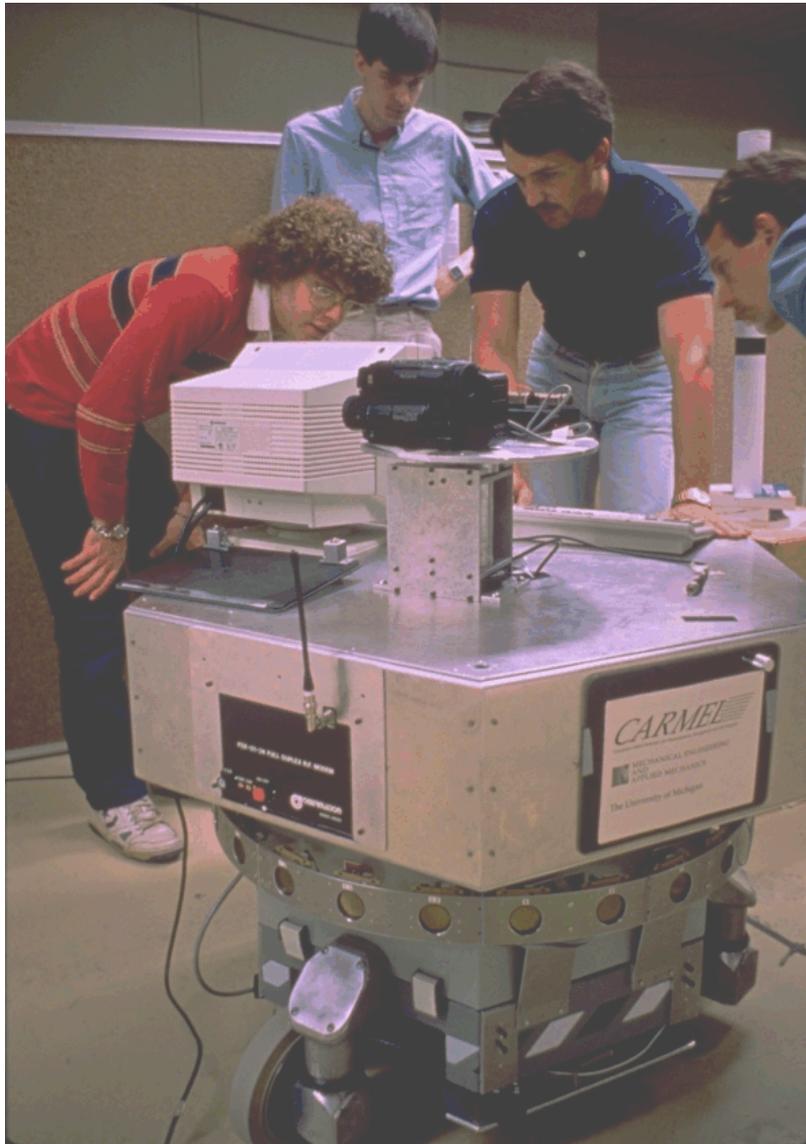


# Part II

## Systems and Methods for Mobile Robot Positioning



Tech-Team leaders Chuck Cohen, Frank Koss, Mark Huber, and David Kortenkamp (left to right) fine-tune CARMEL in preparation of the 1992 Mobile Robot Competition in San Jose, CA. The efforts paid off: despite its age, CARMEL proved to be the most agile among the contestants, winning first place honors for the University of Michigan.

# CHAPTER 5

## ODOMETRY AND OTHER DEAD-RECKONING METHODS

Odometry is the most widely used navigation method for mobile robot positioning. It is well known that odometry 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 which increase proportionally with the distance traveled by the robot. Despite these limitations, 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. Odometry is used in almost all mobile robots, for various reasons:

- Odometry data can be fused with absolute position measurements to provide better and more reliable position estimation [Cox, 1991; Hollingum, 1991; Byrne et al., 1992; Chenavier and Crowley, 1992; Evans, 1994].
- Odometry can be used in between absolute position updates with landmarks. Given a required positioning accuracy, increased accuracy in odometry allows for less frequent absolute position updates. As a result, fewer landmarks are needed for a given travel distance.
- Many mapping and landmark matching algorithms (for example: [Gonzalez et al., 1992; Chenavier and Crowley, 1992]) assume that the robot can maintain its position well enough to allow the robot to look for landmarks in a limited area and to match features in that limited area to achieve short processing time and to improve matching correctness [Cox, 1991].
- In some cases, odometry is the only navigation information available; for example: when no external reference is available, when circumstances preclude the placing or selection of landmarks in the environment, or when another sensor subsystem fails to provide usable data.

### 5.1 Systematic and Non-Systematic Odometry Errors

Odometry is based on simple equations (see Chapt. 1) that are easily implemented and that utilize data from inexpensive incremental wheel encoders. However, odometry is also based on the assumption that wheel revolutions can be translated into linear displacement relative to the floor. This assumption is only of limited validity. One extreme example is wheel slippage: if one wheel was to slip on, say, an oil spill, then the associated encoder would register wheel revolutions even though these revolutions would not correspond to a linear displacement of the wheel.

Along with the extreme case of total slippage, there are several other more subtle reasons for inaccuracies in the translation of wheel encoder readings into linear motion. All of these error sources fit into one of two categories: *systematic errors* and *non-systematic errors*.

#### Systematic Errors

- Unequal wheel diameters.
- Average of actual wheel diameters differs from nominal wheel diameter.

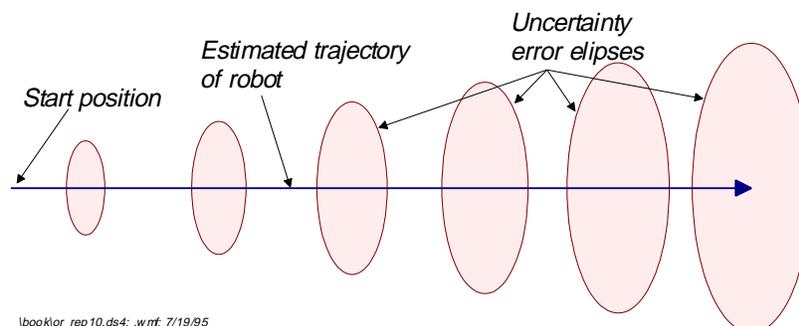
- Actual wheelbase differs from nominal wheelbase.
- Misalignment of wheels.
- Finite encoder resolution.
- Finite encoder sampling rate.

### Non-Systematic Errors

- Travel over uneven floors.
- Travel over unexpected objects on the floor.
- Wheel-slippage due to:
  - slippery floors.
  - overacceleration.
  - fast turning (skidding).
  - external forces (interaction with external bodies).
  - internal forces (castor wheels).
  - non-point wheel contact with the floor.

The clear distinction between systematic and non-systematic errors is of great importance for the effective reduction of odometry errors. For example, systematic errors are particularly grave because they accumulate constantly. On most smooth indoor surfaces systematic errors contribute much more to odometry errors than non-systematic errors. However, on rough surfaces with significant irregularities, non-systematic errors are dominant. The problem with non-systematic errors is that they may appear unexpectedly (for example, when the robot traverses an unexpected object on the ground), and they can cause large position errors. Typically, when a mobile robot system is installed with a hybrid odometry/landmark navigation system, the frequency of the landmarks is determined empirically and is based on the worst-case systematic errors. Such systems are likely to fail when one or more large non-systematic errors occur.

