Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Information Sciences 171 (2005) 447–474

INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL

www.elsevier.com/locate/ins

Intelligent mobile manipulator navigation using adaptive neuro-fuzzy systems ☆

Jean Bosco Mbede [a,b,c,*], Pierre Ele [b], Chantal-Marguerite Mveh-Abia [b], Youssoufi Toure [c], Volker Graefe [d], Shugen Ma [a]

[a] *Faculty of Engineering, Ibaraki University, Hitachi, Ibaraki 316-8511, Japan*
[b] *Ecole Nationale Supérieure Polytechnique, Université de Yaoundé I, BP 8390 Yaounde, Cameroun*
[c] *Laboratoire de Vision et Robotique, Université d'Orléans, 18020 Bourges, France*
[d] *Faculty of Aerospace Engineering, The University of the Armed Forces of Munich, 85577 Neubiberg, Germany*

Accepted 27 September 2004

Abstract

The work presented in this paper deals with the problem of autonomous and intelligent navigation of mobile manipulator, where the unavailability of a complete mathematical model of robot systems and uncertainties of sensor data make the used of approximate reasoning to the design of autonomous motion control very attractive.

A modular fuzzy navigation method in changing and dynamic unstructured environments has been developed. For a manipulator arm, we apply the robust adaptive fuzzy reactive motion planning developed in [J.B. Mbede, X. Huang, M. Wang, Robust neuro-fuzzy sensor-based motion control among dynamic obstacles for robot manipu-

☆ This paper was presented in part at the 2004-IEEE International Conference on Robotics and Automation (ICRA 2004), New Orleans, April 2004.

* Corresponding author. Address: Faculty of Engineering, Ibaraki University, Hitachi, Ibaraki 316-8511, Japan. Tel.: +81 294 38 50 92; fax: +81 294 38 51 11.

E-mail address: mbede@dse.ibaraki.ac.jp (J.B. Mbede).

doi:10.1016/j.ins.2004.09.014

lators, IEEE Transactions on Fuzzy Systems 11 (2) (2003) 249–261]. But for the vehicle platform, we combine the advantages of probabilistic roadmap as global planner and fuzzy reactive based on idea of elastic band. This fuzzy local planner based on a computational efficient processing scheme maintains a permanent flexible path between two nodes in network generated by a probabilistic roadmap approach. In order to consider the compatibility of stabilization, mobilization and manipulation, we add the input of system stability in vehicle fuzzy navigation so that the mobile manipulator can avoid stably unknown and/or dynamic obstacles. The purpose of an integration of robust controller and modified Elman neural network (MENN) is to deal with uncertainties, which can be translated in the output membership functions of fuzzy systems.

Keywords: Autonomous mobile manipulator; Fuzzy reactive navigation; Robust neuro-fuzzy controllers

1. Introduction

Mobile manipulator is nowadays a widespread term to refer to robots built from a robotic arm mounted on a mobile platform. Such systems combine the advantages of mobile platforms and robotic arms and reduce their drawbacks. For instance, the mobile platform extends the arm workspace, whereas an arm offers much operational functionality. Applications for such systems abound in mining, construction, forestry, planetary exploration, and the military. Although appeared very early in robotics history [13], a host of issues related to mobile manipulators have mainly been studied in recent years [3,23,24]. These include dynamic and static stability, force development and application, control in the presence of base compliance, dynamic coupling issues [12,17]. Particularly, motion planning has generally been treated within the framework of optimal control. The complete trajectory is preplanned before execution. On these issues several papers have appeared in literature: Desai and Kumar [4] have formulated the problem as an optimal control problem. Pin et al. [15] performed local optimisation at velocity level. Huang et al. [8] decoupled the motion of the vehicle from that of the manipulator and optimised each one using different criteria. Recently, Tanner et al. [21] proposed a closed loop motion-planning scheme that achieves convergence and obstacle avoidance based on the potential field method. Because generally the precedent method might fail to achieve the task by getting trapped in local minima, Brock and Khatib [2] combine the advantages of reactive controllers (fastness) and the advantages of planners (find a solution whenever one exists) by an elastic band or elastic strip formulation. Siméon et al. [19] provided a probabilistic scheme in order to compute the visibility roadmaps. In most of the previous works, the robot workspaces are highly engineered and completely determined at design time.

The efficacy of motion control algorithms need to be validated in face of uncertainty and imprecision that are inherent of real-time navigation and motion control based on poor and noisy sensory data.

The fuzzy control is credited with being an adequate methodology for designing robust controllers that are able to deliver a satisfactory performance in face of large amounts of noise in the input and of variability in the parameters. This methodology attempts to build the fuzzy rules and the (inputs and outputs) membership functions based on expert knowledge. Significant amounts of research and development have been performed through the observation of the actions of a human operator controlling the mobile robots [7,9,18,22], and in the areas of autonomous motion control of manipulator [1,6,10,11]. But very little has been reported on the topic of coupling these two functions for the purpose of developing an autonomous navigation of mobile manipulator in dynamic unstructured environments. The uncertainties associated with changing and dynamic unstructured environments cannot be handled by using only precise and crisp membership functions of fuzzy sets. Erden et al. [5] identified the base, elbow and shoulder as three different agents to control a mobile manipulator using fuzzy control. They explored the use of genetic algorithms to turn the membership functions of fuzzy controllers.

In this paper, we attempt to develop an on-board intelligent motion controller based on fuzzy obstacle avoidance scheme, that enable the mobile manipulator to navigate autonomously to specified goals, in presence of uncertainties, while not exposing the robot to any undue risk. We couple the fuzzy reactive navigation algorithm of manipulator arm developed in [10] with autonomous navigation of mobile platform. The autonomous navigation of mobile platform combines the advantages of probabilistic roadmap as global planner and the advantages of fuzzy reactive planner based on idea of elastic band. The proposed intelligent local planners have many advantages: adaptation to fast changes in environment and real-time sensor-based path planning application. These attributes make them suitable for on-line dynamic obstacle avoidance. Furthermore, it is necessary to consider the stability problem (overturn prevention), because the mobile manipulator might tip over when the vehicle is moving at a high speed without taking into the optimal configuration of the arm or in environment with disturbance, and when the arm is exerting large forces to move heavy payload. The purpose of the system stability input of fuzzy navigation of vehicle control is to maintain the system stable during its motion.

Unlike fuzzy-genetic motion planning [5], robust fuzzy obstacle avoidance should be capable of dealing with structured and unstructured uncertainties in the output of intelligent systems. To accomplish this task, we proposed an integration of robust terms and modified Elman neural network (MENN) to approximate such uncertainties. We preferred this MENN to a multilayer neural network, because there is not yet a well-defined procedure for the selection of optimum multilayer neural network architecture for a given problem. The common technique is to try a few architectures that give the required approximation while keeping the complexity of the network at a low level. The MENN is a recurrent neural network used by Pham and Liu [14] to identify the second order nonlinear dynamic system. The proposed MENN controller differs from Pham and Liu's learning techniques in the sense that the adaptation algorithm is computed on-line using the operational data of actual robot movements based on the trajectory planning given by fuzzy navigation. To keep the weights of the MENN from growing without bounds, we use a robust term in control law. In real-time world systems, sensor-based motion control becomes essential to deal with model uncertainties and unexpected obstacles.

The rest of this paper is organised as follows. In Section 2, we introduce a modular model of mobile manipulator containing actuator dynamic. Before presenting the design of robust neuro-fuzzy controller in Section 4, we first describe the navigation techniques based on combination of two fuzzy local planners and global probabilistic roadmap approach, and derive the coordination motion control algorithm in Section 3. Section 5 presents our simulation results. Finally, Section 6 gives some concluding remarks.

2. Modular model containing actuator dynamics

The equations of motion containing the actuator dynamics for a mobile manipulator, which the skeleton model is shown in Fig. 1, are given by

$$\begin{aligned} &M_{\mathrm{a}}(q_{\mathrm{a}})\ddot{q}_{\mathrm{a}} + C_{\mathrm{a}}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}) + G_{\mathrm{a}}(q_{\mathrm{a}}, q_{\mathrm{v}}) + F_{\mathrm{a}}(\dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}) \\ &\quad = \tau_{\mathrm{a}} - R_{\mathrm{a}}(q_{\mathrm{a}}, q_{\mathrm{v}})\ddot{q}_{\mathrm{v}}, \end{aligned} \tag{1}$$

$$\begin{aligned} &M_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}})\ddot{q}_{\mathrm{v}} + C_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}) + G_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}}) + F(\dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}) \\ &\quad = E_{\mathrm{v}}\tau_{\mathrm{v}} - A^{\mathrm{T}}\lambda - R_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}})\ddot{q}_{\mathrm{a}}, \end{aligned} \tag{2}$$

$$\tau_{\mathrm{a}} = N_{\mathrm{a}}K_{\mathrm{a}}^{\mathrm{T}}I_{\mathrm{a}}, \tag{3}$$

$$r_{\mathrm{a}}I_{\mathrm{a}} + L_{\mathrm{a}}\dot{I}_{\mathrm{a}} + N_{\mathrm{a}}K_{\mathrm{a}}^{\mathrm{e}}\dot{q}_{\mathrm{a}} = u_{\mathrm{a}}, \tag{4}$$

$$\dot{q}_{\mathrm{v}} = \begin{bmatrix} \cos\theta_{\mathrm{v}} & -d\sin\theta_{\mathrm{v}} \\ \sin\theta_{\mathrm{v}} & d\cos\theta_{\mathrm{v}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \pi r_{\mathrm{v}}N_{\mathrm{v}}K_{\mathrm{v}}^{\mathrm{m}}(u_{\mathrm{r}} + u_{\mathrm{l}}) \\ 2\pi r_{\mathrm{v}}N_{\mathrm{v}}K_{\mathrm{v}}^{\mathrm{m}}\frac{(u_{\mathrm{r}} - u_{\mathrm{l}})}{b} \end{bmatrix}, \tag{5}$$

where $q_{\mathrm{a}}, q_{\mathrm{v}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}, \ddot{q}_{\mathrm{a}}, \ddot{q}_{\mathrm{v}}$ are the coordinates, velocities and accelerations of arm robot and its vehicle, respectively, $M_{\mathrm{a}}(q_{\mathrm{a}})$ is the inertia matrix of manipulator, $C_{\mathrm{a}}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})$ represents coriolis and centrifugal terms of manipulator, and coriolis and centrifugal terms caused by the angular motion of the platform, τ_{a} is the input torque/force for the manipulator, and $R_{\mathrm{a}}(q_{\mathrm{a}}, q_{\mathrm{v}})$ is the inertia matrix

