Robust Real-time Control of Autonomous Mobile Robot by Using Ultrasonic and Infrared sensors

Van-Quyet Nguyen, Shim-Byoung Kyun, and Sung-Hyun Han

Dept. of Mechanical System Engineering, Graduate School, Kyungnam University,
449 Wolyoung-dong, Masan, Gyeongsangnam-do, Korea, 631-701
(Tel: 82-55-249-2590; Fax: 82-55-249-2617)
(Email address: afi.quyet@gmail.com)

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.

Keywords: Ultrasonic sensor, IR sensor, Autonomous mobile robot, Obstacle avoidance, Real-time control

I. 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. The most difficulty of process is the real environment that robots are operated. It is unstructured environments and large uncertainties.

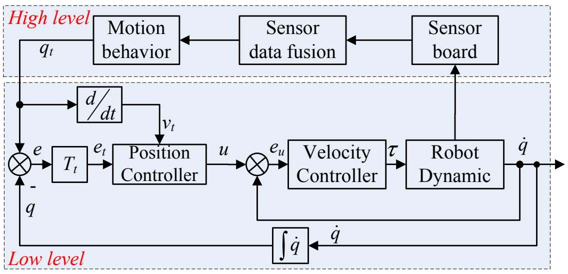


Fig. 1 Two-level control of Mobile robot

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 (Fig. 1) with arbitrary responsibility. A low-level 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 [2, 3, 4]. A high-level subsystem contains path planning and obstacle avoidance algorithms based on ultrasonic and IR sensors data as linguistic variables to design of fuzzy logic controller.

II. DAYNAMIC 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 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, 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 *xCy* and the world frame. The kinetic energy of the whole structure is given by the following equation:

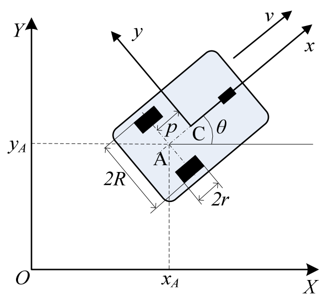


Fig. 2 A typical three-wheel vehicle.

$$T = T_t + T_r + T_m \tag{1}$$

where: T_t - the kinetic energy that is the consequence of pure translation of the entire vehicle, T_r - the kinetic energy of rotation of the vehicle in the *XOY* plane, T_m - the kinetic energy of rotation of the wheels and rotors of the DC motors. Total kinetic energy of the vehicle can be calculated in terms of $\dot{\theta}_R$ and $\dot{\theta}_L$:

$$T(\dot{\theta}_R,\dot{\theta}_L)=\left[\frac{mr^2}{8}+\frac{(I_A+mp^2)r^2}{8R^2}+\frac{I_0}{2}\right]\dot{\theta}_R + \left[\frac{mr^2}{8}+\frac{(I_A+mp^2)r^2}{8R^2}+\frac{I_0}{2}\right]\dot{\theta}_L + \left[\frac{mr^2}{4}+\frac{(I_A+mp^2)r^2}{4R^2}\right]\dot{\theta}_R\dot{\theta}_L \quad (2)$$

Now, the Lagrange equations are applied:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}_R}\right)-\frac{\partial L}{\partial\dot{\theta}_R}=\tau_R-k\dot{\theta}_R$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}_L}\right)-\frac{\partial L}{\partial\dot{\theta}_L}=\tau_L-k\dot{\theta}_L \quad (3)$$

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:

$$\begin{bmatrix} M & N \\ N & M \end{bmatrix}\begin{bmatrix} \ddot{\theta}_R \\ \ddot{\theta}_L \end{bmatrix}=\begin{bmatrix} \tau_R-k\dot{\theta}_R \\ \tau_L-k\dot{\theta}_L \end{bmatrix} \quad (4)$$

where

$$M=\frac{mr^2}{4}+\frac{(I_A+mp^2)r^2}{4R^2}+I_0$$
$$N=\frac{mr^2}{4}-\frac{(I_A+mp^2)r^2}{4R^2} \quad \mathbf{(5)}$$

III. DESIGNER OF SUBSYSTEM

In this section, we will describe detail about designing of each subsystem: low and high-level respectively. 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 $\dot{q}_t(t)$, 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.

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:

$$e_t=\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}=R_t\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix}=\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_t-x \\ y_t-y \\ \theta_t-\theta \end{bmatrix} \quad (6)$$

The dynamics of the position error derived in (Eq. 6) is postulated as:

$$\dot{e}_1=\omega e_2+u_1$$
$$\dot{e}_2=-\omega e_1+v_t\sin e_3 \quad (7)$$
$$\dot{e}_3=u_2$$

where inputs u_1 and u_2 are new control inputs.

