IEEE Transactions on Robotics and Automation, Vol. 14, No. 6, December 1998, 963-969

Experimental Results from Internal Odometry Error Correction with the OmniMate Mobile Robot[1]

by
Johann Borenstein
The University of Michigan
Advanced Technologies Lab
1101 Beal Avenue
Ann Arbor, MI 48109-2110
Ph.: 313-763-1560, fax: 313-944-1113, email: johannb@umich.edu

ABSTRACT

This paper presents experimental results of test with a new method for detecting and correcting odometry errors without inertial or external-reference sensors. This method, called *Internal Position Error Correction* (IPEC), has been implemented on a new, commercially available mobile robot called "*OmniMate,*" which was specifically designed for the implementation of the IPEC method. The results presented in this paper show that the OmniMate can improve odometric accuracy by one order of magnitude over conventional mobile robots.

The foremost advantage of the OmniMate with IPEC over conventional mobile robots is that the OmniMate's odometry is almost completely insensitive to even severe irregularities of the floor, such as bumps, cracks, or traversable objects. With conventional mobile robots such irregularities can cause large odometry errors with potentially catastrophic effects (i.e., mission failure), thus mandating frequent external registrations to correct for possible odometry errors. The OmniMate with IPEC, on the other hand, detects and corrects such non-systematic odometry errors, thus allowing more reliable travel over larger distances.

1 A shorter version of this paper was presented at the *Seventh Topical Meeting on Robotics and Remote Systems*, Augusta, Georgia, April 27 - May 1st, 1997.

1. INTRODUCTION

Odometry is the most widely used navigation method for mobile robot positioning because it provides good short-term accuracy, is inexpensive, and allows very high sampling rates. However, the fundamental idea of odometry is the integration of incremental motion information over time, which leads inevitably to the accumulation of errors. Particularly, the accumulation of orientation errors will cause large position errors [1], which increase proportionally with the distance traveled by the robot. Nonetheless, most researchers agree that odometry is an important part of a robot navigation system and that navigation tasks will be simplified if odometric accuracy can be improved.

When trying to measure and reduce odometry errors, it is important to understand the distinction between “systematic” and “non-systematic” odometry errors. Systematic errors are those errors that are inherently part of the robot’s kinematic or controller properties, independently from the robot’s environment. Non-systematic errors are those that depend on the robot’s environment and differ from one environment to another. This distinction is important because each one of these two groups affects mobile platforms differently, their remediation differs, and, most important, both groups require different measuring techniques in order to obtain meaningful and comparable experimental data. We categorize odometry errors as follows:

- **Systematic errors**
  - Unequal wheel diameters
  - Average of both wheel diameters differs from nominal diameter
  - Misalignment of wheels
  - Uncertainty about the effective wheelbase (due to non-point wheel contact with the floor)
  - Limited encoder resolution
  - Limited encoder sampling rate

- **Non-systematic errors**
  - Travel over uneven floors
  - Travel over unexpected objects on the floor
  - Wheel-slippage (slippery floors, over-acceleration, skidding in fast turns, etc.)
  - External forces (interaction with external bodies)
  - Internal forces (e.g., castor wheels)
  - Non-point wheel contact with the floor

Some research has examined the use of inertial navigation aids. However, experimental results from the Université Montpellier in France [2] and from the University of Oxford in the U.K. [3], indicate that a purely inertial navigation approach is not realistically advantageous (i.e., too expensive or too inaccurate) for mobile robot applications.

Another, quite promising solution, is the use of very accurate fiber optic gyros. These devices have become reasonably inexpensive during the last few years. The limitation of these sensors is their inherent drift rate. Komoriya and Oyama [4] tested the Optical Fiber Gyro OFG-3 from Hitachi [5] and found its drift rate to be quite linear with 0.00317°/s (11.4°/hr). One potential problem with fiber optic gyros in mobile robot applications is the *minimum detectable rotation* rate, which is 0.05°/s for the Hitachi OFG-3. Another laser gyro, the Andrew 3ARG-A, [6] has similar specifications. If either gyro was installed on a robot with a systematic error (see Section 3) of 1° per 10 meter linear travel, then neither gyro would detect this systematic error at speeds lower than 0.5 m/s [7].

1.1 The OmniMate Design

The OmniMate is a Multi-degree-of-Freedom (MDOF) mobile platform with omni-directional motion capabilities. The design of the OmniMate is based on an earlier prototype, called the "CLAPPER." The CLAPPER was invented and built at the University of Michigan, where the unique odometry error correction methods discussed in this paper were first implemented and demonstrated (see [1]).

The OmniMate is made from two differential-drive TRC *LabMate* platforms (here called "trucks") as shown in Figure 1 and Figure 2. The front truck can rotate around rotational joint A, which is bolted to the bottom of a rigid loading deck. The rear truck can rotate around rotational joint B, which is welded to a slider assembly. The slider assembly can move linearly along slider bars that are welded at their ends to the bottom of the loading deck. Rotary encoders mounted on joints A and B measure the relative rotation between each truck and the loading deck, while a linear encoder measures the position of the linear slider assembly, from which the distance between the center points of the two trucks can be computed. Additional joints not shown in Figure 2 allow for limited pitch, roll, and yaw motion of the trucks relative to each other, to accommodate uneven ground. Four optical incremental encoders mounted on the four drive wheels provide raw odometry data.

Because of the linear slider the two trucks can freely move relative to each other. This UM design is called "compliant linkage." The purpose of the compliant linkage is to absorb the inevitable momentary controller errors that can cause wheel slippage in conventional, rigidly-built MDOF mobile robots, as reported by Reister [8], West and Asada [9], Hirose and Amano [10], or Pin and Killough [11].

Figure 1 shows how the OmniMate design provides a completely flat, 180×90 cm (72×36 in) loading deck that is available exclusively for the end-user's payload. A 24-Volt battery pack, designed to power user-installed equipment and the onboard control computer, is installed underneath the loading-deck. In addition, each of the trucks is individually powered by a 24-Volt battery pack installed inside of each truck. Control and feedback signals to and from the trucks are passed through slip-ring assemblies. The onboard motion control computer is a 486/100 MHz PC-compatible single board computer.

