Robust Real-time Control of Autonomous Mobile Robot Based on Ultrasonic and Infrared sensors

Van-Quyet Nguyen*, Sung-Hyun Han+

(논문접수일 2009. 11. 03, 심사완료일 2010. 02. 09)

초음파 및 적외선 센서 기반 자율 이동 로봇의 견실한 실시간 제어

노연판쿠웨트*, 한성현+

Abstract

This paper presents a new approach to obstacle avoidance for mobile robot in unknown or partially unknown environments. The method combines two navigation subsystems: low level and high level. The low level subsystem takes part in the control of linear, angular velocities using a multivariable PI controller, and the nonlinear position control. The high level subsystem uses ultrasonic and IR sensors to detect the unknown obstacle include static and dynamic obstacle. This approach provides both obstacle avoidance and target-following behaviors and uses only the local information for decision making for the next action. Also, we propose a new algorithm for the identification and solution of the local minima situation during the robot's traversal using the set of fuzzy rules. The system has been successfully demonstrated by simulations and experiments.

Key Words : Ultrasonic sensor(초음파센서), IR sensor(적외선센서), Autonomous mobile robot(자율이동로봇), Obstacle avoidance (장애물회피), Real-time control(실시간제어)

1. Introduction

Autonomous robots are intelligent machines capable of performing tasks in the word by themselves, without explicit human control over their movements[1]. The autonomy implies that the robot is capable of reacting to static obstacles and unpredictable dynamic events. To achieve this level of robustness, methods need to be developed to provide solutions to localization, navigation, planning and control.

*   Dept. of Mechanical System Engineering, Graduate School, Kyungnam University
+   Corresponding author, Division of Mechanical System and Automation Engineering, Kyungnam University
      (shhan@kyungnam.ac.kr)
      Address: 449 Wolyoung-dong, Masan, Gyeongsangnam-do, Korea, 631-701

The robot has to find a trajectory between the starting configuration and the goal configuration in a static or dynamic environment containing some obstacles. To this end, the robot needs the capability to build a map of the environment, which is essentially a repetitive process of moving to a new position, sensing the environment, updating the map, and planning subsequent motion.

The most difficulty of process is the real environment that robots are operated. It is unstructured environments and large uncertainties because of these conditions: (1) General environment is incomplete, uncertain, and approximate. (2) Perceptually acquired information is usually unreliable. (3) The effects of control actions are not completely reliable, e.g. the wheels of a mobile robot may slip. (4) A real-world environment typically has complex and unpredictable dynamics: objects can move, other agents can modify the environment, and apparently stable features may change with time.

This paper proposes a new method of designing a robust autonomous mobile robot control system. This system provides the mobile robot that may navigate in a priority unknown indoor environment using sonar and IR sensor information. To achieve these requirements, the system is hierarchically organized into two distinct separated subsystems with arbitrary responsibility. A low-level[1] includes both velocity controller and position controller. Velocity controller is developed using the standard PI multivariable control law, a dynamic model of a mobile robot and actuators. The position control law has to be nonlinear in order to ensure stability of the error, that is, its convergence to zero[5,8,10]. A high-level
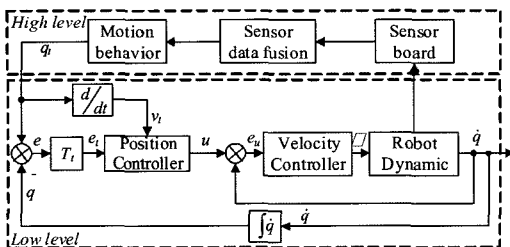
subsystem contains path planning and obstacle avoidance algorithms based on ultrasonic and IR sensors. All the modules are designed using fuzzy logic controller.

2. Dynamic equation motion of mobile robot

Consider a typical three-wheeled vehicle, two co-axial wheels provide both steering and power, while a single wheel is a passive caster shown in Fig. 2. The driving wheels now must be differentially driven. One problem in control of wheeled vehicle relates to the number of controllable degree of freedom (DOF). The usual goal of vehicle control is to reach a goal position (x, y) with a given orientation θ. Note that this implies control over 3 DOFs. However, we can control only 2 DOFs: vehicle motion and steering wheel angle. Vehicles with fewer controllable degrees of freedom than the actual vehicle *of freedom are known as non-holonomic. In this section,* a dynamic model of a nonholonomic mobile robot with viscous friction will be derived first. A typical representation of a nonholonomic mobile robot is shown in Fig. 2.

