymbol{I}_n \end{bmatrix}, \tag{5}$$

where $\boldsymbol{q} = [\boldsymbol{q}_v^T, \boldsymbol{q}_r^T]^T \in \Re^{n+s+3}$ is the augmented state vector; $\boldsymbol{q}_v = [\boldsymbol{\zeta}, \ \varphi, \ \boldsymbol{\gamma}]^T \in \Re^{s+3}$ and $\boldsymbol{q}_r \in \Re^n$ represent the generalized coordinates of the mobile platform and n-link manipulator, respectively; $\boldsymbol{\zeta} = [x_o, y_o]^T$ denotes the coordinates of a reference point P_o on the mobile platform; φ is the heading angle according to Fig. 1; $\boldsymbol{\gamma} = [\boldsymbol{\gamma}^{1T}, \boldsymbol{\gamma}^{2T}]^T \in \Re^s$ represents the steering coordinates of independent steering wheels; $\boldsymbol{G}(\varphi, \boldsymbol{\gamma}) \in \Re^{(s+3)\times(m+s)}$ is the kinematic matrix; and $\boldsymbol{\mu}(t) = [\boldsymbol{\omega}_m^T(t), \ \boldsymbol{\omega}_s^T(t)]^T \in \Re^{m+s}$ is a vector of angular velocities of robot wheels. The vectors $\boldsymbol{\omega}_m(t) \in \Re^m$ and $\boldsymbol{\omega}_s(t) \in \Re^s$ are related to velocities of mobility and steering coordinates of the mobile platform through the kinematic matrix $\boldsymbol{G}(\varphi, \boldsymbol{\gamma})$, respectively. The velocity vector $\boldsymbol{\mu}(t)$ is transformed to a pseudo-velocity vector $\boldsymbol{v}(t) \in \Re^{m+s}$ by the transformation matrix $\boldsymbol{X}(\boldsymbol{q}_v) \in \Re^{(m+s)\times(m+s)}$. The vector $\boldsymbol{v} = [\boldsymbol{v}^T, \dot{\boldsymbol{q}}_r^T]^T \in \Re^p$ includes system velocity signals, where $p = m + s + n$. $\boldsymbol{M}_a(\boldsymbol{q}) \in \Re^{p\times p}$ is a symmetric positive-definite inertia matrix of the mobile manipulator; $\boldsymbol{C}_a(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \Re^{p\times p}$ is the centripetal and Coriolis matrix; $\boldsymbol{B}_v(\boldsymbol{q}_v) \in \Re^{(m+s)\times(m+s)}$ is the input transformation matrix; $\boldsymbol{D}_a \in \Re^{p\times p}$ stands for viscous friction and damping coefficients matrix; and $\boldsymbol{D}_c \in \Re^{p\times p}$ represents Coulomb friction matrix. The term $\boldsymbol{\tau}_d(t) = [\boldsymbol{\tau}_{dv}^T, \boldsymbol{\tau}_{dr}^T]^T \in \Re^p$ denotes bounded time-varying disturbances and unmodeled dynamics.

$\boldsymbol{\tau}_a = [\boldsymbol{\tau}_v^T, \boldsymbol{\tau}_r^T]^T \in \Re^p$ denotes the torque input vector of the mobile manipulator. The kinematic models (1) and (2) and dynamic Eq. (3) are integrated into the following state space representation:

$$\dot{\boldsymbol{x}} = \overbrace{\begin{bmatrix} \boldsymbol{H}(\boldsymbol{q}_v)\,\boldsymbol{v} \\ 0 \end{bmatrix}}^{f(\boldsymbol{x})} + \overbrace{\begin{bmatrix} 0 \\ \boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{B}_a(\boldsymbol{q}) \end{bmatrix}}^{g(\boldsymbol{x})} \boldsymbol{\tau}_a + \overbrace{\begin{bmatrix} 0 \\ -\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{C}_a(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{v} - \boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{D}_a\boldsymbol{v} \end{bmatrix}}^{\xi(\boldsymbol{x})} + \underbrace{\begin{bmatrix} 0 \\ -\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{D}_c\,\mathrm{sgn}(\boldsymbol{v}) - \boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{\tau}_d(t) \end{bmatrix}}_{\eta(t,\boldsymbol{x})}, \tag{6}$$

where $\boldsymbol{x} = [\boldsymbol{q}^T, \boldsymbol{v}^T]^T \in \Re^{p+n+s+3}$ represents the state vector; $\boldsymbol{f}(\boldsymbol{x}) \in \Re^{p+n+s+3}$, $\boldsymbol{g}(\boldsymbol{x}) \in \Re^{(p+n+s+3)\times p}$, and $\boldsymbol{\xi}(\boldsymbol{x})$, $\boldsymbol{\zeta}(\boldsymbol{x}) \in \Re^{p+n+s+3}$ are smooth vector fields where the kinematic matrix $\boldsymbol{H}(\boldsymbol{q}_v)$ is defined as follows:

$$\boldsymbol{H}(\boldsymbol{q}_v) = \begin{bmatrix} \boldsymbol{G}(\varphi,\boldsymbol{\gamma})\boldsymbol{X}(\boldsymbol{q}_v) & 0 \\ 0 & \boldsymbol{I}_n \end{bmatrix}, \tag{7}$$

$$\boldsymbol{G}(\varphi,\boldsymbol{\gamma}) = \begin{bmatrix} \boldsymbol{R}^T(\varphi)\boldsymbol{Q}(\boldsymbol{\gamma}) & \boldsymbol{0} \\ \boldsymbol{b}^T(\boldsymbol{\gamma}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_s \end{bmatrix}, \tag{8}$$

where the kinematic matrix $\boldsymbol{Q}(\boldsymbol{\gamma}) \in \Re^{2\times m}$ and the vector $\boldsymbol{b}(\boldsymbol{\gamma}) \in \Re^m$ may be extracted from Table I of ref. [32] for a type (*m*,*s*) mobile platform.

2.2. *Input–output model of type (m,s) WMMs*

In this section, a second-order input–output model is developed to solve the tracking control of WMMs. Motivated by the works in refs. [1, 32], the following smooth epimorphism transformation is introduced to develop the input–output model:

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{\zeta} + \boldsymbol{R}^T(\varphi)\boldsymbol{d}(\ell_{dj}, \theta_{dj}, \boldsymbol{\gamma}^2) \\ \boldsymbol{\gamma}^1 \\ \boldsymbol{q}_m \end{bmatrix}, \quad j = 1, 2, \ldots, n, \tag{9}$$

where $\boldsymbol{d}(\ell_{dj}, \theta_{dj}, \boldsymbol{\gamma}^2) \in \Re^2$ is a vector representing the coordinates of a virtual reference point $P_R = (x_R, y_R)$ in the front of the WMM in the body-fixed frame, which is given by

$$\boldsymbol{d} = \begin{bmatrix} x_b + \sum_{j=1}^{n} \ell_{dj} \cos\left(\gamma + \sum_{k=1}^{j} \theta_{dk}\right) & y_b + \sum_{j=1}^{n} \ell_{dj} \sin\left(\gamma + \sum_{k=1}^{j} \theta_{dk}\right) \end{bmatrix}^T, \tag{10}$$

and $\boldsymbol{R}(\bullet) \in \Re^{2\times 2}$ is a rotation matrix between earth-fixed frame $\{O_E, X_E, Y_E\}$ and body-fixed frame $\{O_B, X_B, Y_B\}$; ℓ_{dj} and θ_{dk} specify the desired configuration of *n*-link manipulator associated with the mobile manipulator, according to Fig. 1; and (x_b, y_b) is the coordinates of the arm base on the WMM. In addition, the steering variables $\boldsymbol{\gamma}^1 \in \Re^{m+s-2}$ and $\boldsymbol{\gamma}^2 \in \Re^{2-m}$ can be extracted from Table 2 of ref. [32] for a type (*m*,*s*) mobile platform. From the time derivative of the output equation $y \in \Re^p$ in (9) and replacing (1) and (2), one gets

$$\dot{\boldsymbol{y}} = L_f\boldsymbol{h} + L_\xi\boldsymbol{h} + L_\eta\boldsymbol{h} + \left(L_g\boldsymbol{h}\right)\boldsymbol{\tau}_a = \underbrace{\frac{\partial \boldsymbol{h}(\boldsymbol{q})}{\partial \boldsymbol{q}}\boldsymbol{H}(\boldsymbol{q}_v)}_{\boldsymbol{J}(\boldsymbol{q})}\boldsymbol{v}(t), \tag{11}$$

where $L_f\boldsymbol{h}(\boldsymbol{x}) = \nabla\boldsymbol{h}(\boldsymbol{x})\boldsymbol{f}$, $L_g\boldsymbol{h}(\boldsymbol{x}) = \nabla\boldsymbol{h}(\boldsymbol{x})\boldsymbol{g}$, $L_\xi\boldsymbol{h}(\boldsymbol{x}) = \nabla\boldsymbol{h}(\boldsymbol{x})\boldsymbol{\xi}$, and $L_\eta\boldsymbol{h}(\boldsymbol{x}) = \nabla\boldsymbol{h}(x)\boldsymbol{\eta}$ show the Lie derivatives of $\boldsymbol{h}$ along the direction of the vectors $\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{\xi}$, and $\boldsymbol{\eta}$, respectively; $\nabla\boldsymbol{h}(x)$ represents the gradient of $\boldsymbol{h}$ with respect to $\boldsymbol{x}$. Since $\dot{\boldsymbol{y}}$ in (11) is not related to the actuator input, we differentiate (11) once again to obtain the following input–output model:

$$\begin{aligned}\ddot{\boldsymbol{y}} &= L_f^2\boldsymbol{h}(\boldsymbol{x}) + L_\xi L_f\boldsymbol{h}(\boldsymbol{x}) + L_\eta L_f\boldsymbol{h}(\boldsymbol{x}) + L_g L_f\boldsymbol{h}(\boldsymbol{x})\boldsymbol{\tau}_a \\ &= (\partial(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{v})/\partial\boldsymbol{q})\,\boldsymbol{H}(\boldsymbol{q}_v)\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{x})\boldsymbol{\tau}_a + L_\xi L_f\boldsymbol{h}(\boldsymbol{x}) + L_\eta L_f\boldsymbol{h}(\boldsymbol{x}),\end{aligned} \tag{12}$$

where

$$\begin{aligned}&L_f^2\boldsymbol{h}(\boldsymbol{x}) = (\partial(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{v})/\partial\boldsymbol{q})\,\boldsymbol{H}(\boldsymbol{q}_v)\boldsymbol{v}, \\ &L_\xi L_f\boldsymbol{h}(\boldsymbol{x}) = -\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{C}_a(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{v} - \dot{\boldsymbol{J}}(\boldsymbol{q})\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{D}_a\boldsymbol{v}, \\ &L_\eta L_f\boldsymbol{h}(\boldsymbol{x}) = -\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{D}_c\mathrm{sgn}(\boldsymbol{v}) - \dot{\boldsymbol{J}}(\boldsymbol{q})\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{\tau}_d(t), \\ &\boldsymbol{D}(\boldsymbol{x}) := L_g L_f\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}_a^{-1}(\boldsymbol{q})\boldsymbol{B}_a(\boldsymbol{q})\end{aligned} \tag{13}$$

2.3. Control objectives and mathematical background

The following tracking problem is addressed in this paper:

Definition 1. Given a smooth bounded desired trajectory $\boldsymbol{y}_d(t) : [0, \infty) \to \Re^{m+s+n}$, which is created by a reference model whose dynamic equation is given by

$$\dot{\boldsymbol{x}}_d(t) = \begin{bmatrix}\boldsymbol{H}(\boldsymbol{q}_{vd})\,\boldsymbol{v}_d \\ \boldsymbol{0}\end{bmatrix} + \begin{bmatrix}0 \\ \boldsymbol{I}_p\end{bmatrix}\boldsymbol{\tau}_{ad}, \quad \boldsymbol{y}_d(t) = \boldsymbol{h}(\boldsymbol{\eta}_d(t)), \tag{14}$$

the control objective of this paper is to design a feedback control law for a type (*m*,*s*) WMM system (1)–(5) such that it makes the tracking errors, $\boldsymbol{e}(t) := \boldsymbol{y}(t) - \boldsymbol{y}_d(t) \in \Re^p$, uniformly ultimately bounded (UUB) and exponentially converge to a small neighborhood of the origin with the transient and steady-state prescribed performance specifications in the presence of model uncertainties, unknown parameters, and external disturbances.

