Trajectory Tracking and Obstacle Avoidance of Car-Like Mobile Robots in an Intelligent Space Using Mixed H_2/H_∞ Decentralized Control

Chih-Lyang Hwang and Li-Jui Chang

***Abstract*—In this paper, the trajectory tracking and obstacle avoidance of a car-like mobile robot (CLMR) within an intelligent space via mixed H_2/H_∞ decentralized control is developed. To obtain obstacle avoidance and trajectory tracking, two distributed charge-coupled device (CCD) cameras are established to realize the pose of the CLMR and the position of the obstacle. Based on the authority of these two CCD cameras, a suitable reference command for the proposed controller of the CLMR is planned online by the information of the CCD camera with higher authority. Many of the problems encountered by classic mobile robots are solved. Because an arbitrarily smooth trajectory can be approximated by a set of piecewise lines, the trajectory tracking of the piecewise line is addressed. The features of the proposed control are smaller energy consumption with bounded tracking error, attenuation of output disturbance, and improvement of control performance. The overall system contains two processors (i.e., personal computer and digital signal processor) with multiple sampling rates. Finally, a sequence of experiments including the comparison among the proposed control, PID control, different microprocessor control system, and different initial poses are carried out to confirm the usefulness of the suggested control system.**

***Index Terms*—Car-like mobile robot, decentralized control, intelligent space, mixed H_2/H_∞ optimization, microprocessor control, obstacle avoidance, trajectory tracking.**

I. Introduction

Recently, distributed control applications within an intelligent space (a space with various distributed sensors to obtain the information about the corresponding environment) are gaining importance. Such intelligent spaces are able to monitor what is occurring within themselves, to build their own models, to communicate with their inhabitants, and to act on the basis of decisions they make (e.g., [1]–[6]). In an intelligent space, many of the problems encountered by classic mobile robots are solved. For example, one of the biggest problems in classic mobile robot is the localization (e.g., [7]). Because the mobile robot does not have a fixed basis, a localization method with bounded errors is required. Unfortunately, most methods are weak in terms of disturbances, e.g., the slip of wheels, collisions, parameter changes. Secondly, the navigation of a mobile robot in such an environment needs to have information about the environment in a certain form. However, to handle such information and sensors for localization, high computational power is required by each robot. For instance, an online trajectory planning and obstacle avoidance developed by Yang and Meng [8] requires high computational power. Hence, it is difficult to implement these algorithms in the microprocessor of a mobile robot. Thirdly, different types of mobile robots in an environment need different software prepared for each robot, according to the resources that the robot possesses. Hence, a mobile robot tuned for a different environment is not easy and instant (e.g., [9]). Fourthly, the interference with each sensor and the countermeasure of the signal must be considered. Otherwise, the reliability of sensor-based control is reduced (e.g., [10]).

Based on this concept about the intelligent space, two distributed charge-coupled device (CCD) cameras are employed to provide the poses (i.e., position and orientation) of the mobile robot and the position of the obstacle. After the image processing in a personal computer (PC), the corresponding information is applied to plan a reference command for the controller in a car-like mobile robot (CLMR). This online planning trajectory is then transmitted to the CLMR by a wireless device. If the monitoring region becomes larger, the number of the CCD camera should be increased or the active CCD camera should be considered. In Section II, the controller design for the CLMR is described.

For simplifying the controller design, a linear discrete-time dynamic model for every subsystem in the nominal condition (i.e., one motor is for the steering orientation, and the other motor is for the forward or backward motion) is first obtained by the recursive least-squares parameter estimation. However, every subsystem assumed to be a linear dynamic system will be a challenge to design a decentralized control for the mobile robots (see [11], [12]). Due to the existence of interactions between two subsystems, modeling errors, and external loads caused by different operations, friction, or slippage, there are not many efficient decentralized controls for the mobile robot (e.g., [13], [14]).

Mixed H_2/H_∞ control problem has become a popular research topic in recent years (e.g., [15]–[17]). The controller design for the CLMR is based on our previous paper [17]. These important concepts are introduced next. A reference model is first employed to shape the transient response of the desired trajectory. The output error is defined as the difference between output of the reference model and that of the CLMR. The H_2-norm of the output error and weighted control input is then minimized to obtain a control such that small energy consumption with bounded tracking error for every subsystem is achieved [18]. The above design concept is important for the control problem of the CLMR with battery power. However, an output disturbance caused by the interactions between two subsystems, modeling errors resulting from linear system assumption, and external loads brought about by the motion in different road conditions, often deteriorates system performance or even results in instability. In this situation, the H_∞-norm of weighted sensitivity of the ith subsystem between output disturbance and system output is accomplished [19]. Although the effect of output disturbance for every subsystem is reduced or partially rejected, a better performance of CLMR can be improved by a switching control [19], [20]. Due to space constraint, the interested readers can refer to [17].

The overall control system includes two processors with multiple sampling rates. One PC using a sampling time 65 ms is applied to capture the image of the CLMR and obstacle in order to plan a reference command for the CLMR, and then, to transmit it to the CLMR. The other is a DSP implementing in the CLMR with a sampling time 10 ms to control two dc motors. Because an arbitrary smooth trajectory can be approximated by a set of piecewise lines, the trajectory tracking of the piecewise line is addressed. The authors think that this is the first paper that discusses the corresponding subject. It includes two modes: one is the approach mode, and the other is the fine-tune mode. Owing to the possible existence of (dynamic) obstacle in an intelligent space, the ways to judge the category of obstacle, and then, to avoid the corresponding obstacles are also discussed. Finally, the comparison between the proposed control and PID control of trajectory tracking in the presence (or absence) of obstacle is presented. The evaluation of different microprocessor control systems and different initial poses in intelligent space is also given to consolidate the usefulness of the proposed control system.

II. System Description

Fig. 1 shows the experimental setup of a CLMR in an intelligent space. The overall system contains a CLMR (including two dc motors,

Manuscript received March 1, 2006; revised December 15, 2006. Recommended by Guest Editors H.-P. Huang and F.-T. Cheng. This work was supported by the National Science Council of Taiwan, R.O.C., under Grant NSC93-2218-E-036-003.

The authors are with the Department of Electrical Engineering, Tamkang University, Taipei 25137, Taiwan, R.O.C. (e-mail: clhwang@mail.tku.edu.tw).

Digital Object Identifier 10.1109/TMECH.2007.897281

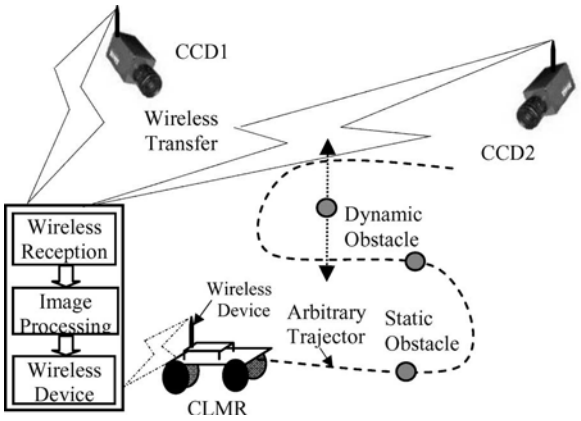

Fig. 1. Block diagram of the overall system.