The robot has two driving wheels mounted on the same axis and a free front wheel. The two driving wheels are independently driven by two actuators to achieve both transition and orientation. The position of the mobile robot in the world frame *XOY* can be defined by the position of the mass center of the mobile robot system,
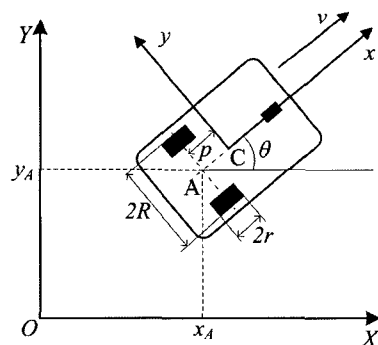

Fig. 1 Two-level control of Mobile robot

Fig. 2 A typical three-wheel vehicle.

denoted by C, or alternatively by the position A, which is the center of the mobile robot gear, and θ the angle between the robot's body frame $_xC_y$ and the world frame. The kinetic energy of the whole structure is given by the following equation:

$$W = W_t + W_r + W_m \tag{1}$$

where: W_t - the kinetic energy that is the consequence of pure translation of the entire vehicle, W_r - the kinetic energy of rotation of the vehicle in the XOY plane, W_m - the kinetic energy of rotation of the wheels and rotors of the DC motors

The values of energy terms introduced can be expressed by Eq. 2:

$$
\begin{aligned}
W_t &= \frac{1}{2}mv_c^2 = \frac{1}{2}m\left(\dot{x}_c^2 + \dot{y}_c^2\right) \\
W_r &= \frac{1}{2}I_A\dot{\theta}^2 \\
W_m &= \frac{1}{2}I_0\left(\dot{\theta}_R^2 + \dot{\theta}_L^2\right)
\end{aligned} \tag{2}
$$

where m is the mass of the entire vehicle, v_c is the linear velocity of the vehicle's center of mass C, I_A is the moment of inertia of the entire vehicle considering point A, θ is the angle that represents the orientation of the vehicle (Fig. 2), I_0 is the moment of inertia of the rotor/wheel complex, and $\dot{\theta}_R$ and $\dot{\theta}_L$ are the angular velocities of the right-hand and left-hand wheels, respectively. Further, the components of the velocity of the point A can be expressed in terms of $\dot{\theta}_R$ and $\dot{\theta}_L$:

$$
\begin{aligned}
\dot{x}_A &= \frac{1}{2}r\left(\dot{\theta}_R + \dot{\theta}_L\right)\cos\theta \\
\dot{y}_A &= \frac{1}{2}r\left(\dot{\theta}_R + \dot{\theta}_L\right)\sin\theta \\
\dot{\theta} &= \frac{r}{2R}\left(\dot{\theta}_R - \dot{\theta}_L\right)
\end{aligned} \tag{3}
$$

Since where p is the distance between points A and C, it is obvious that the equations below follow:

$$
\begin{aligned}
\dot{x}_C &= \frac{r}{2}\left(\dot{\theta}_R + \dot{\theta}_L\right)\cos\theta - p\dot{\theta}\sin\theta \\
\dot{y}_C &= \frac{r}{2}\left(\dot{\theta}_R + \dot{\theta}_L\right)\sin\theta + p\dot{\theta}\cos\theta
\end{aligned} \tag{4}
$$

By substituting terms in Eq. 1 with expressions in Eq. 2-4, the total kinetic energy of the vehicle can be calculated in terms of $\dot{\theta}_R$ and $\dot{\theta}_L$:

$$
\begin{aligned}
W\left(\dot{\theta}_R, \dot{\theta}_L\right) &= \left[\frac{mr^2}{8} + \frac{\left(I_A + mp^2\right)r^2}{8R^2} + \frac{I_0}{2}\right]\dot{\theta}_R \\
&+ \left[\frac{mr^2}{8} + \frac{\left(I_A + mp^2\right)r^2}{8R^2} + \frac{I_0}{2}\right]\dot{\theta}_L \\
&+ \left[\frac{mr^2}{4} + \frac{\left(I_A + mp^2\right)r^2}{4R^2}\right]\dot{\theta}_R\dot{\theta}_L
\end{aligned} \tag{5}
$$

Now, the Lagrange equations are applied:

$$
\begin{aligned}
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_R}\right) - \frac{\partial L}{\partial \theta_R} &= \tau_R - k\dot{\theta}_R \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_L}\right) - \frac{\partial L}{\partial \theta_L} &= \tau_L - k\dot{\theta}_L
\end{aligned} \tag{6}
$$

Here, τ_R and τ_L are right and left actuation torques, $k\dot{\theta}_R$ and $k\dot{\theta}_L$ are the viscous friction torques of right and left wheel-motor systems, respectively. Finally, the dynamic equations of motion can be expressed as:

$$
\begin{bmatrix} M & N \\ N & M \end{bmatrix}\begin{bmatrix} \ddot{\theta}_R \\ \ddot{\theta}_L \end{bmatrix} = \begin{bmatrix} \tau_R - k\dot{\theta}_R \\ \tau_L - k\dot{\theta}_L \end{bmatrix} \tag{7}
$$

where

$$
\begin{aligned}
M &= \frac{mr^2}{4} + \frac{\left(I_A + mp^2\right)r^2}{4R^2} + I_0 \\
N &= \frac{mr^2}{4} - \frac{\left(I_A + mp^2\right)r^2}{4R^2}
\end{aligned} \tag{8}
$$