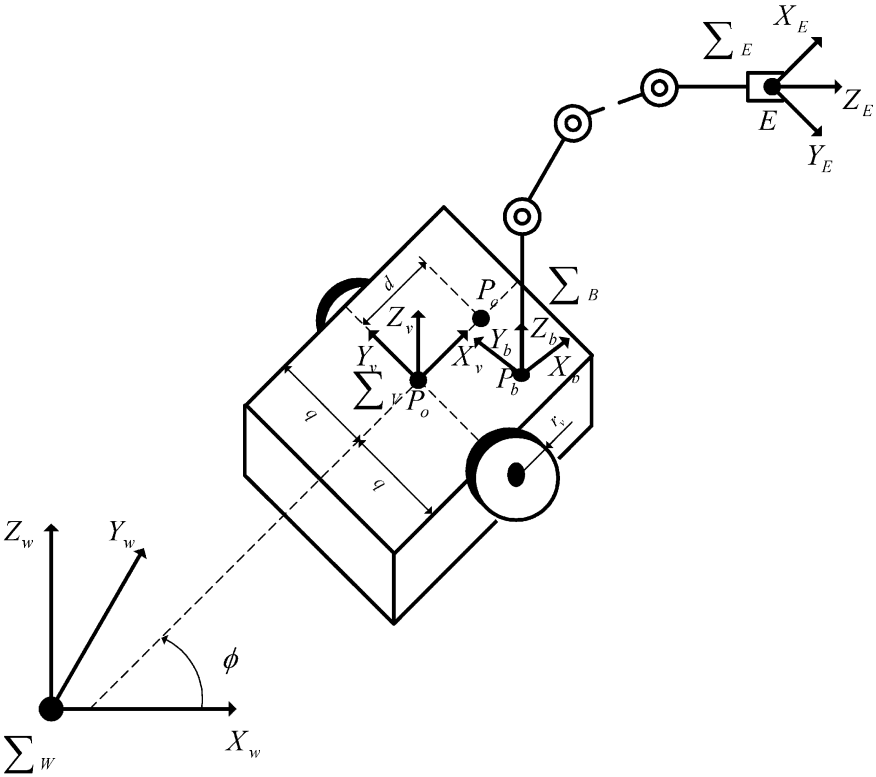

Fig. 1. Skeleton model of mobile manipulator.

which represents the effect of the vehicle dynamics on the arm. The generalized joint torque equation (3) contains the torque constant of the *j*th motor K_a and the gear ratio of the *j*th joint N_a. The voltage equation (4) of the armature circuit of the *j*th motor of the robot arm contains the resistance of armature circuit r_a, L_a the inductance of armature current, the armature current I_a, the voltage constant of the motor K_a^e, and the armature voltage u_a. The mobile platform's motion equations (2) and (5) contain the voltage constant of the motor K_v^m, the armature voltage of the right wheel u_r, the armature voltage of the left wheel u_l, the number of rotation of both motors N_v, the radius of each driven wheel r_v, the distance between the driven wheels and the axis of symmetry b, the angular velocity of wheel $\dot{\theta}_v$, where d is the distance from P_0 (the intersection of the axis of symmetry with the driving wheel axis) to P_c (the center of mass of the platform). $M_v(q_a, q_v)$ represents the mass inertia matrix of platform and the inertia term due to the presence of the manipulator, $C_v(q_a, q_v, \dot{q}_a, \dot{q}_v)$ is the velocity dependent terms of the platform, and the coriolis and centrifugal terms due to the presence of the manipulator, τ_v is the input torque/force to the vehicle, E_v is a constant matrix, A is a Jacobian matrix associated with the contact surface geometry, λ denotes the vector of Lagrange multiplier corresponding to the kinematic constraints, and $R_v(q_a, q_v)$ represents the inertia matrix which reflects the dynamic effect of the arm motion on the vehicle.

3. Mobile manipulator navigation

For mobile manipulator path planning, we describe a new paradigm to the learning motion planning approach, which combines a global roadmap with fuzzy local planners. In the learning phase, which is closely related to the works done by [20], a probabilistic roadmap is built up by repeatedly generating random free configurations and trying to connect these to other configurations by some simple motion planning algorithm. The network thus formed is stored in a graph G. After the learning is done, a query consists of trying to connect the given start and goal configurations to some nodes (in the same connected component) of G, with paths, which are feasible for the robot. Next, local planners can transform the path between these nodes in G into a feasible path. Two local planners have been implemented for our mechanical system. The first one is applied to arm manipulator path. This local planner is based on fuzzy rules for idea of artificial potential fields using analytic harmonic functions. The second local planer takes into account the nonholonomic constraints of mobile platform in Reeds and Shepp metric system [16]. This last local planner is also based on fuzzy rules but for idea of the concept of elastic band in Reeds and Shepp metric system, which consists in maintaining a permanent flexible path between two nodes in G.

3.1. Reactive fuzzy navigation of arm manipulator

In this subsection, we realize a collision avoidance based on the local intelligent dynamic motion-planning algorithm presented in Fig. 2 where a current position of robot arm $X_A = h(q_a)$ is the transformation from joint coordinates to Cartesian coordinates, and X_G is a set of points given by operator (goals) and/or imposed by system stability.

It is important to note that the case where $d_a > d_{\min}$, is extremely rare because it is a moment when the obstacles suddenly surround the robot. The Fuzzy decision is described in [10].

3.2. Reactive fuzzy navigation of mobile platform

The mobile platform reactive navigation is based on local intelligent dynamic motion planning algorithm presented in Fig. 3. The fuzzy system, which

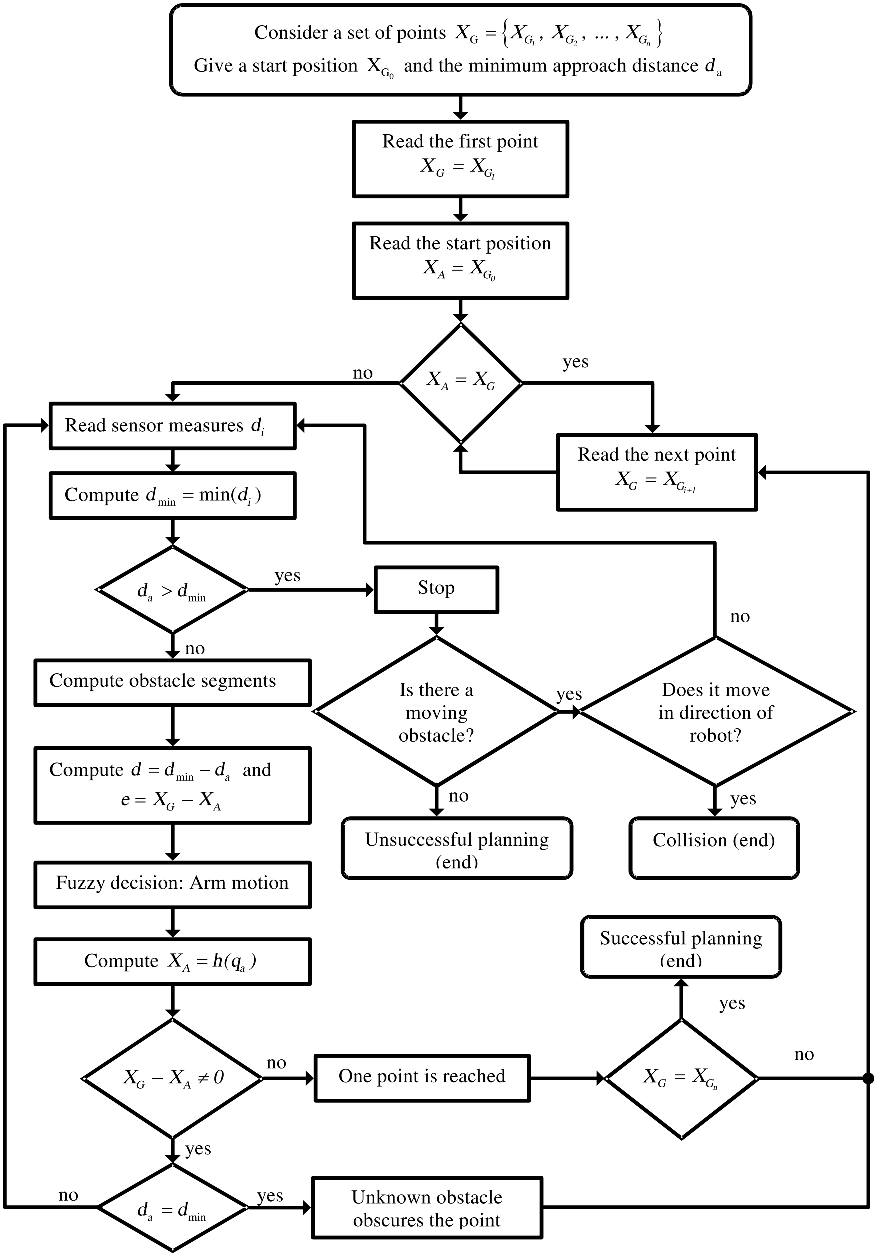
Consider a set of points $X_G = \{X_{G_1}, X_{G_2}, \dots, X_{G_n}\}$
Give a start position X_{G_0} and the minimum approach distance d_a
Read the first point $X_G = X_{G_1}$
Read the start position $X_A = X_{G_0}$
$X_A = X_G$
no
yes
Read sensor measures d_i
Read the next point $X_G = X_{G_{i+1}}$
Compute $d_{\min} = \min(d_i)$
$d_a > d_{\min}$
yes
Stop
no
no
Compute obstacle segments
Is there a moving obstacle?
yes
Does it move in direction of robot?
Compute $d = d_{\min} - d_a$ and $e = X_G - X_A$
no
yes
Unsuccessful planning (end)
Collision (end)
Fuzzy decision: Arm motion
Successful planning (end)
Compute $X_A = h(q_a)$
yes
no
$X_G - X_A \neq 0$
no
One point is reached
$X_G = X_{G_n}$
yes
no
$d_a = d_{\min}$
yes
Unknown obstacle obscures the point

Fig. 2. Arm manipulator local intelligent dynamic motion planning algorithm.