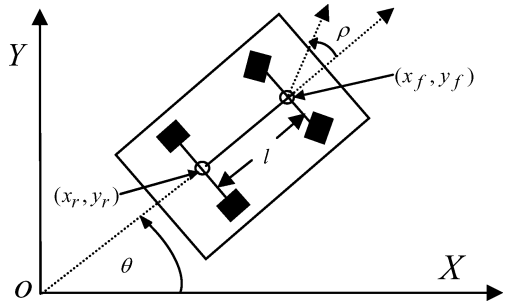

Fig. 2. Kinematics of the CLMR.

TABLE I
BASIC SPECIFICATIONS OF CLMR

Car-Like CLMR	Length	38.7*cm*
	Width	29.5*cm*
	Height	15*cm*
	Weight	5.5 *kg*
Front- and Rear-Wheel	Diameter	12.7*cm*
	Thickness	4*cm*
	Wheelbase	25.5*cm*

one microprocessor, one wireless device, and mechanism), two CCDs, and one PC (including image processing card and wireless device). The CLMR with two wheels driving system is depicted in Fig. 2. The rear wheels are fixed parallel to the car chassis, and allowed to roll or spin but not slip; two front wheels are parallel, and can simultaneously turn to the right or left. Front-wheel and rear-wheel are individually driven by the same permanent magnet dc motor. The only difference is their gear ratio; one is 190:1, and the other is 36:1. Table I shows the basic specifications of the CLMR. For a CLMR size and shape, its location in the 2-D cartesian workspace W can be uniquely determined by the spatial position (x, y) of the base point and the orientation angle θ with respect to the base point (see Fig. 2). The kinematics constraint of a nonholonomic mobile robot is described as (see, e.g., [13], [14], [21])

$$-\dot{x}(t)\sin\theta(t) + \dot{y}(t)\cos\theta(t) = 0. \tag{1a}$$

The velocity parameters of the CLMR are expressed as

$$\dot{x}_r(t) = v\cos(\theta), \dot{y}_r(t) = v\sin(\theta), \text{ and } \dot{\theta}(t) = v\tan(\rho)/l \tag{1b}$$

where (x_r, y_r) is the position of the rear wheel center of the CLMR, $\theta(t)$ is the angle between the orientation of the CLMR and the X-direction, ρ is the steering angle with respect to the frame of the CLMR (the counterclockwise direction is defined as positive steering angle), v is the speed of the longitude (or rear wheel), and l is the distance of the wheelbase. Similarly, the kinematics of the front wheel of the CLMR is described as

$$\dot{x}_f(t) = v\cos(\theta), \dot{y}_f(t) = v\sin(\theta), \text{ and } \dot{\theta}(t) = v\tan(\rho)/l. \tag{1c}$$

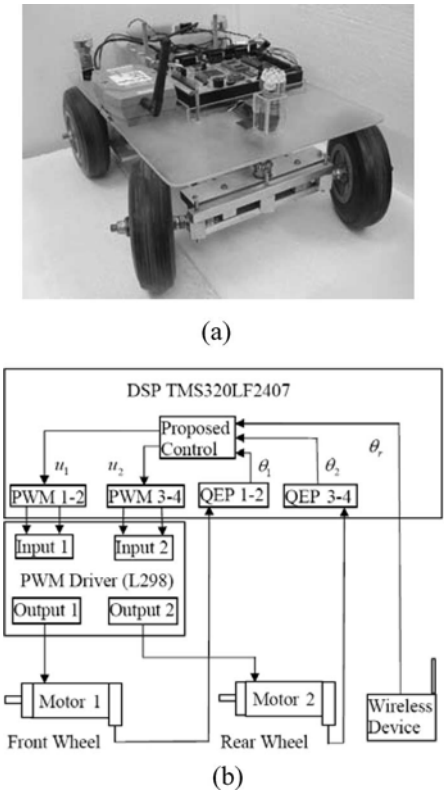

Fig. 3. Realization of the CLMR.

The specifications of the dc motor system are described as: 1) motor (A-max32 motor of Maxon Company)—graphite brushes, power 20 W, no load speed 6420 r/min, maximum continuous torque 0.0473 N·m, maximum continuous current 1.35 A, torque constant 0.0351 N·m/A, mechanical time constant 0.014 s, rotor inertia 43 g·cm^2, length 61.5 mm, and mass 242 g; 2) gear box (GP-32 C of Maxon Company)—gear ratio 190:1 (front-wheel) and 36:1 (rear-wheel), maximum permissible radial load (12 mm from flange) 140 N, maximum continuous torque at gear output 6 N·m, maximum efficiency 70%, mass inertia 0.7 g·cm ^{2}, length 43 mm, and weight 194 g; 3) digital encoder—$2 + 1$ index channel, 500 counts per turn, maximum operating frequency 100 kHz, length 18.3 mm, and weight 90 g; and 4) drive—L298, which is an integrated monolithic circuit with a high voltage (up to 46 V) and current (up to 4 A) dual full-bridge driver.

The realization of the CLMR is shown in Fig. 3. The core of the CLMR is the DSP of TMS320LF2407. It includes general purpose (GP) timers, analog/digital converter (ADC), full-compare/PWM units, capture units, quadrature-encoders pulse (QEP), series port such as series communication interface (SCI), series peripheral interface (SPI), control array network (CAN), and interface of joint test action group (JTAG).

The wireless device SST-2450 of ICP DAS Company is a spread-spectrum radio modem controlling an RS-232/RS-485 interface port.

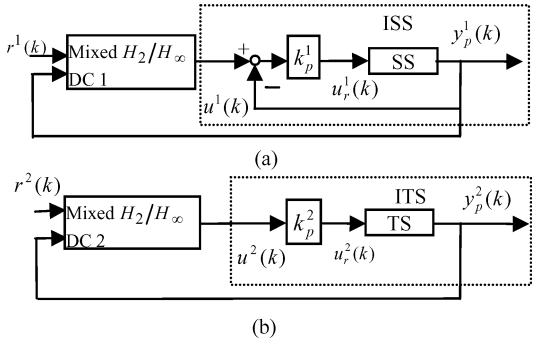

Fig. 4. Improved CLMR (ICLMR) by a proportional (feedback or forward) gain.