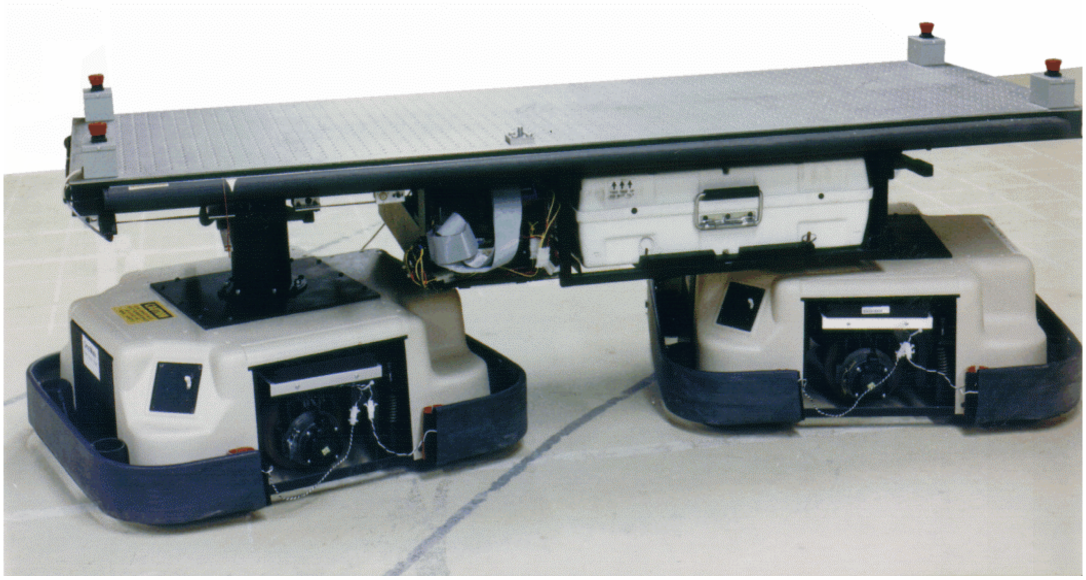

Figure 1: The OmniMate is based on two TRC LabMate "trucks" connected by a compliant linkage. This design provides an uncluttered 180×90 cm (72×36-in) loading deck for up to 114 Kg (250 lbs.) of payload.

1.2 The OmniMate Control System

The onboard computer controls and coordinates the motion of the two trucks in a user-transparent manner. This means that the user (or a high-level control program) commands the desired translation and rotation of the vehicle only with respect to the loading-deck, without worrying about the motion of the two trucks that would result in the desired motion of the loading-deck.

Another function of the control system is to perform the *Internal Position Error Correction* (IPEC),

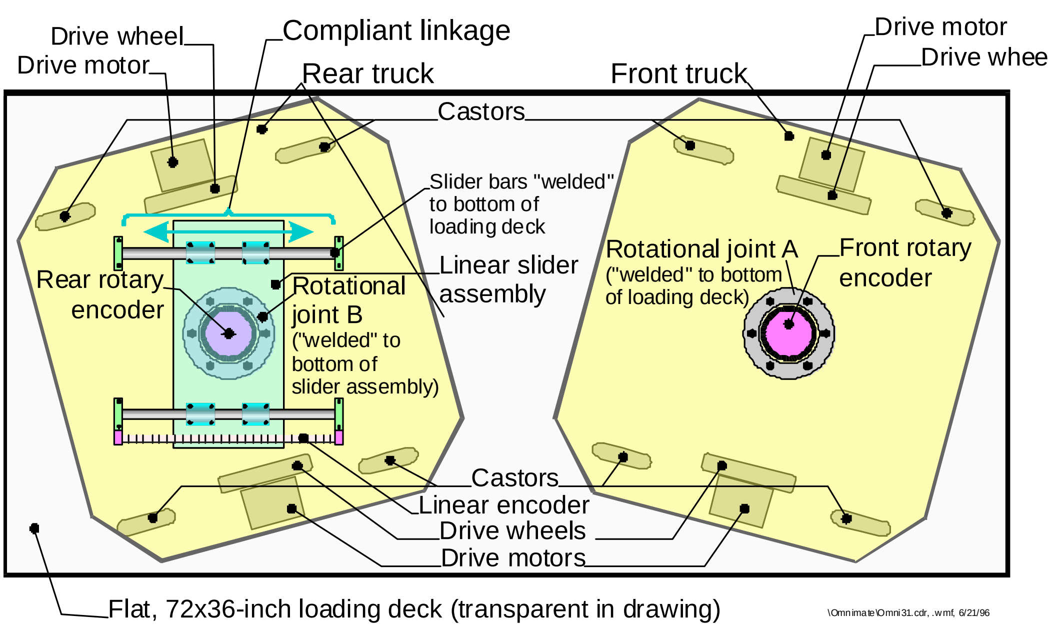

Figure 2: Schematic diagram of the OmniMate mobile robot.

which is capable of detecting and automatically correcting odometry errors caused by bumps, cracks, or other irregularities on the floor. The IPEC method is described in detail in [1] and is reviewed briefly in Section 2. Section 3 provides and discusses the experimental results obtained with the OmniMate.

2. Internal Position Error Correction

Internal Position Error Correction (IPEC) allows two mobile robots to mutually correct their odometry errors, even while both robots are in continuous, fast motion. To implement this method, it is required that both robots can measure their relative distance and bearing continuously and accurately. The two trucks of the OmniMate and the encoders installed on the compliant linkage are designed to meet this requirement.

2.1 Principle of Operation

The principle of operation of IPEC is based on the concept of *error growth rate,* which was first presented by Borenstein [1]. According to this concept one can distinguish between "fast-growing" and "slow-growing" odometry errors. For example, when a differentially steered robot traverses a floor irregularity it will immediately experience an appreciable orientation error (i.e., a *fast-growing* error). The resulting lateral displacement error, however, is initially very small (i.e., a *slow-growing* error), but grows in an unbounded fashion as a consequence of the orientation error. The internal error correction algorithm (explained below) performs relative position measurements with a sufficiently fast update rate (for example: 15 ms in the OmniMate) to allow each truck to detect the *fast-growing* errors in orientation, while relying on the fact that the lateral position errors accrued by both trucks during the sampling interval were small.

To better understand how the IPEC method makes use of the *error growth rate* concept, consider, for example, the typical case in Figure 3, in which both trucks are initially aligned while traveling forward. In this initial configuration (i.e., before truck A encounters a bump) truck A can expect the center of truck B straight behind it, on line L_e, and the rotary encoder at joint A will confirm this expectation. After traversing a bump Truck A's orientation will change[2] — a fact unknown to Truck A's odometry computation. Truck A is therefore still expecting Truck B along the extension of line L_e. However, because of the physically incurred rotation of Truck A, the absolute encoder on truck A will report that truck B is now actually seen along line L_m. The angular difference between L_e and L_m is the thus measured odometry orientation error of Truck A, $\Delta\theta_m$, which can be corrected immediately.