3. Design subsystems' controller

In this section, we will describe detail about designing of each subsystem: low and high level respectively. The

low level velocity controller is developed using the standard PI multivariable control law. And the position control law has to be nonlinear in order to ensure stability of the error – it is convergence to zero. Some of the control parameters are continuous time functions, and usually the backstepping method[5,7,8] was used for their adjustment. In order to achieve the optimal parameter values, we used a genetic algorithm. Both controllers are based on a dynamic model of a differential drive mobile robot having angular velocities as main variables. The high-level subsystem uses ultrasonic and IR sensors to detect the unknown obstacle include static and dynamic obstacle by applying fuzzy logic.

The function of this controller is to implement a mapping between the known information (e.g. reference position, velocity and sensor information) and the actuator commands designed to achieve the robot task. For a mobile robot, the controller design problem can be described as follows: given the reference position $q_t(t)$ and velocity $q_t(t)$, design a control law for the actuator torques so that the mobile robot velocity may track a given reference control path with a given smooth velocity. Let the velocity and position of the reference robot (Fig. 3) be given as:

$$q_t = \begin{bmatrix} x_t & y_t & \theta_t \end{bmatrix}^T \tag{9}$$

where $\dot{x}_t = v_t \cos\theta_t; \dot{y}_t = v_t \sin\theta_t; \dot{\theta}_t = \omega_t$ and $v_t > 0$ is the reference linear velocity and ω_t is the reference angular velocity.

3.1 Position control

The trajectory tracking problem for a mobile robot is based on a virtual reference robot that has to be tracked (Fig. 3). The tracking position error between the reference robot and the actual robot can be expressed in the robot frame as:

$$e_t = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = T_t \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_t - x \\ y_t - y \\ \theta_t - \theta \end{bmatrix} \tag{10}$$
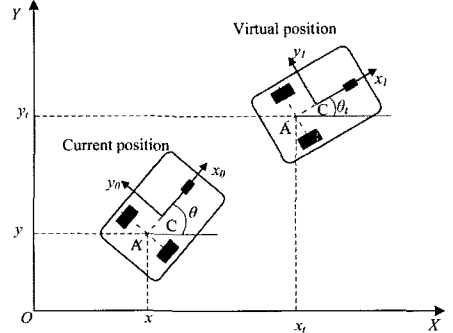

Fig. 3 The concept of tracking of a virtual reference robot

The dynamics of the position error derived in (10) is postulated as:

$$\dot{e}_1 = \omega e_2 + u_1$$
$$\dot{e}_2 = -\omega e_1 + v_t \sin e_3$$
$$\dot{e}_3 = u_2 \tag{11}$$

where inputs u_1 and u_2 are new control inputs.

In this paper, we propose the following control inputs in the velocity control loop:

$$u_1 = v_t \cos e_3 + \frac{k_1 e_1}{\sqrt{k_4 + e_1^2 + e_2^2}}$$
$$u_2 = \omega_t + \frac{k_2 v_t e_2}{\sqrt{k_5 + e_1^2 + e_2^2}} + \frac{k_3 v_t \sin e_3}{\sqrt{k_6 + e_3^2}} \tag{12}$$

where k_1, k_2, k_3, k_4, k_5 and k_6 are positive parameters.

Eq. (12) is a modified backstepping control law given first in[10]. The modification consists of the introduction of denominators. In[10], Lyapunov's stability theory was used to prove that the control law that is considered provides a uniformly bounded norm of error $\| e_t(t) \|$. However, usage of this law is justified by simulation results.

The key problem in such a control design is to obtain control coefficients k1 to k6. To solve this problem, a genetic algorithm is used to find the optimal values of those coefficients.

3.2 Velocity control

The dynamics of the velocity controller is given by the following equations in the Laplace domain:

$$\tau(s) = \begin{bmatrix} \tau_R(s) \\ \tau_L(s) \end{bmatrix} = \frac{1}{r}\begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} e_v(s) \\ e_\omega(s) \end{bmatrix} \quad (13)$$

where $e_v(s)$ is the linear velocity error and $e_\omega(s)$ is the angular velocity error. Transfer functions $f_j(s)$ are chosen to represent PI controllers:

$$f_1(s) = K_1\left(1 + \frac{1}{T_{i1}s}\right)R$$
$$f_2(s) = K_2\left(1 + \frac{1}{T_{i2}s}\right)R \quad (14)$$