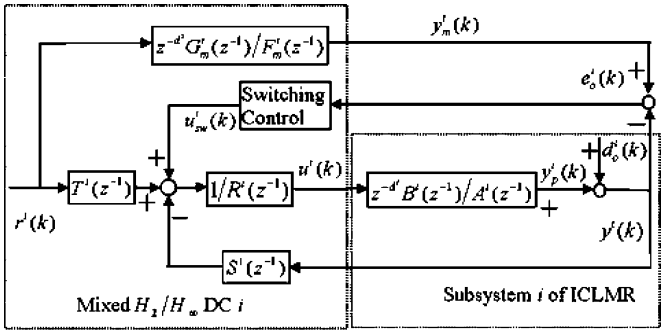

Fig. 5. Control block diagram of the ith subsystem of the ICLMR.

It is used for data acquisition and transmission between the PC and the CLMR. Its operating frequency range is between 2410.496 and 2471.936 MHz. The channel spacing is 4.096 MHz.

The Matrox Meteor-II card is applied as an image processing card with wireless reception. It is a monochrome and component RGB analog frame grabber for standard and nonstandard video acquisition. It is also available in a PCI or PC/104-Plus form factor, both of which can use a Matrox Meteor-II MJPEG module for compression and decompression of monochrome and color images. The board features six software-selectable input channels on which two components of red-green-blue (RGB) or six monochrome cameras can be attached. It supports acquisition from one camera at a time or simultaneous acquisition from up to three gen-locked RS-170/CCIR cameras (i.e., RS-170RGB); it supports both single and dual-tap configurations. The software Matrox MIL-Lite 6.0 developed by Matrox possesses many modules, which can be used to recognize the image.

III. System Modeling, Control Concept and Problem Formulation

A. System Modeling of CLMR

It is assumed that a CLMR is expressed as

$$Y_p(k) = F(Y_p(k-1), \ldots, Y_p(k-n_y), U(k-1), \ldots, U(k-n_u)) \tag{2}$$

where $Y_p(k) = \left[y_p^1(k)\ \ y_p^2(k)\right]^T, U(k) = \left[u^1(k)\ \ u^2(k)\right]^T$ are the system output and control input of the CLMR, respectively, and $F(\cdot)$ denotes an unknown function. In order to design a decentralized controller for a CLMR, two dynamic models are required. Before modeling the dc motors, a proportional feedback gain $k_p^1 = 65$ V/rad for steering subsystem [see Fig. 4(a)] is employed to adjust its pole and dc gain. It is called "improved steering subsystem (ISS or ICLMR1)". Similarly, a forward gain $k_p^2 = 100$ V/rad in Fig. 4(b) is applied to obtain a unit dc gain of angular velocity subsystem. It is called "improved translating subsystem (ITS or ICLMR2)". The pseudorandom binary signal (PRBS) with suitable amplitude and period is employed to drive each subsystem of the CLMR at one time. Then, the input and output pairs of data $\{u^i(k), y_p^i(k)\}, i = 1, 2$ are achieved. These input/output pairs are individually fed into the least-squares parameter estimation algorithm (e.g., [22]). After the model verification, an appropriate learned model for the ith subsystem is expressed as

$$y_p^i(k) = z^{-d^i} B^i(z^{-1}) u^i(k)/A^i(z^{-1}), \quad i = 1, 2 \tag{3}$$

where

$$A^i(z^{-1}) = 1 + a_1^i z^{-1} + a_2^i z^{-2},\ B^i(z^{-1}) = b_0^i + b_1^i z^{-1},\ d^i = 1 \tag{4a}$$

$$a_1^1 = -1.351, a_2^1 = 0.455,\ b_0^1 = 0.04558, b_1^1 = 0.05471 \tag{4b}$$

$$a_1^2 = -0.655, a_2^2 = -0.1208, b_0^2 = 0.0581, b_1^2 = 0.06972. \tag{4c}$$

These two subsystems are stable and in nonminimum phase, and the polynomials $A^i(z^{-1})$ and $B^i(z^{-1})$ for $i = 1, 2$ are coprime. Based on the approximation theory, the CLMR (2) is approximated by the aforementioned subsystems with output disturbance $d_o^i(k), i = 1, 2, \ldots, n$ given by

$$\begin{aligned} Y_p(k) &= F(Y_p(k-1), \ldots, Y_p(k-n_y), U(k-1), \ldots, U(k-n_u)) \\ &= \begin{pmatrix} z^{-d^1} B^1(z^{-1}) u^1(k)/A^1(z^{-1}) + d_o^1(k) \\ z^{-d^2} B^2(z^{-1}) u^2(k)/A^2(z^{-1}) + d_o^2(k) \end{pmatrix} \end{aligned} \tag{5a}$$

where the output disturbance is relatively bounded as

$$\left|d_o^i(k)\right| \le \alpha_0^i + \sum\nolimits_{j=1}^{2} \alpha_j^i \left|u^j(k-1)\right| \tag{5b}$$

where α_0^i and $\alpha_j^i, i, j = 1, 2$ are bounded. In summary, two individual linear subsystems in the face of output disturbance are employed to represent the dynamics of the front wheel and rear wheel of the CLMR system.

B. Concept of Controller Design

Because the trajectory tracking of a CLMR includes different setpoint positions control for the front wheel and velocities control for the rear wheel, the reference command to be tracked for the ith subsystem, i.e., $r^i(k)$, is assigned as

$$r^i(k) = G_r^i(z^{-1})\delta(k)/F_r^i(z^{-1}) \tag{6}$$

where $\delta(k)$ is a Dirac delta function; $G_r^i(z^{-1})$ and $F_r^i(z^{-1})$ for $i = 1, 2$ are coprime. In this paper, $G_r^i(z^{-1}) = 1$ and $F_r^i(z^{-1}) = 1 - z^{-1}$. The proposed control of the ith subsystem of the CLMR is assumed as (see Fig. 5)

$$R^i(z^{-1}) u^i(k) = -S^i(z^{-1}) y^i(k) + T^i(z^{-1}) r^i(k) + u_{\mathrm{sw}}^i(k) \tag{7a}$$

$$y^i(k) = y_p^i(k) + d_o^i(k) \tag{7b}$$

where $d_o^i(k)$ is caused by the interactions, modeling error, and external load; $y^i(k)$ is the output of the ith subsystem; the polynomials $R^i(z^{-1})$, $S^i(z^{-1})$, and $T^i(z^{-1})$ are found to achieve an equivalent control of the ith subsystem; and $u_{\mathrm{sw}}^i(k)$ is the switching control of

the ith subsystem. Assume that the input/output relationship between $r^i(k)$ and $y_m^i(k)$, for a stable reference model, is written as

$$y_m^i(k) = z^{-d^i} G_m^i(z^{-1})/F_m^i(z^{-1}) r^i(k) \tag{8}$$

where $G_m^i(z^{-1})$ and $F_m^i(z^{-1})$ for $i = 1, 2$ are coprime, and $F_m^i(z^{-1})$ is a stable monic polynomial, i.e., $F_m^i(0) = 1$. The purpose of using the reference model is to shape the response of the ith closed-loop subsystem.