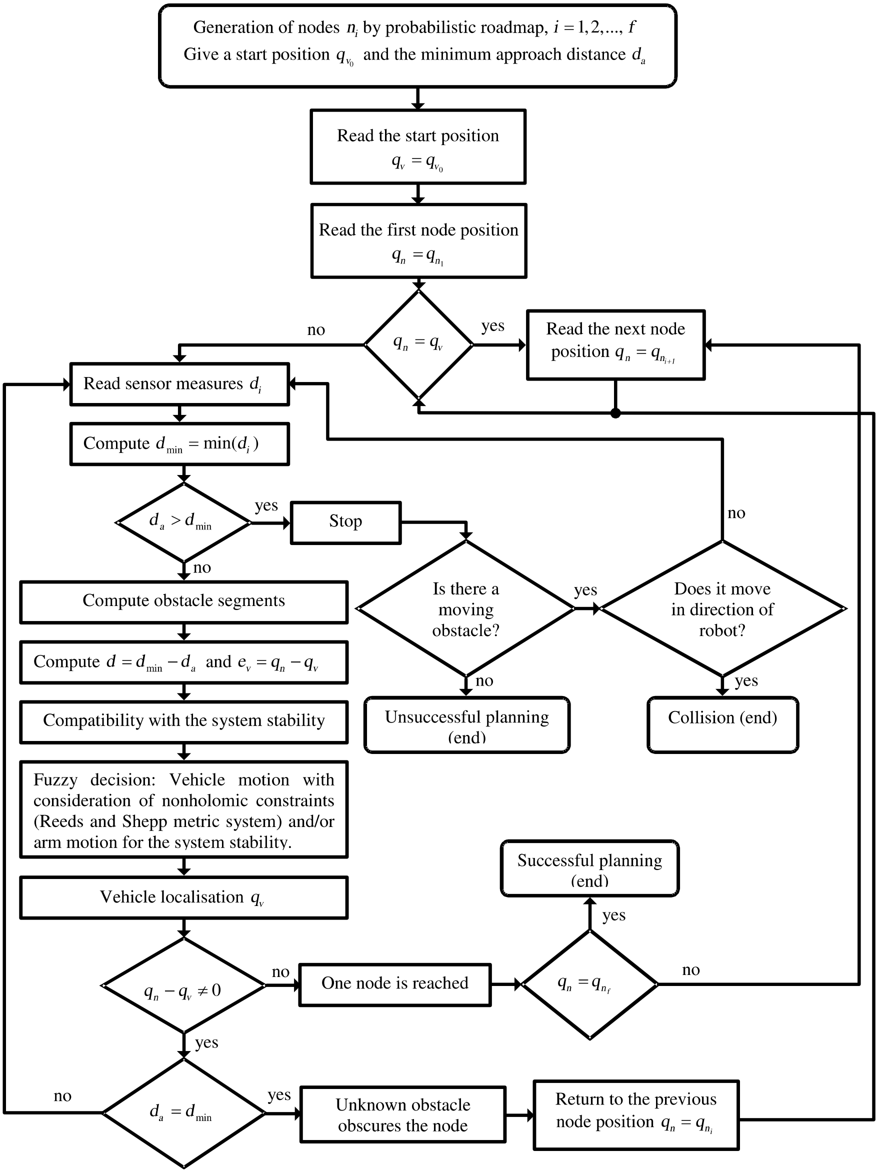
Generation of nodes n_i by probabilistic roadmap, $i = 1,2,...,f$
Give a start position q_{v_0} and the minimum approach distance d_a
Read the start position $q_v = q_{v_0}$
Read the first node position $q_n = q_{n_1}$
$q_n = q_v$
no
yes
Read the next node position $q_n = q_{n_{i+1}}$
Read sensor measures d_i
Compute $d_{\min} = \min(d_i)$
$d_a > d_{\min}$
yes
Stop
no
Is there a moving obstacle?
yes
Does it move in direction of robot?
no
Compute obstacle segments
Compute $d = d_{\min} - d_a$ and $e_v = q_n - q_v$
no
Unsuccessful planning (end)
yes
Collision (end)
Compatibility with the system stability
Fuzzy decision: Vehicle motion with consideration of nonholomic constraints (Reeds and Shepp metric system) and/or arm motion for the system stability.
Successful planning (end)
Vehicle localisation q_v
yes
$q_n - q_v \neq 0$
no
One node is reached
$q_n = q_{n_f}$
no
yes
no
$d_a = d_{\min}$
yes
Unknown obstacle obscures the node
Return to the previous node position $q_n = q_{n_i}$

Fig. 3. Vehicle local intelligent dynamic motion planning algorithm.

makes decision about the vehicle motion, has three types of input variables: sensor data represented by the distance d_v that is the difference between the minimum distance $d_{\min}$ of sensor data and the given minimum approach distance d_a. The difference e_v between the present localization q_{1_i} of vehicle and the position q_{v_i} of next node n_i is the second input of our fuzzy reactive navigation. The last input considers the system stability, mobilization, and manipulation. The force F_v that approximate the virtual spring forces attached to control points on consecutive configurations of the robot along the elastic strip, and the correction position X_{G_i} of arm manipulator that maintain the system stable during the vehicle motion are the both output variables of the vehicle robust fuzzy controller.

3.2.1. Fuzzification

The fuzzification module transforms numerical variables in fuzzy sets, which can be manipulated by the controller.

The distance d_v is partitioned into six fuzzy sets: far left (FL)/far back (FB), medium left (ML)/medium back (MB), close left (CL)/close back (CB), close right (CR)/close front (CF), medium right (MR)/medium front (MF), and far right (FR)/far front (FF). When the obstacle is in the left or back of the vehicle the distance d_v is negative, and when it is in right or front of the mobile platform, the distance d_v is positive. Its fuzzy membership functions are asymmetric and shown in Fig. 4 where $\wp$ is the maximum ultrasonic range.

The difference e_v is partitioned into five fuzzy sets: big negative (BN), small negative (SN), zero (Z), small positive (SP), and big positive (BP). Its fuzzy membership functions are symmetric and shown in Fig. 5 where χ denotes the maximum distance between two nodes.

The arm position q_s for the system stability during the vehicle motion is partitioned into three fuzzy sets: left (LF)/back (BC), medium left (ML)/medium back (MB), and right (RG)/front (FR). Its fuzzy membership functions are symmetric and shown in Fig. 6 where ϑ is the bound of stability space of arm manipulator during vehicle motion.

3.2.2. Rule base

The rule base is generalized as follows: R^i: If $e_v(k)$ is $\mu_1^i(e_v(k))$ and…and $e_v(k-n+1)$ is $\mu_n^i(e_v(k-n+1))$ and $d_v(k)$ is and…and $d_v(k-m+1)$ is

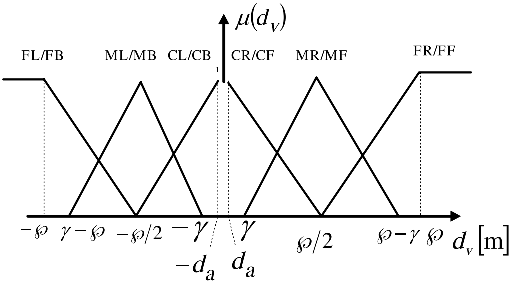

Fig. 4. Fuzzy membership functions for d_v.

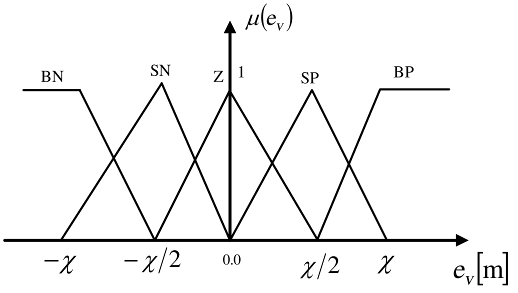

Fig. 5. Fuzzy membership functions for e_{v}.

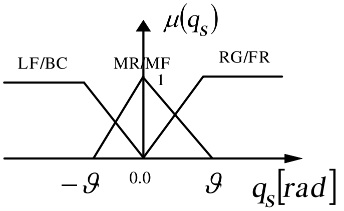

Fig. 6. Fuzzy membership functions for q_{s}.

$\mu^i_m(e_{\mathrm{v}}(k-m+1))$ and q_{s} (k) is $\mu^i_1(q_{\mathrm{s}}(k))$ and…and $q_{\mathrm{s}}(k-s+1)$ is $\mu^i_n(q_{\mathrm{s}}(k-s+1))$ Then $F_{vi}(k+1)$ is r^i and X_{G_i} is p^i.

The outputs of fuzzy rule base are virtual spring forces F_{v} and stability position of manipulator arm X_G. The torque output F_{v} is partitioned into thirteen fuzzy sets: forward left very big (FLVB)/backward left very big (BLVB), forward left big (FLB)/backward left big (BLB), forward left small (FLS)/backward left small (BLS), forward left very small (FLVS)/backward left very small (BLVS), forward right very big (FRVB)/backward right very big (BRVB), forward right big (FRB)/backward right big (BRB), forward right small (FRS)/backward right small (BRS), forward right very small (FRVS)/backward right very small (BRVS), forward strait very big (FSVB)/backward strait very big (BSVB), forward strait big (FSB)/backward strait big (BSB), forward strait small (FSS)/backward strait small (BSS), forward strait very small (FSVS)/backward strait very small (BSVS), and zero (Z). The stability position X_G is partitioned into three fuzzy sets: left (LF), zero (Z), and right (RG).

For example, e_{v} is BP, d_{v} is FF, and q_{s} is MR then F_{v} is FSVB and X_G is Z.

3.2.3. Defuzzification

Two output fuzzy sets result from the inference process, one for the force F_{v}, and the other for the correction position X_{G_i} of arm manipulator. The defuzzification module obtains, from these sets, numerical values. It was chosen the center of gravity (COG) method because it takes into account, better than any other method, the distribution of the resultant fuzzy set. The COG method, applied to fuzzy singletons, simplifies the computation of inference mechanism and reduces processing time without degradation of the defuzzified value. This one is obtained from:

$$u = \frac{\sum_i \alpha_i s_i}{\sum_i \alpha_i}, \tag{6}$$

where α_i is the degree of activation of the ith rule and s_i is the output singleton.

3.3. Cooperative motion planning

Considering the compatibility of high velocity of the vehicle, manipulation of the manipulator and stabilization motion of all the system, it is necessary to coordinate the two local planners. Then, based on the assurance of system stability, the mobile manipulator should execute tasks with an optimal configuration. The vehicle motion mainly affects system stability: for example, the robot falls over easily if the velocity or acceleration of the vehicle is high. Since the range of motion of the manipulator will not be large given an end-point trajectory, the manipulator has only limited ability to aid stability. Therefore; in order to ensure system stability, it is effective to plan vehicle local motion first; then plan manipulator local motion. However, if considering only vehicle motion and system stability while planning vehicle motion, the manipulator

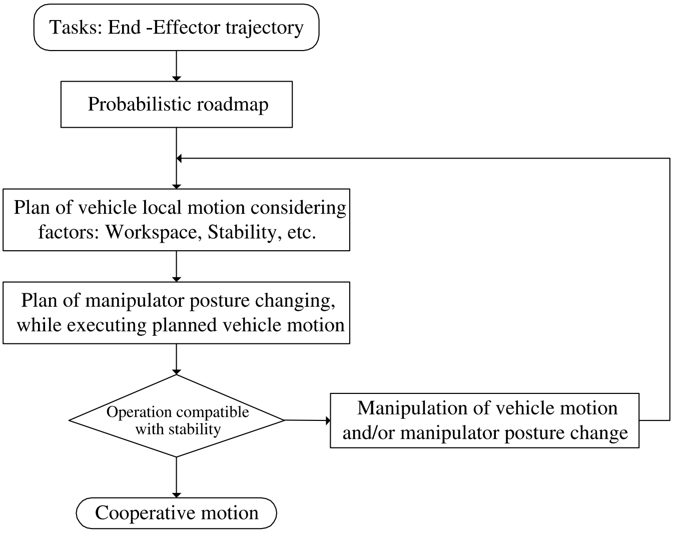
Tasks: End -Effector trajectory
Probabilistic roadmap
Plan of vehicle local motion considering factors: Workspace, Stability, etc.
Plan of manipulator posture changing, while executing planned vehicle motion
Operation compatible with stability
Manipulation of vehicle motion and/or manipulator posture change
Cooperative motion

Fig. 7. Algorithm of cooperative motion planning.

configuration will not be suitable to execute end-point tasks. Conversely, if considering only optimal configuration for executing end-point tasks while planning manipulator motion, the velocity and the acceleration of the vehicle may oscillate easily, and maintenance of system stability will become impossible. Therefore, we propose the coordination algorithm as shown in Fig. 7.