From Eqs. 7 and 13, we can write another form as:

$$\begin{bmatrix} Ms+k & Ns \\ Ns & Ms+k \end{bmatrix}\begin{bmatrix} \omega_R(s) \\ \omega_L(s) \end{bmatrix} = \frac{1}{r}\begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} u_1(s)-v(s) \\ u_2(s)-\omega(s) \end{bmatrix} \quad (15)$$

where

$$\omega_R(s) = \frac{v + R\omega}{r}; \quad \omega_L(s) = \frac{v - R\omega}{r}$$

and then (15) can be transformed to:

$$\begin{bmatrix} Ms+k & Ns \\ Ns & Ms+k \end{bmatrix}\begin{bmatrix} v(s)+R\omega(s) \\ v(s)-R\omega(s) \end{bmatrix} = \begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} u_1(s)-v(s) \\ u_2(s)-\omega(s) \end{bmatrix} \quad (16)$$

and further to:

$$\begin{bmatrix} \mu_1(s) & \mu_2(s) \\ \mu_1(s) & -\mu_2(s) \end{bmatrix}\begin{bmatrix} v(s) \\ \omega(s) \end{bmatrix} = \begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \quad (17)$$

where

$$\mu_1(s) = \frac{(M+N)s^2 + (k+K_1T_{i1})s + K_1}{s}$$
$$\mu_2(s) = \frac{R(M-N)s^2 + (Rk+K_2T_{i2})s + K_2}{s} \quad (18)$$

The following equations could be easily derived from (17):

$$v(s) = \frac{f_1(s)}{\mu_1(s)}u_1(s) = F_1u_1(s)$$
$$= \frac{K_1T_{i1}s + K_1}{(A+B)s^2 + (k+K_1T_{i1})s + K_1}u_1(s)$$
$$\omega(s) = \frac{f_2(s)}{\mu_2(s)}u_2(s) = F_2u_2(s)$$
$$= \frac{K_2T_{i2}s + K_2}{R(A-B)s^2 + (Rk+K_2T_{i2})s + K_2}u_2(s) \quad (19)$$

It is obvious that transfer functions F_1 and F_2 are static with gains equal to "1", which completes the proof. The velocity control loop structure is shown in Fig. 1, as an inner loop. From the simulation results obtained (Figs. 4 and 5), it can be seen that the proposed PI controller successfully tracks the given linear and angular velocity profiles. The controller parameters used for this simulation
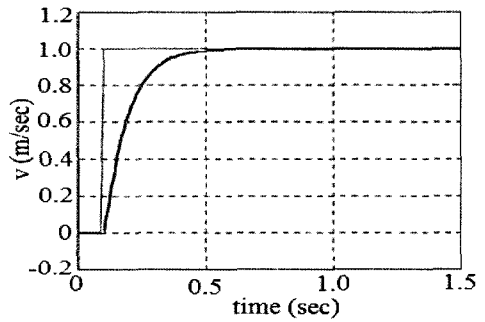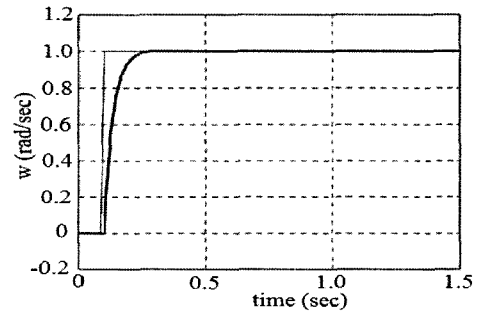

Fig. 4 Linear velocity step response.

Fig. 5 Angular velocity step response.

are $K_1 = 89.68$, $K_2 = 27.32$, $T_{i1} = 15.18$, and $T_{i2} = 18.57$.

3.3 Design of hybrid position controller

It has been noticed that, during the preliminary simulations, at the beginning of tracking the control torques increase rapidly if the initial position of the reference robot does not belong to the straight line, determined with the robot and its initial orientation.

For that purpose, the following control law, which provides velocity servo inputs, is proposed:

$$u_1^*(t) = \beta(t)u_1(t)$$
$$u_2^*(t) = \beta(t)u_2(t) + [1 - \beta(t)]\omega_s(t) \qquad (20)$$

Function $\omega_s(t)$ is generated as the output of the following system (Fig. 6). The input $s(t)$ is given by:

$$s(t) = \text{sgn}\{atan2[e_y(t), e_x(t)] - \theta(t)\} \qquad (21)$$

The function $\beta(t)$ satisfies the following differential equation:

$$b_0 \frac{d^2\beta(t)}{dt^2} + b_1 \frac{d\beta(t)}{dt} + \beta(t) = \varepsilon(t) \qquad (22)$$

where $\varepsilon(t)$ is close to a step function and is given by:

$$\varepsilon(t) = \begin{cases} 1, & if \ \exists t_1 \in [0, t]: \theta(t_1) = atan2[e_y(t), e_x(t)] \\ 0, & othewise \end{cases} \qquad (23)$$

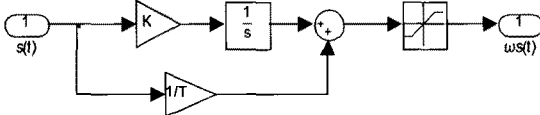This way, the robot does not start tracking the virtual robot instantly; it first rotates around its own axis with an increasing angular velocity $\omega_s(t)$.