It is noteworthy that many researchers develop algorithms that estimate the position uncertainty of a dead-reckoning robot (e.g., [Tonouchi et al., 1994; Komoriya and Oyama, 1994].) With this approach each computed robot position is surrounded by a characteristic “error ellipse,” which indicates a region of uncertainty for the robot’s actual position (see Figure 5.1) [Tonouchi et al., 1994; Adams et al., 1994]. Typically, these ellipses grow with travel distance, until an absolute position measurement reduces the growing uncertainty and thereby “resets” the size of the error ellipse. These error estimation techniques must rely on error estimation parameters derived from observations of the vehicle’s dead-reckoning performance. Clearly, these parameters can take into account only systematic errors, because the magnitude of non-systematic errors is unpredictable.



lbooklor\_rep10.ds4: .wrf, 7/19/85

**Figure 5.1:** Growing “error ellipses” indicate the growing position uncertainty with odometry. (Adapted from [Tonouchi et al., 1994].)

## 5.2 Measurement of Odometry Errors

One important but rarely addressed difficulty in mobile robotics is the *quantitative* measurement of odometry errors. Lack of well-defined measuring procedures for the quantification of odometry errors results in the poor calibration of mobile platforms and incomparable reports on odometric accuracy in scientific communications. To overcome this problem Borenstein and Feng [1995a; 1995c] developed methods for quantitatively measuring systematic odometry errors and, to a limited degree, non-systematic odometry errors. These methods rely on a simplified error model, in which two of the systematic errors are considered to be dominant, namely:

- the error due to unequal wheel diameters, defined as

$$E_d = D_R/D_L \quad (5.1)$$

where  $D_R$  and  $D_L$  are the *actual* wheel diameters of the right and left wheel, respectively.

- The error due to uncertainty about the effective wheelbase, defined as

$$E_b = b_{\text{actual}}/b_{\text{nominal}} \quad (5.2)$$

where  $b$  is the wheelbase of the vehicle.

### 5.2.1 Measurement of Systematic Odometry Errors

To better understand the motivation for Borenstein and Feng's method (discussed in Sec. 5.2.1.2), it will be helpful to investigate a related method first. This related method, described in Section 5.2.1.1, is intuitive and widely used (e.g., [Borenstein and Koren, 1987; Cybermotion, 1988; Komoriya and Oyama, 1994; Russell, 1995]), but it is a fundamentally unsuitable benchmark test for differential-drive mobile robots.

#### 5.2.1.1 The Unidirectional Square-Path Test — A Bad Measure for Odometric Accuracy

Figure 5.2a shows a 4×4 meter unidirectional square path. The robot starts out at a position  $x_0, y_0, \theta_0$ , which is labeled **START**. The starting area should be located near the corner of two perpendicular walls. The walls serve as a fixed reference before and after the run: measuring the distance between three specific points on the robot and the walls allows accurate determination of the robot's absolute position and orientation.

To conduct the test, the robot must be 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 — 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 errors* caused by odometry and denoted  $\epsilon_x, \epsilon_y$ , and  $\epsilon_\theta$ .

$$\begin{aligned}\epsilon x &= x_{abs} - x_{calc} \\ \epsilon y &= y_{abs} - y_{calc} \\ \epsilon \theta &= \theta_{abs} - \theta_{calc}\end{aligned}\quad (5.3)$$

where

$\epsilon x, \epsilon y, \epsilon \theta$  = position and orientation errors due to odometry

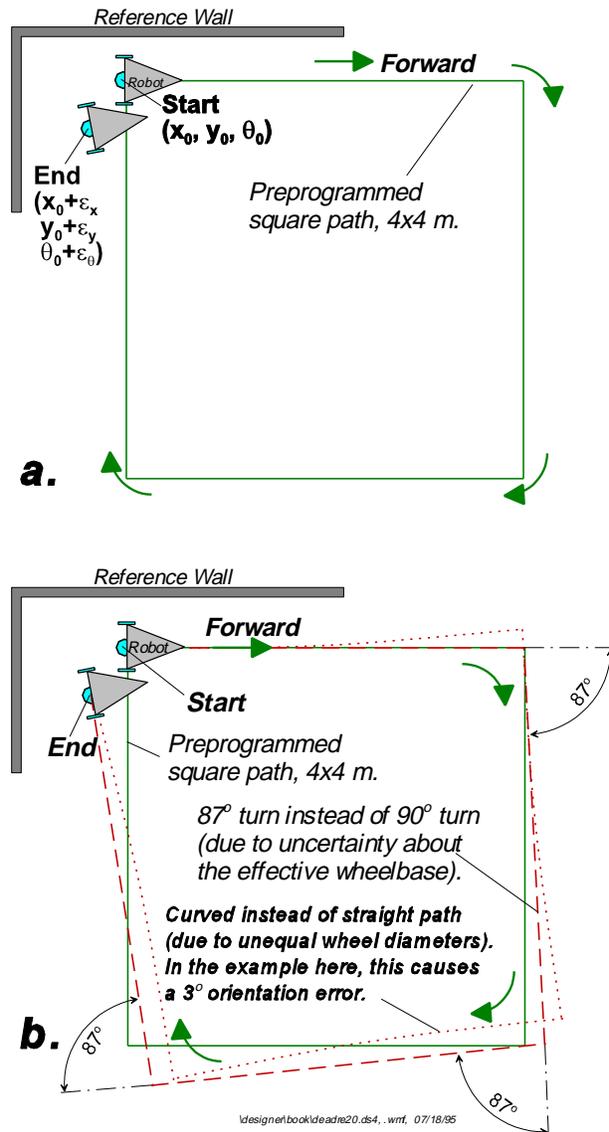
$x_{abs}, y_{abs}, \theta_{abs}$  = absolute position and orientation of the robot

$x_{calc}, y_{calc}, \theta_{calc}$  = position and orientation of the robot as computed from odometry.

The path shown in Figure 5.2a comprises of four straight-line segments and four pure rotations about the robot's centerpoint, at the corners of the square. The robot's end position shown in Figure 5.2a visualizes the odometry error.

While analyzing the results of this experiment, the experimenter may draw two different conclusions: The odometry error is the result of unequal wheel diameters,  $E_d$ , as shown by the slightly curved trajectory in Figure 5.2b (dotted line). Or, the odometry error is the result of uncertainty about the wheelbase,  $E_b$ . In the example of Figure 5.2b,  $E_b$  caused the robot to turn 87 degrees instead of the desired 90 degrees (dashed trajectory in Figure 5.2b).

As one can see in Figure 5.2b, either one of these two cases *could* yield approximately the same position error. The fact that two different error mechanisms might result in the same overall error may lead an experimenter toward a serious mistake: correcting only one of the two error sources in software. This mistake is so serious because it will yield apparently “excellent” results, as shown in the example in Figure 5.3. In this example, the experimenter began “improving” performance by adjusting the wheelbase  $b$  in the control software. According to the dead-reckoning equations for differential-drive vehicles (see Eq. (1.5) in Sec. 1.3.1), the experimenter needs only to increase the value of  $b$  to make the robot turn more in each nominal 90-degree turn. In doing so, the experimenter will soon have adjusted  $b$  to the seemingly “ideal” value that will cause the robot to turn 93 degrees, thereby effectively compensating for the 3-degree orientation error introduced by each slightly curved (but nominally straight) leg of the square path.



**Figure 5.2:**

The unidirectional square path experiment.

a. The nominal path.