Lemma 1.[36]**:** *Assume that a rational function $H_j(s)$ of the complex variable $s = \sigma + j\omega$ is real for real s and is not identically zero for all s. Let n* be the relative degree of $H_j(s)$ with $|n^*| \le 1$. Then, $H_j(s)$ is strictly positive real (SPR) if and only if the following conditions hold: (i) $H_j(s)$ is analytic in* $\mathrm{Re}[s] > 0$*; (ii)* $\mathrm{Re}[H_j(j\omega)] > 0$, $\forall\omega \in (-\infty, \infty)$*; (iii) when* $n^* = 1$, $\lim_{|\omega|\to\infty} \omega^2\mathrm{Re}[H_j(j\omega)] > 0$*; when* $n^* = -1$, $\lim_{|\omega|\to\infty} H_j(j\omega)/j\omega > 0$.

Lemma 2.[37]**:** *Consider a linear system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$, $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$. Then, this system is SPR if and only if: (a) for any symmetric positive-definite $\boldsymbol{Q}$, there exists a symmetric positive-definite $\boldsymbol{P}$ solution of the Lyapunov equation $\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q}$, and (b) the matrices $\boldsymbol{B}$ and $\boldsymbol{C}$ satisfy $\boldsymbol{B}^T\boldsymbol{P} = \boldsymbol{C}$.*

Lemma 3.[38]**:** *The inequality $h\,\|\boldsymbol{x}\| - \boldsymbol{x}^{\mathrm{T}}h\,\boldsymbol{Tanh}(\nu h\boldsymbol{x}/\gamma_t) \le n\gamma_t$ holds for any $\gamma_t > 0$, and for any $\forall\boldsymbol{x} \in \Re^n$ and $h \in \Re$ where ν is a constant that satisfies $\nu = e^{-(\nu+1)}$, that is, $\nu = 0.2785$.*

2.4. MLNN approximation

In this section, MLNNs are introduced to approximate unknown nonlinearities of the WMM. From a review of refs. [37, 39], this technique has been effectively used to estimate unknown nonlinear dynamics of robotic systems. Figure 2 shows a simple structure of a three-layer NN whose output can be expressed in the following form:

$$y_i = \sum_{j=1}^{N_h}\left[w_{ij}\bar{\sigma}\left(\sum_{k=1}^{N_i} v_{jk}x_k + \theta_{vj}\right) + \theta_{wi}\right], \quad i = 1, 2, \ldots, N_o, \tag{15}$$

where N_h, N_i, and N_o denote the number of hidden-layer, input-layer, and output-layer neurons, respectively; w_{ij} and v_{jk} show the NN weights; the parameters θ_{wi} and θ_{vj} denote threshold offsets; and $\bar{\sigma}(.)$ is a sigmoid activation function given by $\bar{\sigma}(x) = 1/(1 + e^{-x})$ or $\bar{\sigma}(x) = \tanh(x)$. One may write (15) in the following matrix form:

$$\boldsymbol{y} = \boldsymbol{W}^T\boldsymbol{\sigma}(\boldsymbol{V}^T\boldsymbol{x}), \tag{16}$$

where $\boldsymbol{W}^T \in \Re^{N_o\times(N_h+1)}$ and $\boldsymbol{V}^T \in \Re^{N_h\times(N_i+1)}$ are weight matrices whose first columns include thresholds θ_{wi} and θ_{vj}, respectively. In addition, $\boldsymbol{x} = \left[1, x_1, \ldots, x_{N_i}\right]^T \in \Re^{N_i+1}$, $\boldsymbol{y} = \left[y_1, y_2, \ldots, y_{N_o}\right]^T \in \Re^{N_o}$, and $\boldsymbol{\sigma}(\boldsymbol{V}^T\boldsymbol{x}) = \left[1, \bar{\sigma}(\boldsymbol{V}_{r_1}^T\boldsymbol{x}), \ldots, \bar{\sigma}(\boldsymbol{V}_{r_{N_h}}^T\boldsymbol{x})\right]^T \in \Re^{N_h+1}$, where $\boldsymbol{V}_{r_j}^T$, $j = 1, 2, \ldots, N_h$, denotes *j*-th

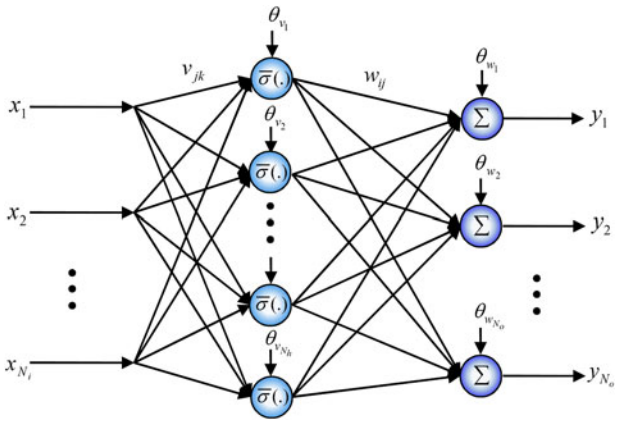

Fig. 2. Structure of an MLNN.

row of matrix $\boldsymbol{V}^T$. For a given continuous function $\boldsymbol{f}(\boldsymbol{x}) : U \to \Re^{N_o}$, where $U \subset \Re^{N_i}$ is a compact set, there exist ideal weights, thresholds, and some number of hidden-layer neurons such that $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{W}^{*T}\boldsymbol{\sigma}(\boldsymbol{V}^{*T}\boldsymbol{x}) + \boldsymbol{e}_w(\boldsymbol{x})$ where $\boldsymbol{e}_w(\boldsymbol{x})$ represents NN functional approximation error vector that is bounded over the compact set, that is, $\|\boldsymbol{e}_w(\boldsymbol{x})\| \le B_w$, $\forall x \in U$ where B_w is an unknown positive constant. These ideal NN weight matrices $\boldsymbol{W}^{*T} \in \Re^{N_o \times (N_h+1)}$ and $\boldsymbol{V}^{*T} \in \Re^{N_h \times (N_i+1)}$ are defined as follows:

$$\left(\boldsymbol{W}^*,\ \boldsymbol{V}^*\right) := \arg \min_{(W,\,V)} \left\{ \sup_{x \in U} \left\| \boldsymbol{W}^T\boldsymbol{\sigma}(\boldsymbol{V}^T\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}) \right\| \right\}$$

However, since the values of ideal NN weights $\boldsymbol{W}^*$ and $\boldsymbol{V}^*$ are not known, $\boldsymbol{f}(\boldsymbol{x})$ is replaced by its estimation $\hat{\boldsymbol{f}}(\boldsymbol{x}) = \hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x})$ in most control applications where $\hat{\boldsymbol{W}}$ and $\hat{\boldsymbol{V}}$ are estimated NN weight matrices that are updated by some appropriate update rules that will be designed in Section 3.

The following assumptions are presented to facilitate controller design and stability analysis:

Assumption 1. *The following assumptions are essential to meet the above control objectives:*

A1: *Measurements of state vector* $\boldsymbol{x} \in \Re^{p+n+s+3}$ *are available for feedback in real time.*

A2: *The desired trajectory* $\boldsymbol{y}_d(t)$ *is chosen such that* $\boldsymbol{y}_d(t)$, $\dot{\boldsymbol{y}}_d(t)$, *and* $\ddot{\boldsymbol{y}}_d(t)$ *are all bounded signals in the sense that* $\sup_{t\ge 0} \left\|\boldsymbol{y}_d(t)\right\| < B_{dp}$, $\sup_{t\ge 0} \left\|\dot{\boldsymbol{y}}_d(t)\right\| < B_{dv}$, *and* $\sup_{t\ge 0} \left\|\ddot{\boldsymbol{y}}_d(t)\right\| < B_{da}$, *where* B_{dp}, B_{dv}, *and* B_{da} *are unknown positive constants.*

A3: *The time-varying disturbance and unmodeled dynamics vector, that is,* $\boldsymbol{\tau}_d(t) \in \Re^{m+s+n}$ *is assumed to be bounded in the sense that* $\|\boldsymbol{\tau}_d(t)\| \le B_{\tau d}$ *where* $B_{\tau d}$ *is an unknown positive constant.*

A4: *The ideal NN weights are bounded on a compact set* $U \subset \Re^{N_i}$ *such that* $\left\|\boldsymbol{W}^*\right\|_F \le W_M$ *and* $\left\|\boldsymbol{V}^*\right\|_F \le V_M$, *where* W_M *and* V_M *are unknown positive constants.*[37,39]

2.5. Prescribed performance function

In order to guarantee some predefined specifications in the transient and steady-state performance behavior of the control system, the *prescribed performance control* strategy[21] is utilized in this paper. For this purpose, the performance specifications are described by decreasing bounded and strictly positive performance functions $\mu_j(t) : \Re^+ \to \Re^+$, $j = 1, 2, \ldots, p$. Based on refs. [21–25], we may obtain the prescribed performance control objectives if the following conditions hold for the tracking errors, that is, $e_j(t)$, $j = 1, 2, \ldots, p$:

$$-a_j\mu_j(t) < e_j(t) < b_j\mu_j(t), \quad j = 1, 2, \ldots, p, \quad \forall t > 0 \tag{17}$$

where a_j, b_j, $j = 1, 2, \ldots, p$ are some positive constants that are chosen by the expert designer and $\mu_j(t)$ shows an exponentially decaying performance function that is selected as follows:[21]

$$\mu_j(t) = \left(\mu_{j0} - \mu_{j\infty}\right) \exp(-\lambda_j t) + \mu_{j\infty}, \tag{18}$$

where μ_{j0}, $\mu_{j\infty}$, and $\lambda_j, j = 1, 2, \ldots, p$, denote positive parameters such that $\mu_{j0} = \mu_j(0)$, $\mu_j(t) \leq \mu_{j0}$, and $\mu_{j\infty} = \lim_{t\to\infty} \mu_j(t)$. In (17), $b_j\mu_{j0}$ and $-a_j\mu_{j0}$ define the upper and lower bounds for the maximum overshoot and undershoot values of j-th element of tracking errors, respectively; λ_j defines a lower bound for the convergence rate of j-th element of tracking errors; and $\mu_{j\infty}$ represents the allowable steady-state tracking error. Therefore, if the above parameters are adjusted properly, the specifications of the steady-state and transient performance can be determined in advance. In order to obtain the prescribed performance for tracking errors $e_j(t)$, $j = 1, 2, \ldots, p$, a smooth, strictly monotonic increasing error transformation S_j is introduced to transform the constrained errors $e_j(t)$, $j = 1, 2, \ldots, p$, to unconstrained ones, which are denoted by $\varepsilon_j(t)$, $j = 1, 2, \ldots, p$, in the sense that $-a_j < S_j(\varepsilon_j) < b_j$, $\forall \varepsilon_j \in L_\infty$, $\lim_{\varepsilon_j \to +\infty} S_j(\varepsilon_j) = b_j$, and $\lim_{\varepsilon_j \to -\infty} S_j(\varepsilon_j) = -a_j$. Therefore, we have

$$e_j(t) = \mu_j(t) S_j\left(\varepsilon_j(t)\right), \quad j = 1, 2, \ldots, p \tag{19}$$