3.4 Sensory systems

It is very important for mobile robot to know environment by using multi sensory information. So that both an accurate sensor model and a reasonable sensor fusion were proposed and applied in our model. There are many researches were successful implemented with autonomous robot working in the common environment that surrounding environment is described as static or known. Unfortunately, it is not useful for K-RONI to complete tasks in the real world, because the surrounding environment in the real world is dynamic, unknown and time-varying. There are a lot of tasks where robots are required to move safely with unknown and dynamic obstacles. Reaching the destination safely and executing the assignment for K-RONI is an essential and important factor. We can detect the obstacles around the environment and construct the new environment map by analyzing the distances mounted around K-RONI. We are going to discuss what difference from sensors used as Ultrasonic and IR sensors.

3.4.1 Ultrasonic Sensors

Ultrasonic sensors provide good range information, long-range detection, the drawback of that is the unstable system and slow reaction rate. However, uncertainties in ultrasonic sensors caused by the specular reflection from the environment make them less attractive. The information of sensory signals received with the ultrasonic sensors for the system to detect and avoid the obstacles. There are seven ultrasonic sensors are distributed around the frontal
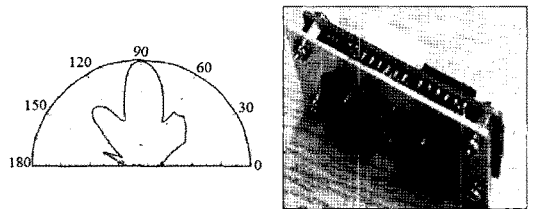

Fig. 6 Producing $\omega_s(t)$

Fig. 7 Ultrasonic sensors

of K-RONI for environment exploration. The firing rate and sequence of the ultrasonic sensors are controlled by a microprocessor on sonar sensory board. The beam width for each sensor is 36 degree, and the range to an object is determined by the time of flight of an acoustic signal generated by the transducer and reflected by the object. K-RONI can use the sensors to detect the range information from 0.5m to 1.5m. But the obstacles whose the distance to K-RONI is shorter than 0.5m are recognized as a kind of uncertain information, and which causes the K-RONI not to detect exactly whether there are uncertain obstacles around this area. IR sensor will be compensated the error above.

3.4.2 IR Sensors

IR sensors can overcome almost disadvantages of the ultrasonic sensor. It has fast reaction rate, and stable system. But the drawback is the short-range detection. The principle of the reflection IR sensor is following. There is one transmitter and one receiver in the sensor. The transmitter always launches light signal. When the receiver receives the signal, it means the sensor detect an obstacle whose the range information is shorter than 0.5m, and represented as a signal "0". On the contrary, that means non-Obstacle exists, represented as a signal "1". The IR sensor can reliably detect obstacles in three separate quadrants, and the range is adjustable down to only a few inches. An obstacle can be detected on the left, the right or the center shown in fig. 8. However, the longest detection distance of the IR sensors is about 0.5m, when the distance information detected is longer than 0.5m that means the sensor failed to detect if there are any obstacles in the environment.

Main purpose of using the IR sensors is to detect the dynamic obstacle. When the IR sensors detect the dynamic obstacle is navigating slowly close to it, the mobile robot can pause and wait this dynamic obstacle to depart. Therefore, obstacle can be detected near by the Robot.
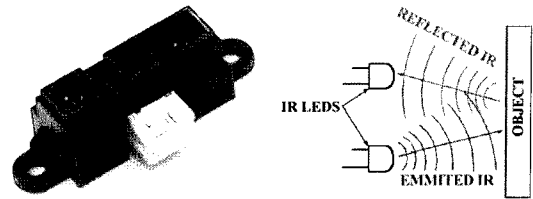

Fig. 8 IR sensor

Table 1 IR sensor specification

Sensor type	Reflective IR
IR LED type	Narrow focus 10°
I/O required	2 outputs, 1 input
Measurement range	100-550mm
Input voltage	5 Vdc
Current requirements	8 mA

3.4.3 Cooperation of Ultrasonic Sensors and IR Sensors

By overview of advantages and disadvantages of ultrasonic and IR sensor, we propose a method of combination two kind of these sensors. When K-RONI is planning to navigate in the surroundings, there are usually the static and dynamic obstacles stopping it from going to the given position. In order to make K-RONI avoid these uncertain obstacles successfully, the ultrasonic sensors and the IR sensors are used to detect if there exists obstacles simultaneously when K-RONI navigates in the real environment. If the distance between the obstacle and K-RONI is longer than 0.5m, the ultrasonic sensors are used to detect the obstacles; however the distance is shorter than 0.5m, the IR sensors are prior to be used.

For example, when K-RONI navigates in the environment, the ultrasonic sensors could detect the obstacle first with its long detection distance. And if this obstacle is static, the sensor can detect and avoid it continuously with safe distance. But this obstacle is dynamic, and it is close to K-RONI suddenly. This kind of unexpected behavior led the ultrasonic not to recognize this range information exactly. Finally, K-RONI used the IR sensors

to execute the obstacle avoidance in order to go to the given position.