2) In a differential-drive vehicle the controller tries to keep the number of pulses read from each wheel encoder equal, to accomplish straight-line motion. When one of the wheels travels over a bump, some of its rotation is spent on traveling up and down the bump, while the unaffected wheel travels forward. As a result, the wheel that traversed the bump accomplishes less forward motion than the other wheel, resulting in a small turn of the vehicle into the direction of the bump.

The above explanation of the IPEC method requires four additional clarifications:

1. The *actual* odometry orientation error of truck A is the angle $\Delta\theta_a$ between line L_e and L_a and not $\Delta\theta_m$. However, the discrepancy between lines L_a and L_m, caused by the lateral displacement of truck A, e_{lat}, during the sampling interval is much smaller in reality than in the exaggerated drawing of Figure 3. Borenstein [1] shows that this discrepancy is negligible in practice.
2. One should note that even if Truck B encountered a bump at the same time, the resulting rotation of Truck B would not affect the orientation error measurement. This is so because the pure rotation of truck B does not immediately affect the geometric configuration in Figure 3.
3. It can be shown easily that the principle of operation explained above works similarly well even when both trucks are not aligned but, rather, face arbitrary directions.
4. The experimental results presented in Section 4 illustrate the effectiveness of IPEC with regard to certain non-systematic errors, such as irregularities on the floor. In informal experiments, not reported here, we observed that IPEC also corrects other non-systematic *orientation* errors, such as those caused by wheel slippage or external or internal forces. Thus, if, for example, one wheel slipped on a slippery surface, then IPEC would considerably reduce

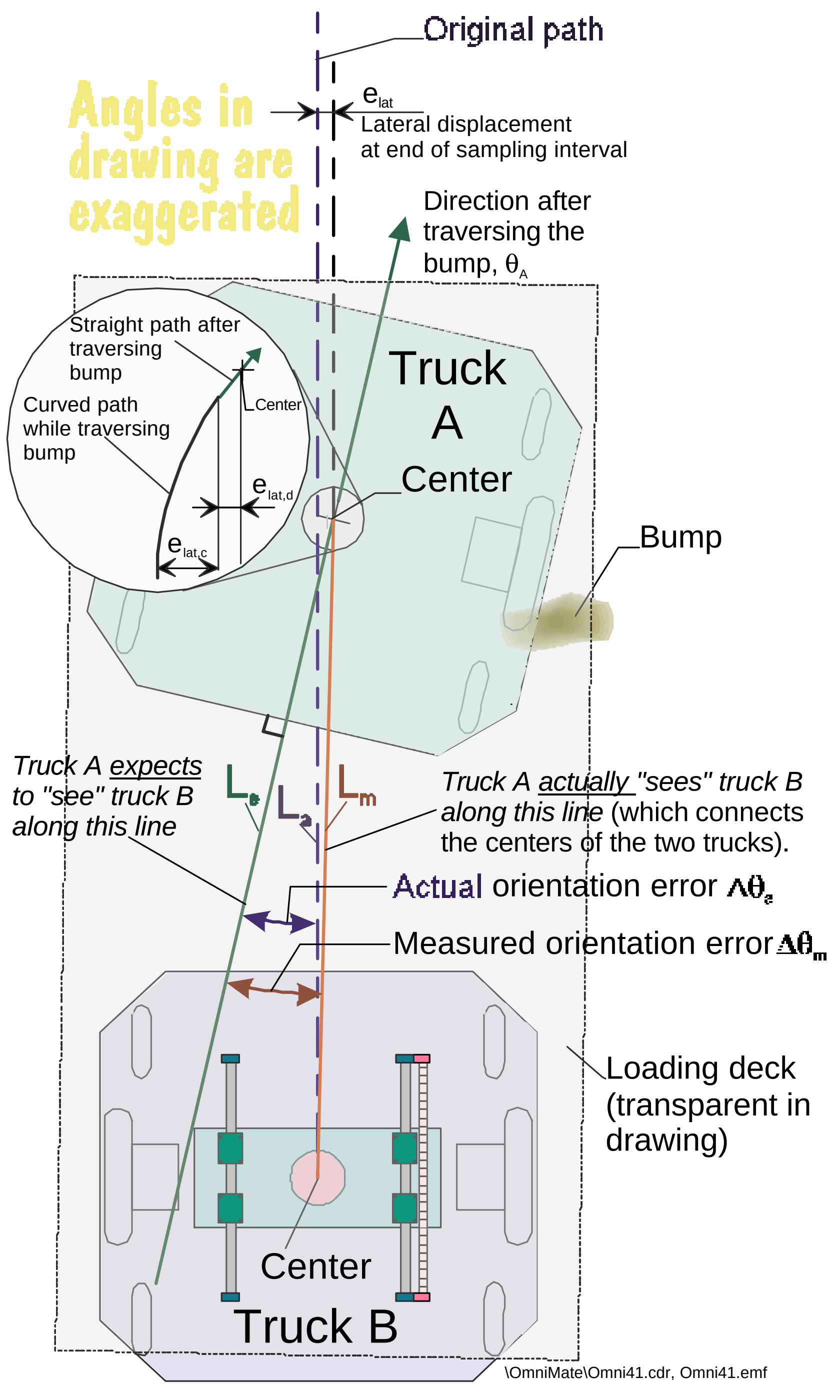

Figure 3: After traversing a bump, the resulting change of orientation of Truck A can be measured relative to Truck B.

the resulting orientation error. If, however, both wheels slipped by about the same amount, then IPEC would not detect or correct the resulting position error.

2.2 Implementation of the IPEC Algorithm

In this section we describe step-by-step how the IPEC algorithm is implemented on the OmniMate platform.

1. At sampling interval *i*, compute the position and orientation of truck A, $x_{A,i}$, $y_{A,i}$, $\theta_{A,i}$, respectively, from conventional odometry.
2. Compute the direction of line L_m, which connects the centers of truck A and truck B. This direction is denoted θ_0.

$$q_{0,i} = \frac{y_{A,i} - y_{B,i-1}}{x_{A,i} - x_{B,i-1}} \qquad (1)$$

3. Compute the angle between the direction of motion of truck A, $\theta_{A,i}$, and the direction of line L_m, $\theta_{0,i}$, . This angle is denoted θ_{expect}.

$$\theta_{expect,i} = \theta_{0,i} - \theta_{A,i}; \qquad (2)$$

Note that if there are no odometry errors, then θ_{expect} should be equal to the *true* angle between truck A and line $L_{m.}$ This angle, denoted α, is measured by the encoder located at point A.