From the fact that $S_j(\varepsilon_j)$ is smooth strictly monotonic increasing transformation and that $\mu_j(t) \geq \mu_{j\infty} > 0$ from (18), the inverse transformation exists and is given by

$$\varepsilon_j(t) := T_j\left(\hat{e}_j(t)\right) = S_j^{-1}\left(\hat{e}_j(t)\right) : \Omega_j \to \Re, \quad j = 1, 2, ..., p \tag{20}$$

where $\hat{e}_j(t) := e_j(t)/\mu_j(t)$ defines j-th normalized element of the constrained tracking error vector, $\Omega_j := \left\{\hat{e}_j(t) : \hat{e}_j \in \left(-a_j, b_j\right)\right\}$, $T_j(0) = 0, j = 1, 2, ..., p$. If $-a_j\mu_j(0) < e_j(0) < b_j\mu_j(0)$, and the tracking controller is designed such that it can guarantee $\varepsilon_j(t) \in L_\infty$, $\forall t \geq 0$, then $-a_j < S_j(\varepsilon_j) < b_j$ or $\hat{e}_j(t) \in \Omega_j$, and thus, the constrained errors $e_j(t)$, $j = 1, 2, \ldots, p$, satisfy (17). Also, if $\lim_{t\to\infty} \varepsilon_j(t) = 0$, we get $\lim_{t\to\infty} e_j(t) = 0$, which leads to the control objectives of Definition 1. To design the prescribed performance controller, we need to differentiate (20) with respect to time:

$$\dot{\varepsilon}_j(t) := \frac{1}{\mu_j(t)} \frac{\partial T_j(\hat{e}_j(t))}{\partial \hat{e}_j(t)} \left(\dot{e}_j(t) - \frac{\dot{\mu}_j(t)}{\mu_j(t)} e_j(t)\right), \quad j = 1, 2, \ldots, p \tag{21}$$

which can be stated in the following compact form:

$$\dot{\boldsymbol{\varepsilon}}(t) = \boldsymbol{R}_T \dot{\boldsymbol{e}}(t) + \boldsymbol{\Phi} \boldsymbol{e}(t), \tag{22}$$

where $\boldsymbol{\varepsilon} := [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_p]^T$ and $\boldsymbol{e} := [e_1, e_2, \ldots, e_p]^T$, $\boldsymbol{\Phi} = \boldsymbol{R}_T \boldsymbol{\Lambda}$, $\boldsymbol{R}_T$, and $\boldsymbol{\Lambda}$ are defined as

$$\boldsymbol{R}_T = diag\left[\frac{1}{\mu_1}\frac{\partial T_1(\hat{e}_1(t))}{\partial \hat{e}_1(t)}, \frac{1}{\mu_2}\frac{\partial T_2(\hat{e}_2(t))}{\partial \hat{e}_2(t)}, \ldots, \frac{1}{\mu_p}\frac{\partial T_p(\hat{e}_p(t))}{\partial \hat{e}_p(t)}\right], \tag{23}$$

$$\boldsymbol{\Lambda} = diag\left[-\frac{\dot{\mu}_1(t)}{\mu_1(t)}, -\frac{\dot{\mu}_2(t)}{\mu_2(t)}, \ldots, -\frac{\dot{\mu}_p(t)}{\mu_p(t)}\right], \tag{24}$$

From a review of ref. [22], an example of error transformation (20) is given by:

$$\varepsilon_j := T_j(\hat{e}_j(t)) = \frac{1}{2p_j} \ln\left(b_j\hat{e}_j(t) + a_j b_j\right) - \frac{1}{2p_j} \ln\left(a_j b_j - a_j\hat{e}_j(t)\right), \tag{25}$$

In the next section, main results of this paper are presented.

3. Main Results

3.1. Controller design

In this section, an adaptive neural feedback linearizing controller is proposed for the general class of type (*m*,*s*) mobile manipulators with a prescribed performance. From (12), the following nonlinear control law is introduced:

$$\boldsymbol{\tau}_a = \hat{\boldsymbol{D}}^{-1}(\boldsymbol{x})\boldsymbol{\tau}_\ell, \tag{26}$$

where $\boldsymbol{\tau}_\ell$ is a control law that is designed in the sequel, and $\hat{\boldsymbol{D}}(\boldsymbol{x})$ shows the best approximation of $\boldsymbol{D}(\boldsymbol{x})$, which is guessed by the expert designer. The substitution of (26) into (12) leads to:

$$\begin{aligned}\ddot{\boldsymbol{y}} = {} & \boldsymbol{\tau}_\ell + L_\xi L_f \boldsymbol{h}(\boldsymbol{x}) + L_\eta L_f \boldsymbol{h}(\boldsymbol{x}) \\ & + (\partial(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{v})/\partial \boldsymbol{q})\, \boldsymbol{H}(\boldsymbol{q}_v)\boldsymbol{v} + \left(\boldsymbol{D}(\boldsymbol{x})\hat{\boldsymbol{D}}^{-1}(\boldsymbol{x}) - \boldsymbol{I}_p\right)\boldsymbol{\tau}_\ell\end{aligned} \tag{27}$$

From (22), one obtains:

$$\dot{\boldsymbol{e}}(t) = \boldsymbol{R}_T^{-1}\dot{\boldsymbol{\varepsilon}}(t) - \boldsymbol{R}_T^{-1}\boldsymbol{\Phi}\boldsymbol{e}(t) \tag{28}$$

Then, the time derivative of (22) and the substitution of (28) gives:

$$\ddot{\boldsymbol{\varepsilon}}(t) = \boldsymbol{R}_T\ddot{\boldsymbol{y}}(t) - \boldsymbol{R}_T\ddot{\boldsymbol{y}}_d(t) + \left(\dot{\boldsymbol{R}}_T + \boldsymbol{\Phi}\right)\boldsymbol{R}_T^{-1}\dot{\boldsymbol{\varepsilon}}(t) - \left(\dot{\boldsymbol{R}}_T + \boldsymbol{\Phi}\right)\boldsymbol{R}_T^{-1}\boldsymbol{\Phi}\boldsymbol{e}(t) + \dot{\boldsymbol{\Phi}}\boldsymbol{e}(t), \tag{29}$$

By replacing (27) in (29), one gets:

$$\ddot{\boldsymbol{\varepsilon}}(t) = \boldsymbol{R}_T\boldsymbol{\tau}_\ell(t) - \boldsymbol{R}_T\ddot{\boldsymbol{y}}_d(t) + \boldsymbol{N}(\boldsymbol{x}_w) + \boldsymbol{R}_T L_\eta L_f \boldsymbol{h}(\boldsymbol{x}), \tag{30}$$

where $\boldsymbol{N}(\boldsymbol{x}_w)$ includes uncertain nonlinearities given by

$$\begin{aligned}\boldsymbol{N}(\boldsymbol{x}_w) := {} & \boldsymbol{R}_T L_\xi L_f \boldsymbol{h}(\boldsymbol{x}) + \left(\dot{\boldsymbol{R}}_T + \boldsymbol{\Phi}\right)\boldsymbol{R}_T^{-1}\dot{\boldsymbol{\varepsilon}}(t) - \left(\dot{\boldsymbol{R}}_T + \boldsymbol{\Phi}\right)\boldsymbol{R}_T^{-1}\boldsymbol{\Phi}\boldsymbol{e}(t) + \dot{\boldsymbol{\Phi}}\boldsymbol{e}(t) \\ & + \boldsymbol{R}_T\left(\partial(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{v})/\partial \boldsymbol{q}\right)\boldsymbol{H}(\boldsymbol{q}_v)\boldsymbol{v} + \boldsymbol{R}_T\left(\boldsymbol{D}(\boldsymbol{x})\hat{\boldsymbol{D}}^{-1}(\boldsymbol{x}) - \boldsymbol{I}_p\right)\boldsymbol{\tau}_\ell\end{aligned} \tag{31}$$

Then, the controller $\boldsymbol{\tau}_\ell$ is proposed as follows:

$$\begin{cases}\boldsymbol{\tau}_\ell(t) = \ddot{\boldsymbol{y}}_d(t) + \boldsymbol{\tau}_{PID}(t) - \boldsymbol{\tau}_{NN}(t) - \boldsymbol{\tau}_{ARC}(t), \\ \boldsymbol{\tau}_{PID} = -\boldsymbol{R}_T^{-1}\boldsymbol{K}_P\,\boldsymbol{\varepsilon}(t) - \boldsymbol{R}_T^{-1}\boldsymbol{K}_I\int_0^t \boldsymbol{\varepsilon}(\tau)\,d\tau - \boldsymbol{R}_T^{-1}\boldsymbol{K}_D\,\dot{\boldsymbol{\varepsilon}}(t), \\ \boldsymbol{\tau}_{NN} = \boldsymbol{R}_T^{-1}\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) + k_n\boldsymbol{R}_T^{-1}\left(\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\right)\boldsymbol{Z}, \\ \boldsymbol{\tau}_{ARC} = \boldsymbol{R}_T^{-1}\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\upsilon\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right),\end{cases} \tag{32}$$

where $\boldsymbol{K}_P, \boldsymbol{K}_I, \boldsymbol{K}_D \in \Re^{p\times p}$ denote PID gains; $\boldsymbol{Z}(t)$ is defined in the sequel; and $\boldsymbol{Y}$ is given by

$$\boldsymbol{Y}(\boldsymbol{e}, \boldsymbol{\mu}) := [\|\boldsymbol{R}_T(\boldsymbol{e}, \boldsymbol{\mu})\|\,, 1], \tag{33}$$

In (32), $\hat{\boldsymbol{W}}$, $\hat{\boldsymbol{V}}$, and $\hat{\boldsymbol{\alpha}}$ are tuned online by the following adaptive laws:

$$\dot{\hat{\boldsymbol{W}}} = \mathrm{Proj}_{\hat{\boldsymbol{W}}}\left(\boldsymbol{\Gamma}_W\left(\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - \boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right)\boldsymbol{Z}^T - \delta_W\boldsymbol{\Gamma}_W\hat{\boldsymbol{W}}\right), \quad \hat{\boldsymbol{W}}(0) \in \Omega_W, \tag{34}$$

$$\dot{\hat{\boldsymbol{V}}} = \mathrm{Proj}_{\hat{\boldsymbol{V}}}\left(\boldsymbol{\Gamma}_V\boldsymbol{x}_w\boldsymbol{Z}^T\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - \delta_V\,\boldsymbol{\Gamma}_V\hat{\boldsymbol{V}}\right), \quad \hat{\boldsymbol{V}}(0) \in \Omega_V, \tag{35}$$

$$\dot{\hat{\boldsymbol{\alpha}}} = \mathrm{Proj}_{\hat{\boldsymbol{\alpha}}}\left(\boldsymbol{\Gamma}_\alpha\,\boldsymbol{Y}^T\|\boldsymbol{Z}(t)\| - \delta_\alpha\boldsymbol{\Gamma}_\alpha\,(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0)\right), \quad \hat{\boldsymbol{\alpha}}(0) \in \Omega_\alpha, \tag{36}$$

In the above equations, the matrices $\boldsymbol{\Gamma}_W \in \Re^{(N_h+1)\times(N_h+1)}$, $\boldsymbol{\Gamma}_V \in \Re^{(N_i+1)\times(N_i+1)}$, and $\boldsymbol{\Gamma}_\alpha \in \Re^{2\times 2}$ represent adaptation gains that adjust the learning rate; $\delta_W, \delta_V, \delta_\alpha \in \Re^+$ denote small positive design parameters; and $\boldsymbol{\alpha}_0$ is *a priori* estimation of $\boldsymbol{\alpha}$. Moreover, $\Omega_{\boldsymbol{W}} := \left\{\hat{\boldsymbol{W}} \in \Re^{(N_h+1)\times N_o} : \mathrm{tr}\left\{\hat{\boldsymbol{W}}^T\hat{\boldsymbol{W}}\right\} \le W_m\right\}$, $\Omega_V := \left\{\hat{\boldsymbol{V}} \in \Re^{(N_i+1)\times N_h} : \mathrm{tr}\left\{\hat{\boldsymbol{V}}^T\hat{\boldsymbol{V}}\right\} \le V_m\right\}$, $\Omega_\alpha := \left\{\hat{\boldsymbol{\alpha}} \in \Re^3 : \left\|\hat{\boldsymbol{\alpha}}\right\| \le \alpha_m\right\}$, and W_m, V_m and α_m are positive constants. By employing the universal approximation property of MLNNs, unknown nonlinearity $\boldsymbol{N}(\boldsymbol{x}_w)$ in (30) may be expressed as:

$$\boldsymbol{N}(\boldsymbol{x}_w) := \boldsymbol{W}^{*T}\boldsymbol{\sigma}(\boldsymbol{V}^{*T}\boldsymbol{x}_w) + \boldsymbol{e}_w(\boldsymbol{x}_w), \tag{37}$$

where $\boldsymbol{x}_w := [1,\ \boldsymbol{x}^T, \boldsymbol{e}^T, \dot{\boldsymbol{e}}^T, \boldsymbol{\tau}_\ell^T, \dot{\boldsymbol{\varepsilon}}^T, \boldsymbol{\mu}^T, \dot{\boldsymbol{\mu}}^T, \ddot{\boldsymbol{\mu}}^T]^T$ and $\|\boldsymbol{e}_w(\boldsymbol{x}_w)\| \le B_w$. From refs. [37, 39], if weight estimation errors are defined as $\tilde{\boldsymbol{W}} = \boldsymbol{W}^* - \hat{\boldsymbol{W}}$ and $\tilde{\boldsymbol{V}} = \boldsymbol{V}^* - \hat{\boldsymbol{V}}$, it is easy to prove that:

$$\begin{aligned} \boldsymbol{W}^{*T}\boldsymbol{\sigma}(\boldsymbol{V}^{*T}\boldsymbol{x}_w) - \hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) + \boldsymbol{e}_w(\boldsymbol{x}_w) &= \tilde{\boldsymbol{W}}^T\left(\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - \boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right) \\ &\quad + \hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\tilde{\boldsymbol{V}}^T\boldsymbol{x}_w + \boldsymbol{n}, \end{aligned} \tag{38}$$

where $\boldsymbol{\sigma}'$ is given by

$$\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) = [\boldsymbol{0}_{N_h\times 1},\ \mathrm{diag}[\sigma_1',\ \sigma_2',\ \ldots,\ \sigma_{N_h}']]^T \in \Re^{(N_h+1)\times N_h} \tag{39}$$

with $\sigma_j' = d\bar{\sigma}(s)/ds|_{s=\hat{V}_{r_j}^T x_w}$, $j = 1, 2, \ldots, N_h$, and

$$\boldsymbol{n} = \tilde{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\boldsymbol{V}^{*T}\boldsymbol{x}_w + \boldsymbol{W}^{*T}O(\tilde{\boldsymbol{V}}^T\boldsymbol{x}_w)^2 + \boldsymbol{\varepsilon}_w(\boldsymbol{x}_w), \tag{40}$$

which is bounded in the following form:

$$\|\boldsymbol{n}\| \le \left\|\boldsymbol{V}^*\right\|_F \left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F \|\boldsymbol{x}_w\| + \left\|\boldsymbol{W}^*\right\|_F \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\| + a, \tag{41}$$

where a is a positive constant. For an activation function in the form of $\bar{\sigma}(x) = 1/(1+e^{-x})$, it is easy to prove that $\sigma_j'(\hat{\boldsymbol{V}}_{r_j}^T\boldsymbol{x}_w) = \bar{\sigma}(\hat{\boldsymbol{V}}_{r_j}^T\boldsymbol{x}_w)[1 - \bar{\sigma}(\hat{\boldsymbol{V}}_{r_j}^T\boldsymbol{x}_w)]$, which gives the following result:

$$\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) = [\boldsymbol{0}_{N_h\times 1},\ \boldsymbol{\Sigma}_{\hat{V}}[\boldsymbol{I}_{N_h} - \boldsymbol{\Sigma}_{\hat{V}}]]^T, \tag{42}$$

where $\boldsymbol{\Sigma}_{\hat{V}}$ is defined as follows:

$$\boldsymbol{\Sigma}_{\hat{V}} = \mathrm{diag}\left[\bar{\sigma}(\hat{\boldsymbol{V}}_{r_1}^T\boldsymbol{x}_w),\ \bar{\sigma}(\hat{\boldsymbol{V}}_{r_2}^T\boldsymbol{x}_w),\ \ldots,\ \bar{\sigma}(\hat{\boldsymbol{V}}_{r_{N_h}}^T\boldsymbol{x}_w)\right]. \tag{43}$$

By replacing (32) into (30), one obtains:

$$\ddot{\boldsymbol{\varepsilon}}(t) = -\boldsymbol{K}_P\,\boldsymbol{\varepsilon}(t) - \boldsymbol{K}_D\,\dot{\boldsymbol{\varepsilon}}(t) - \boldsymbol{K}_I\int_0^t \boldsymbol{\varepsilon}(\tau)\,d\tau + \boldsymbol{u}_R(t), \tag{44}$$

$$\begin{aligned} \boldsymbol{u}_R = &-\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - k_n\left(\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\right)\boldsymbol{Z} \\ &-\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right) + \boldsymbol{N}(\boldsymbol{x}_w) + \boldsymbol{R}_T L_\eta L_f\boldsymbol{h}(\boldsymbol{x}), \end{aligned} \tag{45}$$

whose j-th element gives the following error equation:

$$\ddot{\varepsilon}_j + k_{dj}\,\dot{\varepsilon}_j + k_{pj}\,\varepsilon_j + k_{ij}\int_0^t \varepsilon_j(\tau)\,d\tau = u_{Rj}, \quad j = 1,\ 2,\ \ldots,\ p \tag{46}$$

To drive online update rules, the following filtered error signal is defined for j-th output:

$$z_j(t) = \dot{\varepsilon}_j(t) + \beta_{1j}\varepsilon_j(t) + \beta_{2j}\int_0^t \varepsilon_j(\tau)\,d\tau, \quad j = 1,\ 2,\ \ldots,\ p \tag{47}$$

The factors β_{1j} and β_{2j} are selected such that the transfer function from the output $z_j(t)$ to the input u_{Rj} is SPR:

$$H_j(s) = (s^2 + \beta_{1j}s + \beta_{2j})/(s^3 + k_{dj}\,s^2 + k_{pj}s + k_{ij}) \tag{48}$$

By applying SPR conditions of Lemma 1 to (48), it is easy to show that $H_j(s)$ is SPR if $k_{pj} < k_{dj}^2$, $k_{pj}\,k_{dj} > k_{ij}$, $k_{pj}^2 > 2k_{ij}\,k_{dj}$, $\beta_{1j} = k_{pj}/k_{dj}$, and $\beta_{2j} = k_{ij}/k_{dj}$. According to Lemma 2, there subsequently exist positive definite matrices $\boldsymbol{P}_j$ and $\boldsymbol{Q}_j$ such that $\boldsymbol{A}_j^T\boldsymbol{P}_j + \boldsymbol{P}_j\boldsymbol{A}_j = -\boldsymbol{Q}_j$ and $\boldsymbol{P}_j\boldsymbol{B}_j = \boldsymbol{C}_j^T$ where matrices $\boldsymbol{A}_j$, $\boldsymbol{B}_j$ and $\boldsymbol{C}_j$ are defined by minimal state-space realization of (46) and (47) as $\dot{\boldsymbol{X}}_j = \boldsymbol{A}_j\boldsymbol{X}_j + \boldsymbol{B}_j\boldsymbol{u}_{Rj}$ and $z_j = \boldsymbol{C}_j\boldsymbol{X}_j$ where $\boldsymbol{X}_j = \left[\int_0^t \varepsilon_j(\tau)\,d\tau,\ \ \varepsilon_j,\ \ \dot{\varepsilon}_j\right]^T$ is the state variable and

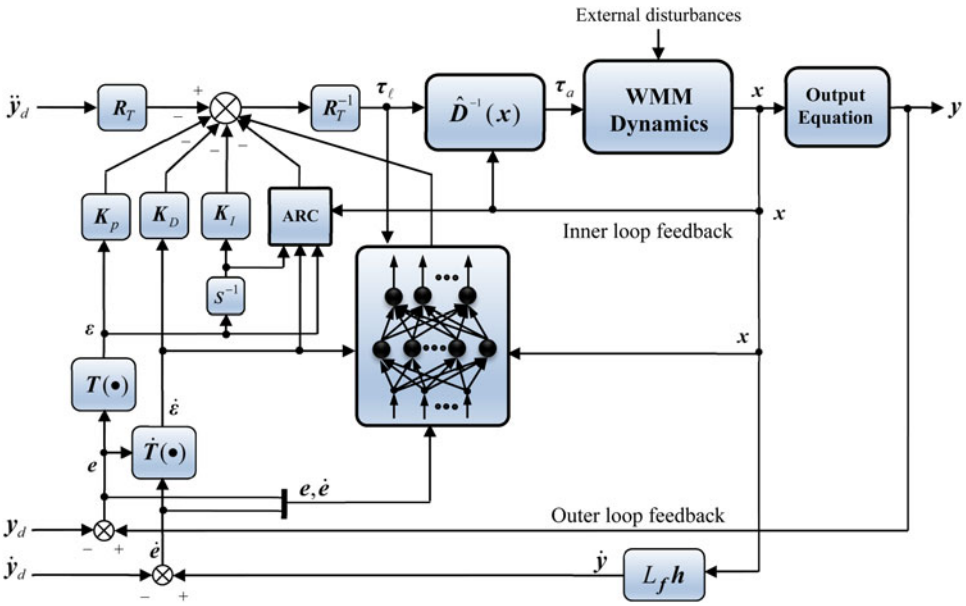

Fig. 3. Block diagram of the proposed WMM control system.

$$\boldsymbol{A}_j = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -k_{ij} & -k_{pj} & -k_{dj} \end{bmatrix}, \quad \boldsymbol{B}_j = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \boldsymbol{C}_j = \begin{bmatrix} \beta_{2j} & \beta_{1j} & 1 \end{bmatrix}. \tag{49}$$

As a result, the state space representation of the entire system is written as:

$$\begin{cases} \dot{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{B}\boldsymbol{u}_R, \\ \boldsymbol{Z} = \boldsymbol{C}\boldsymbol{X} = \dot{\boldsymbol{\varepsilon}} + \boldsymbol{\beta}_1\boldsymbol{\varepsilon} + \boldsymbol{\beta}_2 \int_0^t \boldsymbol{\varepsilon}(\tau)\, d\tau \in \Re^p \end{cases} \tag{50}$$

where $\boldsymbol{A} = diag[\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_p]$, $\boldsymbol{B} = diag[\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_p]$, and $\boldsymbol{C} = diag[\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_p]$ are block diagonal matrices, and $\boldsymbol{X} = [\boldsymbol{X}_1^T, \boldsymbol{X}_2^T, \ldots, \boldsymbol{X}_p^T]^T$, $\boldsymbol{\varepsilon} := [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_p]^T$. Moreover, $\boldsymbol{\beta}_1 := diag[\beta_{11}, \beta_{12}, \ldots, \beta_{1p}]$ and $\boldsymbol{\beta}_2 := diag[\beta_{21}, \beta_{22}, \ldots, \beta_{2p}]$ denote controller gain matrices. Now, the Lyapunov equation is expressed for the entire system as follows:

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q}, \quad \boldsymbol{P}\boldsymbol{B} = \boldsymbol{C}^T, \tag{51}$$

where $\boldsymbol{P} = diag[\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_p]$ and $\boldsymbol{Q} = diag[\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_p]$. Figure 3 illustrates a block diagram of the proposed adaptive NN control system. As shown in this figure, the inner loop feedback controller (26) partially linearizes the WMM nonlinear dynamics. However, since the main parts of unknown nonlinearities could not be cancelled in the inner loop, the effective combination of MLNN and ARC blocks compensates them in the outer loop. According to Fig. 3, the constrained errors, that is, e and $\dot{e}$, are transformed into the unconstrained ones, that is, $\boldsymbol{\varepsilon}$ and $\dot{\boldsymbol{\varepsilon}}$, by the transformation blocks $\boldsymbol{T}$ and $\dot{\boldsymbol{T}}$, which are employed to generate the PID part of the controller, that is, $\boldsymbol{\tau}_{PID}(t)$. Finally, the outer loop controller, that is, $\boldsymbol{\tau}_\ell(t)$ in (32), is generated by the sum of PID, ARC, MLNN, and the desired acceleration $\ddot{\boldsymbol{y}}_d(t)$.