In the intelligent system, in order to plan K-RONI to execute works in the static and dynamic environment, there are five IR sensors and seven ultrasonic sensors arrangement on the robot. With these sensors arrangement, we can measure the distance from the center of K-RONI to the obstacles. Hence, the multi-sensor integration technique is applied in the robotics. This technique acquires sensory information to reconstructs the dynamic environment.

3.4.4 Obstacle avoidance behavior

The obstacle avoidance algorithm of K-RONI robot is emphasized the relation between IR and ultrasonic sensors. The Ultrasonic sensors of K-RONI are primary ones to receive the range information. When the front six pieces of sensors detect the obstacle and K-RONI begins to change direction toward free-space areas, at the same time, the range information values start to increase gradually, that means K-RONI commences to navigate toward the free-space areas for avoiding the obstacles. The information signal of IR sensors changed from "1" to "0", that means the obstacle was detected by sensors.

When the Sensor Fusion Model received the information of the IR and Ultrasonic sensors, they will be sent to the Robot Motion Behaviors. And the Robot Motion Behaviors will be divided into five conditions by using varied sensors modes when the K-RONI met different of obstacles in the navigating path (Fig. 4).

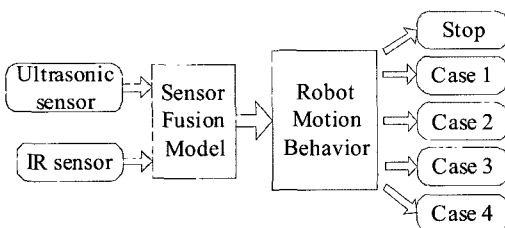There are two states to describe the situation which

Fig. 9 Integration of IR and ultrasonic sensors for obstacle avoidance

K-RONI run into obstacles as follow:

$\varepsilon \neq 0$: Ultrasonic sensor detected obstacle
$\gamma \neq 0$: IR sensors detected obstacle

· Case 1: $\varepsilon = 0$, and $\gamma = 0$

In this case, it means K-RONI is far away from the obstacles and moved toward the free-space areas when parameter ε and γ is zero.

· Case 2: $\varepsilon \neq 0$, but $\gamma = 0$

This case means the frontal Ultrasonic sensors have detected the obstacle in the navigating path. And the range information received is sent to motion commands. Robot can be avoided with long distance obstacle.

· Case 3: $\varepsilon \neq 0$, and $\gamma \neq 0$

In this case, both the Ultrasonic sensors and IR sensors take turns to be used for detecting the obstacles according to the distance from K-RONI to the obstacles. With the long-range detection and slow reaction rate of the Ultrasonic sensors, it is suitable for detecting the static obstacles. On the contrary, the short-range detection and fast reaction rate of the IR sensors is suitable for detecting the dynamic obstacles.

· Case 4: $\varepsilon = 0$, but $\gamma \neq 0$

In this case, the IR sensors have detected the obstacles near K-RONI. The range information is sent to the motion agent starts to figure out the free-space areas after receiving the motion commands.

4. Simulations and Experimental results

To show a usefulness of the proposed approach, a series of simulation and experiments has been conducted by using an arbitrarily constructed environment including static and dynamic obstacles. Almost simulations are

performed in the MATLAB / SIMULINK software. In this paper, we used a mobile robot with the following parameters: $M = 12kg$, $I_A = 1kg\ m^2$, $r = 0.04m$, $R = 0.250m$, $p = 0.02m$ and $I_0 = 0.001kg\ m^2$.

The proposed hybrid controller was tested by using an ordinary backstepping algorithm[10] and a hybrid backstepping algorithm. Fig. 10 and 11 are illustrated the errors of x, y axis, and orientation respectively. It also can be concluded that satisfactory tracking results are obtained using both control strategies. However, better tracking of

the reference trajectory is achieved in the case of an ordinary controller, especially at the beginning of transient process.

However, the hybrid controller ensures that much smaller values of the control input torques are needed to obtain the reference position and orientation trajectories shown in Fig. 12 and 13. Consequently, the required power of the DC motors is also much smaller in the case of a hybrid controller shown in Fig. 14 and 15.