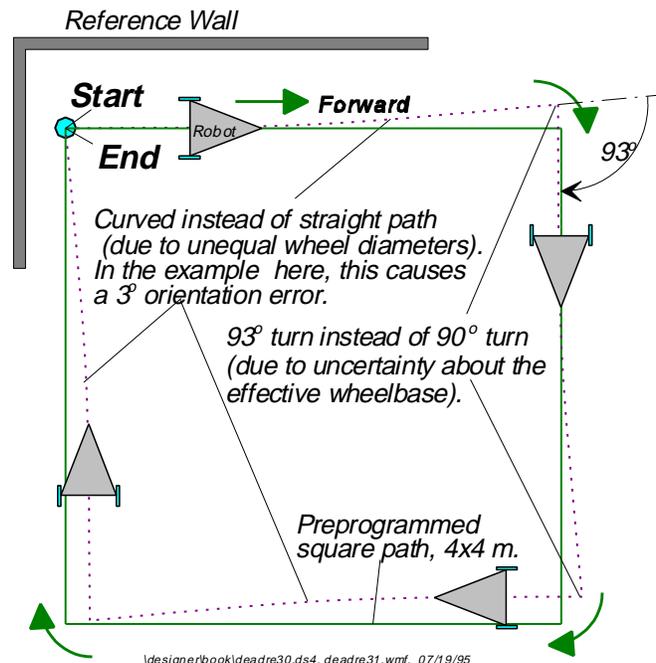
b. Either one of the two significant errors  $E_b$  or  $E_d$  can cause the same final position error.

One should note that another popular test path, the “figure-8” path [Tsumura et al., 1981; Borenstein and Koren, 1985; Cox, 1991] can be shown to have the same shortcomings as the uni-directional square path.

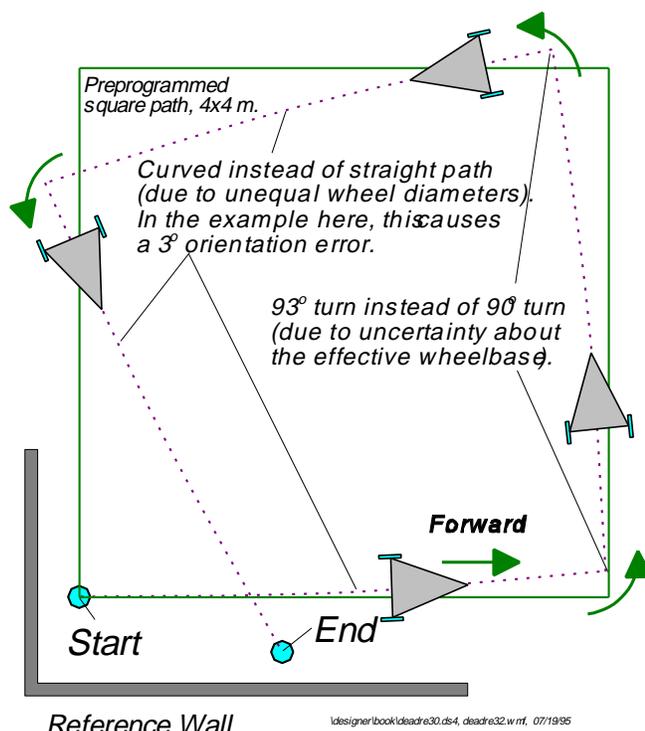
### 5.2.1.2 The Bidirectional Square-Path Experiment

The detailed example of the preceding section illustrates that the unidirectional square path experiment is unsuitable for testing odometry performance in differential-drive platforms, because it can easily conceal two mutually compensating odometry errors. To overcome this problem, Borenstein and Feng [1995a; 1995c] introduced the *bidirectional square-path* experiment, called *University of Michigan Benchmark* (UMBmark). UMBmark requires that the square path experiment be performed in both clockwise and counterclockwise direction. Figure 5.4 shows that the concealed dual error from the example in Figure 5.3 becomes clearly visible when the square path is performed in the opposite direction. This is so because the two dominant systematic errors, which may compensate for each other when run in only one direction, add up to each other and increase the overall error when run in the opposite direction.

The result of the bidirectional square-path experiment might look similar to the one shown in Figure 5.5, which presents actual experimental results with an off-the-shelf TRC *LabMate* robot [TRC] carrying an evenly distributed load. In this experiment the robot was programmed to follow a 4×4 meter square path, starting at (0,0). The stopping positions for five runs each in clockwise (cw) and counterclockwise (ccw) directions are shown in Figure 5.5. Note that Figure 5.5 is an enlarged view of the target area. The results of Figure 5.5 can be interpreted as follows:



**Figure 5.3:** The effect of the two dominant systematic dead-reckoning errors  $E_b$  and  $E_d$ . Note how both errors may cancel each other out when the test is performed in only one direction.



**Figure 5.4:** The effect of the two dominant systematic odometry errors  $E_b$  and  $E_d$ : when the square path is performed in the opposite direction one may find that the errors add up.

- The stopping positions after cw and ccw runs are clustered in two distinct areas.
- The distribution within the cw and ccw clusters are the result of non-systematic errors, such as those mentioned in Section 5.1. However, Figure 5.5 shows that in an uncalibrated vehicle, traveling over a reasonably smooth concrete floor, the contribution of *systematic* errors to the total odometry error can be notably larger than the contribution of non-systematic errors.

After conducting the UMBmark experiment, one may wish to derive a single numeric value that expresses the odometric accuracy (with respect to systematic errors) of the tested vehicle. In order to minimize the effect of non-systematic errors, it has been suggested [Komoriya and Oyama, 1994; Borenstein and Feng, 1995c] to consider the center of gravity of each cluster as representative for the systematic odometry errors in the cw and ccw directions.

The coordinates of the two centers of gravity are computed from the results of Equation (5.3) as

$$x_{c.g.,cw/ccw} = \frac{1}{n} \sum_{i=1}^n \epsilon x_{i,cw/ccw} \quad (5.4)$$

$$y_{c.g.,cw/ccw} = \frac{1}{n} \sum_{i=1}^n \epsilon y_{i,cw/ccw}$$

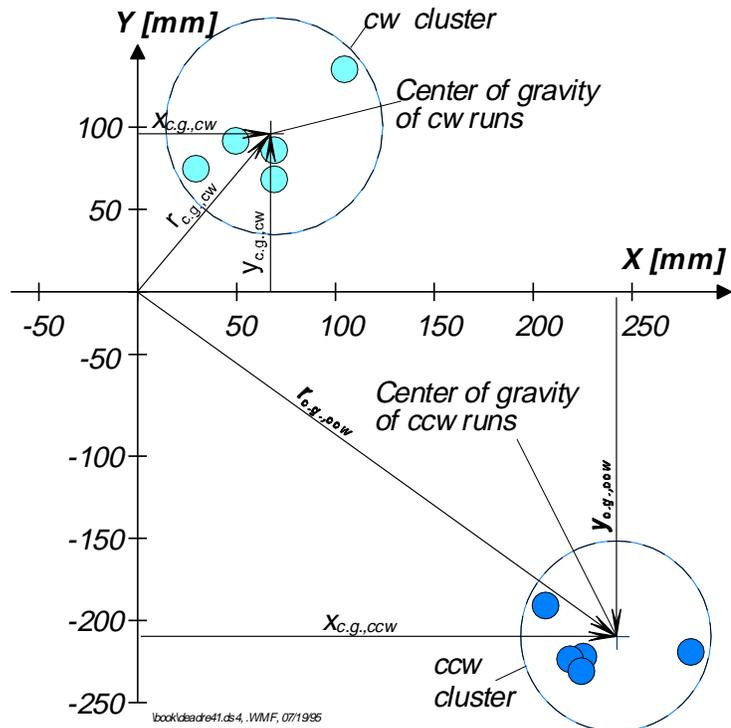
where  $n = 5$  is the number of runs in each direction.

The absolute offsets of the two centers of gravity from the origin are denoted  $r_{c.g.,cw}$  and  $r_{c.g.,ccw}$  (see Fig. 5.5) and are given by

$$r_{c.g.,cw} = \sqrt{(x_{c.g.,cw})^2 + (y_{c.g.,cw})^2} \quad (5.5a)$$

and

$$r_{c.g.,ccw} = \sqrt{(x_{c.g.,ccw})^2 + (y_{c.g.,ccw})^2} \quad (5.5b)$$



**Figure 5.5:** Typical results from running UMBmark (a square path run in both cw and ccw directions) with an uncalibrated vehicle.