3.2. Stability analysis

In this section, the following theorem is presented to summarize the principal results of the high-performance tracking control of a general class of WMMs:

Theorem 1. *Consider the mathematical motion model of type (m,s) WMMs that are denoted by (1)–(5). Provided that the reference trajectory $\boldsymbol{y}_d(t)$ is selected to be bounded $\forall t > 0$, if Assumptions A1–A4 hold, the gain conditions $k_{pj} < k_{dj}^2$, $k_{pj}\, k_{dj} > k_{ij}$, $k_{pj}^2 > 2k_{ij}\, k_{dj}$, $k_n > 0.5$, $\beta_{1j} = k_{pj}/k_{dj}$, and $\beta_{2j} = k_{ij}/k_{dj}$ are fulfilled, and the initial tracking errors satisfy $-a_j\rho_j(0) < e_j(0) < b_j\rho_j(0)$, $j = 1, 2, \ldots, p$, then the following controller*

$$\begin{cases} \boldsymbol{\tau}_a = \hat{\boldsymbol{D}}^{-1}(\boldsymbol{x})\Big(\ddot{\boldsymbol{y}}_d(t) - \boldsymbol{R}_T^{-1}\Big(\boldsymbol{u}_{PID}(t) + \hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) + k_n\boldsymbol{h}\boldsymbol{Z} + \boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right)\Big)\Big), \\ \boldsymbol{h} = \left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2, \\ \boldsymbol{u}_{PID} = \boldsymbol{K}_P\,\boldsymbol{\varepsilon}(t) + \boldsymbol{K}_I\int_0^t\boldsymbol{\varepsilon}(\tau)\,d\tau + \boldsymbol{K}_D\,\dot{\boldsymbol{\varepsilon}}(t), \\ \dot{\hat{\boldsymbol{W}}} = \mathrm{Proj}_{\hat{\boldsymbol{W}}}\left(\boldsymbol{\Gamma}_W\left(\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - \boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right)\boldsymbol{Z}^T - \delta_W\boldsymbol{\Gamma}_W\,\hat{\boldsymbol{W}}\right), \quad \hat{\boldsymbol{W}}(0) \in \Omega_W \\ \dot{\hat{\boldsymbol{V}}} = \mathrm{Proj}_{\hat{V}}\left(\boldsymbol{\Gamma}_V\boldsymbol{x}_w\boldsymbol{Z}^T\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) - \delta_V\,\boldsymbol{\Gamma}_V\hat{\boldsymbol{V}}\right), \quad \hat{\boldsymbol{V}}(0) \in \Omega_V \\ \dot{\hat{\boldsymbol{\alpha}}} = \mathrm{Proj}_{\hat{\boldsymbol{\alpha}}}\left(\boldsymbol{\Gamma}_\alpha\,\boldsymbol{Y}^T\,\|\boldsymbol{Z}(t)\| - \delta_\alpha\boldsymbol{\Gamma}_\alpha\,(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0)\right), \quad \hat{\boldsymbol{\alpha}}(0) \in \Omega_\alpha \end{cases} \tag{52}$$

ensures that: (i) all the signals in the closed-loop control system are bounded; (ii) the tracking errors $\int_0^t \varepsilon_j(\tau)\,d\tau$, ε_j, *and* $\dot{\varepsilon}_j$, $j = 1, 2, \ldots, p$, *are UUB and* ε_j *exponentially converges to a small ball containing the origin; and (iii) the tracking errors* $e_j(t)$, $j = 1, 2, \ldots, p$, *are UUB and exponentially converge to neighborhoods of the zero with prescribed transient and steady-state performance specifications.*

Proof. Consider the following Lyapunov function:

$$V = 0.5\left(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X} + \mathrm{tr}\left\{\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma}_W^{-1}\tilde{\boldsymbol{W}}\right\} + \mathrm{tr}\left\{\tilde{\boldsymbol{V}}^T\boldsymbol{\Gamma}_V^{-1}\tilde{\boldsymbol{V}}\right\} + \tilde{\boldsymbol{\alpha}}^T\boldsymbol{\Gamma}_\alpha^{-1}\tilde{\boldsymbol{\alpha}}\right), \tag{53}$$

where $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}}$. By recalling (45), (50), and (51), the time derivative of (53) yields:

$$\begin{aligned} \dot{V} &= 0.5\boldsymbol{X}^T\left(\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}\right)\boldsymbol{X} + \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{B}\boldsymbol{u}_R - \mathrm{tr}\left\{\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma}_W^{-1}\dot{\hat{\boldsymbol{W}}}\right\} - \mathrm{tr}\left\{\tilde{\boldsymbol{V}}^T\boldsymbol{\Gamma}_V^{-1}\dot{\hat{\boldsymbol{V}}}\right\} - \tilde{\boldsymbol{\alpha}}^T\boldsymbol{\Gamma}_\alpha^{-1}\dot{\hat{\boldsymbol{\alpha}}} \\ &= -0.5\boldsymbol{X}^T\boldsymbol{Q}\boldsymbol{X} - \boldsymbol{Z}^T\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right) + \boldsymbol{Z}^T\left(\boldsymbol{W}^T\boldsymbol{\sigma}(\boldsymbol{V}^T\boldsymbol{x}_w) - \hat{\boldsymbol{W}}^T\boldsymbol{\sigma}(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w) + \boldsymbol{e}_w\right) \\ &\quad -k_n\left(\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\right)\|\boldsymbol{Z}\|^2 + \boldsymbol{Z}^T\boldsymbol{R}_T\boldsymbol{L}_\eta\boldsymbol{L}_f\boldsymbol{h}(\boldsymbol{x}) \\ &\quad -\mathrm{tr}\left\{\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma}_W^{-1}\dot{\hat{\boldsymbol{W}}}\right\} - \mathrm{tr}\left\{\tilde{\boldsymbol{V}}^T\boldsymbol{\Gamma}_V^{-1}\dot{\hat{\boldsymbol{V}}}\right\} - \tilde{\boldsymbol{\alpha}}^T\boldsymbol{\Gamma}_\alpha^{-1}\dot{\hat{\boldsymbol{\alpha}}} \end{aligned} \tag{54}$$

Considering the adaptation rules in (34)–(36), applying ((38) and recalling the properties of the projection-type adaptive laws from refs. [36, 39], (54) can be written as:

$$\begin{aligned} \dot{V} \le &-0.5\boldsymbol{X}^T\boldsymbol{Q}\boldsymbol{X} - \boldsymbol{Z}^T\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right) + Z^T\left(\boldsymbol{n} + \boldsymbol{R}_T\boldsymbol{L}_\eta\boldsymbol{L}_f\boldsymbol{h}(\boldsymbol{x})\right) \\ &-k_n\left(\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\right)\|\boldsymbol{Z}\|^2 + \delta_W\mathrm{tr}\left\{\tilde{\boldsymbol{W}}^T\,\hat{\boldsymbol{W}}\right\} \\ &+\delta_V\mathrm{tr}\left\{\tilde{\boldsymbol{V}}^T\,\hat{\boldsymbol{V}}\right\} - \tilde{\boldsymbol{\alpha}}^T\boldsymbol{\Gamma}_\alpha^{-1}\mathrm{Proj}_{\hat{\alpha}}\left(\boldsymbol{\Gamma}_\alpha\,\boldsymbol{Y}^T\,\|\boldsymbol{Z}(t)\|\right) + \delta_\alpha\tilde{\boldsymbol{\alpha}}^T(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0) \end{aligned} \tag{55}$$

By recalling (40), Assumptions A3 and A4, the structural properties of WMMs[29] and the fact $\left\|L_\eta L_f\boldsymbol{h}(x)\right\| \le \beta$, where β is an unknown positive constant, it is easy to prove that

$$\begin{aligned} \boldsymbol{Z}^T\boldsymbol{n} + \boldsymbol{Z}^T\boldsymbol{R}_T\boldsymbol{L}_\eta\boldsymbol{L}_f\boldsymbol{h}(\boldsymbol{x}) \le &0.5\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2\|\boldsymbol{Z}\|^2 + 0.5\left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\|\boldsymbol{Z}\|^2 \\ &+\|\boldsymbol{Z}\|\,\boldsymbol{Y}\boldsymbol{\alpha} + 0.5\left\|\boldsymbol{W}^*\right\|_F^2 + 0.5\left\|\boldsymbol{V}^*\right\|_F^2 \end{aligned} \tag{56}$$

where $\boldsymbol{Y} := [\|\boldsymbol{R}_T\|\,,\,1]$. By considering (56), $\boldsymbol{\alpha} = \tilde{\boldsymbol{\alpha}} + \hat{\boldsymbol{\alpha}}$, taking the following into account

$$\delta_W\mathrm{tr}\{\tilde{\boldsymbol{W}}^T\hat{\boldsymbol{W}}\} \le -(1 - 0.5/\kappa^2)\delta_W\left\|\tilde{\boldsymbol{W}}\right\|_F^2 + 0.5\kappa^2\delta_W\left\|\boldsymbol{W}^*\right\|_F^2, \tag{57}$$

$$\delta_V\mathrm{tr}\left\{\tilde{\boldsymbol{V}}^T\hat{\boldsymbol{V}}\right\} \le -(1 - 0.5/\kappa^2)\delta_V\left\|\tilde{\boldsymbol{V}}\right\|_F^2 + 0.5\kappa^2\delta_V\left\|\boldsymbol{V}^*\right\|_F^2, \tag{58}$$

$$\delta_\alpha\tilde{\boldsymbol{\alpha}}^T(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0) \le -\delta_\alpha(1 - 0.5/\kappa^2)\,\|\tilde{\boldsymbol{\alpha}}\|^2 + 0.5\delta_\alpha\kappa^2\,\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_0\|^2, \tag{59}$$

and further simplifications, one obtains:

$$\begin{aligned}\dot{V} \leq & -0.5\lambda_{\min}(\boldsymbol{Q})\,\|\boldsymbol{X}\|^2 + \|\boldsymbol{Z}\|\,\boldsymbol{Y}\hat{\boldsymbol{\alpha}} - \boldsymbol{Z}^T\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right)\\ & -(k_n-0.5)\left(\left\|\hat{\boldsymbol{W}}^T\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\right\|_F^2\|\boldsymbol{x}_w\|^2 + \left\|\boldsymbol{\sigma}'(\hat{\boldsymbol{V}}^T\boldsymbol{x}_w)\hat{\boldsymbol{V}}^T\boldsymbol{x}_w\right\|^2\right)\|\boldsymbol{Z}\|^2\\ & +\|\boldsymbol{Z}\|\,\boldsymbol{Y}\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}^T\boldsymbol{\Gamma}_\alpha^{-1}\mathrm{Proj}_{\hat{\boldsymbol{\alpha}}}\left(\boldsymbol{\Gamma}_\alpha\,\boldsymbol{Y}^T\,\|\boldsymbol{Z}(t)\|\right) - (1-0.5/\kappa^2)\delta_W\left\|\tilde{\boldsymbol{W}}\right\|_F^2\\ & -(1-0.5/\kappa^2)\delta_V\left\|\tilde{\boldsymbol{V}}\right\|_F^2 - \delta_\alpha(1-0.5/\kappa^2)\,\|\tilde{\boldsymbol{\alpha}}\|^2 + 0.5\kappa^2\delta_W\left\|\boldsymbol{W}^*\right\|_F^2\\ & +0.5\kappa^2\delta_V\left\|\boldsymbol{V}^*\right\|_F^2 + 0.5\left\|\boldsymbol{W}^*\right\|_F^2 + 0.5\left\|\boldsymbol{V}^*\right\|_F^2 + 0.5\delta_\alpha\kappa^2\,\|\boldsymbol{\alpha}-\boldsymbol{\alpha}_0\|^2\end{aligned} \tag{60}$$

By recalling the fact that $\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\|\boldsymbol{Z}\| - \boldsymbol{Z}^T\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\,\boldsymbol{Tanh}\left(\nu\boldsymbol{Y}\hat{\boldsymbol{\alpha}}\boldsymbol{Z}/\gamma_t\right) \leq p\gamma_t, \forall\,\mu_t > 0$ from Lemma 3, choosing $k_n > 0.5$, and considering the projection technique properties,[36,39] one obtains the following inequality:

$$\dot{V} \leq -0.5\lambda_{\min}(\boldsymbol{Q})\,\|\boldsymbol{X}\|^2 - c_w\sum_{j=1}^{N_h+1}\sum_{k=1}^{N_o}\left|\tilde{w}_{jk}\right|^2 - c_v\sum_{j=1}^{N_i+1}\sum_{k=1}^{N_h}\left|\tilde{v}_{jk}\right|^2 - c_\alpha\,\|\tilde{\boldsymbol{\alpha}}\|^2 + p\gamma_t + \rho, \tag{61}$$

where $c_w = (1-0.5/\kappa^2)\delta_W$, $c_v = (1-0.5/\kappa^2)\delta_V$ and $c_\alpha = \delta_\alpha(1-0.5/\kappa^2)$ and

$$\rho := 0.5\left(\kappa^2\delta_W + 1\right)\left\|\boldsymbol{W}^*\right\|_F^2 + 0.5\left(\kappa^2\delta_V + 1\right)\left\|\boldsymbol{V}^*\right\|_F^2 + 0.5\delta_\alpha\kappa^2\,\|\boldsymbol{\alpha}-\boldsymbol{\alpha}_0\|^2, \tag{62}$$

which can be written as follows:

$$\dot{V}(t) \leq -\boldsymbol{c}_m\,\|\boldsymbol{E}(t)\|^2 + p\gamma_t + \rho, \tag{63}$$

where $\boldsymbol{c}_m := \min\{0.5\lambda_{\min}(\boldsymbol{Q})\,1, c_w, c_v, c_\alpha\}$, $\boldsymbol{E} := [\boldsymbol{X}^T, \tilde{w}_{11}, \ldots, \tilde{w}_{(N_h+1)N_o}, \tilde{v}_{11}, \ldots, \tilde{v}_{(N_i+1)N_h}, \tilde{\boldsymbol{\alpha}}^T]^T$. On the contrary, the Lyapunov function (53) is bounded as follows:

$$\lambda_{\min}(\boldsymbol{A})\,\|\boldsymbol{E}(t)\|^2 \leq V(t) \leq \lambda_{\max}(\boldsymbol{A})\,\|\boldsymbol{E}(t)\|^2, \tag{64}$$

where $\boldsymbol{A} := 0.5\,blockdiag\{\boldsymbol{P}, \boldsymbol{\Gamma}_W^{-1}, \boldsymbol{\Gamma}_V^{-1}, \boldsymbol{\Gamma}_\alpha^{-1}\}$. From (64), inequality (63) becomes

$$\dot{V}(t) + \boldsymbol{c}_m V(t)/\lambda_{\max}(\boldsymbol{A}) \leq p\gamma_t + \rho \tag{65}$$