The integration of sonar data is obtained by a low-level fusion. The building of occupancy maps is well suited to path planning and obstacle avoidance. The minimum distance obtained within the cone of each sensor is
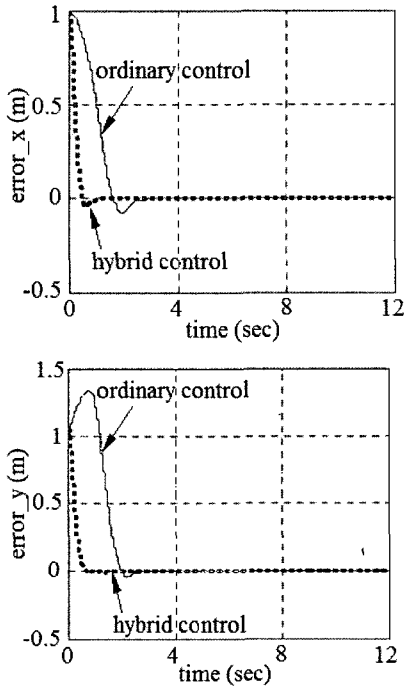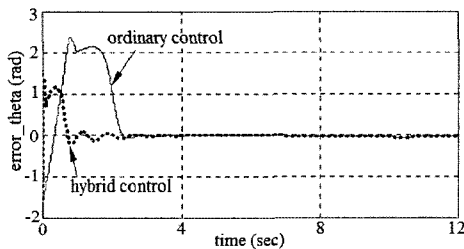

Fig. 10 x, y axis errors respectively
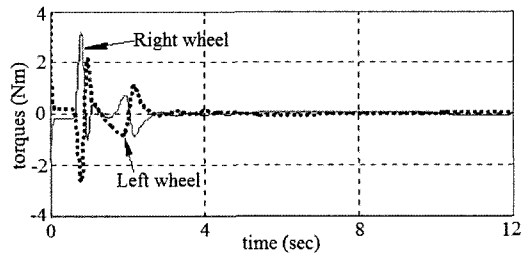

Fig. 11 Orientation error

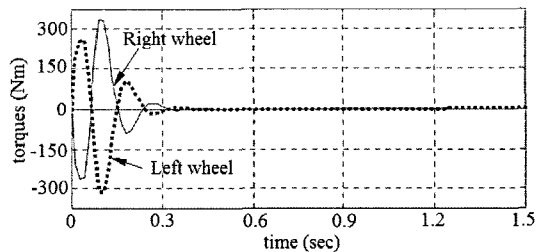Fig. 12 Control torques - hybrid controller

Fig. 13 Control torques - ordinary controller

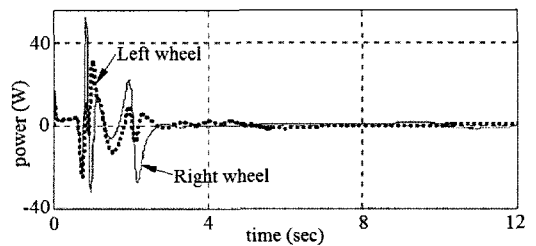Fig. 14 Power of DC motors - hybrid controller

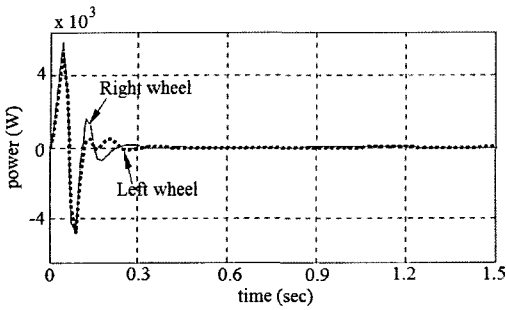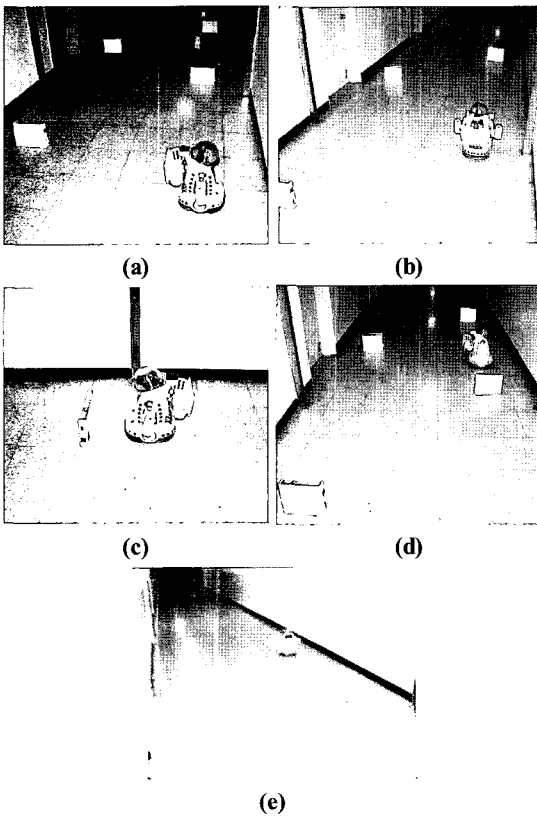Fig. 15 Power of DC motors − ordinary controller

(a) (b)

(c) (d)

(e)

Fig. 16 Experimental environment of K-RONI

considered as the distance of the obstacle from that sensor, and it is available as an input to the algorithm. In these simulations, the positions of all the obstacles in the workspace are unknown to the robot. The robot is aware of its start and finish positions, static and dynamic obstacles will be occurred. Experimental results obtained

in several different working scenarios are presented in Fig. 16.