Finally, the larger value among  $r_{c.g., cw}$  and  $r_{c.g., ccw}$  is defined as the "*measure of odometric accuracy for systematic errors*":

$$E_{\max, \text{syst}} = \max(r_{c.g., cw} ; r_{c.g., ccw}) \quad (5.6)$$

The reason for not using the *average* of the two centers of gravity  $r_{c.g., cw}$  and  $r_{c.g., ccw}$  is that for practical applications one needs to worry about the *largest* possible odometry error. One should also note that the final orientation error  $\epsilon\theta$  is not considered explicitly in the expression for  $E_{\max, \text{syst}}$ . This

is because all systematic orientation errors are implied by the final position errors. In other words, since the square path has fixed-length sides, systematic orientation errors translate directly into position errors.

### 5.2.2 Measurement of Non-Systematic Errors

Some limited information about a vehicle's susceptibility to non-systematic errors can be derived from the spread of the return position errors that was shown in Figure 5.5. When running the UMBmark procedure on smooth floors (e.g., a concrete floor without noticeable bumps or cracks), an indication of the magnitude of the non-systematic errors can be obtained from computing the estimated standard deviation  $\sigma$ . However, Borenstein and Feng [1994] caution that there is only limited value to knowing  $\sigma$ , since  $\sigma$  reflects only on the interaction between the vehicle and a certain floor. Furthermore, it can be shown that from comparing  $\sigma$  from two different robots (even if they traveled on the same floor), one cannot necessarily conclude that the robots with the larger  $\sigma$  showed higher susceptibility to non-systematic errors.

In real applications it is imperative that the *largest possible disturbance* be determined and used in testing. For example, the estimated standard deviation of the test in Figure 5.5 gives no indication at all as to what error one should expect if one wheel of the robot inadvertently traversed a large bump or crack in the floor. For the above reasons it is difficult (perhaps impossible) to design a generally applicable quantitative test procedure for non-systematic errors. However, Borenstein [1994] proposed an easily reproducible test that would allow comparing the susceptibility to non-systematic errors of different vehicles. This test, called the *extended UMBmark*, uses the same bidirectional square path as UMBmark but, in addition, introduces artificial bumps. Artificial bumps are introduced by means of a common, round, electrical household-type cable (such as the ones used with 15 A six-outlet power strips). Such a cable has a diameter of about 9 to 10 millimeters. Its rounded shape and plastic coating allow even smaller robots to traverse it without too much physical impact. In the proposed extended UMBmark test the cable is placed 10 times under one of the robot's wheels, during motion. In order to provide better repeatability for this test and to avoid mutually compensating errors, Borenstein and Feng [1994] suggest that these 10 bumps be introduced as evenly as possible. The bumps should also be introduced during the first straight segment of the square path, and always under the wheel that faces the inside of the square. It can be shown [Borenstein, 1994b] that the most noticeable effect of each bump is a fixed orientation error in the direction of the wheel that encountered the bump. In the TRC *LabMate*, for example, the orientation error resulting from a bump of height  $h = 10$  mm is roughly  $\Delta\theta = 0.44^\circ$  [Borenstein, 1994b].

Borenstein and Feng [1994] proceed to discuss which measurable parameter would be the most useful for expressing the vehicle's susceptibility to non-systematic errors. Consider, for example, Path A and Path B in Figure 5.6. If the 10 bumps required by the *extended UMBmark* test were concentrated at the beginning of the first straight leg (as shown in exaggeration in Path A), then the return position error would be very small. Conversely, if the 10 bumps were concentrated toward the end of the first straight leg (Path B in Figure 5.6), then the return position error would be larger. Because of this sensitivity of the return position errors to the exact location of the bumps it is not a good idea to use the return position error as an indicator for a robot's susceptibility to non-systematic errors. Instead, the return orientation error  $\epsilon\theta$  should be used. Although it is more difficult to measure small angles, measurement of  $\epsilon\theta$  is a more consistent quantitative indicator for

comparing the performance of different robots. Thus, one can measure and express the susceptibility of a vehicle to non-systematic errors in terms of its *average absolute orientation error* defined as

$$\epsilon\theta_{avg}^{nonsys} = \frac{1}{n} \sum_{i=1}^n |\epsilon\theta_{i,cw}^{nonsys} - \epsilon\theta_{avg,cw}^{sys}| + \frac{1}{n} \sum_{i=1}^n |\epsilon\theta_{i,ccw}^{nonsys} - \epsilon\theta_{avg,ccw}^{sys}| \quad (5.7)$$

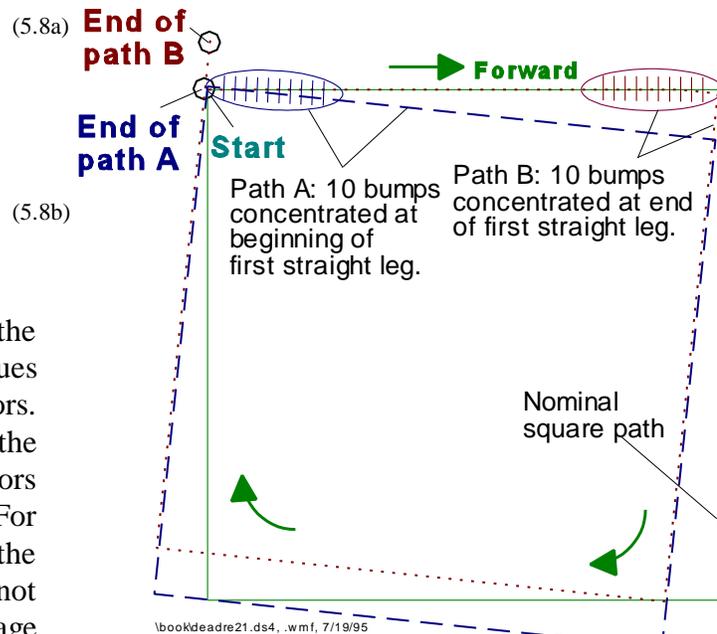
where  $n = 5$  is the number of experiments in cw or ccw direction, superscripts “sys” and “nonsys” indicate a result obtained from either the regular UMBmark test (for systematic errors) or from the extended UMBmark test (for non-systematic errors). Note that Equation (5.7) improves on the accuracy in identifying non-systematic errors by removing the systematic bias of the vehicle, given by

$$\epsilon\theta_{avg,cw}^{sys} = \frac{1}{n} \sum_{i=1}^n \epsilon\theta_{i,cw}^{sys} \quad (5.8a)$$

and

$$\epsilon\theta_{avg,ccw}^{sys} = \frac{1}{n} \sum_{i=1}^n \epsilon\theta_{i,ccw}^{sys} \quad (5.8b)$$

Also note that the arguments inside the Sigmas in Equation (5.7) are absolute values of the bias-free return orientation errors. This is because one would want to avoid the case in which two return orientation errors of opposite sign cancel each other out. For example, if in one run  $\epsilon\theta = 1^\circ$  and in the next run  $\epsilon\theta = -1^\circ$ , then one should not conclude that  $\epsilon\theta_{avg}^{nonsys} = 0$ . Using the average absolute return error as computed in Equation (5.7) would correctly compute  $\epsilon\theta_{avg}^{nonsys} = .1$ . By contrast, in Equation (5.8) the actual arithmetic average is computed to identify a fixed bias.



**Figure 5.6:** The return *position* of the extended UMBmark test is sensitive to the exact location where the 10 bumps were placed. The return *orientation* is not.