Solving the differential inequality (65) yields

$$0 \leq V(t) \leq V(0)e^{-\boldsymbol{c}_m t/\lambda_{\max}(A)} + (\lambda_{\max}(\boldsymbol{A})(p\gamma_t+\rho)/\boldsymbol{c}_m)\left(1 - e^{-c_m t/\lambda_{\max}(\boldsymbol{A})}\right), \quad \forall t \in [0,\ \infty) \tag{66}$$

From (66), it is clear that the Lyapunov function is bounded as follows:

$$V(t) \leq \max\{V(0),\ \lambda_{\max}(\boldsymbol{A})(p\gamma_t+\rho)/\boldsymbol{c}_m\}, \quad \forall t \geq 0 \tag{67}$$

which together with (64) gives the following result:

$$\|\boldsymbol{E}(t)\| \leq \left(\max\{V(0),\ \lambda_{\max}(\boldsymbol{A})(p\gamma_t+\rho)/\boldsymbol{c}_m\}/\lambda_{\min}(\boldsymbol{A})\right)^{1/2} \tag{68}$$

Therefore, inequality (66) implies that $\dot{V}(t)$ is strictly negative outside the compact set

$$S_E = \left\{\boldsymbol{E}(t)\,\middle|\,\|\boldsymbol{E}(t)\| \leq \left(\max\{V(0),\ \lambda_{\max}(\boldsymbol{A})(p\gamma_t+\rho)/\boldsymbol{c}_m\}/\lambda_{\min}(\boldsymbol{A})\right)^{1/2}\right\} \tag{69}$$

Thus, $\|\boldsymbol{E}(t)\|$ decreases with time whenever $\boldsymbol{E}(t)$ is outside the compact set S_E, and hence $\|\boldsymbol{E}(t)\|$ is UUB. This discussion implies that tracking errors are UUB and exponentially converge to a small area containing the origin and $\boldsymbol{\varepsilon}$, $\int_0^t \boldsymbol{\varepsilon}(\tau)\,d\tau$, $\dot{\boldsymbol{\varepsilon}}$, $\tilde{w}_{11}, \ldots, \tilde{w}_{(N_h+1)N_o}, \tilde{v}_{11}, \ldots, \tilde{v}_{(N_i+1)N_h}, \hat{\boldsymbol{\alpha}} \in L_\infty$. Since $-a_j\mu_j(0) < e_j(0) < b_j\mu_j(0)$, $j = 1, 2, \ldots, p$, $\varepsilon(t) \in L_\infty$, $\forall t \geq 0$, and $\varepsilon(t)$ converge to a small

region around zero, then the tracking errors $e_j(t)$, $j = 1, 2, \ldots, p$, satisfy (17) and converge to small regions around zero with prescribed performance specifications. This completes the proof.

3.3. Parameter tuning guidelines

One of the main challenges in achieving a desirable performance of the proposed control system is parameter tuning. The adjustment of controller parameters is usually carried out by the trial-and-error method based on some rules, which may be extracted from the stability analysis and theoretical developments. The following guidelines are suggested for parameter tuning:

(i) The PPF parameters a_j, b_j, and μ_{j0}, $j = 1, 2, \ldots, p$, are initially chosen large enough to ensure that $-a_j\mu_{j0} < e_j(0) < b_j\mu_{j0}$, $j = 1, 2, \ldots, p$, $\forall t > 0$.
(ii) The convergence speed λ_j is initially set to a small value and then increased gradually. The parameters $\mu_{j\infty}$, $j = 1, 2, \ldots, p$, determine the bounds of the steady-state errors, which are adjusted to large values and then reduced.
(iii) The PID gains k_{pj}, k_{ij}, and k_{dj}, $j = 1, 2, \ldots, p$, should be tuned such that $k_{pj} < k_{dj}^2$, $k_{pj}\, k_{dj} > k_{ij}$, and $k_{pj}^2 > 2k_{ij}\, k_{dj}$, which ensures that $H_j(s)$ in (48) is SPR.
(iv) A larger adaptive gain $\boldsymbol{\Gamma}_\alpha$ in (36) enhances parameter adaptation, which, in turn, improves the robustness and tracking accuracy at the expense of a high control action and more chattering. However, the very large values of $\boldsymbol{\Gamma}_\alpha$ may lead to instability.
(v) The parameter γ_t in the adaptive robust controller $\boldsymbol{\tau}_{ARC}$ in (32) is used to make a trade-off between final tracking accuracy and control signal smoothness. High values of γ_t make control signals smoother at the expense of a larger ultimate bound and less tracking accuracy based on (68).
(vi) The hidden layer neuron number is initially set to be small. Then, the neuron number is gradually increased to get a desirable performance. When the tracking performance is not improved anymore, the increment of neuron number will be stopped. It should be noted that a very large number of neurons might lead to more computational complexity, and tracking performance may be reduced due to an overestimation.
(vii) The larger values of $\boldsymbol{\Gamma}_W$ and $\boldsymbol{\Gamma}_V$ in NN adaptation laws (34) and (35) enhance the learning rate. However, high adaptation gains may lead to instability.
(viii) The small values of δ_W, δ_V and δ_α reduce ρ in (62) that increases the final tracking accuracy at the expense of the smaller robustness of adaptive rules (34)–(36). Therefore, a trade-off may be conducted between robustness and final tracking accuracy. The initial value of the weight matrix $\hat{\boldsymbol{W}}$ is set to zero and $\hat{\boldsymbol{V}}(0)$ is randomly chosen.

3.4. Discussion

In the previous sections, a tracking controller is presented for all types of WMMs based on the approximation-based feedback linearization technique. From a comparative viewpoint, most of the available research works, including refs. [1–20], only concentrated on tracking control design for a special type of WMMs, such as type (2,0) WMMs, that is, a mobile manipulator with a differential drive mobile platform, or a type (3,0) WMM, that is, a mobile manipulator with an omnidirectional mobile platform,[12] while this paper addresses the tracking problem of a general class of n-link WMMs with a type (m,s) mobile platform. As frequently reported in the literature, approximation-based techniques, including robust and adaptive laws[2–9] and Fuzzy and neural networks,[10–16] have been widely used to deal with parametric and non-parametric uncertainties. However, it is very difficult to acquire a desired transient performance due to parameter adaptation and robust control action in previous works. In contrast to refs. [2–20], a neural adaptive robust tracking controller with a guaranteed prescribed transient and steady-state performance is designed for WMMs for the first time by employing the PPF technique.[21,22] This design strategy helps the user to freely adjust the maximum overshoot/undershoot, convergence rate, and final tracking accuracy in advance. The tracking performance of the proposed controller is illustrated in the next section by numerical simulations.

4. Numerical Simulations

In this section, the tracking performance of the proposed controller is evaluated for a type (2,0) WMM illustrated by Fig. 4. The generalized coordinates of the WMM are shown by $\boldsymbol{q} = [x_o, y_o, \varphi, \theta_1, \theta_2]^T$.

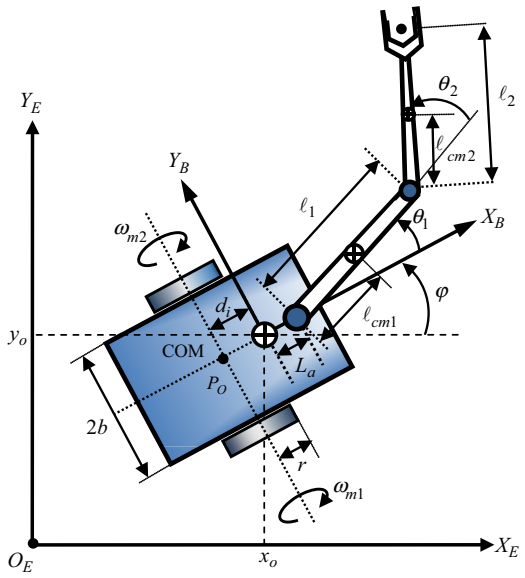

Fig. 4. Planar configuration of a type (2,0) two-link WMM.

The motion model of the robot, its parameter definitions, and their attributed values are obtained from ref. [40]. It is assumed that the mobile manipulator is equipped with a localization and mapping algorithm to estimate its posture in the environment.[41,42]

The controller parameters are chosen based on gain conditions of Theorem 1 and trial and error. In this simulation, parameters $\boldsymbol{K}_P = \text{diag}[50, 50, 50, 50]$, $\boldsymbol{K}_D = \text{diag}[50, 50, 50, 50]$, $\boldsymbol{K}_I = \text{diag}[4, 4, 4, 4]$, $\boldsymbol{\beta}_1 = \text{diag}[1, 1, 1, 1]$, $\boldsymbol{\beta}_2 = 0.08\text{diag}[1, 1, 1, 1]$, $a_1 = a_2 = b_1 = b_2 = 4$, $a_3 = a_4 = b_3 = b_4 = 2.5$, $p_j = 2$, $j = 1, \ldots, 4$, $\mu_{j0} = 1$, $j = 1, \ldots, 4$, $\mu_{j\infty} = 0.05$, $j = 1, \ldots, 4$, and $\lambda_j = 0.2$, $j = 1, \ldots, 4$ are selected to demonstrate controller performance. An MLNN with parameters $N_h = 6$, $N_i = 15$, $N_o = 4$, $\boldsymbol{\Gamma}_W = 2\boldsymbol{I}_4$, $\boldsymbol{\Gamma}_V = 2\boldsymbol{I}_4$, $\delta_W = \delta_V = 0.005$, and $k_n = 1$ are used. For simplicity, the projection is not applied to this simulation. The adaptive robust controller parameters are also set to $\boldsymbol{\Gamma}_\alpha = 0.25\text{diag}[1, 10^{-6}]$, $\delta_\alpha = 0.25$, and $\gamma_t = 200$. The signal $\boldsymbol{\tau}_{di}(t) = 0.25\sin(0.05t)[1, 1, 1, 1]^T + 0.8\text{sign}(v(t))$ is used to simulate external disturbance. The torque signals are saturated within $|\tau_{aj}| \leq 20\,Nm$, $j = 1, \ldots, 4$ to simulate actuator saturation. The output vector is chosen as $\boldsymbol{y} = [(\boldsymbol{\zeta} + \boldsymbol{R}^T(\varphi)\boldsymbol{d})^T, \boldsymbol{q}_m^T]^T$ for a type (2,0) WMM such that

$$\boldsymbol{d} = \left[x_b + \ell_{d1}\cos(\theta_{d1}) + \ell_{d2}\cos(\theta_{d1} + \theta_{d2}),\quad y_b + \ell_{d1}\sin(\theta_{d1}) + \ell_{d2}\sin(\theta_{d1} + \theta_{d2})\right]^T, \tag{70}$$

where $\ell_{d1} = 0.514$ m, $\ell_{d2} = 0.362$ m, and $\theta_{d1} = \theta_{d2} = \pi/4$. The initial posture of the robot is set to $\boldsymbol{q}_1(0) = [2, 0, 0, -\pi/4, -\pi/4]^T$. Figures 5–9 demonstrate the tracking performance of a desired circular trajectory. Figure 5 shows the x–y plot of WMM trajectory and its desired trajectory. It illustrates that the mobile manipulator end effector successfully tracks the desired trajectory.