4. Determine the difference between θ_{expect} and α, which is denoted $\Delta\theta_m$. The IPEC method assumes that $\Delta\theta_m$ is the rotational odometry error of truck A.

$$\Delta\theta_{m,i} = \alpha_i - \theta_{expect,i} \qquad (3)$$

5. Correct θ_A by the amount of the thus found rotational odometry error, $\Delta\theta_m$.

$$\theta'_{A,i} = \theta_{A,i} + \Delta\theta_{m,i}; \qquad (4)$$

6. Compute the position and orientation of the rear truck from the kinematic chain, *not* from odometry:

$$x_{B,i} = x_{A,i} - L_i\cos\theta_{0,i}$$

and (5)

$$y_{B,i} = y_{B,i} - L_i\sin\theta_{0,i}$$

$$\theta_{B,i} = \theta_{0,i} + \beta_i \qquad (6)$$

where β is the relative angle between truck B and the direction of line L_m, measured by the encoder located at point B.

2.3 Comparison with other Odometry Improvements

The idea of mobile robots referencing each other has also been suggested by others. For example, Sugiyama [12] proposed to use two robots that could measure their positions mutually. When one of the robots moves to another place, the other remains still, observes the motion, and determines the first robot's new position. In other words, at any time one robot localizes itself with reference to a fixed object: the standing robot. Another approach was tested in simulations by Kurazume and Nagata [13]. This approach uses multiple cooperating mobile robots divided into two groups. When the members of group A move, the members of group B remain stationary and provide positioning beacons that the group A robots can use for absolute positioning. After a while, group A robots stand still and group B robots move.

In both cases the efficiency of the robots is severely limited by the stop-and-go approach. In contrast, the IPEC method described here works while both components of the robot are in motion — and independently of their speed (within the OmniMate's range of operating speeds – i.e., up to 0.5 m/s).

One other method for improving odometric accuracy is based on the separation of drive wheels from measurement wheels. One of the earliest implementations of this method is given in [14]. This method, while effective in reducing many systematic errors, is ineffective with most non-systematic errors.

3. EXPERIMENTAL RESULTS

In this section we present the experimental results of the odometric accuracy tests that were performed according to procedures called the "*UMBmark*" and "*extended UMBmark.*" With this procedure the robot is programmed to follow an approximate square-path, and, upon returning to the starting position, the difference between the robot's actual position and orientation is compared to the position and orientation measured by odometry. The results are the return position errors, εx and εy, and the return orientation error, $\varepsilon\theta$. One key-requirement of the UMBmark procedure is that the test be run several times in clockwise (cw) and in counter-clockwise (ccw) direction. A more detailed, formal description of these procedures is given in [15].

During the tests the OmniMate was equipped with what we call the "sonar calibrator," a device that uses three ultrasonic sensors to measure the distance between three points on the robot to two L-shaped walls. With the sonar calibrator the absolute position of the vehicle can be measured at the beginning and end of each run fully automatically, and, subsequently, the onboard computer can compute the *return position/orientation errors*. Using the sonar calibrator the UMBmark test with multiple laps can be run fully automatically in each direction. The accuracy of the sonar calibrator is about 0.3% of the distance of each sonar from the nearest wall and ±0.2° in orientation.

In a typical test the robot is programmed to traverse the four legs of the square path. The path will return the vehicle to the starting area, but because of odometry and controller errors, not precisely to the starting position. Since this test aims at determining odometry errors and not controller errors, the vehicle does not need to be programmed to return to its starting position precisely **C** returning approximately to the starting area is sufficient. Upon completion of the square path, the experimenter again measures the absolute position of the vehicle, using the fixed walls as a reference. These absolute measurements are then compared to the position and orientation of the vehicle as computed from odometry data. The result is a set of *return position/orientation errors* caused by odometry and denoted εx, εy, and $\varepsilon\theta$:

$$\begin{aligned} \varepsilon_x &= x_{abs} - x_{calc} \\ \varepsilon_y &= y_{abs} - y_{calc} \\ \varepsilon_\theta &= \theta_{abs} - \theta_{calc} \end{aligned} \tag{7}$$

where

ε_x, ε_y, ε_θ **C** Position and orientation errors due to odometry.
x_{abs}, y_{abs}, θ_{abs} **C** Absolute position and orientation of the robot.
x_{calc}, y_{calc}, θ_{calc} **C** Position and orientation of the robot as computed from odometry.

Following is a brief summary of the actually employed test procedure, which differed slightly from the formal UMBmark procedure.

3.1 Summary of the Test Procedure

Experiments were performed in sets of 10 laps along a rectangular path with rounded corners. The total length of the rectangular path (i.e., for one lap) was 18.5 meters (60 ft) and the platform performed a total of four 90°-turns in each lap. Four sets of fully automatic runs were performed:

Set 1: 10 laps with IPEC, cw

Set 2: 10 laps with IPEC, ccw

Set 3: 9 laps[3] without IPEC, cw

Set 4: 9 laps[3] without IPEC, ccw

In each of the four sets the first five laps were run without bumps (i.e., on marginally smooth concrete floor, with no bumps of more than 1 mm height). The remaining laps were run with artificial 9-mm diameter bumps placed under the OmniMate's wheels, as follows:

Lap #6: 20 bumps under the inside wheel of the front truck.

Lap #7: 20 bumps under the outside wheel of the front truck.

Lap #8: 10 bumps each under the inside wheels of the front truck and the rear truck, for a total of 20 bumps).

[3] See note following explanation of Lap #10 for reason why Lap #10 was omitted in runs without IPEC.

Lap #9: 10 bumps each under outside wheels of the front truck and the rear truck, for a total of 20 bumps.

Note that in Laps #6 through #9 the bumps were introduced manually and they were approximately evenly spaced along the whole path. However, no great accuracy in the spatial distribution of the bumps is required, because each of our cable-bumps introduces a rotational error of the same magnitude, no matter where along the path it is introduced. While the magnitude of the return position error varies depending on the placement of the bumps, the return orientation error, which is the only result of interest in the extended UMBmark test, does not. Rather, the return position error is the result only of the accumulation of individual orientation errors incurred along the path.

Lap #10: 20 bumps placed randomly under all wheels.