The output error of the ith subsystem is defined as

$$e_o^i(k) = y_m^i(k) - y^i(k). \tag{9}$$

Then, the response of $e_o^i(k)$ from the inputs $r^i(k)$, $d_o^i(k)$, and $u_{\rm sw}^i(k)$ is obtained from (5)–(9), i.e.,

$$e_o^i(k) = L^i(z^{-1}) r^i(k) - V^i(z^{-1}) d_o^i(k) - D^i(z^{-1}) u_{\rm sw}^i(k) \tag{10a}$$

where

$$L^i(z^{-1}) = z^{-d^i}\left[G_m^i(z^{-1})/F_m^i(z^{-1}) - B^i(z^{-1}) T^i(z^{-1})/A_c^i(z^{-1})\right] \tag{10b}$$

$$V^i(z^{-1}) = A^i(z^{-1}) R^i(z^{-1})/A_c^i(z^{-1}) \tag{10c}$$

$$D^i(z^{-1}) = z^{-d^i} B^i(z^{-1})/A_c^i(z^{-1}). \tag{10d}$$

The polynomial $A_c^i(z^{-1})$ denotes the characteristic polynomial of the ith closed-loop subsystem

$$A_c^i(z^{-1}) = A^i(z^{-1}) R^i(z^{-1}) + z^{-d^i} B^i(z^{-1}) S^i(z^{-1}). \tag{11}$$

The following two cost functions in the H_2-norm and H_∞-norm spaces are applied to design the proposed control.

$$J_1 = \left\| E_o^i(z^{-1}) + z^{-d^i} W_1^i(z^{-1}) U^i(z^{-1}) \right\|_2$$

and

$$J_2 = \left\| W_2^i(z^{-1}) V^i(z^{-1}) \right\|_\infty \tag{12}$$

where the weighted functions $W_1^i(z^{-1})$ and $W_2^i(z^{-1})$ denote two suitable rational functions; $E_o^i(z^{-1})$ and $z^{-d^i} U^i(z^{-1})$ are the pulse transfer functions of $e_o^i(k)$ and $u^i(k - d^i)$, respectively. In brief, the proposed controller design for a CLMR is described as follows.

1) The equivalent control [i.e., the polynomials $R^i(z^{-1})$, $S^i(z^{-1})$, and $T^i(z^{-1})$] is obtained by the satisfaction of the following two requirements.
   a) For $d_o^i(k) = u_{\rm sw}^i(k) = 0$, the cost function J_1 is minimized.
   b) For $u_{\rm sw}^i(k) = 0$, the cost function J_2 is simultaneously minimized.
2) The switching control [i.e., $u_{\rm sw}^i(k)$] is also designed to enhance system performance.

C. Problem Formulation

The image system to detect the response of the CLMR is introduced here. First, three light emitter diodes (LEDs) are set at suitable places; three corresponding points on the image plane to represent these three positions with respect to the world coordinate, i.e., (x_1, y_1), (x_2, y_2), and (x_3, y_3) in Fig. 6] are attained. Based on (13), the pose of the geometry center of the CLMR in the world coordinate at the kth sampling interval [i.e., $(x(k), y(k), \theta(k))$] are then accomplished

$$\theta(k) = \tan^{-1}\{[(y_2+y_3)/2 - y_1]/[(x_2+x_3)/2 - x_1]\} \tag{13a}$$

$$(x(k), y(k)) = ((x_2 + x_3)/4 + x_1/2, (y_2 + y_3)4 + y_1/2). \tag{13b}$$

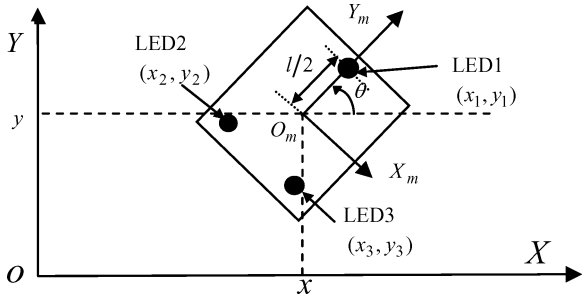

Fig. 6. Three LEDS for estimating the pose of CLMR.

It is not necessary that the experiments must be in a dark room. The purpose of this arrangement is to simplify the procedure of the image processing. In this paper, the interpolation method to obtain the world coordinate (OXY) from the image coordinate $(O_i X_i Y_i)$ is used. This is a real-world plane grabbed by the CCD with the coordinate $(O_c X_c Y_c)$; the point (320 480) pixel on the image plane is defined as (0, 0) cm on the world coordinate. Because a CCD does not face squarely to a plane in the world coordinate, an image of plane in the world coordinate becomes a trapezoid. The visible area by two CCDs possesses two trapezoid areas—the first trapezoid has the length of upper side and lower side as 244 and 144 cm, respectively, and a depth of 300 cm while the second trapezoid has the length of upper side and lower side as 241 and 141 cm, respectively, and a depth of 295 cm; an overlapped region with 24 cm; and an offset of 12 cm (i.e., these two trapezoids are not symmetric).

To begin with, two dc servo motors for the front and rear wheels of the CLMR are controlled by individual mixed H_2/H_∞ decentralized control. Although two dc motors are successfully controlled by decentralized control, the dynamics of the CLMR is different from that of the dc motor, e.g., slip, friction, coupling effect, and external load. In this situation, the dynamics of the two dc motors is in the face of uncertainties. However, the proposed control containing the optimal and enhanced robustness can cope with these uncertainties (see the third paragraph of Section I).

The main goal of this study is to investigate a mixed H_2/H_∞ decentralized control for a CLMR in an intelligent space. These experiments are categorized into the following six cases:

1) to track a trajectory of piecewise lines of the CLMR;
2) to track the same trajectory of part 1) with two static obstacles;
3) to track the same trajectory of part 1) with one static obstacle and one dynamic obstacle in the X-direction;
4) to compare the proposed control and PID control;
5) to compare TMS320LF2407 by TI with PWM interface to drive the two motors, and ARNUX 7525A by ANCHER using Linux operating system and DA interface to drive the two motors; and
6) to provide different initial poses for the proposed control system.

IV. Experiment

There are two divisions to discuss the experiment. Section IV-A is experimental preliminaries. Section IV-B discusses experimental results.

A. Experimental Preliminaries

1) Pose Estimation: There is an overlap between the ends of two trapezoid areas with width of 24 cm. The control authority in this overlap is considered here. The authority is based on the distinguishable error; higher distinguishable error indicates higher control authority.

