A Model-Reference Adaptive Motion Controller for a Differential-Drive Mobile Robot

L. Feng, Y. Koren and J. Borenstein
Mobile Robot Laboratory
Department of Mechanical Engineering and Applied Mechanics
The University of Michigan
Ann Arbor, MI 48109-2125

Abstract

This paper describes the design and implementation of a model-reference adaptive motion controller for a differential-drive mobile robot. This controller uses absolute position information to modify control parameters in real time to compensate for motion errors. Robot motion errors are classified into internal and external errors. Cross-coupling control method is used to compensate for the internal errors that can be detected by wheel encoders. The adaptive controller provides compensation for external errors. The adaptive controller is analyzed, and its stability and convergence are discussed. Experiments are conducted to evaluate the control system and the results show significant improvements over conventional controllers.

1. Introduction

Currently most mobile robot research has concentrated on the application of mobile platforms to perform intelligent tasks, rather than on the development of methodologies for analyzing, designing, and controlling mobile systems. However, improved motion control systems will enable the application of mobile robots to tasks requiring accurate trajectory tracking even in unstructured environments [1].

In this paper, we introduce an adaptive controller with the following two improvements over a conventional motion controller:

1. Adaptation to the changes of robot parameters and environment [2]. Some robot parameters change with robot operation, e.g., the drive wheel diameters change with the load and load distribution.

2. Direct control of the most significant error [3]. In this paper, we will introduce the *most significant error*, which is defined as the error that has the largest impact on the motion accuracy. Our controller controls the most significant error directly in addition to control errors in individual drive loop.

The basic idea of adaptive control is to eliminate the effects of variations in the controlled system parameters by estimating these parameters in real time and using the estimates in the control process, or by generating correction signals to compensate for errors [4]. Adaptive control has been used to accurately control robot manipulators in the case where their parameters are not precisely known or they change with robot operation [5, 6]. However, application of adaptive control in mobile robot has not been widely studied. A self-tuning navigation algorithm has been suggested by Banta [2]. His algorithm is aimed at correcting motion errors caused by miscalibration, uneven tire wear and wheel misalignment. Banta's algorithm employs a least-square method for the parameter estimation and the estimates are used to adjust the control of the robot.

Our paper will concentrate on the problem of adaptive motion control of a mobile robot in the case where the physical model that describes the motion of the robot is not well known or it changes with robot operation. The proposed controller dynamically adjusts its parameters according to the robot's operating and the environmental conditions. In addition, Cross-coupling control is used to control the orientation error by coordinating the motion of the two drive loops. In the following discussion, we limit us to the case of a differential-drive mobile robot, although the methodology is applicable to other robot types.

In the next Section, we will discuss error sources in robot motion and introduce a model for the vehicle-environment system. In Section 3, we will discuss the adaptive motion control for mobile robot. In Section 4, the performances of the proposed controller will be evaluated by experiments. Finally, conclusions are drawn in the Section 5.

2. Motion Error Sources for Mobile Robots and Robot-environment System Modeling

We can classify error sources into internal and external errors. Internal errors are the errors that can be detected by wheel encoders. External errors are the errors that only become apparent when robot wheels interact with the environment and can only be detected by absolute robot motion measurements [7]. External errors can be further divided into systematic errors and non-systematic errors. Systematic errors exist over a long period of time without changing their characteristics. Non-systematic errors happen in a random fashion and can only be described in a statistical sense at best.

2.1 Motion error sources

The main internal error sources are: (1) *Different drive loop parameters*, e. g., different time constants and loop gains. (2) *Different disturbances acting on the different drive loops*, e. g., different bearing friction.

The main external systematic error sources are: (1) *Different wheel diameters*. When the two drive wheels have different diameters and same angular speeds, the robot will follow a circular instead of a straight line path. A difference in the wheel diameters can be caused by load or its distribution changes and uneven wear of the wheels. (2) *Wheel misalignment*. The effect of the misalignment of the drive wheels is the robot constantly pulling to one side. Causes for this error include manufacturing tolerance and load distribution. (3) *Contact area*. When the wheel contact with the floor, there is a contact area, rather than a contact point. This causes an uncertainty about the wheelbase.