Note that the tests performed in Laps #6 - #10 differ from the procedure originally described as the extended UMBmark in Borenstein and Feng [15]. The reason for this change is that the original extended UMBmark test was designed for basic differential-drive mobile robots. Although the extended UMBmark test could be performed with the OmniMate without modification, we noticed during experimentation with the OmniMate that skeptical observers often asked if the placement of bumps under the wheels of the other truck would have any negative impact on the odometry error correction. To diffuse these concerns we modified the “with-bumps” procedure to include bumps under both the left and right wheels of the front- and the rear truck. Lap #10, with bumps placed randomly under all four wheels, was omitted in the runs without error correction, because the effects of random bumps cancel each other out and thus produce a misleading result: a zero- or near-zero error.

As a consequence to this change in the testing procedure it is not meaningful to compute the *average* return orientation error as prescribed by Borenstein and Feng [15]. Instead, we consider the *worst* orientation error from among any one of laps #5 through #10 as the representative worst error of runs with bumps. For completeness, we also note the worst position error, although this data is not meaningful for comparison purposes.

3.2 Data Collection

Our experiments were performed under certain conditions and premises that were not explicitly addressed by the original UMBmark procedures. Here is a summary:

- Each set of 10 laps with error correction was performed fully automatically under computer control. Thus, the 10 laps in each set were consecutive runs, not a selection of hand-picked runs. Furthermore, the two sets of cw and ccw runs were run immediately one after the other, without changing any of the robot’s parameters between sets.
- Laps #5 through #9 of the “no-error correction,” “with bumps” runs were made under human operator joystick control, since the OmniMate would otherwise exit the experimentation area due to the large position errors.

- We deviated from the prescribed extended UMBmark protocol by placing not 10, but rather 20 bumps under the robot's wheels to produce more noticeable errors and to highlight the unique error resilience of the OmniMate.
- A multi-Degree-of-Freedom vehicle like the OmniMate can be programmed to execute a rectangular path in an infinite number of poses (sets of momentary position and orientation). For example, the robot can be programmed to move straight forward for one leg of the rectangular path, then rotate on the spot through 90°, then move straight forward again for the second leg, and so forth. Alternatively, the robot can perform the 90°-turns in a "follow-the leader"-type mode, in which the rear truck tries to follow the exact same trajectory of the front truck. For our experiments we chose yet a different mode, in which we programmed the OmniMate to imitate the kinematic behavior of a car: the rear truck (similar to the rear axle of a car) loosely faced in the direction of the centerpoint of the front truck. This configuration was the easiest to program and involved the fewest parameters that might have influenced the IPEC error correction system. In informal experiments we ran the robot in other, arbitrary modes of turning under R/C joystick control. The resulting odometry errors were generally slightly worse, but not significantly so.

Figure 4 shows the experimental set-up used in all tests. Also shown are the traces of the centerpoints of the front and rear truck as recorded and plotted by the onboard computer for one particular lap. In each lap the robot started and finished in the area labeled "Start/Finish" in Figure 4. To produce the rectangular path we pre-programmed the four corner points as via-points for the OmniMate's "*pass_by*" command. The *pass_by* command implies that another motion command will follow and, to produce smooth, continuous motion, does not stop the robot at the *pass_by* location. Instead, the control program executes the next motion command as soon as the centerpoint of the front truck comes within a tolerance range of 50 cm (20 in) from the via-point. This is why in Figure 4 the trace of the front truck doesn't touch the via-points. The four via-points don't form a rectangle exactly, since the exact shape of the trajectory is irrelevant in the UMBmark test. What is important is that the robot turns through a total of 360° in each lap. The somewhat irregular placement of the via points was mandated by the need to keep the robot as far away from obstacles as possible, to allow for the large path deviations in runs without error correction.

One device worthwhile mentioning is the "*sonar calibrator*," which is used to measure the absolute position and orientation of the OmniMate before and after a run. The sonar calibrator comprises three Polaroid sonars: two sonars installed at the corners of the left side and one sonar in the rear of the OmniMate. The sonar calibrator uses the four stationary walls as absolute references. For cw runs the walls on the left-hand side and bottom of Figure 4 are used, while for ccw runs the walls at the center and top of Figure 4 serve as absolute references. Because of the relatively large distance between the rear sonar and the respective rear walls, measurements in y-direction are less accurate, on the order of ±3 cm (1.2 in). The accuracy of the side-facing sonars is better, on the order of ±3 mm (0.12 in), because of the shorter distance to the respective reflector walls. Using the two side-facing sonars, the sonar calibrator can determine the robot's true orientation with respect to the reflector walls with an accuracy of about ±0.2°.

At the beginning of each lap the OmniMate determined its absolute position with the sonar calibrator and initialized the odometry system with that data. Traveling at a maximum speed of 0.3 m/s (11.8 in/s) during straight segments, the robot slowed down near via-points. Then, when within the tolerance range of 50 cm (20 in) and while still moving, the robot would begin to align itself with the direction to the next via-point and, simultaneously, aim its front truck toward that new via-point. At the end of each run the sonar calibrator measured the robot's actual position and compared the result to the vehicle's internal position, based on odometry. The error, expressed as εx, εy, and $\varepsilon\theta$, was automatically recorded, the internal position (i.e., odometry) was reset to the actually measured one, and the robot continued with the next lap.

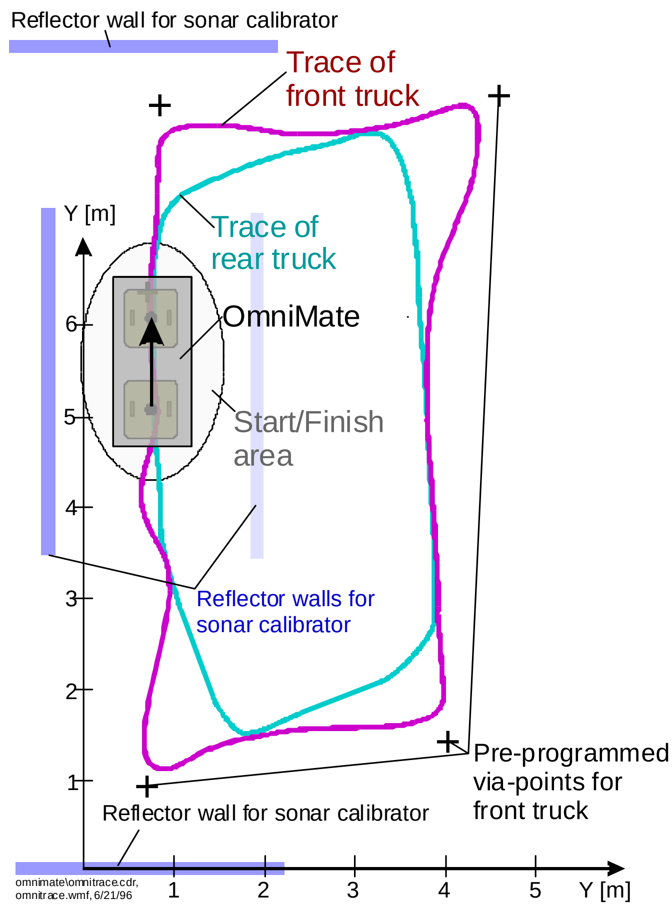

Figure 4: Experimental setup for the OmniMate experiments. Traces of the front and rear trucks as recorded and plotted by the onboard computer are also shown.