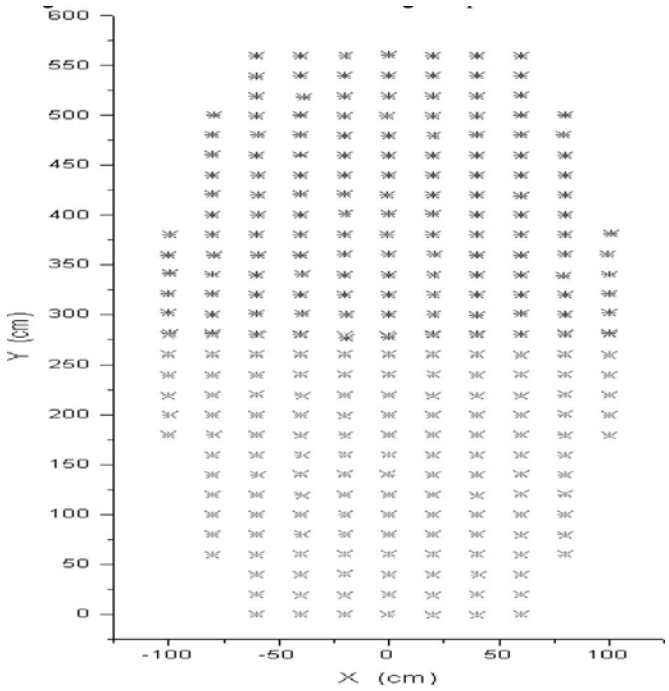

Fig. 7. Comparison between the real position and estimated position by CCD of world coordinate (+ real position visible by CCD1; × estimated position by CCD1; + real position visible by CCD2; * estimated position by CCD2).

Higher authority of one CCD is in charge of the CLMR. The centerline of the overlap is set as the equal control authority for the corresponding two CCDs. Finally, the real position of the CLMR in the world coordinate and the estimation by the image of CCD1 and CCD2 is shown in Fig. 7. The maximum estimation error is about 2 cm, which is acceptable, and merely occurs in the periphery of the trapezoid.

According to the restriction of grabbing frequency for a CCD, the sampling time for image processing and trajectory planning is set as 65 ms. While the CLMR is in the first trapezoid area, the pose of the CLMR or the position of (dynamic) obstacle is estimated by CCD1. After trajectory planning, the reference command for the CLMR is sent by a wireless device. Similarly, as the CLMR is in the second trapezoid area, the pose of the CLMR or the position of obstacle is estimated by CCD2; the planned reference command is also sent by the same wireless device. However, when the CLMR is inside the overlap region, two CCDs can grab the image, and the pose of the CLMR or obstacle is estimated either by CCD1 or CCD2.

2) Tracking Mode of Trajectory: Tracking mode of trajectory is of two kinds: a) approach mode; and b) fine-tune mode. The approach mode is to drive the CLMR in the neighborhood of piecewise line trajectory. The fine-tune mode is applied to force the CLMR on the piecewise line trajectory as near as possible. It is well known that the relative angle between the orientation of the CLMR and that of the piecewise line trajectory is the index for the desired steering angle (i.e., ρ_d). Certainly, it is also affected by whether the CLMR is in the left or right region of the piecewise line. However, the tracking mode of trajectory for the left or right region is analogous. For brevity, only the motion of the CLMR in left region is explained in Fig. 8. The content of large circle with dash–dotted line of Fig. 8 indicates the concept of fine tune mode. After the CLMR is in the fine tune mode, the trajectory tracking of piecewise line is illustrated in Fig. 9. The turning angle δ_{i-1} of the $(i-1)$th segment is first defined as the relative angle between the ith segment and the $(i-1)$th segment. The turning point p_{i-1} or

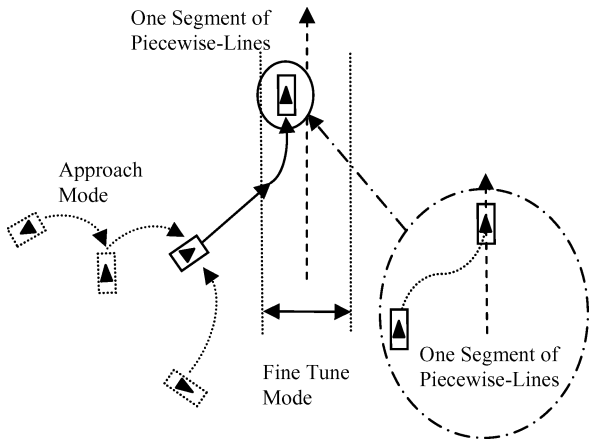

Fig. 8. Illustration of trajectory tracking mode.

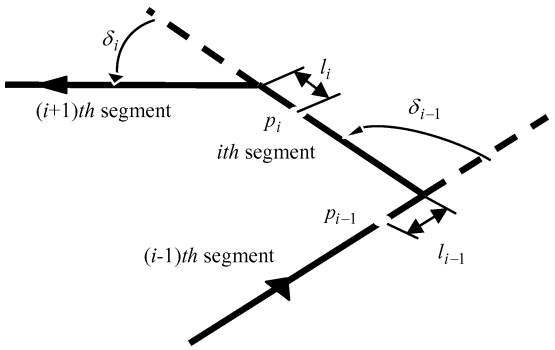

Fig. 9. Illustration of tracking a trajectory of piecewise lines.

turning length l_{i-1} of the $(i-1)$th segment in Fig. 9 is represented as the point or the position, starting to turn into the ith segment. Based on the analysis of our system, the following relation $l_{i-1} = c_2(\delta_{i-1})^{c_1}$, where $c_1 = 0.6$ cm, $c_2 = 41.5$ cm, and the unit of δ_{i-1} is in radian, is employed to track a trajectory of piecewise line. Due to the existence of minimum turning radius, the turning angle δ_i should not be over $90\,^\circ$; otherwise, the tracking performance will deteriorate.

3) Strategy of Obstacle Avoidance: The strategy for the avoidance of static and dynamic obstacles is introduced. When a CLMR is in the presence of static and dynamic obstacles, the strategy can be simultaneously employed to avoid the obstacles. The strategy for the avoidance of static obstacle in Fig. 10 is described as follows.

a) As the distance between the corresponding point of the CLMR and the obstacle is smaller than $d_{1,\min}$ (refer to Fig. 11 for the illustration of d_1), the CLMR starts avoiding the corresponding obstacle. That is, the operation of the CLMR is in a mode of static obstacle avoidance. It is not limited to one static obstacle.

b) As the minimum distance (refer to Fig. 12) between the corresponding point of the CLMR and the obstacle is greater than $d_{2,\min}$, then $\rho = 0$ so that the CLMR moves in a straight line. The minimum distance in the front and back sections in Fig. 12 is set to zero so that the CLMR is in the mode of the static obstacle avoidance.

c) As $|\beta - \alpha| > 90^\circ$, the operation of the CLMR returns to the tracking mode of trajectory.