External nonsystematic errors include: (1) *Wheel slippage*. Slippage is a big problem in dead-reckoning, it is not a biased error and it can happen in a very short time period. However, slippage normally happens only when a robot moves on a curved path and when a robot accelerates or decelerates. (2) *Floor roughness*. When a robot travels over a rough floor, the wheels move up and down over bumps. Part of the motion recorded by wheel encoders is the vertical distance required to clear bumps. Surface roughness causes the traveled distance to be overestimated.

Under normal conditions, motion error is largely due to systematic external errors. External nonsystematic errors are random in nature and there are no good ways to qualitatively predict these errors.

2.2 Error Decomposition

Mobile robot motion errors can be decomposed as shown in Fig. 1: The first is the orientation error e_θ, which is defined as the difference between the real robot orientation and the desired robot orientation. It is the *most significant error* because it will result in a contour error, which grows with the distance traveled without a bound. The contour error e_c is defined as the distance between the actual robot position and the desired robot position in the direction perpendicular to the direction of travel. The contour error is the direct result of the orientation error. The third error is the tracking error e_t, which is the distance between the actual position and the desired position in the direction of travel. The tracking error does not have a very significant effect on robot motion accuracy.

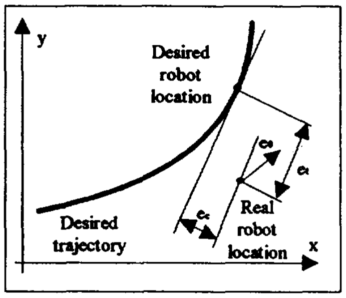

Fig. 1 Motion error decomposition

The main problem with robot motion errors is that they can grow without a bound, and that they increase nonlinearly with the distance traveled because of the *accumulation* of the orientation error. The key task is therefore to control the growth of the orientation error. The unbounded growth of motion error is in most part caused by systematic external errors under normal conditions.

2.3 Robot-Environment System Modeling

A fundamental characteristic of adaptive motion control is that both the robot and its environment are included in the system model. A dead-reckoning model aimed at improving path following accuracy by introducing error terms was introduced by Banta [2]. In Banta's work, robot position (x_k, y_k) and orientation θ_k at time k are given by:

$$x_k = x_{k-1} + \Delta u_k \cos((\theta_k + \theta_{k-1})/2) + \beta_x \Delta u_k \quad (1)$$

$$y_k = y_{k-1} + \Delta u_k \sin((\theta_k + \theta_{k-1})/2) + \beta_y \Delta u_k \quad (2)$$

$$\theta_k = \theta_{k-1} + \Delta\theta_k + \beta_\theta \Delta u_k \quad (3)$$

Where Δn_k^L and Δn_k^R are the measured angular displacements of the left and right wheels in each sampling period, $\Delta u_k = (\Delta n_k^L + \Delta n_k^R)/2$ is the distance traveled by the robot in each sampling period, $\Delta\theta_k = (\Delta n_k^R - \Delta n_k^L)/b_w$ is the orientation change during a sampling period, and β_x, β_y, and β_θ are the error coefficients. The errors are assumed to be proportional to the distance traveled by the robot.

Although this model is very simple, it is able to represent very important error sources and it has been experimentally proved to provide good results [2]. This model will be used in our adaptive motion control. The error model given by Eq. (3) indicates that the orientation error is a linear function of the distance traveled by the robot. Our experimental results support this assumption. We can think of the composite effect of all the error sources as an *effective difference in wheel diameters*.

3. Adaptive Motion Control

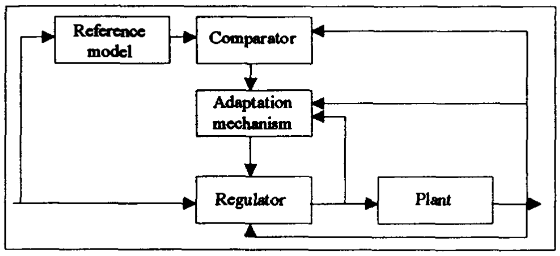

Fig. 2 A model-reference adaptive control system

In a model-reference adaptive control (MRAC) system (Fig. 2), there are four main components: (1) a reference model that specifies desired performances; (2) an adjustable system whose performances should be as close as possible to that of the reference model; (3) a comparator that forms the error between the states or the outputs of the reference model and of the adjustable system; (4) an adaptation mechanism that processes the error in order to modify accordingly the control or the parameters of the adjustable system.