3.3 Test Results

The quantitative results of our tests are summarized in Table I. Graphical representations of these results are shown in Figs. 5 through 8.

Table I includes results required by the UMBmark test and some additional results that are not required by the UMBmark test. As mentioned before, the UMBmark procedure is designed to allow developers to compare the performance of different robots, or to help developers fine-tune the odometry parameters of a single robot. However, we recognize that end-users of mobile robots may want to know the worst accuracy of a robot for a given test. These worst-accuracy results are printed on shaded background in Table I.

It is interesting to note that for the variety of results shown in Table I the IPEC error correction method provides consistently one order of magnitude greater accuracy than that obtained from running the same vehicle without IPEC. The graphical representation of our results in Figures 6 through 9 confirms this observation visually.

Table I: Summary of numeric results. Results printed on shaded background are not required by the UMBmark test but are included to demonstrate the *worst* accuracy of the OmniMate with and without IPEC error correction in each type of experiment.

Worst of cw and ccw runs:	$r_{c.g.}$ [mm]		$r_{individual}$ [mm]		$\lvert e2_{avrg}\rvert$ [deg]		$\lvert e2_{individual}\rvert$ [deg]		$\lvert e\,q_i^{nonsys} - e\,q_{avrg}^{sys}\rvert$ [deg]	
	With IPEC	Without IPEC	With IPEC	Without IPEC	With IPEC	Without IPEC	With IPEC	Without IPEC	With IPEC	Without IPEC
No bumps	25	205	30	275	0.3	3.7	0.4	5.3	Not meaningful	
With bumps	Not meaningful		44	465	0.8	8.0	1.3	11.3	1.2	13.3

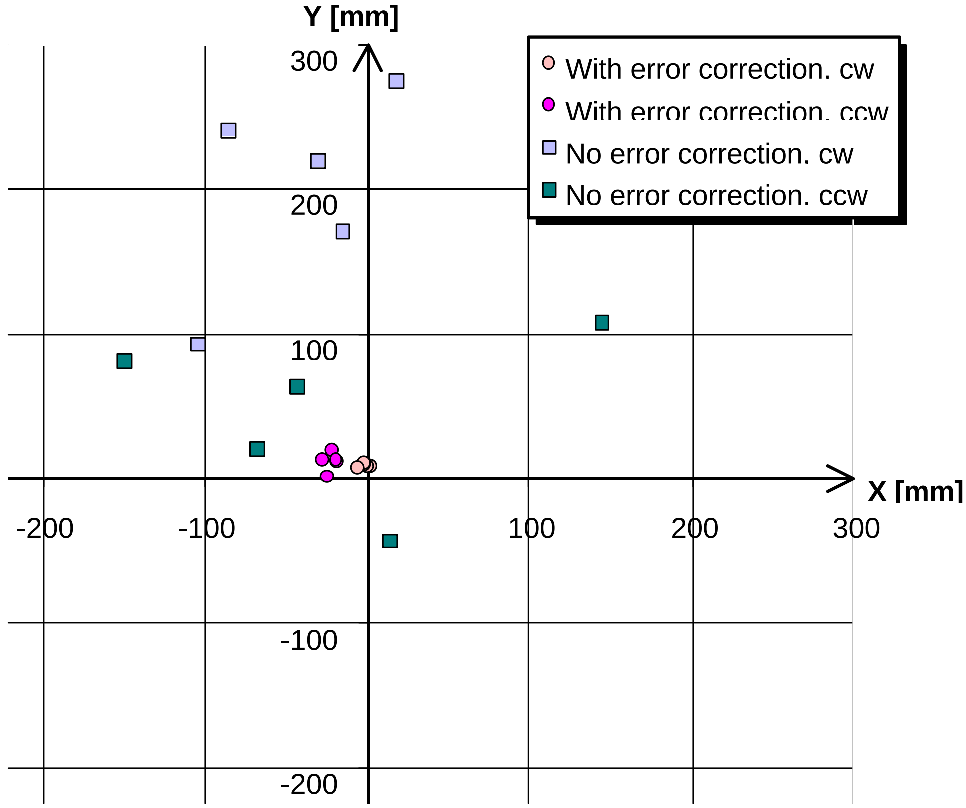

Figure 5: OmniMate return position errors after completing the 18.5 m rectangular path of Figure 4 on a smooth concrete floor ***without*** bumps.