Similarly, the strategy for the avoidance of dynamic obstacle in Fig. 13 is depicted as follows.

a) As the value $\bar{y}$ of the CLMR is smaller than $y_{\min}$, the CLMR stops for the multiple of sampling time (i.e., $h = 65$ ms).

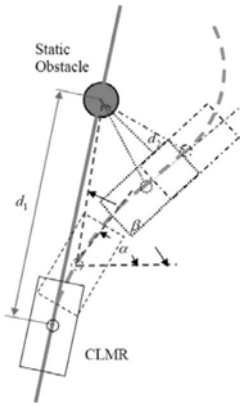

Fig. 10. Strategy for the avoidance of static obstacle.

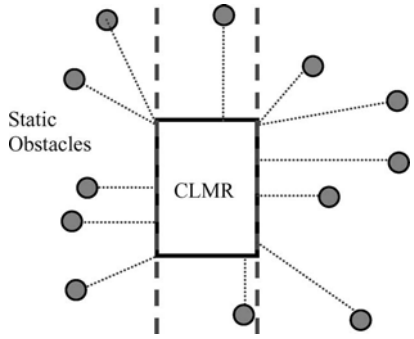

Fig. 11. Illustration of starting to avoid obstacle in Fig. 10.

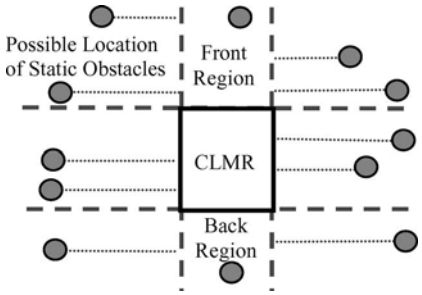

Fig. 12. Illustration of the relation between minimum distance and distance d_2 in Fig. 10.

b) The CLMR starts to track the specific trajectory when either of the following conditions is satisfied.
   i) $\bar{x} > x_{\text{depature}}$, where $\bar{x}$ denotes the distance beween the intersection of trajectory and the center of the CLMR, departing from the intersection, and x_{depature} represents an assigned distance to prevent the CLMR from bumping against dynamic obstacle.
   ii) $\bar{\bar{x}} > x_{\text{arrival}}$, where $\bar{\bar{x}}$ denotes the distance between the intersection of trajectory and the center of the CLMR arriving at the intersection, and x_{arrival} denotes an assigned distance to prevent the CLMR from bumping against dynamic obstacle.

In the beginning, the images grabbed by two CCDs for a time interval are accomplished to judge whether the visible area (or intelligent space) encounters with the static or dynamic obstacle. If the obstacle is static, the corresponding position (x, y) is applied for the trajectory planning to avoid the obstacle (see Figs. 10–12). If the obstacle is dynamic, its

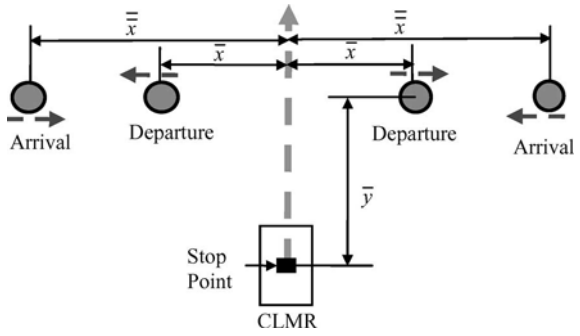

Fig. 13. Strategy for the avoidance of dynamic obstacle.

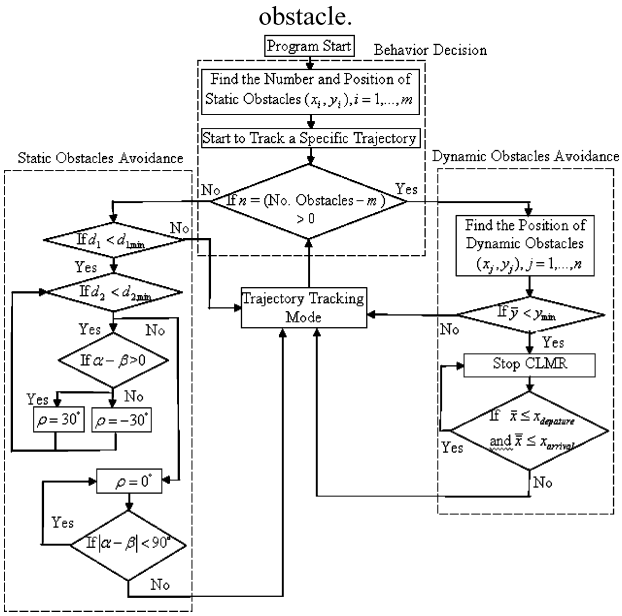

Fig. 14. Flowchart of the trajectory tracking and obstacle avoidance.

position is estimated in every sampling interval (i.e., $h = 65$ ms) for the avoidance of dynamic obstacle in Fig. 13. In this paper, the parameters for the (dynamic) obstacle avoidance are set as follows: $d_{1,\min} = 55$ cm, $d_{2,\min} = 10$ cm, $x_{\text{departure}} = 20$ cm, $x_{\text{arrival}} = 60$ cm, and $y_{\min} = 60$ cm. To gain the concept of this paper, a flow chart for trajectory tracking and obstacle avoidance is depicted in Fig. 14.

B. Experimental Results

1) Performance of Two DC Servo Motors: The CLMR is first raised to evaluate the control performance of two dc motors. The responses for different desired steering angles (i.e., ρ_d) and desired forward or backward velocities (i.e., v_d) by using the following parameters for the equivalent control: $G_m^1(z^{-1})/F_m^1(z^{-1}) = (0.1909 + 0.2291z^{-1})/(1 - 0.7z^{-1} + 0.12z^{-2}), W_1^1(z^{-1}) = 0.003/(1 + 0.3z^{-1})$, $W_2^1(z^{-1}) = (1 - 0.92z^{-1})/(1 - 0.85z^{-1})$, $G_m^2(z^{-1})/F_m^2(z^{-1}) = (0.0545 + 0.0655z^{-1})/(1 - 1.3z^{-1} + 0.42z^{-2}), W_1^2(z^{-1}) = 0.002/(1 + 0.95z^{-1})$, $W_2^2(z^{-1}) = (1 - 0.95z^{-1})/(1 - z^{-1})$, and the following parameters for the switching control: $\lambda_0^i = \lambda_{11}^i = \lambda_{12}^i = 0.001$, $\lambda_2^i = 0.01$, $\varepsilon^i = 0.7$, $\lambda^i = 0.01$, $\xi^i(k) = |, \xi_1^i(k) + 1.25g_1^i(k)\{1 - 0.995e^{-1000|e_o^i(k)|}\}$, are satisfactory (refer to [17]). For simplicity, these are omitted.

2) Performance of the CLMR in an Intelligent Space: The experiments described in this paper are categorized into the six cases as mentioned in Section III-C.