In all of the cases in which the target is surrounded by cluttered obstacles, which form the convex loop, the proposed algorithm helps the robot to find the target. The algorithm works very well in the case when a cluttered environment consists of walls in the form of long loops, which is always convex in order to current robot location.

During the process of obstacle avoidance, K-RONI will have many different situation of obstacle avoidance as the difference environmental varieties which IR sensor and ultrasonic sensor sensed. A new problem arises if the environment consists of concave obstacles and the robot gets into an infinite loop or into a local minimum. An even more complex problem arises when the obstacles are long and have many bends and kinks; the target attracting and goal repulsing behavior conflict and the robot also gets itself into an infinite loop. These results clearly show that this system is able to navigate in complex environments such as mazes.

5. Conclusions

Our approach presented a new mobile robot navigation strategy based on the computing technologies(such as fuzzy logic and other reasoning techniques) in an a priori unknown environment. Firstly, a dynamic model of a mobile robot with nonholonomic constraints is derived. The special feature of this model is that the main variables are the angular velocities of the wheels. Due to this approach, the impossibility of lateral motion is embedded into the model. In addition, such a model is easily simulated. The proposed navigation system consists of three control subsystems.

The low-level control of robot velocities is designed using PI controllers. The purpose of medium-level nonlinear backstepping position controller is adjusted by a genetic algorithm. The high-level control system consists of integrated two kinds of sensors including ultrasonic and IR sensors to avoid the unknown obstacles both of static

and dynamic obstacle and navigate K-RONI to the given position in the unknown environment. Hence, a new approach is applied in robotics and approved by the results of simulation and experiments.

References

(1) Bekey, G. A., 2005, *Autonomous Robots: From Biological Inspiration to Implementation and Control*, The MIT press.

(2) Ren, W., Sun, J. S., Beard, R. W., and McLain, T. W., 2005, "Nonlinear tracking control for nonholonomic mobile robots with input constraints: an experimental study," *American Control Conference, proc. of the 2005*, vol. 7, pp. 4923~4928.

(3) Shim, H. S., and Sung, Y. G., 2004, "Stability and four-posture control for nonholonomic mobile robots," *Robotics and Automation, IEEE Trans.* on vol. 20, Issue 1, pp. 148~154.

(4) Kuc, T. Y., S. M., 1999, "Robust velocity learning and position tracking of nonholonomic mobile robots," *IEEE International Conf.* on vol. 6, pp. 643~648.

(5) Hu, T., Yang, S.X., 2001, "Real-time motion control of a nonholonomic mobile robot with unknown dynamics," *proc. of the Computational Kinematics Conf.*, Seoul, Korea.

(6) Kobayashi, H., and Yanagida, M., 1995, "Moving object detection by an autonomous guard robot," *Proc. of the 4th IEEE International Workshop on Robot and Human Communication*, pp. 323~326.

(7) Ramaswamy, S.A.P., Balakrishnan, S.N., 2008, "Formation control of car-like mobile robots: A Lyapunov function based approach," *American Control Conf.*, pp. 657~662.

(8) Tanner, H.G., Kyriakopoulos, K.J., 2003, "Backstepping for nonsmooth systems," *Automatica no. 39*, vol. 5, pp. 1259~1265.

(9) Um, D., Stankovic, B., Giles, K., Hammond, T., Lumelsky, V., 1998 "A modularized sensitive skin for motion planning in uncertain environments Robotics and Automation," *proc. of IEEE International Conf.* vol. 1, pp. 7~12.

(10) Morin, P., Samson, C., 1997, "Application of backstepping techniques to time varying exponential stabilization of chained systems," *European Journal of Control* vol. 3, no. 1, pp. 15~36.

(11) Hong, J., Choi, Y., and Park, K., 2007, "Mobile robot navigation using modified flexible vector field approach with laser range finder and IR sensor Control," *ICCAS '07 International Conf.*, pp. 721~726.

(12) Buchenberger, M., Jörg, W., and Von Puttkamer, E., 1993, "Laser radar and Sonar-based World Modelling.," *IEEE Conf. on Robotics and Automation*, Atlanta Georgia, pp. 534~541.

(13) Flynn, A. M., 1988, "Combining Sonar and Infrared Sensors for Mobile Robot Navigation," *The International Journal of Robotics Research*, vol. 7, no. 6, pp. 5~14.

(14) Crowley, J. L., 1989, "World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging," *proc. of the 1989 IEEE International Conf. of Robotics and Automation*, pp. 674~680.

(15) Everett, H. R., Gilbreath, G. A., Tran, T., and Nieusma, J. M., 1990, "Modeling the Environment of a Mobile Security Robot," *Technical Document 1835, Advanced Systems Division, Naval Ocean Systems Center*, San Diego, California 92152~5000, June.