### 5.3 Reduction of Odometry Errors

The accuracy of odometry in commercial mobile platforms depends to some degree on their kinematic design and on certain critical dimensions. Here are some of the design-specific considerations that affect dead-reckoning accuracy:

Vehicles with a small wheelbase are more prone to orientation errors than vehicles with a larger wheelbase. For example, the differential drive *LabMate* robot from TRC has a relatively small wheelbase of 340 millimeters (13.4 in). As a result, Gourley and Trivedi [1994], suggest that

odometry with the *LabMate* be limited to about 10 meters (33 ft), before a new “reset” becomes necessary.

- Vehicles with castor wheels that bear a significant portion of the overall weight are likely to induce slippage when reversing direction (the “shopping cart effect”). Conversely, if the castor wheels bear only a small portion of the overall weight, then slippage will not occur when reversing direction [Borenstein and Koren, 1985].
- It is widely known that, ideally, wheels used for odometry should be “knife-edge” thin and not compressible. The ideal wheel would be made of aluminum with a thin layer of rubber for better traction. In practice, this design is not feasible for all but the most lightweight vehicles, because the odometry wheels are usually also load-bearing drive wheels, which require a somewhat larger ground contact surface.
- Typically the synchro-drive design (see Sec. 1.3.4) provides better odometric accuracy than differential-drive vehicles. This is especially true when traveling over floor irregularities: arbitrary irregularities will affect only one wheel at a time. Thus, since the two other drive wheels stay in contact with the ground, they provide more traction and force the affected wheel to slip. Therefore, overall distance traveled will be reflected properly by the amount of travel indicated by odometry.

Other attempts at improving odometric accuracy are based on more detailed modeling. For example, Larsson et al. [1994] used circular segments to replace the linear segments in each sampling period. The benefits of this approach are relatively small. Boyden and Velinsky [1994] compared (in simulations) conventional odometric techniques, based on kinematics only, to solutions based on the dynamics of the vehicle. They presented simulation results to show that for both differentially and conventionally steered wheeled mobile robots, the kinematic model was accurate only at slower speeds up to 0.3 m/s when performing a tight turn. This result agrees with experimental observations, which suggest that errors due to wheel slippage can be reduced to some degree by limiting the vehicle's speed during turning, and by limiting accelerations.

### 5.3.1 Reduction of Systematic Odometry Errors

In this section we present specific methods for reducing systematic odometry errors. When applied individually or in combination, these measures can improve odometric accuracy by orders of magnitude.

#### 5.3.1.1 Auxiliary Wheels and Basic Encoder Trailer

It is generally possible to improve odometric accuracy by adding a pair of “knife-edge,” non-load-bearing *encoder wheels*, as shown conceptually in Figure 5.7. Since these wheels are not used for transmitting power, they can be made to be very thin and with only a thin layer of rubber as a tire. Such a design is feasible for differential-drive, tricycle-drive, and Ackerman vehicles.

Hongo et al. [1987] had built such a set of encoder wheels, to improve the accuracy of a large differential-drive mobile robot weighing 350 kilograms (770 lb). Hongo et al. report that, after careful calibration, their vehicle had a position error of less than 200 millimeters (8 in) for a travel distance of 50 meters (164 ft). The ground surface on which this experiment was carried out was a “well-paved” road.

### 5.3.1.2 The Basic Encoder Trailer

An alternative approach is the use of a trailer with two encoder wheels [Fan et al., 1994; 1995]. Such an *encoder trailer* was recently built and tested at the University of Michigan (see Figure 5.8). This encoder trailer was designed to be attached to a Remotec *Andros V* tracked vehicle [REMOTEC]. As was explained in Section 1.3, it is virtually impossible to use odometry with tracked vehicles, because of the large amount of slippage between the tracks and the floor during turning. The idea of the encoder trailer is to perform odometry whenever the ground characteristics allow one to do so. Then, when the *Andros* has to move over small obstacles, stairs, or otherwise uneven ground, the encoder trailer would be raised. The argument for this part-time deployment of the encoder trailer is that in many applications the robot may travel *mostly* on reasonably smooth concrete floors and that it would thus benefit *most of the time* from the encoder trailer's odometry.

### 5.3.1.3 Systematic Calibration

Another approach to improving odometric accuracy without any additional devices or sensors is based on the careful calibration of a mobile robot. As was explained in Section 5.1, systematic errors are inherent properties of each individual robot. They change very slowly as the result of wear or of different load distributions. Thus, these errors remain almost constant over extended periods of time [Tsumura et al., 1981]. One way to reduce such errors is vehicle-specific calibration. However, calibration is difficult because even minute deviations in the geometry of the vehicle or its parts (e.g., a change in wheel diameter due to a different load distribution) may cause substantial odometry errors.

Borenstein and Feng [1995a; 1995b] have developed a systematic procedure for the measurement and *correction* of odometry errors. This method requires that the UMBmark procedure, described in Section 5.2.1, be run with at least five runs each in cw and ccw direction. Borenstein and Feng define two new error characteristics that are meaningful only in the context of the UMBmark test. These characteristics, called Type A and Type B, represent odometry errors in orientation. A Type A is defined as an orientation error that *reduces (or increases)* the total amount of rotation of the robot during the square-path experiment in *both cw and ccw direction*. By contrast, Type B is defined as an orientation error that *reduces (or increases)* the total amount of rotation of the robot during the square-path experiment in *one direction*, but

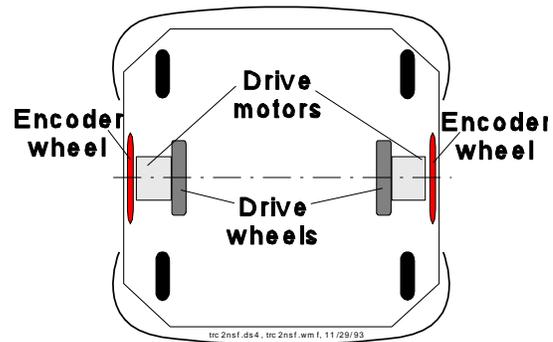


Figure 5.7: Conceptual drawing of a set of encoder wheels for a differential drive vehicle.