4. Mobile manipulator control

4.1. Manipulator control

In this subsection, we have developed an assembled robust neuro-fuzzy controller for inexact arm model knowledge given by (1) and (4) to accomplish a certain motion to reach the prescribed goal q_{ad}. To accomplish this task, we first rewrite Eq. (1) as

$$\begin{aligned}\tau_{\mathrm{a}} = {} & M_{\mathrm{a}}(q_{\mathrm{a}})\ddot{q}_{\mathrm{a}} + C_{\mathrm{a}1}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{a}} + R_{\mathrm{a}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{v}} + C_{\mathrm{a}2}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{v}} \\ & + F(\dot{q}_{\mathrm{a}}) + G(q_{\mathrm{a}}).\end{aligned} \tag{7}$$

Let $a(t) \in R^n$ be

$$a = -\dot{q}_{\mathrm{a}}. \tag{8}$$

With regard to differentiating (8) and invoking (7), it is seen that the robot dynamic links are expressed in terms of a as

$$\begin{aligned}M_{\mathrm{a}}(q_{\mathrm{a}})\dot{a} = {} & -C_{\mathrm{a}1}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})a + C_{\mathrm{a}2}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{v}} + F(\dot{q}_{\mathrm{a}}) + G(q_{\mathrm{a}}) \\ & - \tau_{\mathrm{a}} + R_{\mathrm{a}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{v}}.\end{aligned} \tag{9}$$

Adding and subtracting $M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a}$ and $C_{\mathrm{a}10}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})a$ yields

$$M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a} = -C_{\mathrm{a}10}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})a + f_1(x_1) - \tau_{\mathrm{a}}, \tag{10}$$

where the nonlinear function $f_1(x_1)$, which contains the imperfectly known and difficult to determine robot parameters, is defined as

$$\begin{aligned}f_1(x_1) = {} & (M_{\mathrm{a}0}(q_{\mathrm{a}}) - M_{\mathrm{a}}(q_{\mathrm{a}}))\dot{a} + \begin{pmatrix} C_{\mathrm{a}10}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}}) \\ -C_{\mathrm{a}1}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}}) \end{pmatrix} a \\ & + C_{\mathrm{a}2}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{v}} + F(\dot{q}_{\mathrm{a}}) + G(q_{\mathrm{a}}) + R_{\mathrm{a}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{v}}.\end{aligned} \tag{11}$$

The vector x_1 required to compute $f_1(x_1)$ can be defined as $x_1 = \begin{bmatrix} q_{\mathrm{a}}^{\mathrm{T}} & q_{\mathrm{v}}^{\mathrm{T}} & \dot{q}_{\mathrm{a}}^{\mathrm{T}} & \dot{q}_{\mathrm{v}}^{\mathrm{T}} & \ddot{q}_{\mathrm{v}}^{\mathrm{T}} \end{bmatrix}$, which can be measured.

Now let I_{ad} be a value of I_{a} that stabilizes the dynamics (10) in the form

$$M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a} = -C_{\mathrm{a}10}(q_{\mathrm{a}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})a + f_1(x_1) - NK_T I_{\mathrm{ad}} + NK_T\eta \tag{12}$$

with $\eta = I_{\mathrm{ad}} - I_{\mathrm{a}}$ an error term. Signal $I_{\mathrm{ad}}(t)$ will be selected later. To find the complete error dynamics, differentiate $L\eta$ and substitute from (4) to find

$$L\dot{\eta} = f_2(x_2) + u_{\mathrm{d}} - u_{\mathrm{e}}, \tag{13}$$

where the unknown nonlinear motor function is

$$f_2(x_2) = RI_{\mathrm{a}} + L\dot{I}_{\mathrm{ad}} + NK_{\mathrm{e}}\dot{q}_{\mathrm{a}}. \tag{14}$$

One may select x_2 as $x_2 = \begin{bmatrix} I_{\mathrm{a}}^{\mathrm{T}} & \dot{I}_{\mathrm{ad}}^{\mathrm{T}} & \dot{q}_{\mathrm{a}}^{\mathrm{T}} \end{bmatrix}$.

4.1.1. Manipulator control structure

A suitable approximation-based controller is given by the desirable value of armature current as

$$I_{\mathrm{ad}} = \frac{1}{K_1}(\hat{f}_1 + K_{\mathrm{D}}a + \widetilde{F}_{\mathrm{art}} + \gamma_1) \tag{15}$$

with $\hat{f}_1(x_1)$ an estimate of the unknown robot function $f_1(x_1)$ that is provided by MENN. The signal $\gamma_1(t)$ is required to compensate for the mismatch between $\hat{f}_1(x_1)$ and $f_1(x_1)$. K_{D} is a positive definite symmetric matrix.

Using this control in (12), the closed-loop system becomes

$$M_{\mathrm{a0}}(q_{\mathrm{a}})\dot{a} = -[K_{\mathrm{D}} + C_{\mathrm{a10}}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})]a + \tilde{f}_1(x_1) - \widetilde{F}_{\mathrm{art}} + \gamma_1, \tag{16}$$

where the function approximation error $\tilde{f}_1$ is given by

$$\tilde{f}_1 = f_1 - \hat{f}_1. \tag{17}$$

The second step is the design of the voltage control u_{e} for the open loop system of (13). The control input u_{e} is selected as

$$u_{\mathrm{e}} = \hat{f}_2 + K_P\eta + \gamma_2 \tag{18}$$

with $K_P > 0$ a gain matrix and $\hat{f}_2(x_2)$ an estimate of the unknown function $f_2(x_2)$ that is provided by robust control. The signal $\gamma_2(t)$ is required to compensate for the mismatch between $\hat{f}_2(x_2)$ and $f_2(x_2)$. Substituting (18) into the open loop dynamics of (13), the closed-loop current perturbation dynamics become

$$L\dot{\eta} = \tilde{f}_2(x_2) - K_P\eta, \tag{19}$$

where the function approximation error $\tilde{f}_2$ is given by

$$\tilde{f}_2 = f_2 - \hat{f}_2. \tag{20}$$

Fig. 8 shows the structure of the proposed control strategy.

4.1.2. Recurrent neural network

The MENN, which structure is shown in Fig. 9, is used to approximate nonlinear function $f_1(x_1)$. In this study we use the sigmoid activation function

$$\sigma(x_1) = \frac{1}{1 + \mathrm{e}^{-x_1}}. \tag{21}$$

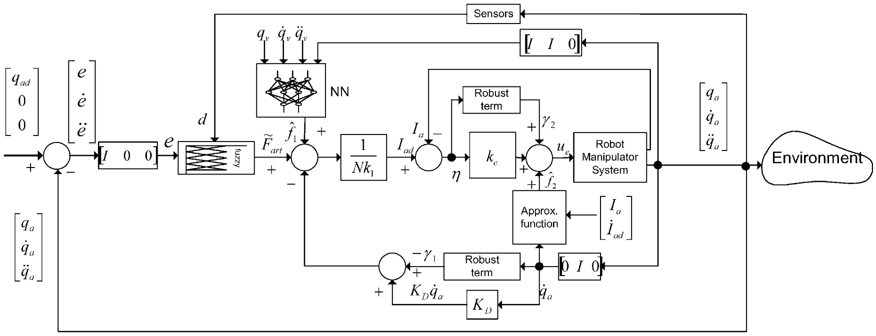
Sensors
q_v $\dot{q}_v$ $\ddot{q}_v$
$[I\ I\ 0]$
NN
Robust term
γ_2
$\begin{bmatrix} q_{ad} \\ 0 \\ 0 \end{bmatrix}$
$\begin{bmatrix} e \\ \dot{e} \\ \ddot{e} \end{bmatrix}$
d
e
$[I\ 0\ 0]$
fuzzy
$\tilde{F}_{art}$
$\hat{f}_1$
$\frac{1}{Nk_1}$
I_{ad}
I_a
η
k_e
u_e
Robot Manipulator System
$\hat{f}_2$
Approx. function
$\begin{bmatrix} I_a \\ \dot{I}_{ad} \end{bmatrix}$
$\begin{bmatrix} q_a \\ \dot{q}_a \\ \ddot{q}_a \end{bmatrix}$
Environment
$\begin{bmatrix} q_a \\ \dot{q}_a \\ \ddot{q}_a \end{bmatrix}$
$-\gamma_1$
Robust term
$[0\ I\ 0]$
$K_D\dot{q}_a$
K_D
$\dot{q}_a$
+
−

Fig. 8. Manipulator controller structure.

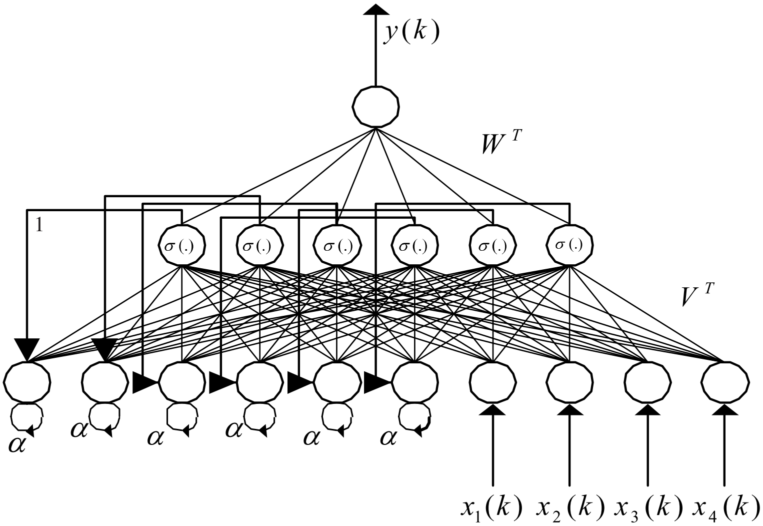

Fig. 9. Modified Elman neural network.