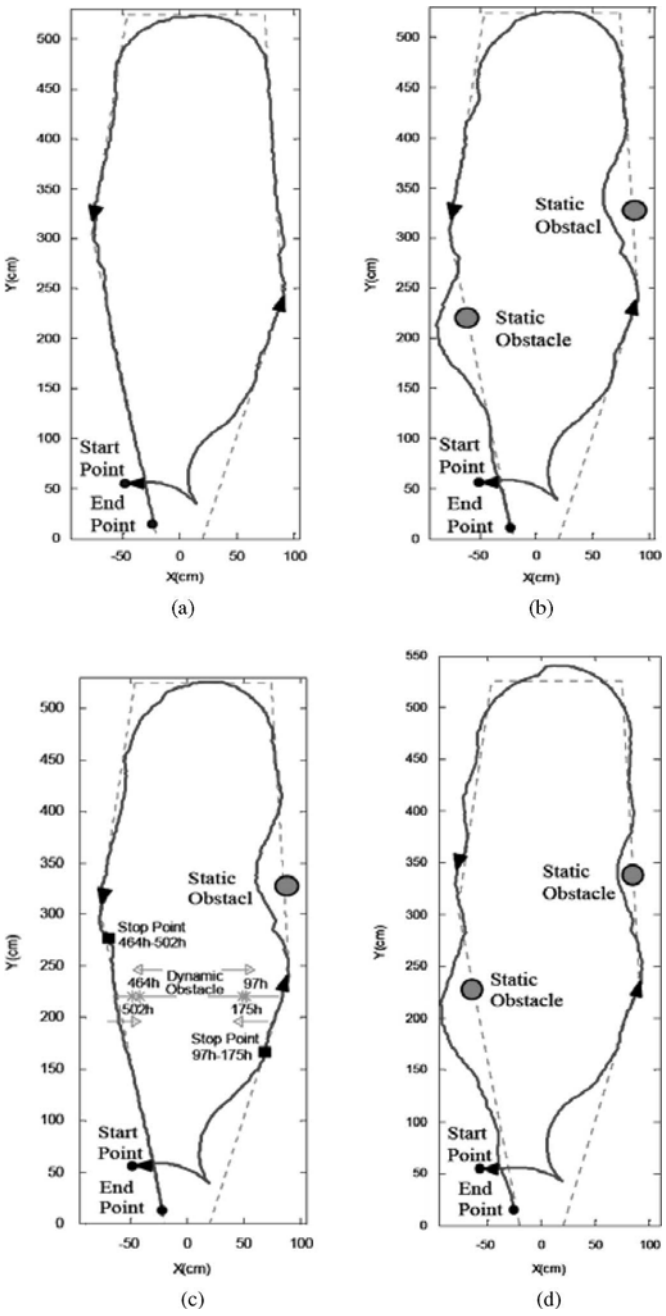

Fig. 15. Responses of the trajectory tracking of the CLMR in the intelligent space for: (a) proposed control without obstacle; (b) proposed control with two static obstacles; (c) proposed control with one static obstacle and one dynamic obstacle; and (d) PID control for the case (b).

The positions of two static obstacles with radius 7 cm are $(X, Y) = (95.4, 327.2)$ and $(-53.8, 219.8)$ cm. The piecewise line trajectory to be tracked is made up by five segments of straight line (i.e., the dash line in Fig. 15). The coordinates of five segments from the start point to the end point are $(20, 0), (90, 240), (74, 525), (-46, 525), (-77, 305)$ and $(-20, 0)$ cm.

Then, the corresponding responses of six the above-mentioned cases are shown in Fig. 15. The response in the vicinity of the overlap region (e.g., $275 < Y < 325$ cm) is often poor due to the discontinuity and poor quality of the image system. Because a minimum turning radius for a CLMR always exists, the tracking performance in the third segment of piecewise line is also poor. Under this circumstance, an omnidirectional mobile robot (e.g., [23]) should be considered. Although the trajectory of piecewise lines to be tracked is in the periphery of intelligent space possessing poor quality for pose estimation, the satisfactory performances in Fig. 15(a)–(c) consolidate the usefulness of our control system. First, a PID control, i.e., $u_{\rm PID}(k) = K_p \left\{ e(k) + T_c \sum e(k)/T_i + T_d \left[e(k) - e(k-1) \right]/T_c \right\}$, where $T_c = 0.01$ s, $K_p = 25.5$, $T_i = \infty, T_d = 0.02$ s for ISS; $K_p = 50.8$, $T_i = 0.078$ s, $T_d = 0.02$ s for the ITS, is applied to control two dc motors. Its unloaded response is almost the same as that of the proposed control. However, its response of trajectory tracking in the presence of two static obstacles in the intelligent space is shown in Fig. 15(d), which is poorer than that shown in Fig. 15(b). The reason is that the huge uncertainties can not be tackled by PID control. The other cases have the same result; for simplicity, these are omitted. In the end, the CLMR uses the microprocessor with embedded Linux platform (ARNUX 7525A by ANCHER) with expansion circuit to drive the two motors. This expansion circuit integrates the following three circuits: 12-bits D/A converter (AD7541A), 16-bits decoder (HCTL2020), and 8-bits A/D converter (ADC0804). The responses for cases in Fig. 15(a)–(c) using the above processor control are almost the same. For brevity, these are left. Finally, different initial poses for cases in Fig. 15(a)–(c) are also investigated. They are acceptable; for simplicity, these are also omitted. If the monitoring region is larger, the number of the CCD camera should be increased or the active CCD camera should be considered.

V. Conclusion

In this paper, a mixed H_2/H_∞ decentralized control for a CLMR in an intelligent space is established to obtain a piecewise line trajectory tracking and obstacle avoidance. A trajectory of piecewise line is employed to approximate an arbitrary trajectory, which is smooth. Many of the problems encountered by classic mobile robots are solved; e.g., localization, information about the environment, high computational power, different software, and sensor-based control for each mobile robot are not required. Due to the discontinuity and poor quality of image system in the neighborhood of the overlap region, its performance is a little poor. Because a minimum turning radius for a CLMR always exists, the tracking performance for the piecewise line with large turning angle is poor. Although the piecewise line to be tracked is in the periphery of intelligent space, possessing poor quality for the pose estimation, the performance is acceptable. In addition, the comparison among the proposed control, PID control, different microprocessor control systems, and different initial poses confirms the effectiveness of the suggested control system. The results of this paper can be extended to the number of CCD cameras greater than two.

References

[1] B. Sinopoli, C. Sharp, L. Schenato, S. Schaffert, and S. S. Satry, "Distributed control applications within sensor networks," *Proc. IEEE*, vol. 91, no. 8, pp. 1235–1246, Aug. 2003.