The time evolution of the constrained tracking errors along with their prescribed performance bounds are demonstrated by Fig. 6. It is seen that the tracking errors lie within PPFs to satisfy our prescribed overshoot, convergence rate, and ultimate accuracy in this simulation. Figure 7 also shows that transformed errors are bounded and converge to neighborhoods of zero in the presence of model uncertainties and external disturbances. The control signals are also illustrated by Fig. 8. Figure 9 demonstrates Frobenius norms of estimated weight matrices and parameter estimates. It should be noted that tracking performance could be improved by better tuning of control parameters.

In order to further evaluate the proposed controller performance, a comparative simulation is carried out. For this purpose, a passivity-based adaptive robust controller is designed as follows, inspired from refs. [37,43]:

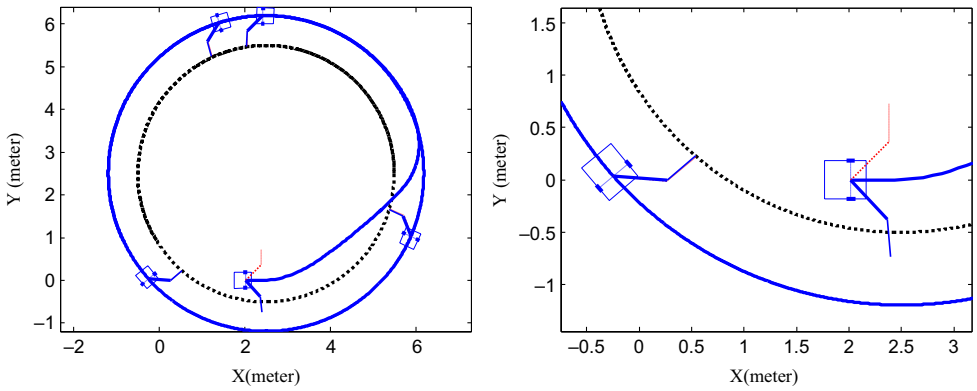

Fig. 5. x–y plot of the WMM and desired trajectories (left) and its magnified view (right).

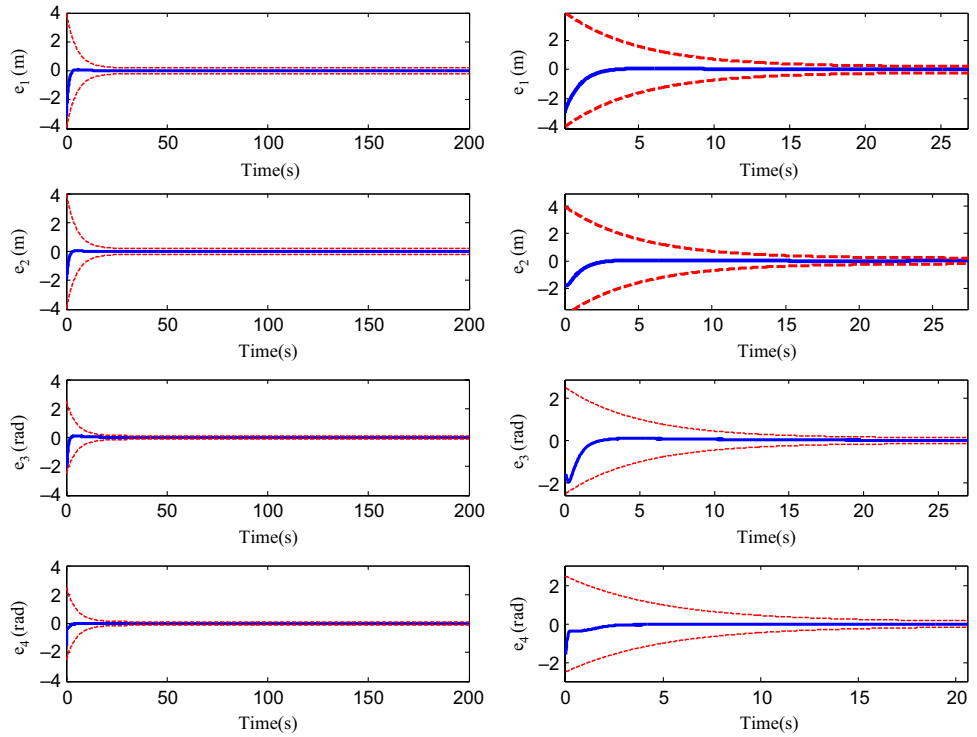

Fig. 6. Time evolution of constrained tracking errors (left) and their magnified views (right).

$$\begin{cases} \boldsymbol{\tau}_a(t) = \boldsymbol{B}^{-1}(\boldsymbol{q})\left(-\boldsymbol{K}\boldsymbol{s}(t) - (\boldsymbol{Y}\hat{\boldsymbol{\theta}})^2\boldsymbol{s}(t)/(\boldsymbol{Y}\hat{\boldsymbol{\theta}}\,\|\boldsymbol{s}(t)\| + \gamma_t)\right), \\ \dot{\hat{\boldsymbol{\theta}}}(t) = \boldsymbol{\Gamma}_\theta \boldsymbol{Y}^T\,\|\boldsymbol{s}(t)\| - \boldsymbol{\sigma}\hat{\boldsymbol{\theta}}(t), \\ \boldsymbol{Y} = \left[1,\ \|\dot{\boldsymbol{y}}_r(t)\|,\ \|\dot{\boldsymbol{y}}_r(t)\|^2,\ \|\ddot{\boldsymbol{y}}_r(t)\|\right], \\ \boldsymbol{s}(t) = \dot{\boldsymbol{y}}(t) - \dot{\boldsymbol{y}}_r(t) = \dot{\boldsymbol{e}}(t) + \boldsymbol{\Lambda}_0\boldsymbol{e}(t) + \boldsymbol{\Lambda}_1\int_0^t \boldsymbol{e}(\tau)d\tau \\ \dot{\boldsymbol{y}}_r(t) = \dot{\boldsymbol{y}}_d(t) - \boldsymbol{\Lambda}_0\boldsymbol{e}(t) - \boldsymbol{\Lambda}_1\int_0^t \boldsymbol{e}(\tau)d\tau, \end{cases} \tag{71}$$

where $\boldsymbol{B}(\boldsymbol{q}) = \boldsymbol{J}^{-T}(\boldsymbol{q})\boldsymbol{B}_a(\boldsymbol{q})$, $\boldsymbol{J}(\boldsymbol{q})$, and $\boldsymbol{B}_a(\boldsymbol{q})$ are defined in Section 2; $\boldsymbol{\Lambda}_0 = \text{diag}[10, 10, 0.5, 0.5]$, $\boldsymbol{\Lambda}_1 = \text{diag}[0.1, 0.1, 0.05, 0.05]$, and $\boldsymbol{K} = \text{diag}[50, 50, 2.5, 2.5]$ are positive gain matrices. In addition, $\gamma_t = 1000$, $\boldsymbol{\Gamma}_\theta = 0.005\text{diag}[5, 1, 0.01, 0.1]$, and $\boldsymbol{\sigma} = 0.5\boldsymbol{\Gamma}_\theta$. For an impartial comparison, both controllers are tuned carefully to obtain their best performance. The parameters of the proposed controller are similar to the previous simulation except that $\mu_{j\infty} = 0.025$ and $\lambda_j = 0.3, j = 1, \ldots, 4$.

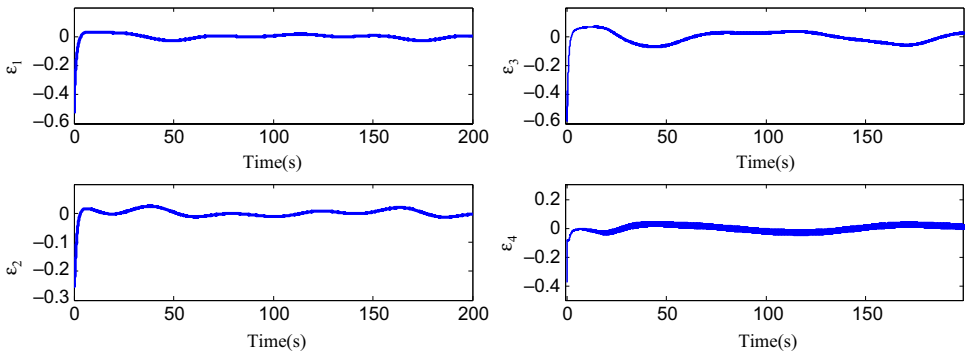

Fig. 7. Time evolution of unconstrained tracking errors.

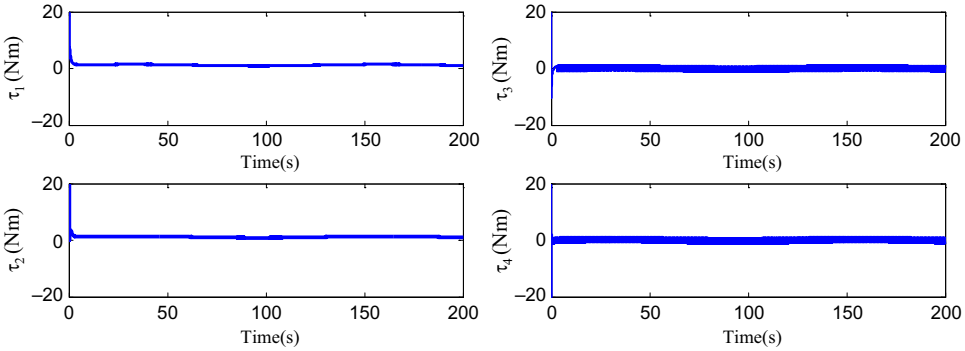

Fig. 8. Time evolution of generated control signals.

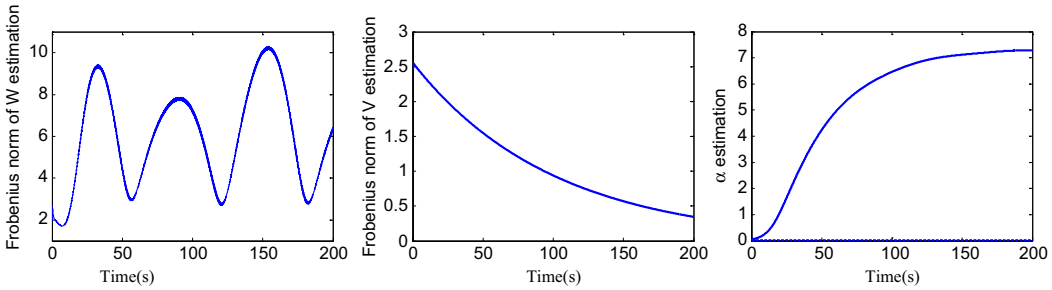

Fig. 9. The Frobenius norms of NN weights and parameter estimates.

Other simulation conditions are intact. The simulation results are illustrated by Fig. 10. As shown in this figure, the proposed control system demonstrates a better performance in both transient and steady-state responses. More simulations with different adjustments verify the performance of the proposed control system.

5. Concluding Remarks

In this paper, an adaptive NN tracking controller is developed based on feedback linearization technique. By introducing a proper set of output equations, an input–output model of the general type (m,s) WMMs is presented. Then, the prescribed performance technique is utilized to satisfy the predefined transient and steady-state response specifications of the tracking control system. The projection-type MLNNs and adaptive techniques are used to compensate all types of model uncertainties. The Lyapunov direct method is used to show that the tracking errors converge to a neighborhood of the origin with a prescribed performance. Numerical simulation results are presented to show the effective performance of the tracking control system for WMMs.