By collecting the MENN weights v_{ij}, v_{ip} into the matrix of weights V^{T}, and weights w_i into the matrix of weights W^{T}, one can write the MENN equation as

$$y = W^{\mathrm{T}}\sigma(V^{\mathrm{T}}x_1) \tag{22}$$

with the vector of activation functions defined by $\sigma(z) = [\,\sigma(z_1) \quad \dots \quad \sigma(z_n)\,]^{\mathrm{T}}$ for a vector $z = V^{\mathrm{T}}x_1$ and the vector $x_1 \equiv [\,1 \quad h_1(k-1) \quad \dots \quad h_1(k-j)$ $h_2(k-1)\dots h_j(k-j)q_{\mathrm{a}}q_{\mathrm{v}}\dot{q}_{\mathrm{a}}\dot{q}_{\mathrm{v}}\ddot{q}_{\mathrm{v}}]$ According to the universal approximation property of neural network, there is a MENN such that

$$f_1(x_1) = W^{\mathrm{T}}\sigma(V^{\mathrm{T}}x_1) + \varepsilon \tag{23}$$

with the approximation error bounded on compact set by $\|\varepsilon\| = \varepsilon_N$ with ε_N a known bound. W and V are ideal target weights that give good approximation to $f_1(x_1)$. They are unknown. All we require is the knowledge that they exist; it is not even required for then to be unique. Define the matrix of all ideal MENN weights as

$$Z \equiv \mathrm{diag}\{W, V\}. \tag{24}$$

Let a MENN estimate of $f_1(x_1)$ be given by

$$\hat{f}_1(x_1) = \widehat{W}^{\mathrm{T}}\sigma(\widehat{V}^{\mathrm{T}}) \tag{25}$$

with $\widehat{V}$, $\widehat{W}$ the actual values of the MENN weights given by the tuning algorithm to be specified. Define the hidden-layer output error for a given x_1 as

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(V^{\mathrm{T}}x_1) - \sigma\left(\widehat{V}^{\mathrm{T}}x_1\right). \tag{26}$$

The Taylor series expansion of $\sigma(x_1)$ for a given x_1 may be written as

$$\sigma(V^{\mathrm{T}}x_1) = \sigma\left(\widehat{V}^{\mathrm{T}}x_1\right) + \sigma'\left(\widehat{V}^{\mathrm{T}}x_1\right)\widetilde{V}^{\mathrm{T}}x_1 + \mathrm{O}\left(\widetilde{V}^{\mathrm{T}}x_1\right)^2 \tag{27}$$

with Jacobian matrix $\sigma'(\hat{z}) \equiv \frac{\mathrm{d}\sigma(z)}{\mathrm{d}z}\|_{z=\hat{z}}$ and $\mathrm{O}(z)^2$ denoting term of order two. Denoting $\hat{\sigma}' = \sigma'(\widehat{V}^{\mathrm{T}}x_1)$, we have

$$\tilde{\sigma} = \sigma'\left(\widehat{V}^{\mathrm{T}}x_1\right)\widetilde{V}^{\mathrm{T}}x_1 + \mathrm{O}\left(\widetilde{V}^{\mathrm{T}}x_1\right)^2 = \hat{\sigma}'\widetilde{V}^{\mathrm{T}}x_1 + \mathrm{O}\left(\widetilde{V}^{\mathrm{T}}x_1\right)^2. \tag{28}$$

The importance of this equation is that it replaces $\hat{\sigma}$, which is nonlinear in $\widetilde{V}$, by an expression linear in $\widetilde{V}$ plus higher-order terms. This will allow us to determine tuning algorithms for $\widehat{V}$ in subsequent derivations.

By placing the MENN approximation equation given by (25) into (15), the control input then becomes

$$I_{\mathrm{ad}} = \frac{1}{K_1}\left(\widehat{W}^{\mathrm{T}}\sigma(\widehat{V}^{\mathrm{T}}x_1) + K_{\mathrm{D}}a + \widetilde{F}_{\mathrm{art}} + \gamma_1\right) \tag{29}$$

with $\gamma_1(t)$ a function to be detailed subsequently, providing robustness in the face of higher-order terms in the Taylor series. Using this controller, the closed-loop position error dynamics become

$$\begin{aligned} M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a} = {} & -[K_{\mathrm{D}} + C_{\mathrm{a}10}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})]a - \widetilde{F}_{\mathrm{art}} + W^{\mathrm{T}}\sigma(V^{\mathrm{T}}x_1) \\ & - \widehat{W}^{\mathrm{T}}\sigma(\widehat{V}^{\mathrm{T}}x_1) + \varepsilon + \gamma_1. \end{aligned} \tag{30}$$

Adding and subtracting $W^{\mathrm{T}}\sigma$ yields

$$M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a} = -[K_{\mathrm{D}} + C_0(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})]a - \widetilde{F}_{\mathrm{art}} + \widetilde{W}^{\mathrm{T}}\hat{\sigma} - W^{\mathrm{T}}\tilde{\sigma} + \varepsilon + \gamma_1 \tag{31}$$

with $\hat{\sigma}$ and $\tilde{\sigma}$ defining in (26). Adding and subtracting at this point $\widehat{W}^{\mathrm{T}}\tilde{\sigma}$ yields

$$M_{\mathrm{a}0}(q_{\mathrm{a}})\dot{a} = -[K_{\mathrm{D}} + C_{\mathrm{a}10}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{a}})]a - \widetilde{F}_{\mathrm{art}} + \widetilde{W}^{\mathrm{T}}\hat{\sigma} + \widehat{W}^{\mathrm{T}}\tilde{\sigma} + \widetilde{W}^{\mathrm{T}}\tilde{\sigma} + \varepsilon + \gamma_1. \tag{32}$$

The key step is the use at this point of the Taylor series approximation (27) for $\tilde{\sigma}$, according to which the error system is

$$\begin{aligned} M_{\mathrm{a0}}\dot{a} = {} & -(K_{\mathrm{D}} + C_{\mathrm{a10}}(q_{\mathrm{a}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}}))a - \widetilde{F}_{\mathrm{art}} + \widetilde{W}^{\mathrm{T}}(\tilde{\sigma} - \hat{\sigma}'\widehat{V}^{\mathrm{T}}x_1) \\ & + \widehat{W}^{\mathrm{T}}\hat{\sigma}'\widetilde{V}^{\mathrm{T}}x_1 + w + \gamma_1, \end{aligned} \tag{33}$$

where the disturbance terms are

$$w(t) = \widetilde{W}^{\mathrm{T}}\hat{\sigma}'V^{\mathrm{T}}x_1 + W^{\mathrm{T}}\mathrm{O}(\widetilde{V}^{\mathrm{T}}x_1)^2 + \varepsilon. \tag{34}$$

It is important to note that the MENN reconstruction error $\varepsilon(x)$ and the higher-order terms in the Taylor series expansion of $f_1(x_1)$ all have exactly the same influence as disturbances in the error system; the next bound is required. Its importance is in allowing one disturbance to over bound $w(t)$ each time by a known computable function.

Lemma (Bound on the disturbance term). *The disturbance term (34) is bounded according to*

$$\begin{aligned} & \|w\|(\varepsilon_N + \tau_{dM} + c_3 Z_M) + c_6 Z_M\|\widetilde{Z}\|_F + c_7 Z_M\|\widetilde{Z}\|_F\|a\| \ \textit{or} \\ & \quad \|w\|C_0 + C_1\|\widetilde{Z}\|_F + C_2\|\widetilde{Z}\|_F\|a\| \end{aligned} \tag{35}$$

with C_i known positive constants. Note that C_0 becomes larger with increases in the MENN estimation error ε and the robot dynamics disturbances τ_d.

Let the MENN approximation property (23) hold for the function f_1 (x_1) given in (11) with a given accuracy ε_N for all x_1 inside the ball of radius $r_x > q_M$. Let the initial derivative position error $a(0)$ satisfy

$$a(0) < (r_x - q_M)/(c_0 + c_2). \tag{36}$$

Take the control input of the robot dynamics (7) as (29) with K_{D} gain satisfying

$$K_{\mathrm{D\ min}} > \frac{\left(C_0 + \frac{1}{4k}(kZ_M + C_1)^2\right)(c_0 + c_2)}{r_x - q_M}, \tag{37}$$

where C_0 and C_1 are defined in (32), and $k > 0$ is a small design parameter. Let the robust term be

$$\gamma_1 = -K_z(\|\widehat{Z}\|_F + Z_M)a, \tag{38}$$

where $Z_z > C_2$. Let MENN weight tuning be provided by

$$\dot{\widehat{W}} = G_1\hat{\sigma}a^{\mathrm{T}} - G_1\hat{\sigma}'\widehat{V}^{\mathrm{T}}x_1 a^{\mathrm{T}} - kG_1\|a\|\widehat{W}, \tag{39}$$

$$\dot{\widehat{V}} = G_2 x_1 (\hat{\sigma}'^{\mathrm{T}} \widehat{W} a)^{\mathrm{T}} - kG_2 \|a\| \widehat{V} \tag{40}$$

with constant positive diagonal matrices G_1 and G_2. Then the derivative position error $a(t)$ and MENN weight estimates are uniformly ultimately bounded (UUB). The input to (39) and (40) consists of derivative position error $a(t)$ whereas backpropagation law uses the error between the desired MENN output and the actual MENN output.

4.1.3. Robust term

The control input u_{e} is selected as (18) with the signal $\gamma_2(t)$ defined as

$$\gamma_2 = \frac{\eta \rho_2^2}{\|\eta\| \rho_2 + \varepsilon_2}, \tag{41}$$

where ε_2 is a positive constant controller gain and small, the bounding positive scalar function ρ_2 is defined as $\rho_2 > \|\tilde{f}_2\| = \|f_2 - \hat{f}_2\|$, where the function approximation error $\tilde{f}_2$ is defined in (17).

4.2. Vehicle control

In this subsection, we develop an assembled robust neuro-fuzzy controller for inexact mobile platform model knowledge given by (2) and (5) to accomplish a certain motion. To accomplish this task, we first rewrite Eq. (2) as

$$\begin{aligned} E_{\mathrm{v}}\tau_{\mathrm{v}} - A^{\mathrm{T}}\lambda = {} & M_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{v}} + C_{\mathrm{v}1}(q_{\mathrm{a}},q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{a}} + C_{\mathrm{v}2}(q_{\mathrm{a}},q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{v}} \\ & + F(\dot{q}_{\mathrm{v}}) + R_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{a}}. \end{aligned} \tag{42}$$

Differentiating $A(q_{\mathrm{v}})\dot{q}_{\mathrm{v}} = 0$, substituting this result in (39), and then multiplying by S^{T}, we can eliminate the constraint matrix $A^{\mathrm{T}}(q_{\mathrm{v}})\lambda$. The complete equations of motion of the nonholonomic mobile platform are given by

$$\dot{q}_{\mathrm{v}} = Sv, \tag{43}$$

$$S^{\mathrm{T}}M_{\mathrm{v}}S\dot{v} + S^{\mathrm{T}}(M_{\mathrm{v}}\dot{S} + C_{\mathrm{v}2}S)v + S^{\mathrm{T}}C_{\mathrm{v}1}\dot{q}_{\mathrm{a}} + S^{\mathrm{T}}R_{\mathrm{v}}\ddot{q}_{\mathrm{a}} + \overline{F}_{\mathrm{v}}(v) = S^{\mathrm{T}}E_{\mathrm{v}}\tau_{\mathrm{v}}, \tag{44}$$

where $v(t) \in R^{n-m}$ is a velocity vector. By appropriate definitions we can rewrite (44) as follow

$$\begin{aligned} \bar{\tau}_{\mathrm{v}} = {} & \overline{M}_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}})\dot{v} + \overline{C}_{\mathrm{v}2}(q_{\mathrm{a}},q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})v + \overline{C}_{\mathrm{v}1}(q_{\mathrm{a}},q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{a}} \\ & + \overline{R}_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}})\ddot{q}_{\mathrm{a}} + \overline{F}_{\mathrm{v}}, \end{aligned} \tag{45}$$

where $\overline{M}_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}}) = S^{\mathrm{T}}M_{\mathrm{v}}S$, $\overline{R}_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}}) = S^{\mathrm{T}}R_{\mathrm{v}}(q_{\mathrm{a}},q_{\mathrm{v}})$ are symmetric positive definite inertia matrices, $\overline{C}_{\mathrm{v}2}(q_{\mathrm{a}},q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}}) = S^{\mathrm{T}}(M_{\mathrm{v}}\dot{S} + C_{\mathrm{v}2}S)$, $\overline{C}_{\mathrm{v}1} = S^{\mathrm{T}}C_{\mathrm{v}1}(q_{\mathrm{a}}, q_{\mathrm{v}},\dot{q}_{\mathrm{a}},\dot{q}_{\mathrm{v}})$ are the centripetal and coriolis matrices, $\overline{F}_{\mathrm{v}}(v)$ is the surface friction, and $\bar{\tau}_{\mathrm{v}} = S^{\mathrm{T}}E_{\mathrm{v}}\tau_{\mathrm{v}}$ is the input torque vector. The fuzzy system provides the torque $\widetilde{F}_V$ that is transformed to the desired velocity v_{d}.