The first implementation of MRAC in robotics was done by Dwbowsky et. al [8], where a parametric optimization technique was used and stability was investigated for the uncoupled, linearized system model. Most of the recent research efforts have been focused on stability based methods, particularly, the hyperstability theory [5]. Craig included nonlinearity compensation along with a feedback portion and parameter identification features [9]. In Craig's approach, the plant model does not have to be the same as the real plant and only performance convergence is needed instead of parameter convergence.

The MRAC scheme was chosen for our system for the following reasons:

1. *It requires performance convergence rather than parameter convergence.* The objective is to achieve desired performance. Normally parameter convergence can only be realized when some additional conditions are satisfied [10].

2. *The reference model is used to specify the desired performance and to monitor the state of the robot.* The reference model itself can be adjusted according to the operating conditions and the environment.

3. *Small computational load.* The on-line identification and design procedures of a self-tuning system can be computational intensive and stability problems often occur when the number of the variables to be estimated becomes large [11].

3.1 Cross-Coupling Control

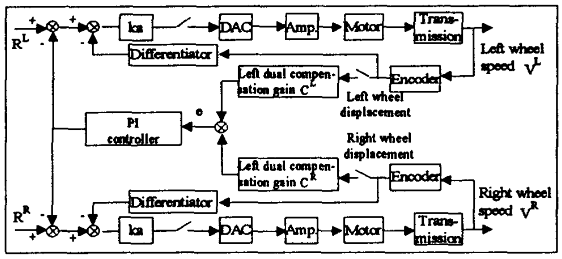

Fig. 3 Cross-coupling motion controller

The objective of cross-coupling (CC) control is to reduce the orientation error by coordinating the control of both drive loops [12]. For example, if a differential-drive mobile robot must follow a straight-line path, then the speed of the two drive wheels should be identical (assuming there are no external errors). In CC control (Fig. 3), in addition to the two conventional proportional control loops for controlling error in each drive loop, there is a proportional and integral controller that is used to control error e, which is proportional to the robot orientation error and is calculated in real time. If there exists orientation error, a correction signal will be generated in addition to the corrections in each drive loop. A thorough analysis of the cross-coupling method was presented in [3] and the conclusions are summarized below:

1. *Cross-coupling control directly reduces (or eliminates) the orientation error by coordinating the velocities of the two drive control loops.* The most important advantage of cross-coupling control is that it directly controls the *most significant error* (orientation error), while conventional controllers attempt to reduce the individual errors in each drive loop. The other advantage of cross-coupling is that the corrections occur in both control loops simultaneously, and the result is short settling time as well as excellent disturbance rejection.

2. *At steady state,* $v^L / v^R = c^L / c^R$ *and the steady state orientation error caused by the continuous disturbances is eliminated.*

3. *Combined cross-coupling and encoder compensation gain.* Dual compensation gains allow the robot to travel on curved paths and to compensate for known external

errors. The cross-coupling gains c_1^L and c_1^R, allow the robot to follow curved paths. For mobile robots, complicated curved paths can often be constructed from linear and circular segments. The encoder compensation gains c_2^L and c_2^R, are used to compensate for known external errors (e. g., different drive wheel diameters). If the left drive wheel diameter, d^L is larger than that of the right drive wheel, d^R and we give the same speed commands to both control loops, then the result is a circular path. However if we set $c_2^L = 1$ and $c_2^R = d^L / d^R$, the error is compensated for.

The encoder compensation gains are used as the adjustable parameters in the adaptive controller. Since there are many factors affecting motion errors and many of them changes with the robot operation and its environment, a fixed set of encoder compensation gains can not provide satisfactory performance over a wide range of operating conditions. An adaptive controller is needed to adjust the compensation gains to compensate for the motion errors. The final dual compensation gain values are the product of the cross-coupling and encoder compensation gains, i. e., $c^L = c_1^L c_2^L$ and $c^R = c_1^R c_2^R$.

In our motion controller, CC control is employed to compensate for internal errors, while adaptive control is used to compensate for external errors by adjusting encoder compensation gains (Fig. 4). Sonars are used to measure the actual robot orientation with respect to a reference (e.g., a section of straight wall).