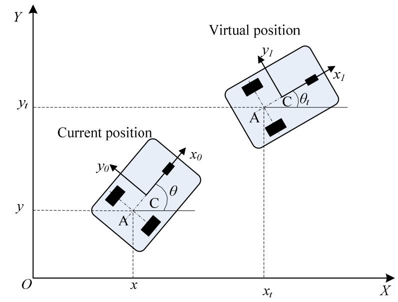


Fig. 3 Tracking of a virtual reference robot

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

$$u_1=v_t\cos e_3+\frac{k_1e_1}{\sqrt{k_4+e_1^2+e_2^2}}$$
$$u_2=\omega_t+\frac{k_2v_te_2}{\sqrt{k_5+e_1^2+e_2^2}}+\frac{k_3v_t\sin e_3}{\sqrt{k_6+e_3^2}} \quad (8)$$

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

3.2. Velocity control

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

$$\tau(s)=\begin{bmatrix} \tau_R(s) \\ \tau_L(s) \end{bmatrix}=\frac{1}{r}\begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} e_v(s) \\ e_\omega(s) \end{bmatrix} \quad \mathbf{(9)}$$

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:

$$f_1(s)=K_1\left(1+\frac{1}{T_{i1}s}\right)R$$
$$f_2(s)=K_2\left(1+\frac{1}{T_{i2}s}\right)R \quad (10)$$

After some transformations, Eq. 9 can express as:

$$\begin{bmatrix} \mu_1(s) & \mu_2(s) \\ \mu_1(s) & -\mu_2(s) \end{bmatrix}\begin{bmatrix} v(s) \\ \omega(s) \end{bmatrix}=\begin{bmatrix} f_1(s) & f_2(s) \\ f_1(s) & -f_2(s) \end{bmatrix}\begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \quad \mathbf{(11)}$$

where

$$\mu_1(s)=\frac{(M+N)s^2+(k+K_1T_{i1})s+K_1}{s}$$
$$\mu_2(s)=\frac{R(M-N)s^2+(Rk+K_2T_{i2})s+K_2}{s} \quad (12)$$

The following equations could be easily derived from (Eq. 11):

$$v(s)=\frac{f_1(s)}{\mu_1(s)}u_1(s)=F_1u_1(s)=\frac{K_1T_{i1}s+K_1}{(A+B)s^2+(k+K_1T_{i1})s+K_1}u_1(s)$$
$$\omega(s)=\frac{f_2(s)}{\mu_2(s)}u_2(s)=F_2u_2(s)=\frac{K_2T_{i2}s+K_2}{R(A-B)s^2+(Rk+K_2T_{i2})s+K_2}u_2(s) \quad \mathbf{(13)}$$

It is obvious that transfer functions F_1 and F_2 are static with gains equal to "1", which completes the proof.

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.

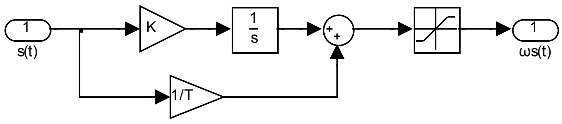


Fig. 4 Producing $\omega_s(t)$

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

$$\begin{aligned} u_1^*(t) &= \beta(t)u_1(t) \\ u_2^*(t) &= \beta(t)u_2(t) + [1-\beta(t)]\omega_s(t) \end{aligned} \tag{14}$$

Function $\omega_s(t)$ is generated as the output of the following system (Fig. 4). The input $s(t)$ is given by:

$$s(t) = \text{sgn}\{atan2[e_y(t), e_x(t)] - \theta(t)\} \tag{15}$$

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

$$b_0 \frac{d^2\beta(t)}{dt^2} + b_1 \frac{d\beta(t)}{dt} + \beta(t) = \varepsilon(t) \tag{16}$$

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

$$\varepsilon(t) = \begin{cases} 1, & if\ \exists t_1 \in [0,t] : \theta(t_1) = atan2[e_y(t), e_x(t)] \\ 0, & \text{othewise} \end{cases} \tag{17}$$

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

3.4.1. 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.

3.4.2. Obstacle avoidance behavior

The obstacle avoidance algorithm of K-RONI robot is emphasized the relation between IR and ultrasonic 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. 5).

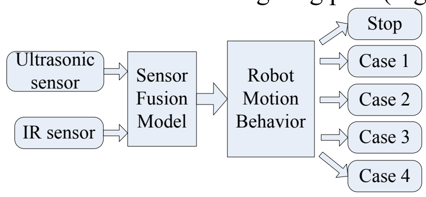


Fig. 5 Integration of IR and ultrasonic sensors for obstacle avoidance

There are two states to describe the situation which 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 $\gamma = 0$*: 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 $\gamma \neq 0$*: Both 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 $\gamma \neq 0$:* 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.