Now let us define the auxiliary velocity tracking error as

$$e_{\mathrm{v}} = v_{\mathrm{d}} - v. \tag{46}$$

Differentiating (46) and using (45), the mobile platform dynamics may be written in terms of the velocity tracking error as

$$\overline{M}_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}})\dot{e}_{\mathrm{v}} + \overline{C}_{\mathrm{v}2}(q_{\mathrm{a}}, q_{\mathrm{v}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})e_{\mathrm{v}} + f(x) = \bar{\tau}_{\mathrm{v}}, \tag{47}$$

where the nonlinear function $f(x)$ which contains the imperfectly known and difficult to determine robot parameters is defined as

$$\begin{aligned} f(x) = {} & \overline{M}_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}})\dot{v}_{\mathrm{d}} + \overline{C}_{\mathrm{v}2}(q_{\mathrm{a}}, q_{\mathrm{v}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})v_{\mathrm{d}} + \overline{C}_{\mathrm{v}1}(q_{\mathrm{a}}, q_{\mathrm{v}}, \dot{q}_{\mathrm{a}}, \dot{q}_{\mathrm{v}})\dot{q}_{\mathrm{a}} \\ & + \overline{R}_{\mathrm{v}}(q_{\mathrm{a}}, q_{\mathrm{v}})\ddot{q}_{\mathrm{a}} + \overline{F}_{\mathrm{v}}(v). \end{aligned} \tag{48}$$

The vector x required to compute $f(x)$ can be defined as $x = [\,v^{\mathrm{T}} \quad v_{\mathrm{d}}^{\mathrm{T}} \quad \dot{v}_{\mathrm{d}}^{\mathrm{T}} \quad \dot{q}_{\mathrm{a}}^{\mathrm{T}} \quad \ddot{q}_{\mathrm{a}}^{\mathrm{T}}\,]$, which can be measured.

4.2.1. Mobile platform control structure

A suitable control input for velocity following as given by the computed-torque like control

$$\bar{\tau}_{\mathrm{v}} = \hat{f} + Ke_{\mathrm{v}} - \gamma \tag{49}$$

with K is diagonal positive definite gain matrix, and $\hat{f}(x)$ an estimate of the robot function $f(x)$ that is provided by the MENN. The robust signal $\gamma(t)$ is required to compensate the unmodeled unstructured disturbances. Using the control in (49), the closed-loop system becomes

$$\overline{M}_{\mathrm{v}}\dot{e}_{\mathrm{v}} + (K + \overline{C}_{\mathrm{v}2})e_{\mathrm{v}} - \tilde{f} - \gamma = 0, \tag{50}$$

where the velocity tracking error is driven by the functional estimation error

$$\tilde{f} = f - \hat{f}. \tag{51}$$

The structure of our mobile platform controller is shown in Fig. 10.

4.2.2. Mobile platform recurrent neural network

In this study, we will use the previous MENN to approximate $f(x)$ for computing the control in (49). By placing into (49) the MENN approximation equation given by (22), the control input then becomes

$$\bar{\tau}_{\mathrm{v}} = \widehat{W}^{\mathrm{T}}\sigma(\widehat{V}^{\mathrm{T}}x) + Ke_{\mathrm{v}} - \gamma \tag{52}$$

with $\gamma(t)$ a function to be detailed subsequently that provides robustness in the face of robot kinematics and hither-order terms in the Taylor series. Using this controller, the closed-loop velocity error dynamics becomes

$$\overline{M}\dot{e}_{\mathrm{v}} = -(K + \overline{C}_{\mathrm{v}2})e_{\mathrm{v}} + W^{\mathrm{T}}\sigma(V^{\mathrm{T}}) - \widehat{W}^{\mathrm{T}}\sigma(\widehat{V}^{\mathrm{T}}) + \gamma. \tag{53}$$

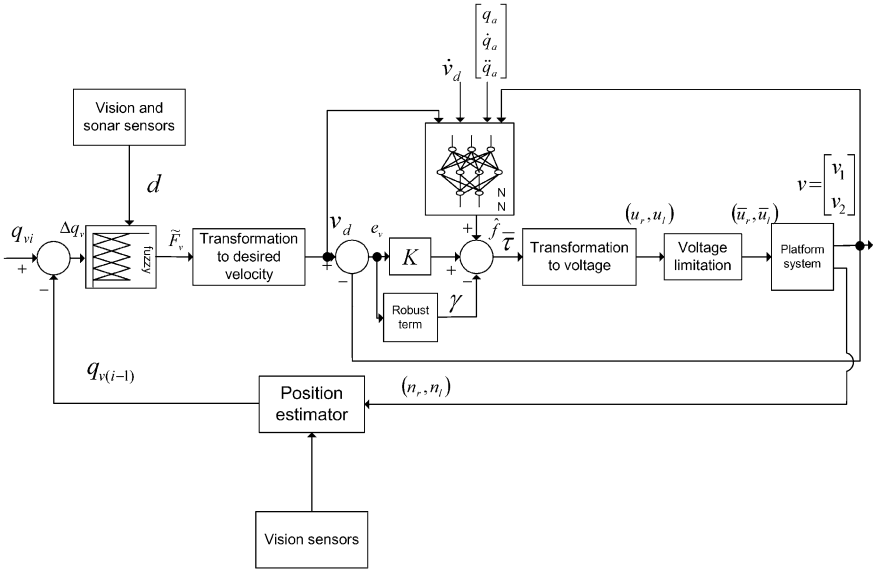

Fig. 10. Mobile platform controller structure.

Adding and subtracting $W^{\mathrm{T}}\hat{\sigma}$ yields

$$\overline{M}\dot{e}_{\mathrm{v}} = -\left(K + \overline{C}_{\mathrm{v}2}\right)e_{\mathrm{v}} + \widetilde{W}^{\mathrm{T}}\hat{\sigma} + W^{\mathrm{T}}\tilde{\sigma} + \varepsilon + \gamma \tag{54}$$

with $\hat{\sigma}, \tilde{\sigma}$ defining in (26). Adding and subtracting now $\widehat{W}^{\mathrm{T}}\tilde{\sigma}$ yields

$$\overline{M}\dot{e}_{\mathrm{v}} = -(K + \overline{C}_{\mathrm{v}2})e_{\mathrm{v}} + \widetilde{W}^{\mathrm{T}}\tilde{\sigma} + \widehat{W}^{\mathrm{T}}\tilde{\sigma} + \widetilde{W}^{\mathrm{T}}\tilde{\sigma} + \varepsilon + \gamma. \tag{55}$$

The key is the use now of the Taylor series approximation (28) for $\tilde{\sigma}$ according to which the error system is

$$\overline{M}\dot{e}_{\mathrm{c}} = -(K + \overline{C}_{\mathrm{v}2})e_{\mathrm{v}} + \widetilde{W}^{\mathrm{T}}(\tilde{\sigma} - \hat{\sigma}'\widehat{V}^{\mathrm{T}}x) + \widehat{W}^{\mathrm{T}}\sigma'\widetilde{V}^{\mathrm{T}} + w + \gamma, \tag{56}$$

where the disturbance terms are

$$w(t) = \widetilde{W}^{\mathrm{T}}\hat{\sigma}'V^{\mathrm{T}}x + W^{\mathrm{T}}\mathrm{O}(\widetilde{V}^{\mathrm{T}}x) + \varepsilon. \tag{57}$$

It is important to note that MENN reconstruction error $\varepsilon(x)$ and the higher-order terms in the Taylor series expansion of $f(x)$ all have exactly the same influence as disturbances in the error system. The next bound is required. Its importance it is in allowing one to over-bound $w(t)$ at each time by a known computable function.

It remains now to show how to select the tuning algorithms for the MENN weights $\widehat{Z}$, and the robust term $\gamma(t)$ so that robust stability and tracking performance are guaranteed. Take the control $\vec{\tau} \in R^r$ for (49) as with robust term

$$\gamma(t) = -K_z(\|\widehat{Z}\|_F + Z_M)e_\mathrm{v} - e_\mathrm{v} \tag{58}$$

and gain $K_z > C_2$ with C_2 the known constant in (35). Let MENN weight tuning be provided by

$$\dot{\widehat{W}} = F\hat{\sigma}e_\mathrm{v}^\mathrm{T} - F\hat{\sigma}'\widehat{V}^\mathrm{T}xe_\mathrm{v}^\mathrm{T} - kF\|e_\mathrm{v}\|\widehat{W}, \tag{59}$$

$$\dot{\widehat{V}} = Gx(\hat{\sigma}'^\mathrm{T}\widehat{W}e_\mathrm{v})^\mathrm{T} - kG\|e_\mathrm{v}t\|\widehat{V}, \tag{60}$$

where F, G are positive definite design parameter matrices, $k > 0$ and the hidden-layer gradient or Jacobian $\hat{\sigma}'$ is easily computed in terms of measurable signals—for the sigmoid activation function it is given by $\hat{\sigma}' \equiv \mathrm{diag}\{\sigma(\widehat{V}^\mathrm{T}x)\}[I - \mathrm{diag}\{\sigma(\widehat{V}^\mathrm{T}x)t\}]$. Then, for large enough control gain K, the velocity tracking error $e_\mathrm{v}(t)$ and the MENN weight estimates $\widehat{V}, \widehat{W}$ are UUB. Moreover, the velocity tracking error may be kept as small as desired by increasing the gain K.

5. Simulations

To validate the theoretical concepts and demonstrate the benefits of the proposed fuzzy obstacle avoidance algorithm, a comprehensive simulation study was carried out using a mobile manipulator that consists of a Mitsubishi manipulator, model RV-M1 (Movemaster EX) and a 3-DOF mobile platform whose parameters are listed in Tables 1 and 2, respectively.