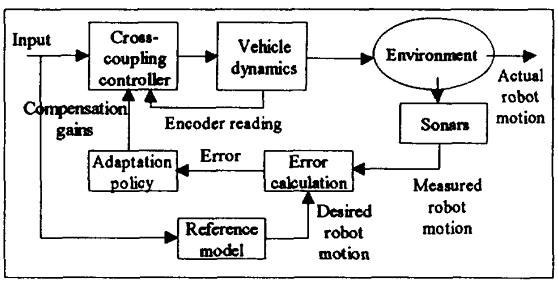

Fig. 4 Proposed adaptive motion control system

3.2 Encoder Gain Adaptation by Hyperstability Method

There are three basic designs for a MRAC system [4], local parametric optimization, Lyapunov Functions, and hyperstability approach. Stability problems are inherent in MRAC design due to their time-varying nonlinear characteristics. Therefore, a satisfactory MRAC system must first be shown to provide stability for the whole system. The adaptive control design based on the use of *hyperstability* and *positivity* concepts, is the most successful approach in the design of model reference adaptive control systems [10]. In our design, the adjustable parameters are the encoder compensation gains and the basic assumptions are: (1) the adaptation takes place only when the robot moves on a straight line with constant speed; (2) the adaptation occurs at a much lower frequency compared to the sampling rate of the cross-coupling control; (3) the two drive loops have the same parameters and there is no disturbance.

The design procedures consist of the following steps [10]: (1) Transform the MRAC system into the form of an equivalent feedback system composed of two blocks, one in the forward path and one in the feedback path as shown in Fig. 5; (2) Find solutions for the part of the adaptation laws which appears in the feedback path of the equivalent system such that the *Popov integral inequality* is satisfied; (3) Find solutions for the remaining part of the adaptation law which appears in the forward path of the equivalent system such that the forward path is a hyperstable block; (4) Specify the adaptation law explicitly for the original MRAC system.

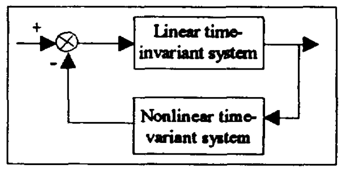

Fig. 5 Equivalent feedback system

In this study, we concentrate on the orientation error of the robot. Based on the orientation error model given in Eq. (3), the reference model is chosen as

$$\theta_m(k) = \theta_m(k-1) + (\Delta u^R(k) - \Delta u^L(k))/b_w \quad (4)$$

Since the adaptation occurs at a much lower frequency, we can assume that at each adaptation step, the cross-coupling loop has reached its steady state [3], i.e.,

$$v^L = hk_a k_b c^R (R^L + R^R)/(1 + hk_a k_b)(c^L + c^R)$$

$$v^R = hk_a k_b c^L (R^L + R^R)/(1 + hk_a k_b)(c^L + c^R)$$

where R^L and R^R are the input velocity commands, v^L and v^R are the wheel speeds, ka is the proportional loop gain, k_b is the motor gain and h is the encoder gain. Since

$$(\Delta u_k^L - \Delta u_k^R)/b_w = (v^R - v^L)\Delta t/b_w$$

$$= hk_a k_b (c^L - c^R)(R^L + R^R)\Delta t/b_w(1 + hk_a k_b)(c^L + c^R)$$

The reference model can be rewritten as

$$\theta_m(k) = \theta_m(k-1) + p_m u_1(k) \quad (5)$$

where $u_1 = hk_a k_b (R^L + R^R)\Delta t/b_w(1 + hk_a k_b)$, Δt is the sampling period of the adaptive loop, and $p_m = (c^L - c^R)/(c^L + c^R)$.

The controlled system can be modeled as

$$\dot{\theta} = (v^R - v^L)/b_w + \beta_\theta (v^R + v^L)/2 \quad (6)$$

Since we are only interested in the ratio of the two speeds v^L and v^R. The effects of b_θ can be absorbed into the encoder compensation gains. The above equation can be rewritten as

$$\theta(k)=\theta(k-1)+p(k+1)u_1(k) \quad (7)$$

where

$p(k+1) = (c^L(k+1)-c^R(k+1))/(c^L(k+1)+c^R(k+1))$.

To simplify the implementation, we can set c^L=1 and only changes c^R. Then we have $p(k+1) = (1-c^R(k+1))/(1-c^R(k+1))$ and c^R can be found by $c^R(k+1)=(1-p(k+1))/(1+p(k+1))$.