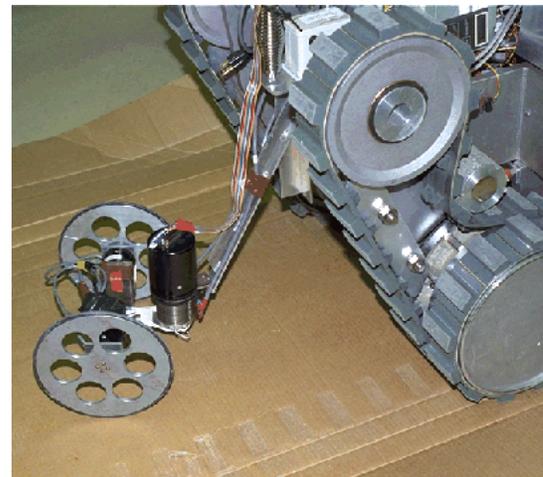
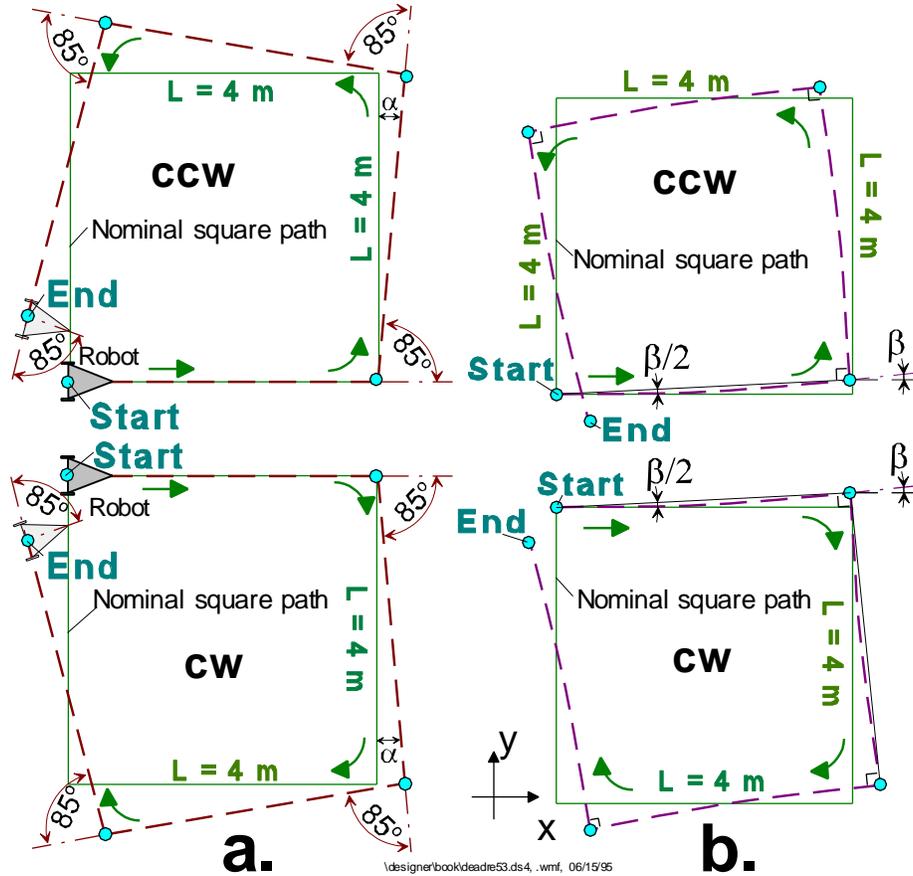


Figure 5.8: A simple encoder trailer. The trailer here was designed and built at the University of Michigan for use with the Remotec's *Andros V* tracked vehicle. (Courtesy of The University of Michigan.)

*increases (or reduces)* the amount of rotation when going in the *other direction*. Examples for Type A and Type B errors are shown in Figure 5.9.



**Figure 5.9:** Type A and Type B errors in the ccw and cw directions. a. Type A errors are caused only by the wheelbase error  $E_b$ . b. Type B errors are caused only by unequal wheel diameters ( $E_d$ ).

Figure 5.9a shows a case where the robot turned four times for a nominal amount of 90 degrees per turn. However, because the actual wheelbase of the vehicle was larger than the nominal value, the vehicle actually turned only 85 degrees in each corner of the square path. In the example of Figure 5.9 the robot actually turned only  $\theta_{\text{total}} = 4 \times 85^\circ = 340^\circ$ , instead of the desired  $\theta_{\text{nominal}} = 360^\circ$ . One can thus observe that in *both the cw and the ccw* experiment the robot ends up turning *less* than the desired amount, i.e.,

$$|\theta_{\text{total, cw}}| < |\theta_{\text{nominal}}| \text{ and } |\theta_{\text{total, ccw}}| < |\theta_{\text{nominal}}| .$$

Hence, the orientation error is of Type A.

In Figure 5.9b the trajectory of a robot with unequal wheel diameters is shown. This error expresses itself in a curved path that adds to the overall orientation at the end of the run in cw direction, but it reduces the overall rotation in the ccw direction, i.e.,

$$|\theta_{\text{total, ccw}}| > |\theta_{\text{nominal}}| \text{ but } |\theta_{\text{total, cw}}| < |\theta_{\text{nominal}}| .$$

Thus, the orientation error in Figure 5.9b is of Type B.

In an actual run Type A and Type B errors will of course occur together. The problem is therefore how to distinguish between Type A and Type B errors and how to compute correction factors for these errors from the measured final position errors of the robot in the UMBmark test. This question will be addressed next.

Figure 5.9a shows the contribution of Type A errors. We recall that Type A errors are caused mostly by  $E_b$ . We also recall that Type A errors cause too much or too little turning at the corners of the square path. The (unknown) amount of erroneous rotation in each nominal 90-degree turn is denoted as  $\alpha$  and measured in [rad].

Figure 5.9b shows the contribution of Type B errors. We recall that Type B errors are caused mostly by the ratio between wheel diameters  $E_d$ . We also recall that Type B errors cause a slightly curved path instead of a straight one during the four straight legs of the square path. Because of the curved motion, the robot will have gained an incremental orientation error, denoted  $\beta$ , at the end of each straight leg.

We omit here the derivation of expressions for  $\alpha$  and  $\beta$ , which can be found from simple geometric relations in Figure 5.9 (see [Borenstein and Feng, 1995a] for a detailed derivation). Here we just present the results:

$$\alpha = \frac{X_{c.g.,cw} + X_{c.g.,ccw}}{-4L} \frac{180^\circ}{\pi} \quad (5.9)$$

solves for  $\alpha$  in [ $^\circ$ ] and

$$\beta = \frac{X_{c.g.,cw} - X_{c.g.,ccw}}{-4L} \frac{180^\circ}{\pi} \quad (5.10)$$

solves for  $\beta$  in [ $^\circ$ ].

Using simple geometric relations, the radius of curvature  $R$  of the curved path of Figure 5.9b can be found as

$$R = \frac{L/2}{\sin\beta/2} . \quad (5.11)$$

Once the radius  $R$  is computed, it is easy to determine the ratio between the two wheel diameters that caused the robot to travel on a curved, instead of a straight path

$$E_d = \frac{D_R}{D_L} = \frac{R + b/2}{R - b/2} . \quad (5.12)$$

Similarly one can compute the wheelbase error  $E_b$ . Since the wheelbase  $b$  is directly proportional to the actual amount of rotation, one can use the proportion:

$$\frac{b_{actual}}{90^\circ} = \frac{b_{nominal}}{90^\circ - \alpha} \quad (5.13)$$

so that

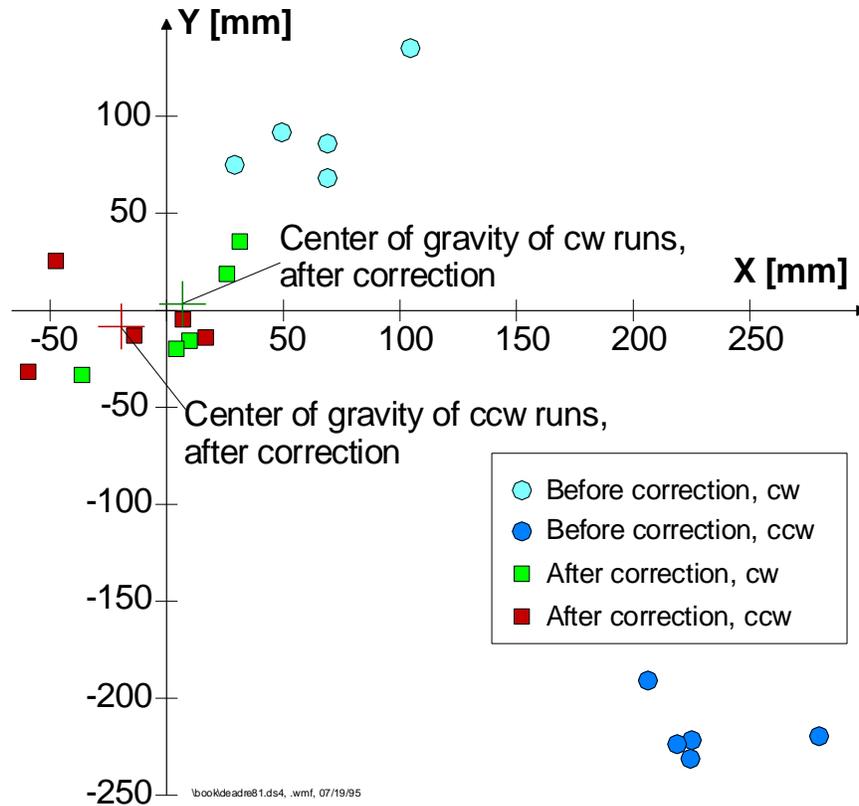
$$b_{actual} = \frac{90^\circ}{90^\circ - \alpha} b_{nominal} \quad (5.14)$$

where, per definition of Equation (5.2)

$$E_b = \frac{90^\circ}{90^\circ - \alpha} . \quad (5.15)$$

Once  $E_b$  and  $E_d$  are computed, it is straightforward to use their values as compensation factors in the controller software [see Borenstein and Feng, 1995a; 1995b]. The result is a 10- to 20-fold reduction in systematic errors.