Table 1
Parameters of the Mitsubishi RV-M1

Degrees of freedom		5 (not including hand)
Electrical drive system		DC servo motors
Base		250 mm × 190 mm
Link operation range and length	Waist J_1 (revolute joint)	300° (maximum 120°/s) and 300 mm length
	Shoulder J_2 (revolute joint)	130° (maximum 72°/s) and 215 mm length
	Elbow J_3 (revolute joint)	110° (maximum 109°/s) and 250 mm length
	Wrist pitch J_4 (revolute joint)	±90° (maximum 100°/s) and 160 mm length
	Wrist roll J_5 (revolute joint)	±180° (maximum 163°/s) and 145 mm length
Maximum payload		1.2 kg including hand
Maximum path velocity		1000 mm/s
Weight		19 kg

Table 2
Parameters of the mobile platform

Dimension	500 mm × 340 mm
Mass including the driving wheel plus the rotor of its motor	100 kg
Electrical drive system	DC servo motors

Simulations were performed on dynamic environment shows in Fig. 11. The environment dimensions are given in meters. The points A, B, C, D, and E are the nodes of graph *G* obtained by the global road map approach. The distance from the control point F to moving obstacle 11 is the maximum distance of obstacle detection. The vehicle fuzzy local navigator who takes into account the nonholonomic constraints of mobile platform in Reeds and Shepp metric system produces the path between two points the precedent points.

Using intelligent motion planning and robust neuro-fuzzy feedback control proposed in the two previous sections, the simulations shown in Fig. 12 have

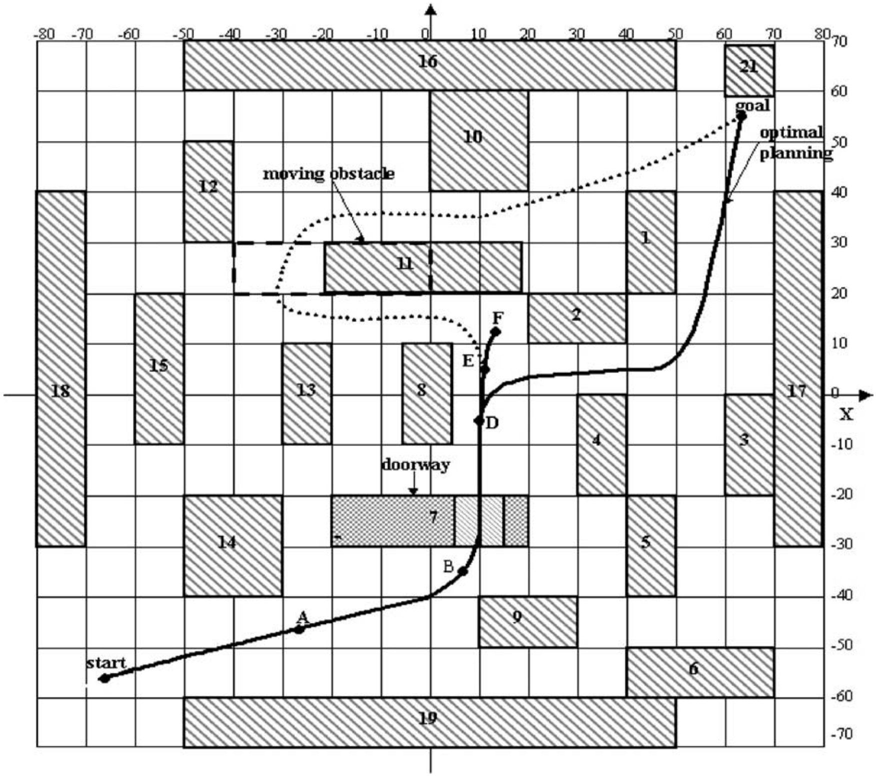

Fig. 11. Dynamic environment.

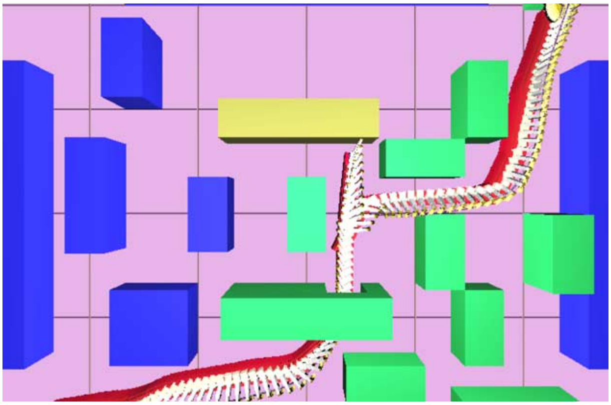

Fig. 12. Mobile manipulator navigation.

been obtained. The mobile platform position, orientation and linear velocity are shown in Figs. 13 and 14, and the arm manipulator joint angles are represented in Fig. 15.

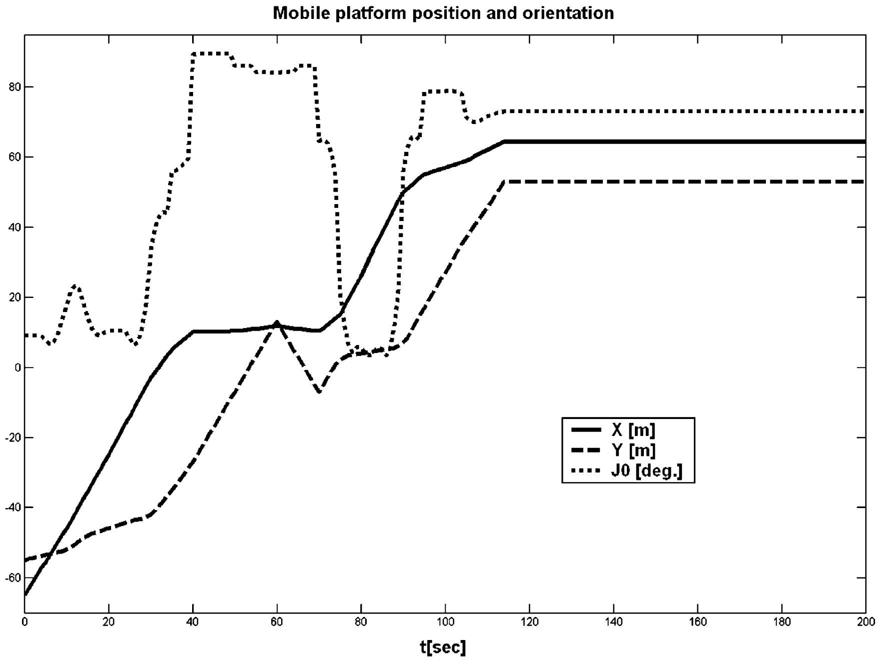

Fig. 13. Mobile platform position and orientation.

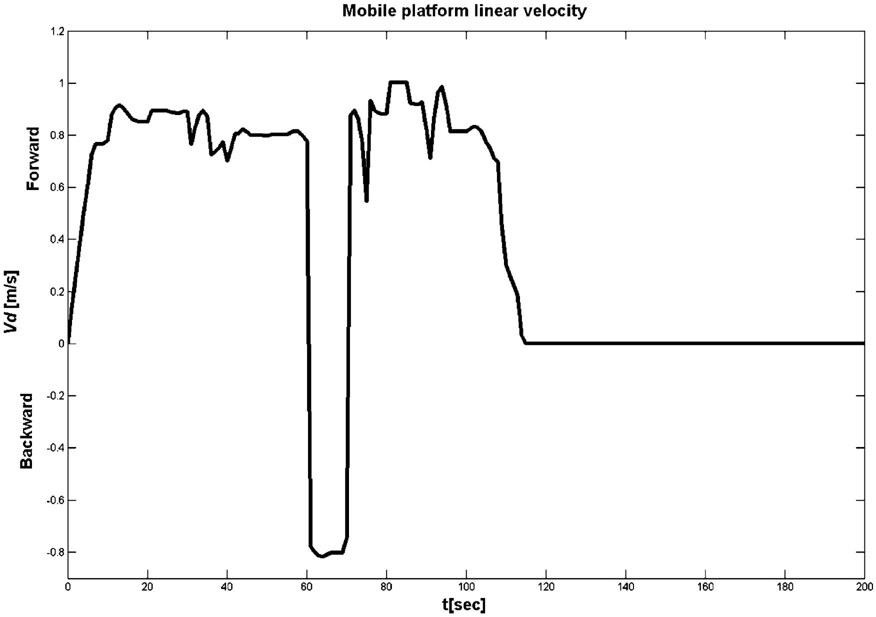

Fig. 14. Mobile platform linear velocity.

Fig. 15 shows that only the waist and shoulder joint angles vary during the vehicle motion. This intelligent phenomenon is allotted to our proposed algorithm, which determines the variation of joints necessary to the stability of over all electromechanical system during the motion of mobile platform. The mobile manipulator at the start position is shown in Fig. 16. In the entry of narrow passage (see Fig. 17), the manipulator is reconfigured to allow the mobile platform to move stable through a narrow passage. From Fig. 14, it can be deduced that the vehicle reduces its velocity during that operation (from $t = 36$ s to $t = 42$ s). Out of the doorway and before the robot meets a dynamic obstacle shown in Fig. 18 (42 s < t < 58 s), the joint angles of arm manipulator do not vary (refer to Fig. 15). At the same time, Fig. 14 shows that the moving platform linear velocity remains constant. The same phenomenon can be also observed when the robot moves back to the point E of the graph.

It can be observed from Fig. 18 that the robot moves back to the node E of the graph G when it meets a moving obstacle at the control point F shown in Fig. 11. This phenomenon is due to the separation of the bubbles between node E and goal point. With the help of global planner at the point E, the robot chooses an optimal path to reach its goal.

Fig. 19 shows the arm manoeuvre at the goal point. It can be observed that the mobile platform brings the arm at the preferred operating region so that the mobile manipulator remains stable when the manipulator moves its payload.

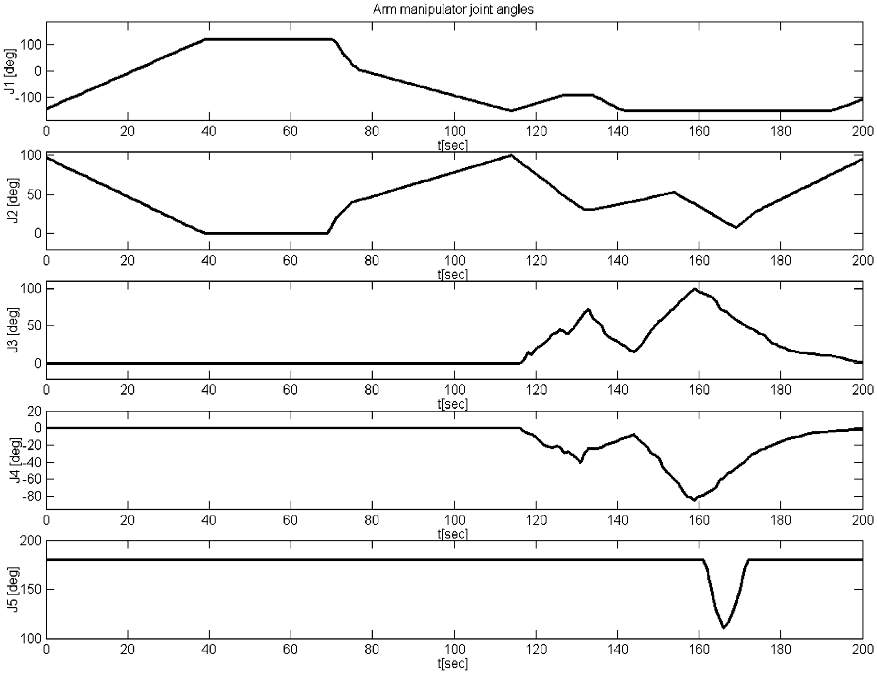

Fig. 15. Arm joint angles.