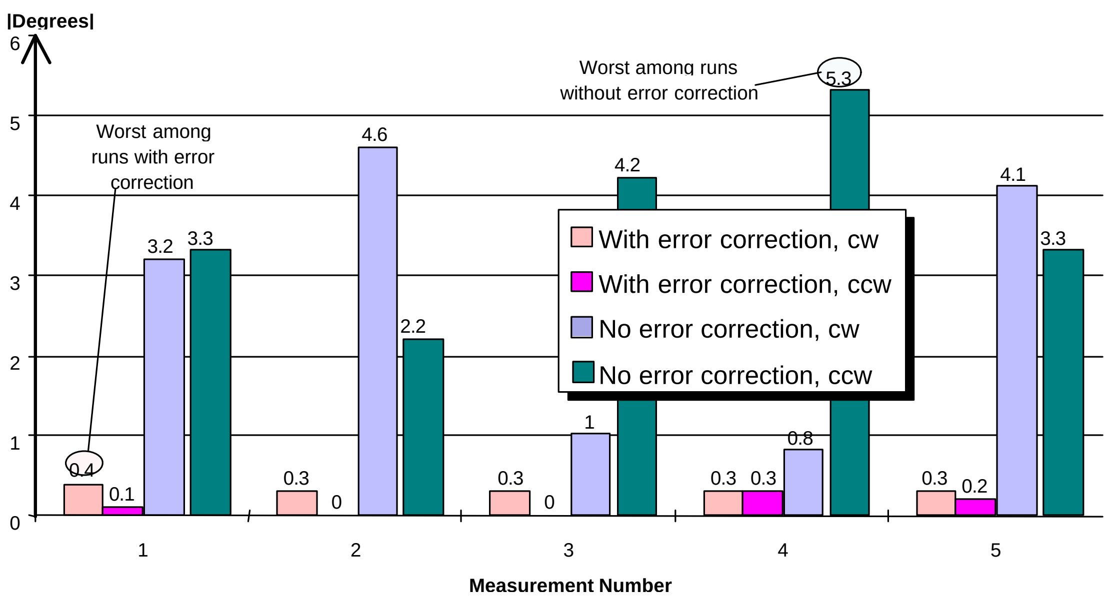

Figure 6: OmniMate return orientation errors after completing the 18.5 m rectangular path of Figure 4 on a smooth concrete floor without bumps.

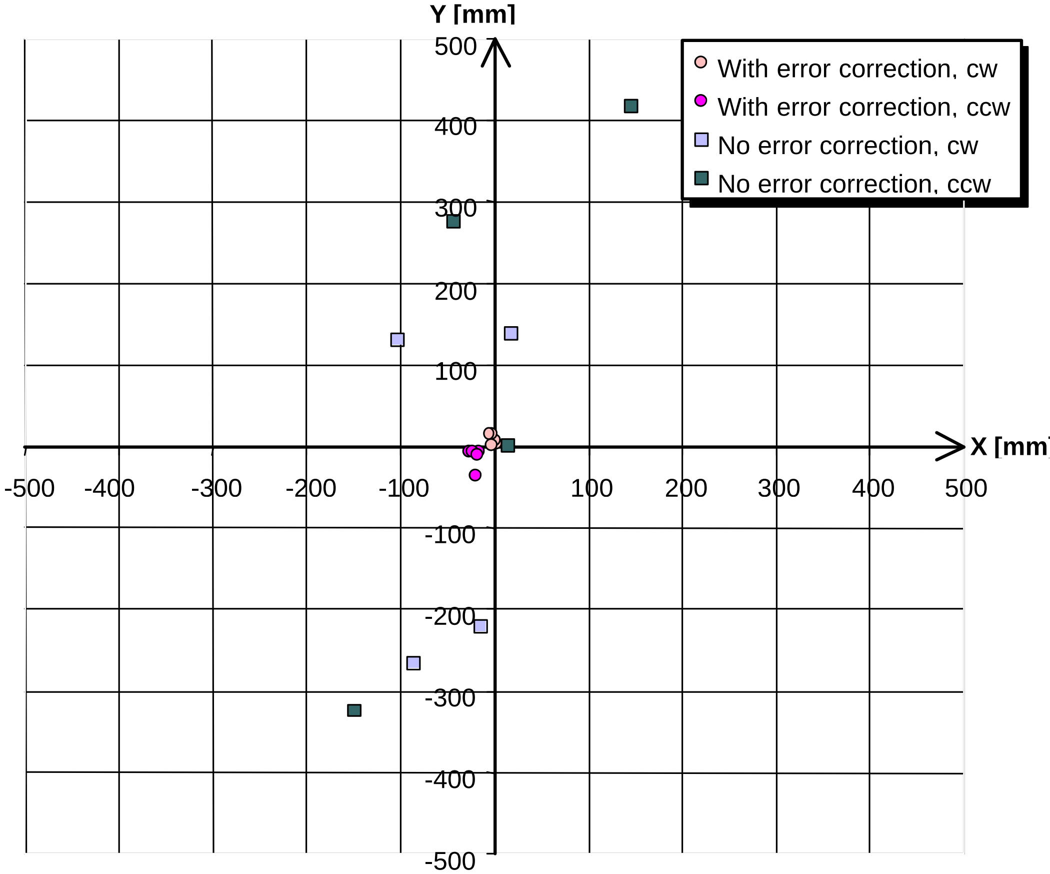

Figure 7: OmniMate return position errors after completing the 18.5 m rectangular path of Figure 4 on a smooth concrete floor ***with 20 artificial 9-mm bumps***.

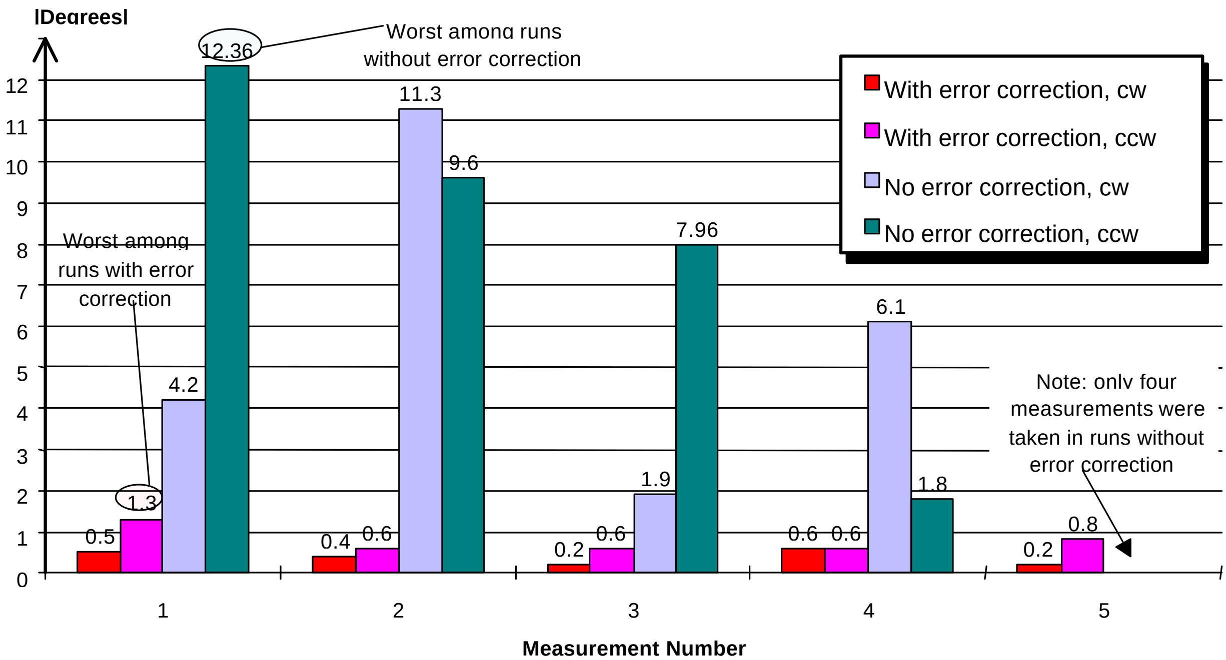

Figure 8: OmniMate return orientation errors after completing the 18.5 m rectangular path of Figure 4 on a smooth concrete floor ***with 20 artificial 9-mm bumps***.