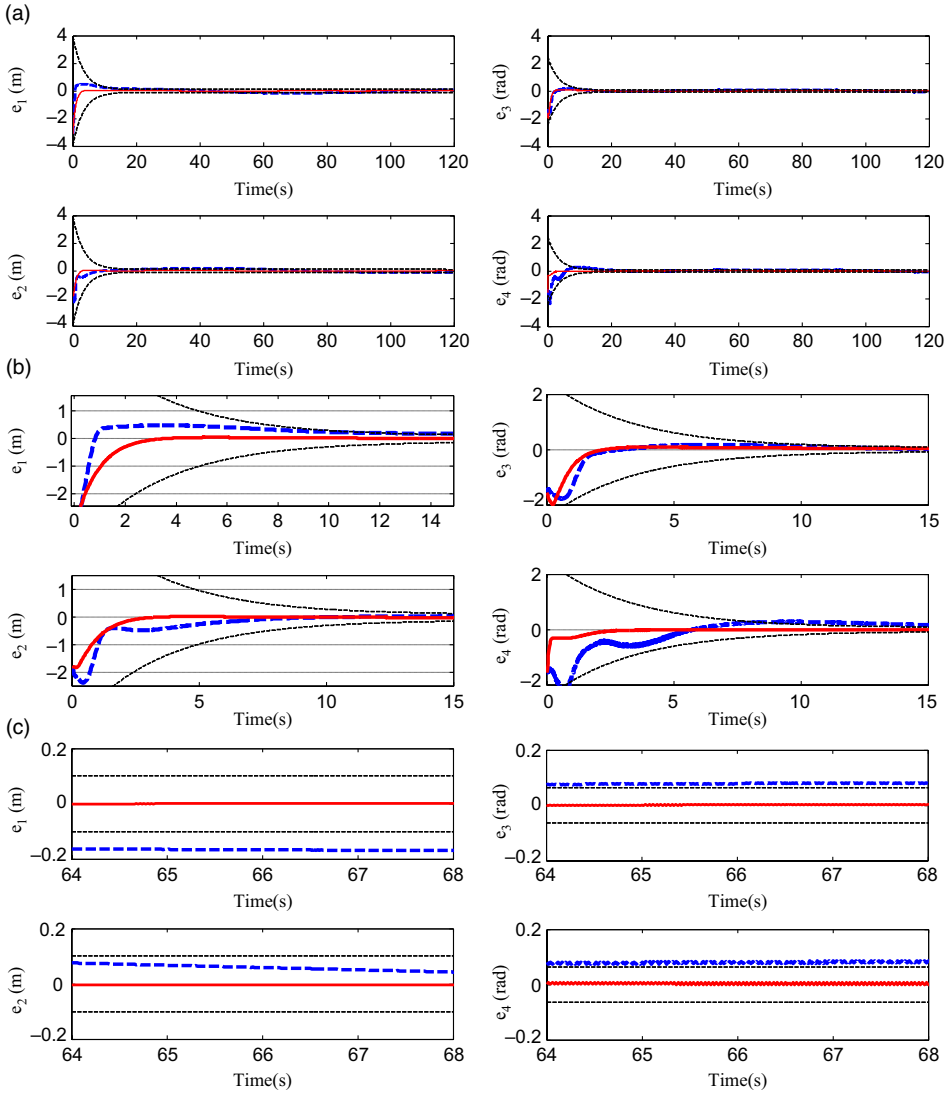

Fig. 10. Comparative simulation results of the proposed controller (red solid signal) and the controller (71) (blue dashed-line signal): (a) the whole error signal, (b) the transient response, and (c) steady-state response.

Future researches can be devoted to the design of an output feedback controller for type (m,s) WMMs as well as for the experimental evaluation of the proposed control system on a real large-scale mobile manipulator. It should be noted that the dynamic model of a small-scale mobile manipulator is not considerable in practice, and the kinematic controllers are sufficient for their motion control according to the literature and to the best of our knowledge. In contrast, the effects of inertia, Coriolis, centripetal and gravity forces, and torques in the large-scale mobile manipulators are significant in practice. Therefore, the proposed controller definitely shows its effectiveness on large-scale mobile manipulators since the controller is capable of compensating the dynamic model nonlinearities due to the neural adaptive robust control design that would preserve the prescribed performance.

Acknowledgments

The authors would like to thank all anonymous reviewers and the Associate Editor for their helpful comments and suggestions for the improvement of the original manuscript. The authors would like to thank all financial support from Najafabad Branch, Islamic Azad University under grant number 1509512080049 under the research project entitled "Coordinated Control of Autonomous Vehicles and Unmanned Cars without Velocity Sensors by Using Adaptive Neural Networks."

References

1. Y. Yamamoto, Control and Coordination of Locomotion and Manipulation of Wheeled Mobile Manipulators. *Ph.D. Thesis* (University of Pennsylvania, 1994).
2. W. Dong, "On trajectory and force tracking control of constrained mobile manipulators with parameter uncertainty," *Automatica* **38**(8), 1475–1484 (2002).
3. A. Mazur, "Hybrid adaptive control laws solving a path-following problem for non-holonomic mobile manipulators," *Int. J. Control* **77**(15), 1297–1306 (2004).
4. Z. Li, S. S. Ge and Z. Wang, "Robust adaptive control of coordinated multiple mobile manipulators," *Mechatronics* **18**(5–6), 239–250 (2008).
5. Z. Li, J. Li and Y. Kang, "Adaptive robust coordinated control of multiple mobile manipulators interacting with rigid environments," *Automatica* **46**(12), 2028–2034 (2010).
6. G. Zhong, Y. Kobayashi, Y. Hoshino and T. Emaru, "System modeling and tracking control of mobile manipulator subjected to dynamic interaction and uncertainty," *Nonlinear Dyn.* **73**(1), 167–182 (2013).
7. J. Peng, J. Yu and J. Wang, "Robust adaptive tracking control for nonholonomic mobile manipulator with uncertainties," *ISA Trans.* **53**(4), 1035–1043 (2014).
8. M. Galicki, "An adaptive non-linear constraint control of mobile manipulators," *Mech. Mach. Theory* **88**, 63–85 (2015).
9. Z. Li, S. S. Ge, M. Adams and W. S. Vijesoma, "Adaptive robust output-feedback motion/force control of electrically driven nonholonomic mobile manipulators," *IEEE Trans. Control Syst. Technol.* **16**(6), 1308–1315 (2008).
10. S. Lin and A. A. Goldenberg, "Neural-network control of mobile manipulators," *IEEE Trans. Neural Networks* **12**(5), 1121–1133 (2001).
11. Z. Li and W. Chen, "Adaptive neural-fuzzy control of uncertain constrained multiple coordinated nonholonomic mobile manipulators," *Eng. Appl. Artif. Intell.* **21**, 985–1000 (2008).
12. D. Xu, D. Zhao, J. Yi and X. Tan, "Trajectory tracking control of omnidirectional wheeled mobile manipulators: Robust neural network-based sliding mode approach," *IEEE Trans. Syst. Man Cybern.-Part B. Cybern.* **39**(3), 788–799 (2009).
13. Z. Li, C. Yang and Y. Tang, "Decentralized adaptive fuzzy control of coordinated multiple mobile manipulators interacting with non-rigid environment," *IET Control Theory Appl.* **7**(3), 397–410 (2013).
14. M.-B. Cheng, W.-C. Su and C.-C. Tsai, "Robust tracking control of a unicycle-type wheeled mobile manipulator using a hybrid sliding mode fuzzy neural network," *Int. J. Syst. Sci.* **43**(3), 408–425 (2012).
15. G. B. Avanzini, A. M. Zanchettin and P. Rocco, "Constrained model predictive control for mobile robotic manipulators," *Robotica* **36**(1), 19–38 (2018).
16. L. Xiao, B. Liao, S. Li, Z. Zhang, L. Ding and L. Jin, "Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator," *IEEE Trans. Ind. Informatics* **14**(1), 98–105 (2018).
17. A. Marino, "Distributed adaptive control of networked cooperative mobile manipulators," *IEEE Trans. Control Syst. Technol.* **26**(5), 1646–1660 (2018). doi: 10.1109/TCST.2017.2720673
18. C.-B. Dai and Y.-C. Liu, "Distributed coordination and cooperation control for networked mobile manipulators," *IEEE Trans. Ind. Electron.* **64**(6), 5065–5074 (2017).
19. B. Sharma, S. Singh, J. Vanualailai and A. Prasad, "Globally rigid formation of n-link doubly nonholonomic mobile manipulators," *Rob. Auton. Syst.* **105**, 69–84 (2018).
20. K.-T. Song, S.-Y. Jiang and M.-H. Lin, "Interactive teleoperation of a mobile manipulator using a shared-control approach," *IEEE Trans. Hum.-Mach. Syst.* **46**(6), 834–845 (2016).
21. C. Bechlioulis and G. Rovithakis, "Robust adaptive control of feedback linearizable MIMO nonlinear systems with prescribed performance," *IEEE Trans. Autom. Control* **53**(9), 2090–2099 (2008).
22. W. Wang, J. Huang and C. Wen, "Prescribed performance bound-based adaptive path-following control of uncertain nonholonomic mobile robots," *Int. J. Adapt. Control Signal Process.* **31**, 805–822 (2017).
23. T. Gao, J. Huang, Y. Zhou and Y.-D. Song, "Robust adaptive tracking control of an underactuated ship with guaranteed transient performance," *Int. J. Syst. Sci.* **48**(2), 272–279 (2017).
24. Y. Karayiannidis and Z. Doulgeri, "Regressor-free prescribed performance robot tracking," *Robotica* **31**(8), 1229–1238 (2013).
25. S. Li and Z. Xiang, "Adaptive prescribed performance control for switched nonlinear systems with input saturation," *Int. J. Syst. Sci.* **49**(1), 113–123 (2018).
26. K. Shojaei, "Saturated output feedback control of uncertain nonholonomic wheeled mobile robots," *Robotica* **33**(1), 87–105 (2015).
27. K. Shojaei, "Neural adaptive output feedback control of wheeled mobile robots with saturating actuators," *Int. J. Adapt. Control Signal Proces.* **29**, 855–876 (2015).
28. K. Shojaei, "Neural adaptive output feedback formation control of type (m, s) wheeled mobile robots," *IET Control Theory Appl.* **11**(4), 504–515 (2017).
29. Z. Li and S. S. Ge, *Fundamentals in Modelling and Control of Mobile Manipulators* (CRC Press, Boca Raton, 2013).
30. K. Shojaei, A. M. Shahri and B. Tabibian, "Design and implementation of an inverse dynamics controller for uncertain nonholonomic robotic systems," *J. Intell. Rob. Syst.* **71**, 65–83 (2013).
31. K. Shojaei, A. Tarakameh and A. M. Shahri, "Adaptive Trajectory Tracking of WMRs Based on Feedback Linearization Technique," *Proceedings of the 2009 IEEE International Conference on Mechatronics and Automation*, Changchun, China (2009), pp. 729–734.

32. D. Wang and G. Xu, "Full-state tracking and internal dynamics of nonholonomic wheeled mobile robots," *IEEE ASME Trans. Mechatron.* **8**(2), 203–214 (2003).
33. G. Campion, G. Bastin and B. d'Andrea-Novel, "Structural properties and classification of kinematic and dyanmic models of wheeled mobile robots," *IEEE Trans. Rob. Autom.* **12**(1), 47–62 (1996).
34. R. Siegwart, I. R. Nourbakhsh and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed. (MIT Press, London, UK, 2011).
35. K. Shojaei and A. M. Shahri, "Output feedback tracking control of uncertain non-holonomic wheeled mobile robots: A dynamic surface control approach," *IET Control Theory Appl.* **6**(2), 216–228 (2012).
36. P. A. Ioannou and J. Sun, *Robust Adaptive Control* (Prentice-Hall, Englewood Cliffs, NJ, 1996).
37. F. L. Lewis, D. M. Dawson and C. T. Abdallah, *Robot Manipulator Control Theory and Practice*. 2nd ed. Revised and Expanded (Marcel Dekker, New York, 2004).
38. M. M. Polycarpou, "Stable adaptive neural control scheme for nonlinear systems," *IEEE Trans. Autom. Control* **41**(3), 447–451 (1996).
39. S. S. Ge, C. C. Hang, T. H. Lee and T. Zhang, *Stable Adaptive Neural Network Control* (Kluwer, Boston, MA, 2001).
40. G. D. White, R. M. Bhatt and V. N. Krovi, "Dynamic redundancy resolution in a nonholonomic wheeled mobile manipulator," *Robotica* **25**(2), 147–156 (2007).
41. K. Shojaei and A. M. Shahri, "Experimental study of iterated Kalman filters for simultaneous localization and mapping of autonomous mobile robots," *J. Intell. Rob. Syst.* **63**(3–4), 575–594 (2011).
42. K. Shojaei and A. M. Shahri, "Iterated Unscented SLAM Algorithm for Navigation of an Autonomous Mobile Robot," *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France (2008).
43. M. Boukattaya, M. Jallouli and T. Damak, "On trajectory tracking control for nonholonomic mobile manipulators with dynamic uncertainties and external torque disturbances," *Rob. Auton. Syst.* **60**, 1640–1647 (2012).