The error is defined as $e(k) = \theta(k) - \theta_m(k)$. The adaptation algorithm is chosen to be the integral algorithm, which is given as

$$p(k+1) = p(k) + \gamma e(k+1)\, u_1(k) \quad (8)$$

where γ is the adaptation gain and e(k+1) can be expressed as:

$$e(k+1) = (\theta_m(k)-\theta(k) + (p_m - p(k))u_1(k))/(1+\gamma u_1^2(k))$$

In order to show the stability and convergence of the system, we first decompose the system into a linear time-invariant system plus a non-linear time-variant feedback system. The system can be rewritten as

$$e(k+1) = \theta_m(k+1)-\theta(k+1)$$

$$= e(k) + p_m u_1(k) - p(k+1)u_1(k) = e(k)+m(k+1)$$

where

$$m(k+1)=p_m u_1(k)-p(k+1)u_1(k) \quad (9)$$

The decomposed system is shown in Fig. 6. In order to prove the stability of the system, the two blocks can be examined separately.

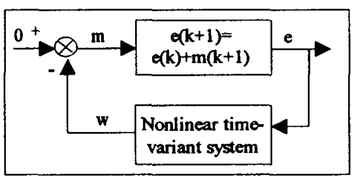

Fig. 6 Resultant equivalent feedback system

The hyperstability theorem states that[10]: *If the feedforward block is such that the feedback system is globally (asymptotically) stable for all feedback blocks satisfying the Popov integral inequality, one then says that the feedback system is (asymptotically) hyperstable and that feedforward block is a hyperstable block.*

It is easy to show that the linear block, whose transfer function is G(z) = z/(z-1), is positive real. Next we need to examine if the non-linear block satisfies the Popov integral inequality [10],

$$\eta(N)=\sum_{k=0}^{N} v'(k)w(k)\geq -r_0^2, (N\geq 0)$$

where v is the input vector, w is the output vector of the feedback block, and r_0^2 is a finite positive constant.

Substituting Eq.(8) into Eq.(9), we get,

$$m(k+1)=(p(k+1)-p_m)u_1(k)$$

$$=(p(0)+\gamma\sum_{i=0}^{k} e(i+1)u_1(i)-p_m)u_1(k)$$

Then $\sum_{k=0}^{N} e(k+1)w(k+1)=\sum_{k=0}^{N} e(k+1)(-m(k+1))$

$$=\sum_{k=0}^{N}(\gamma\sum_{i=0}^{k} e(i+1)u_1(i)+p(0)-p_m)e(k+1)u_1(k)\geq -\frac{1}{2\gamma}(p(0)-p_m)^2$$

The inequality is obtained from the following known inequality [10]: $\sum_{k=0}^{N} f(k)(\alpha\sum_{i=0}^{k} f(i)+c)\geq -\frac{1}{2\alpha}c^2$

From the above analysis, we can observe that this system is always stable no matter what the adaptation gain is. However, this result is obtained by assuming that there is no measurement noise and other disturbances. In practice, a useful adaptation gain will be limited by the measurement accuracy and disturbances.

4. Experimental Investigation

Fig. 7 Experimental vehicle

In this section, we will present an experimental evaluation of the MRAC motion controller. We tested the performance of our adaptive control method on a commercially available differential-drive *LabMate* platform (Fig. 7). Robot orientation measurement is provided by two sonars on one side of the vehicle. In the experiments, the robot moved along a straight wall, while the two sonars measured the distance to the wall. Based on the distance measurements and the distance between the two sonars, robot orientation can be found and used in error compensation. Each sonar is fired at least once every 80 ms. Ten readings from each sonar were gathered and averaged for better accuracy and reliability. Experimental results show that the accuracy of orientation measurement

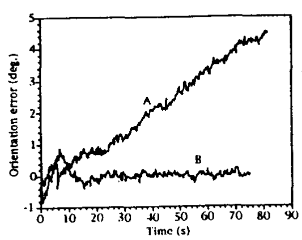

Fig. 8(a)Robot orientation error
A - without compensation
B - with compensation

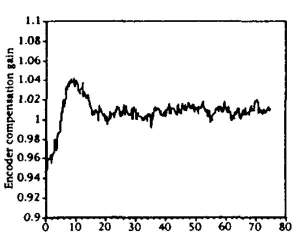

Fig. 8(b)Convergence of the encoder compensation gains was about 1°, while the repeatability of the angle measurement was within 0.2°. It took about 0.7 s to get a valid orientation measurement.