Figure 5.10 shows the result of a typical calibration session.  $D_R$  and  $D_L$  are the effective wheel diameters, and  $b$  is the effective wheelbase.



**Figure 5.10:** Position errors after completion of the bidirectional square-path experiment (4 x 4 m).

Before calibration:  $b = 340.00$  mm,  $D_R/D_L = 1.00000$ .

After calibration:  $b = 336.17$ ,  $D_R/D_L = 1.00084$ .

This calibration procedure can be performed with nothing more than an ordinary tape measure. It takes about two hours to run the complete calibration procedure and measure the individual return errors with a tape measure.

### 5.3.2 Reducing Non-Systematic Odometry Errors

This section introduces methods for the reduction of non-systematic odometry errors. The methods discussed in Section 5.3.2.2 may at first confuse the reader because they were implemented on the somewhat complex experimental platform described in Section 1.3.7. However, the methods of Section 5.3.2.2 can be applied to many other kinematic configurations, and efforts in that direction are subject of currently ongoing research at the University of Michigan.

#### 5.3.2.1 Mutual Referencing

Sugiyama [1993] 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. However, this stop and go approach limits the efficiency of the robots.

#### 5.3.2.2 Internal Position Error Correction

A unique way for reducing odometry errors even further is *Internal Position Error Correction* (IPEC). With this approach two mobile robots mutually correct their odometry errors. However, unlike the approach described in Section 5.3.2.1, the IPEC method works while both robots are in continuous, fast motion [Borenstein, 1994a]. To implement this method, it is required that both robots can measure their relative distance and bearing continuously and accurately. Coincidentally, the MDOF vehicle with compliant linkage (described in Sec. 1.3.7) offers exactly these features, and the IPEC method was therefore implemented and demonstrated on that MDOF vehicle. This implementation is named *Compliant Linkage Autonomous Platform with Position Error Recovery* (CLAPPER).

The CLAPPER's compliant linkage instrumentation was illustrated in Chapter 1, Figure 1.15. This setup provides real-time feedback on the relative position and orientation of the two trucks. An absolute encoder at each end measures the rotation of each truck (with respect to the linkage) with a resolution of 0.3 degrees, while a linear encoder is used to measure the separation distance to within 5 millimeters (0.2 in). Each truck computes its own dead-reckoned position and heading in conventional fashion, based on displacement and velocity information derived from its left and right drive-wheel encoders. By examining the perceived odometry solutions of the two robot platforms in conjunction with their known relative orientations, the *CLAPPER* system can detect and significantly reduce heading errors for both trucks (see video clip in [Borenstein, 1995V].)

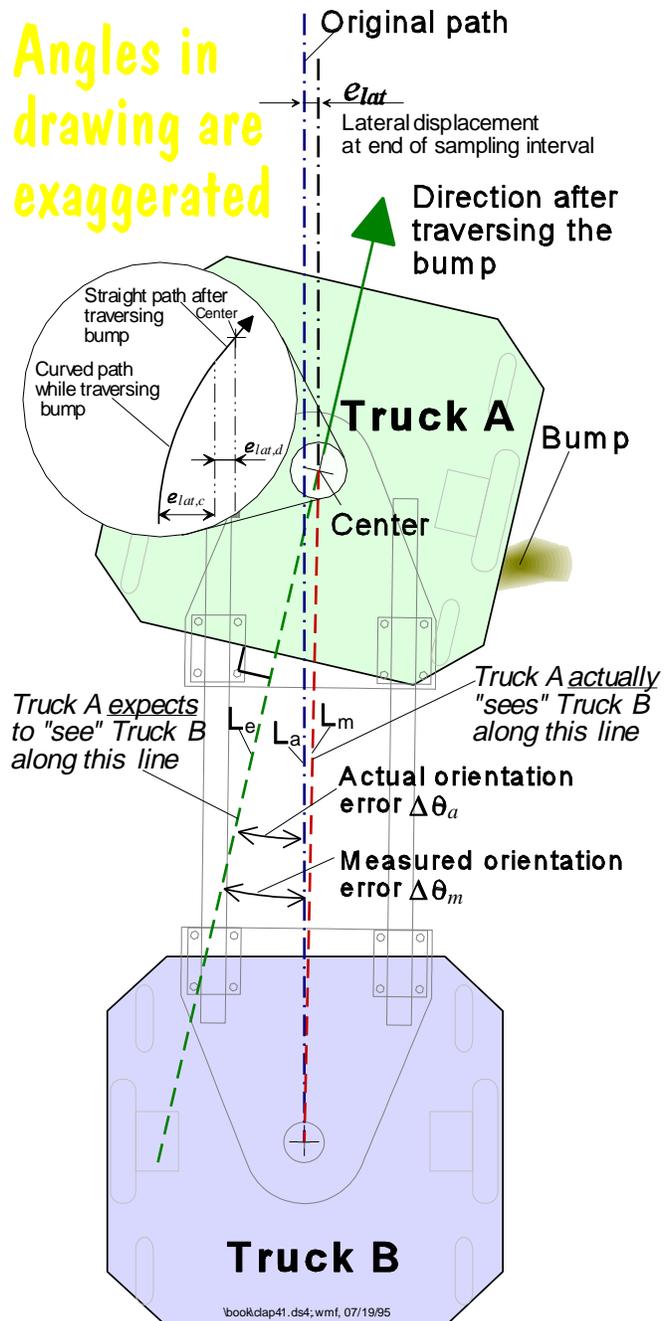
The principle of operation is based on the concept of *error growth rate* presented by Borenstein [1994a, 1995a], who makes a distinction between “fast-growing” and “slow-growing” odometry errors. For example, when a differentially steered robot traverses a floor irregularity it will immediately experience an appreciable orientation error (i.e., a fast-growing error). The associated 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 performs relative position measurements with a sufficiently fast update rate (20 ms) to

allow each truck to detect *fast-growing* errors in orientation, while relying on the fact that the lateral position errors accrued by both platforms during the sampling interval were small.

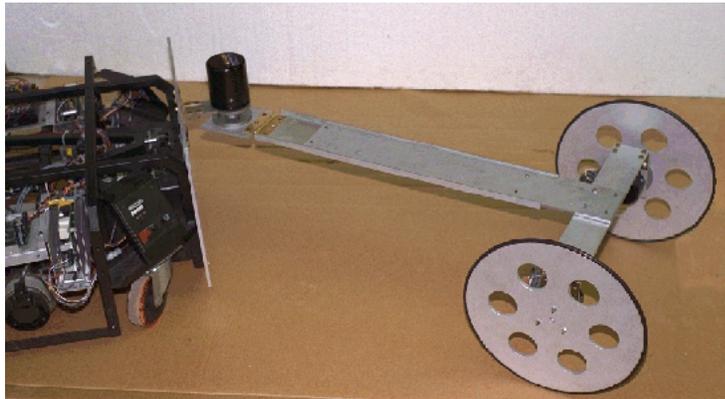
Figure 5.11 explains how this method works. After traversing a bump Truck A's orientation will change (a fact unknown to Truck A's odometry computation). Truck A is therefore expecting to "see" Truck B along the extension of line  $L_e$ . However, because of the physically incurred rotation of Truck 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, which can be corrected immediately. 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.