Discussion

It is evident from the tests without bumps that IPEC corrects not only the non-systematic conventional errors listed in Section I, but also all systematic errors, with the exception of those errors that do not cause an orientation error, such as "Average of both wheel diameters differs from nominal diameter" and equal amounts of slippage on both wheels. However, much as IPEC method corrects conventional systematic errors it also introduces new systematic errors that are unique to the IPEC system but have no relevance in conventional systems (see [1] for a more thorough discussion). The primary sources for new systematic errors are (a) a constant error in measuring α, and (b) a constant error in measuring the link length *L*. In order to minimize these IPEC-specific systematic errors it is necessary to calibrate the vehicle accurately, but we have developed a procedure for doing so in less than two hours. The effectiveness of this procedure is evident in the results in Figures 5 through 8.

Two (somewhat unlikely) conditions will cause errors in the IPEC system. These are:

a. If sufficiently strong external forces are applied onto the OmniMate so that either the front or rear truck slip laterally, then this will cause an error in the measured orientation error $\Delta\theta_m$ (see Figure 3).

b. If both wheels of truck A slipped by the same amount, then this slippage would not be detected. However, the result would only be a translational error, which has usually less impact than an orientation error.

4. Conclusions

This paper presents results of odometric accuracy tests performed with a new, commercially available mobile robot called "OmniMate." The OmniMate provides true omni-directional (i.e., holonomous) motion and its kinematic design eliminates the excessive wheel-slippage often associated with omni-directional platforms. One of the OmniMate's most unique features is its ability to employ *Internal Position Error Correction* (IPEC) to dramatically improve its odometric accuracy.

Using rigorous test procedures called "UMBmark" and "extended UMBmark," the OmniMate and its implementation of IPEC were carefully tested at our lab. The results show an improvement of one order of magnitude in odometric accuracy over the accuracy of a conventional odometry system.

The foremost strength of the IPEC method is its ability to reliably and accurately detect and correct *non-systematic* odometry errors such as those caused by bumps, cracks, or other objects on the floor. In conventional mobile robots the encounter of one or more such irregularities could have a catastrophic effect on the performance of the robot, i.e., cause the mission to fail completely. With the OmniMate and IPEC, on the other hand, floor irregularities have virtually no detrimental effect on the odometric accuracy of the vehicle at all.

Acknowledgment:

This material is based upon work supported by DOE under Award No. DE-FG04-86NE37969.Special thanks to Dr. L. Feng and Mr. J. Berry for designing and installing the electronic components of the OmniMate.

5. REFERENCES

[1] J. Borenstein, "Internal Correction of Dead-reckoning Errors With the Compliant Linkage Vehicle." *Journal of Robotic Systems,* Vol. 12, No. 4, April 1995, pp. 257-273.

[2] J. Vaganay, M. J. Aldon, and A. Fournier, "Mobile Robot Attitude Estimation by Fusion of Inertial Data." *Proceedings of IEEE International Conference on Robotics and Automation*, Atlanta, GA, May 10-15, 1993, pp. 277-282.

[3] B. Barshan and H. F. Durrant-Whyte, "Inertial Navigation Systems for Mobile Robots." *IEEE Transactions on Robotics and Automation*, Vol. 11, No. 3, June 1995, pp. 328-342.

[4] K. Komoriya and E. Oyama, "Position Estimation of a mobile Robot Using Optical Fiber Gyroscope (OFG)." *International Conference on Intelligent Robots and Systems (IROS '94).* Munich, Germany, September 12-16, 1994, pp. 143-149.

[5] HITACHI — Hitachi Cable America, Inc., New York Office, 50 Main Street, White Plains, NY 10606, 914-993-0990.

[6] ANDREW — Andrew Corporation, 10500 W. 153rd Street, Orland Park, IL 60462, 708-349-5294.

[7] J. Borenstein, B. Everett, and L. Feng, *"Navigating Mobile Robots: Systems and Techniques."* A. K. Peters, Ltd., Wellesley, MA, ISBN 1-56881-058-X, February 1996.

[8] D. B., Reister, "A New Wheel Control System for the Omnidirectional HERMIES-III Robot." *Proceedings of the IEEE Conference on Robotics and Automation,* Sacramento, CA, April 7-12, 1991, pp. 2322-2327.

[9] M. West and H. Asada, "Design of a Holonomic Omnidirectional Vehicle." *Proceedings of the 1992 IEEE International Conference on Robotics and Automation.* Nice, France, May 1992, pp. 97-103.

[10] S. Hirose and S. Amano, "The VUTON: High Payload Efficiency Holonomic Omni-Directional Vehicle." *Proceedings of the Sixth International Symposium on Robotics Research (ISRR-93),* Hidden Valley, PA, Oct. 2-5 1993.

[11] F. G. Pin and M. Killough, "A New Family of Omnidirectional and Holonomic Wheeled Platforms for Mobile Robots." *IEEE Transactions on Robotics and Automation,* Vol. 10, No. 4, August 1994, pp. 480-489.

[12] H. Sugiyama, "A Method for an Autonomous Mobile Robot to Recognize its Position in the Global Coordinate System when Building a Map." *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robotics and Systems,* Yokohama, Japan, July 26-30, 1993, pp. 2186-2191.

[13] R. Kurazume and S. Nagata, "Cooperative Positioning with Multiple Robots." *Proceedings of IEEE International Conference on Robotics and Automation,* San Diego, CA, May 8-13, 1994, pp. 1250-1257.

[14] T. Hongo, H. Arakawa, G. Sugimoto, K. Tange, and Y. Yamamoto, "An Automated Guidance System of a Self-Controlled Vehicle." *IEEE Transactions on Industrial Electronics*, Vol. IE-34, No. 1, 1987, pp. 5-10.

[15] J. Borenstein and L. Feng, "UMBmark: A Benchmark Test for Measuring Dead-reckoning Errors in Mobile Robots." *1995 SPIE Conference on Mobile Robots,* Philadelphia, PA, October 22-26, 1995.