In the experiments, the right drive wheel of the robot was covered with three layers of masking tapes. In the first experiment, the robot was instructed to follow a straight line, and the robot orientation was measured using sonars and used for adaptation. One experimental result is shown in Fig. 8. Fig. 8(a) shows the orientation error without (A) and with (B) the adaptive controller. Clearly, when CC control is used alone, it can not compensate for external errors since no external sensory information is used in the CC controller. However, the adaptive motion controller successfully compensated for external errors by adjusting encoder compensation gains based on the absolute orientation measurements. Fig. 8(b) shows the convergence of the adaptation gain. The final value is stored. From the figures, we can observe that the adaptive controller works well under the experimental conditions, and can effectively compensate for motion errors.

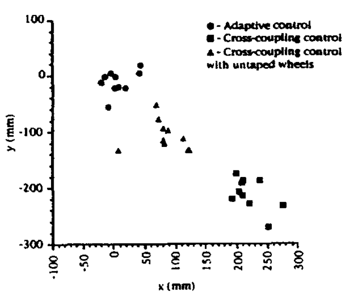

Fig. 9 Motion control accuracy

In the next experiment, the converged value of the encoder compensation gain was used to compensate for the motion error. The robot was instructed to follow a 2mx2m square path (starting at (0,0)) clockwisely with and without the compensation. The final position after completing the path is shown in Fig. 9. We can observe that the compensation is very effective and the motion accuracy improved significantly. We can also observe that the absolute motion accuracy is also improved significantly if we compare the result of the adaptive control with the result of CC control with untapped wheels. The reason is that there are systematic errors even when the wheel is not taped, and the adaptive controller compensates for them as well.

5. Conclusions

We described the design and implementation of an adaptive motion controller on a differential-drive mobile robot. The adaptive controller is used to compensate systematic motion errors through "learning," i.e., it detects errors from absolute position measurements and compensate for these errors. The adaptation algorithm is based on the hyperstability theory to provide good convergence characteristics and stability, and it was found to improve robot motion accuracy significantly.

Acknowledgment: This research is supported by the Department of Energy grant DE-FG02-86NE37969. We appreciate the efforts of Mr. Harry Alter from DOE in coordinating this national project.

6. Reference

[1] P. F. Muir and C. P. Neuman, "Kinematic Modeling of Wheeled Mobile Robot," *Technical Report, CMU-RI-TR-86-12*, The Robotics Institute, Carnegie Mellon University, June 1986.

[2] L. Banta, "A Self Tuning Navigation Algorithm," *Proc. of the 1988 IEEE Inter. Conf. on Robotics and Automation*, pp. 1313-1314, 1988.

[3] L. Feng, Y. Koren and J. Borenstein, "A Cross-Coupling Controller for a Differential-drive Mobile Robot", *IEEE Control Systems Magazine*, Dec. 1993, pp.35-43.

[4] K. J. Astrom and B. Wittenmark, Adaptive Control. Readings, MA: Addison-Wesley, 1989.

[5] J. E. Slotine and W. Li, "On the Adaptive Control of Robot Manipulators," *Proc. of the 1987 IEEE Inter. Conf. on Robotics and Automation*, 1987.

[6] A. J. Koivo and T. Guo, "Adaptive Linear Controller for Robotic Manipulators," *IEEE Trans. on Automatic Control*. Vol. 28, No. 2, 1983.

[7] L. Feng, Y. Fainman, and Y. Koren, "Estimation of Absolute Position of Mobile System by Opto-electronic Processor," *IEEE Trans. on Man, Machine and Cybernetics*, Nov./Dec. 1992.

[8] S. Dubowsky and D. T. DesForges, "The Application of Model-Referenced Adaptive Control to Robotic Manipulators," *ASME J. of Dynamic Systems, Measurement, and Control*, Vol. 101, 1979.

[9] J. J. Craig, Adaptive Control of Mechanical Manipulators. Readings, MA: Addison-Wesley, 1988.

[10] Y. D. Landau, Adaptive Control, The Model Reference Approach. New York: Marcel Dekker, Inc., 1979.

[11] S. Tosunoglu and D. Tesar, "State of the Art in Adaptive Control of Robotic Systems," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 24, No. 5, pp. 552-561, Sept. 1988.

[12] J. Borenstein and Y. Koren, "Motion Control Analysis of a Mobile Robot", *J. of Dynamic Systems, Measurement, and Control*, Vol. 109, pp.73-79, June 1987.