IV. SIMULATION & EXPERIMENTS

To show a usefulness of the proposed approach, a series of simulations 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 = 12 kg, I_A = 1 kg.m^2, r = 0.04 m, R = 0.250 m, p = 0.02 m and I_0 = 0.001 kg.m^2.

Fig. 6 and 7 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.

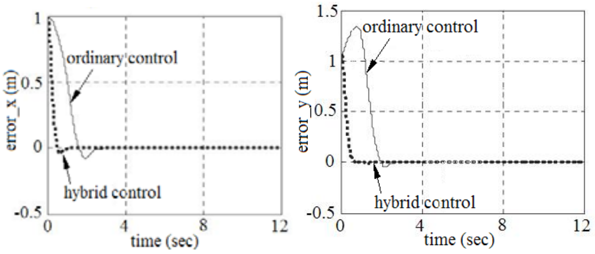


Fig. 6 x, y axis errors respectively

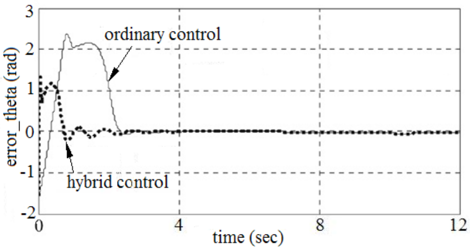


Fig. 7 Orientation error.

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. 8 and 9.

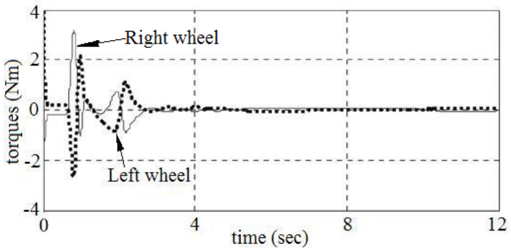


Fig. 8 Control torques – hybrid controller

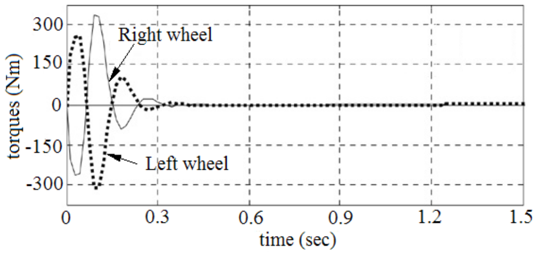
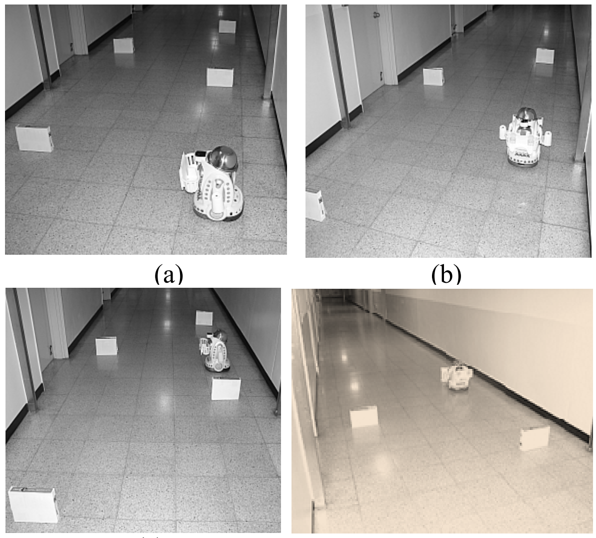


Fig. 9 Control torques – ordinary controller

Experimental results obtained in several different working scenarios are presented in Fig. 10. 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.

V. CONCLUSION

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 two control subsystems. Hence, a new approach is applied in robotics and approved by the results of simulation and experiments.



(a) (b) (c) (d)

Fig. 10 **Experimental environment of K-RONI**

REFERENCES

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

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

[3] H.G. Tanner, K.J. Kyriakopoulos, "Backstepping for nonsmooth systems", Automatica No. 39, Vol. 5, pp. 1259–1265, 2003.

[4] P. Morin, C. Samson, "Application of backstepping techniques to time varying exponential stabilization of chained systems", European Journal of Control Vol. 3, No. 1, pp. 15–36, 1997.

[5] J. Hong, Y. Choi, and K. Park; "Mobile robot navigation using modified flexible vector field approach with laser range finder and IR sensor Control", ICCAS '07 International Conf., pp. 721 – 726, 2007

[6] M. Flynn, "Combining Sonar and Infrared Sensors for Mobile Robot Navigation", The International Journal of Robotics Research, Vol. 7, No. 6, pp. 5-14, 1988.

[7] J. L. Crowley, "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, 1989.