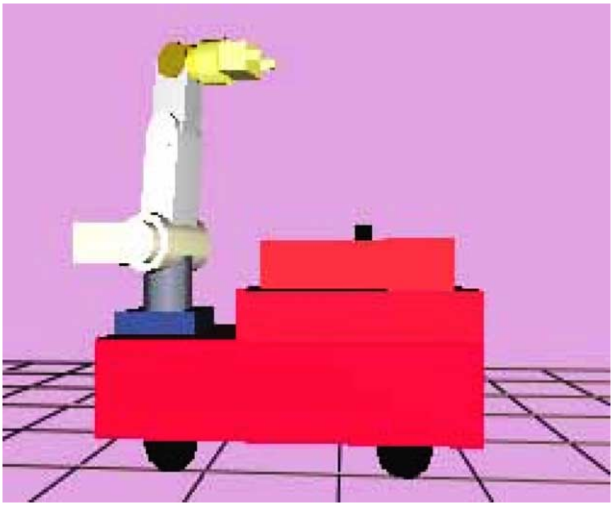

Fig. 16. Mobile manipulator at the start position.

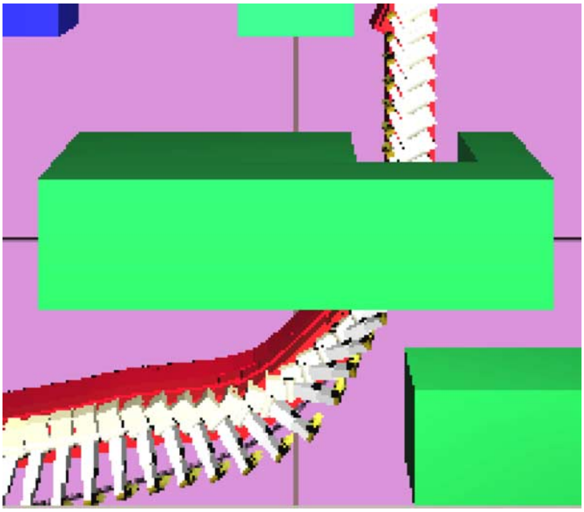

Fig. 17. Robot's navigation in the doorway.

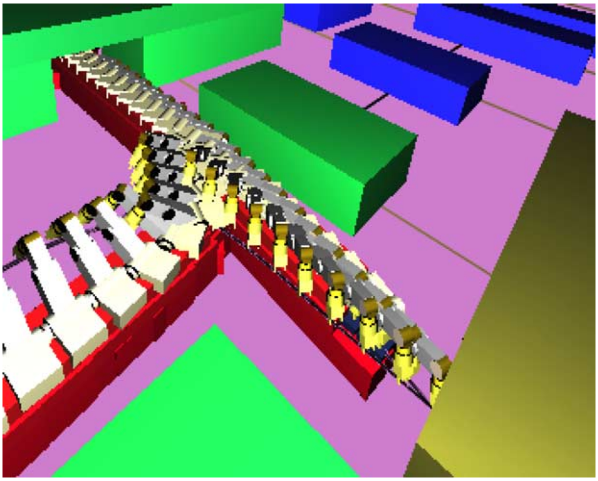

Fig. 18. Moving obstacle avoidance.

As a result of information received by the fuzzy local navigators from different sensors about the form and position of obstacles, the robot successfully completes a complex maneuver avoiding moving and stationary obstacles,

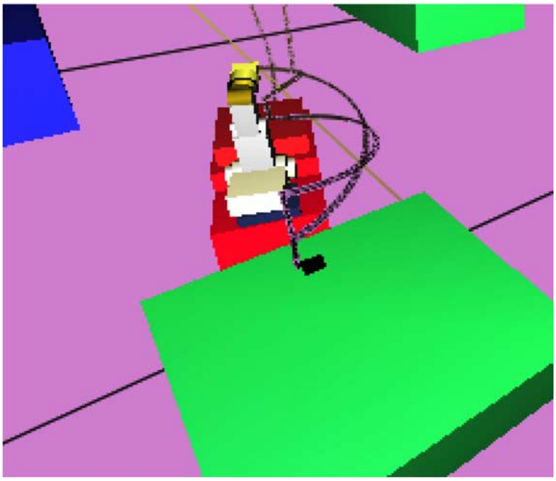

Fig. 19. Mobile manipulator path at the goal point.

and chooses an optimal path to reach its target. The manipulator is positioned at the optimal configuration and the mobile platform reduces its velocity to help the mobile manipulator to be stable during its motion so that it autonomously avoids obstacles and navigates through a doorway. This phenomenon shows the effective intelligent capability of the proposed motion-planning algorithm to deal with dynamic obstacles and makes our proposed intelligent motion planning very attractive.

6. Conclusion

Using a new intelligent motion control strategy that makes possible the integration of fuzzy obstacle avoidance, multisensor-based motions, and robust recurrent neural network control proposed previously, the simulation results, which need to be validated under real conditions, clearly demonstrate that the proposed strategy is an effective approach. Another important result is the effective intelligent motion capability of the proposed scheme. With the help of its "brain" (fuzzy-decision), which makes decision about any action of the robot, it is shown that the robot autonomously and stably chooses an optimal path to reach its target without collision with unknown and/or moving obstacles in unstructured working areas. This brings a high level of autonomy to the overall system, and makes the use of the controller very attractive for real-time fast and nonstop sensor-based guidance of intelligent mobile manipulators.

Acknowledgment

Dr. MBEDE would like to acknowledge the financial support by the Grant-in-Aid from the Japan Society for the Promotion of Science (JSPS).

References

[1] K. Althoefer, B. Krekelberg, D. Husmeier, L.D. Seneviratne, Reinforcement learning in a rule-based navigator for robotic manipulators, Neurocomputing 37 (1–4) (2001) 51–70.

[2] O. Brock, O. Khatib, Elastic strips: a framework for motion generation in human environments, The International Journal of Robotics Research 21 (12) (2002) 1031–1052.

[3] J.H. Chung, S.A. Velinsky, Modeling and control of a mobile manipulator, Robotica 16 (6) (1998) 607–613.

[4] J. Desai, V. Kumar, Motion planning for cooperating mobile manipulators, Journal of Robotic Systems 16 (10) (1999) 557–579.

[5] M.S. Erden, K. Leblebicioglu, U. Halici, Multi-agent system-based fuzzy controller design with genetic tuning for a mobile manipulator robot in the hand over task, Journal of Intelligent and Robotic Systems 39 (2) (2004) 287–306.

[6] Y. Gao, M. Joo Er, Modelling, control, and stability analysis of non-linear systems using generalized fuzzy neural networks, International Journal of Systems Science 34 (6) (2003) 427–438.

[7] H. Hagras, M. Colley, V. Callaghan, Learning and adaptation of an intelligent mobile robot navigator operating in unstructured environment based on a novel online fuzzy-genetic system, Journal of Fuzzy Sets and Systems 141 (1) (2004) 107–160.

[8] Q. Huang, K. Tanie, S. Sugano, Coordinated motion planning for a mobile manipulator considering stability and task constraints, The International Journal of Robotics Research 19 (8) (2000) 732–742.

[9] T. Kondo, K. Ito, A reinforcement learning with evolutionary state recruitment strategy for autonomous mobile robots control, Robotics and Autonomous Systems 46 (2) (2004) 111–124.

[10] J.B. Mbede, X. Huang, M. Wang, Robust neuro-fuzzy sensor-based motion control among dynamic obstacles for robot manipulators, IEEE Transactions on Fuzzy Systems 11 (2) (2003) 249–261.

[11] P. Melin, O. Castillo, A new method for adaptive control of non-linear plants using type-2 fuzzy logic and neural networks, International Journal of General Systems 33 (2–3) (2004) 289–304.

[12] E. Papadopoulos, Y. Gonthier, A framework for large-force task planning of mobile redundant manipulators, Journal of Robotic Systems 16 (3) (1999) 151–162.

[13] V. Pavlov, A. Timofeyev, Construction and stabilization of programmed movements of a mobile robot-manipulator, Engineering Cybernetics 14 (6) (1976) 70–79.

[14] D.T. Pham, X. Liu, Identification of linear and nonlinear dynamic systems using recurrent neural networks, Artificial Intelligence in Engineering 8 (1) (1993) 67–75.

[15] F.G. Pin, K.A. Morgansen, F.A. Tulloch, C.J. Hacker, K.B. Gower, Motion planning for mobile manipulators with a non-holonomic constraint using the FSP (full space parameterization) method, Journal of Robotic Systems 13 (11) (1996) 723–736.

[16] J.A. Reeds, R.A. Shepp, Optimal paths for a car that goes both forward and backwards, Pacific Journal of Mathematics 145 (2) (1990) 367–393.

[17] H. Seraji, A unified approach to motion control of mobile manipulators, International Journal of Robotics Research 17 (2) (1998) 107–118.

[18] H. Seraji, A. Howard, Behavior-based robot navigation on challenging terrain: a fuzzy logic approach, IEEE Transactions on Robotics and Automation 18 (3) (2002) 308–321.

[19] T. Siméon, J.-P. Laumond, C. Nissoux, Visibility-based probabilistic roadmaps for motion planning, Advanced Robotics 14 (6) (2000) 477–492.

[20] P. Svestka, M. Overmars, Probabilistic path planning, in: Robot Motion Planning and Control, in: J.P. Laumond (Ed.), Lecture Notes in Control and Information Sciences, Springer-Verlag, 1998.

[21] H.G. Tanner, S. Loizou, K.J. Kyriakopoulos, Nonholonomic motion planning for mobile manipulators, IEEE Transactions on Robotics and Automation 19 (1) (2003) 53–64.

[22] M. Tedder, D. Chamulak, L.-P. Chen, S. Nair, A. Shvartsman, I. Tseng, C.-J. Chung, An affordable modular mobile robotic platform with fuzzy logic control and evolutionary artificial neural networks, Journal of Robotic Systems 21 (8) (2004) 419–428.

[23] K. Watanabe, K. Sato, K. Izumi, Y. Kunitake, Analysis and control for an omnidirectional mobile manipulator, Journal of Intelligent and Robotic Systems 27 (1) (2000) 3–20.

[24] Y. Yamamoto, X. Yun, A modular approach to dynamic modelling of a class of mobile manipulators, International Journal of Robotic and Automation 12 (2) (1997) 41–48.