[2] T. Yamaguchi, E. Sato, and Y. Takama, "Intelligent space and human centered robotics," *IEEE Trans. Ind. Electron.*, vol. 50, no. 5, pp. 881–889, Oct. 2003.

[3] J. H. Lee and H. Hashimoto, "Controlling mobile robots in distributed intelligent sensor network," *IEEE Trans. Ind. Electron.*, vol. 50, no. 5, pp. 890–902, Oct. 2003.

[4] F. Amigoni, N. Gatti, C. Pinciroli, and M. Roveri, "What planner for ambient intelligence applications?," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, pp. 7–21, Jan. 2005.

[5] F. Doctor, H. Hagras, and V. Callaghan, "A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabited environments," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, pp. 55–65, Jan. 2005.

[6] F. Zhao and L. Guibas, *Wireless Sensor Networks: An Information Processing Approach,* New York: Elsevier, 2005.
[7] K. Briechle and U. D. Hanebeck, "Localization of a mobile robot using relative bearing measurements," *IEEE Trans. Robot. Autom.*, vol. 20, no. 1, pp. 36–44, Feb. 2004.
[8] S. X. Yang and Q. H. Meng, "Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1541–1552, Nov. 2003.
[9] J. Minguez and L. Montano, "Nearness diagram (ND) navigation: Collision avoidance in troublesome scenarios," *IEEE Trans. Robot. Autom.*, vol. 20, no. 1, pp. 45–59, Feb. 2004.
[10] X. Yang, M. Moallem, and R. V. Patel, "A layered goal-oriented fuzzy motion planning strategy for mobile robot navigation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1214–1224, Dec. 2005.
[11] D. D. Sijak, *Large-Scale Dynamic Systems: Stability and Structure.* Amsterdam, The Netherlands: North-Holland, 1978.
[12] E. Freire, T. Bastos Filho, M. Sarcinelli Filho, and R. Carelli, "A new mobile robot control approach via fusion of control signal," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 419–429, Feb. 2004.
[13] T. H. S. Li, S. J. Chang, and Y. X. Chen, "Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot," *IEEE Trans. Ind. Electron.*, vol. 50, no. 5, pp. 867–880, Oct. 2003.
[14] T. C. Lee, C. Y. Tsai, and K. T. Song, "Fast parking control of mobile robots: a motion planning approach with experimental validation," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 5, pp. 661–676, Sep. 2004.
[15] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ problems," *IEEE Trans. Autom. Control*, vol. 34, no. 8, pp. 831–847, Aug. 1989.
[16] B. S. Chen and W. Zhang, "Stochastic H_2/H_∞ control with state-dependent noise," *IEEE Trans. Autom. Control.*, vol. 49, no. 1, pp. 45–57, 2004.
[17] C. L. Hwang and S. Y. Han, "Mixed H_2/H_∞ design for a decentralized discrete variable structure control with application to mobile robots," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 35, no. 4, pp. 736–750, Aug. 2005.
[18] J. Chen, S. Hara, and G. Chen, "Best tracking and regulation performance under control energy constraint," *IEEE Trans. Autom. Control*, vol. 48, no. 8, pp. 1320–1336, Aug. 2003.
[19] C. L. Hwang and C. Jan, "Optimal and reinforced robustness designs of fuzzy variable structure tracking control for a piezoelectric actuator system," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 4, pp. 507–517, Aug. 2003.
[20] Y. Hung, W. Gao, and J. C. Hung, "Variable structure control: A survey," *IEEE Trans. Ind. Electron.*, vol. 40, no. 1, pp. 2–22, Feb. 1993.
[21] D. Wang and G. Xu, "Full-state tracking and internal dynamics of nonholonomic wheeled mobile robots," *IEEE/ASME Trans. Mechatronics*, vol. 8, no. 2, pp. 203–214, Jun. 2003.
[22] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.
[23] R. L. Williams II, B. E. Carter, P. Gallina, and G. Rosati, "Dynamic model with slip for wheeled omni-directional robots," *IEEE Trans. Robot. Autom.*, vol. 18, no. 3, pp. 285–293, Jun. 2002.

Multirobot-Based Nanoassembly Planning with Automated Path Generation

Xiaobu Yuan and Simon X. Yang

***Abstract*—In this paper, a novel approach of automated multirobot nanoassembly planning is presented. This approach uses an improved self-organizing map to coordinate assembly tasks of nanorobots while generating optimized motion paths at run time with a modified shunting neural network. It is capable of synchronizing multiple nanorobots working simultaneously and efficiently on the assembly of swarms of objects in the presence of obstacles and environmental uncertainty. Operation of the presented approach is demonstrated with experiments at the end of the paper.**

***Index Terms*—Multirobot coordination, nanoassembly planning, self-organizing map (SOM), shunting neural networks.**

I. Introduction

Nanotechnology, working on the nanometer scale of molecules and atoms, creates enormous potential in a vast range of new applications. Current research on nanotechnology has been mainly in a few areas. They focus on the development of devices and systems to prepare tools on the invention of methods for the manipulation of nanoparticles [5], [9], and on the formation of strategies that allow the automation of assembly and manufacturing at the nanometer scale [10]. Applications of nanotechnology are also under active investigation [1], [3].

Nevertheless, working on the nanometer scale presents new challenges that researchers have never faced before when they work on machinery parts in the macroworld. Among the list of features that characterize nanoassembly, a complementary pair stands out as the most obvious and essential, i.e., the extreme smallness in size and the massive volume in construction [12]. In contrast to the remarkable progress in nanoscale manipulation [7], little work has been done with respect to the planning of multiple nanorobots.

The massive construction of nanoscale structures in nanoassembly requires large numbers of autonomous nanorobots working together to form desired patterns. The planning of nanoassembly involves different tasks, including object assignment, obstacle detection and avoidance, path finding, and path sequencing. An initial work on nanoassembly planning used a number of existing techniques, and developed a practical method for the planning of single-robot-based nanoassembly [8].

In this paper, an investigation on the automation of multirobot-based nanoassembly is presented. By introducing a modified shunting neural network for automated path generation and designing an improved self-organizing map (SOM) for multirobot coordination, this paper develops a strategy to efficiently synchronize the tasks of multirobot-based nanoassembly. Simulation results demonstrate that this strategy is capable of coordinating multiple nanorobots with equally distributed workload and automatically generated motion paths.

II. Optimized Path Generation

The original model of a biologically inspired neural network was proposed in [4]. Its simplified model takes the form of (1), in which x_i

Manuscript received March 1, 2006; revised December 15, 2006. Recommended by Guest Editors H.-P. Huang and F.-T. Cheng. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

X. Yuan is with the School of Computer Science, University of Windsor, Windsor, ON N9B 3P4, Canada (e-mail: xyuan@uwindsor.ca).

S. X. Yang is with the School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada (e-mail: syang@uoguelph.ca).

Digital Object Identifier 10.1109/TMECH.2007.897282