The compliant linkage in essence forms a pseudo-stable heading reference in world coordinates, its own orientation being dictated solely by the relative translations of its end points, which in turn are affected only by the lateral displacements of the two trucks. Since the lateral displacements are *slow growing*, the linkage rotates only a very small amount between encoder samples. The *fast-growing* azimuthal disturbances of the trucks, on the other hand, are not coupled through the rotational joints to the linkage, thus allowing the rotary encoders to detect and quantify the instantaneous orientation errors of the trucks, even when both are in motion. Borenstein [1994a; 1995a] provides a more complete description of this innovative concept and reports experimental results indicating improved odometry performance of up to two orders of magnitude over conventional mobile robots.

It should be noted that the rather complex kinematic design of the MDOF vehicle is not necessary to implement the IPEC error correction method. Rather, the MDOF vehicle happened to be available at the time and allowed the University of Michigan researchers to implement and verify the validity of the IPEC approach. Currently, efforts are under way to implement the IPEC method on a tractor-trailer assembly, called "*Smart Encoder Trailer*" (SET), which is shown in Figure 5.12. The principle of operation is



**Figure 5.11:** After traversing a bump, the resulting change of orientation of Truck A can be measured relative to Truck B.



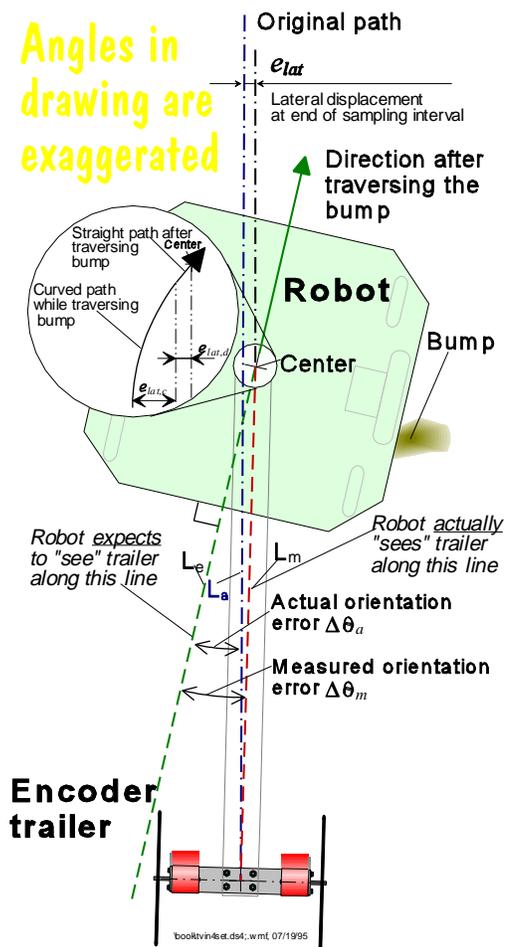
**Figure 5.12:** The University of Michigan's "Smart Encoder Trailer" (SET) is currently being instrumented to allow the implementation of the IPEC error correction method explained in Section 5.3.2.2. (Courtesy of The University of Michigan.)

illustrated in Figure 5.13. Simulation results, indicating the feasibility of implementing the IPEC method on a tractor-trailer assembly, were presented in [Borenstein, 1994b].

#### 5.4 Inertial Navigation

An alternative method for enhancing dead reckoning is inertial navigation, initially developed for deployment on aircraft. The technology was quickly adapted for use on missiles and in outer space, and found its way to maritime usage when the nuclear submarines *Nautilus* and *Skate* were suitably equipped in support of their transpolar voyages in 1958 [Dunlap and Shufeldt, 1972]. The principle of operation involves continuous sensing of minute accelerations in each of the three directional axes and integrating over time to derive velocity and position. A gyroscopically stabilized sensor platform is used to maintain consistent orientation of the three accelerometers throughout this process.

Although fairly simple in concept, the specifics of implementation are rather demanding. This is mainly caused by error sources that adversely affect the stability of the gyros used to ensure correct attitude. The resulting high manufacturing and maintenance costs have effectively precluded any practical application of this technology in the automated guided vehicle industry [Turpin, 1986]. For example, a high-quality *inertial navigation system* (INS) such as would be found in a commercial airliner will have a typical drift of about 1850 meters (1 nautical mile) per hour of operation, and cost between \$50K and \$70K [Byrne et al., 1992]. High-end INS packages used in ground applications have shown performance of better than 0.1 percent of distance traveled, but cost in the neighborhood of \$100K to \$200K, while lower performance versions (i.e., one percent of distance traveled) run between \$20K to \$50K [Dahlin and Krantz, 1988].



**Figure 5.13:** Proposed implementation of the IPEC method on a tractor-trailer assembly.

Experimental results from the Université Montpellier in France [Vaganay et al., 1993a; 1993b], from the University of Oxford in the U.K. [Barshan and Durrant-Whyte, 1993; 1995], and from the University of Michigan indicate that a purely inertial navigation approach is not realistically advantageous (i.e., too expensive) for mobile robot applications. As a consequence, the use of INS hardware in robotics applications to date has been generally limited to scenarios that aren't readily addressable by more practical alternatives. An example of such a situation is presented by Sammarco [1990; 1994], who reports preliminary results in the case of an INS used to control an autonomous vehicle in a mining application.

Inertial navigation is attractive mainly because it is self-contained and no external motion information is needed for positioning. One important advantage of inertial navigation is its ability to provide fast, low-latency dynamic measurements. Furthermore, inertial navigation sensors typically have noise and error sources that are independent from the external sensors [Parish and Grabbe, 1993]. For example, the noise and error from an inertial navigation system should be quite different from that of, say, a landmark-based system. Inertial navigation sensors are self-contained, non-radiating, and non-jammable. Fundamentally, gyros provide angular rate and accelerometers provide velocity rate information. Dynamic information is provided through direct measurements. However, the main disadvantage is that the angular rate data and the linear velocity rate data must be integrated once and twice (respectively), to provide orientation and linear position, respectively. Thus, even very small errors in the rate information can cause an unbounded growth in the error of integrated measurements. As we remarked in Section 2.2, the price of very accurate laser gyros and optical fiber gyros have come down significantly. With price tags of \$1,000 to \$5,000, these devices have now become more suitable for many mobile robot applications.

#### 5.4.1 Accelerometers

The suitability of accelerometers for mobile robot positioning was evaluated at the University of Michigan. In this informal study it was found that there is a very poor signal-to-noise ratio at lower accelerations (i.e., during low-speed turns). Accelerometers also suffer from extensive drift, and they are sensitive to uneven grounds, because any disturbance from a perfectly horizontal position will cause the sensor to detect the gravitational acceleration  $g$ . One low-cost inertial navigation system aimed at overcoming the latter problem included a tilt sensor [Barshan and Durrant-Whyte, 1993; 1995]. The tilt information provided by the tilt sensor was supplied to the accelerometer to cancel the gravity component projecting on each axis of the accelerometer. Nonetheless, the results obtained from the tilt-compensated system indicate a position drift rate of 1 to 8 cm/s (0.4 to 3.1 in/s), depending on the frequency of acceleration changes. This is an unacceptable error rate for most mobile robot applications.

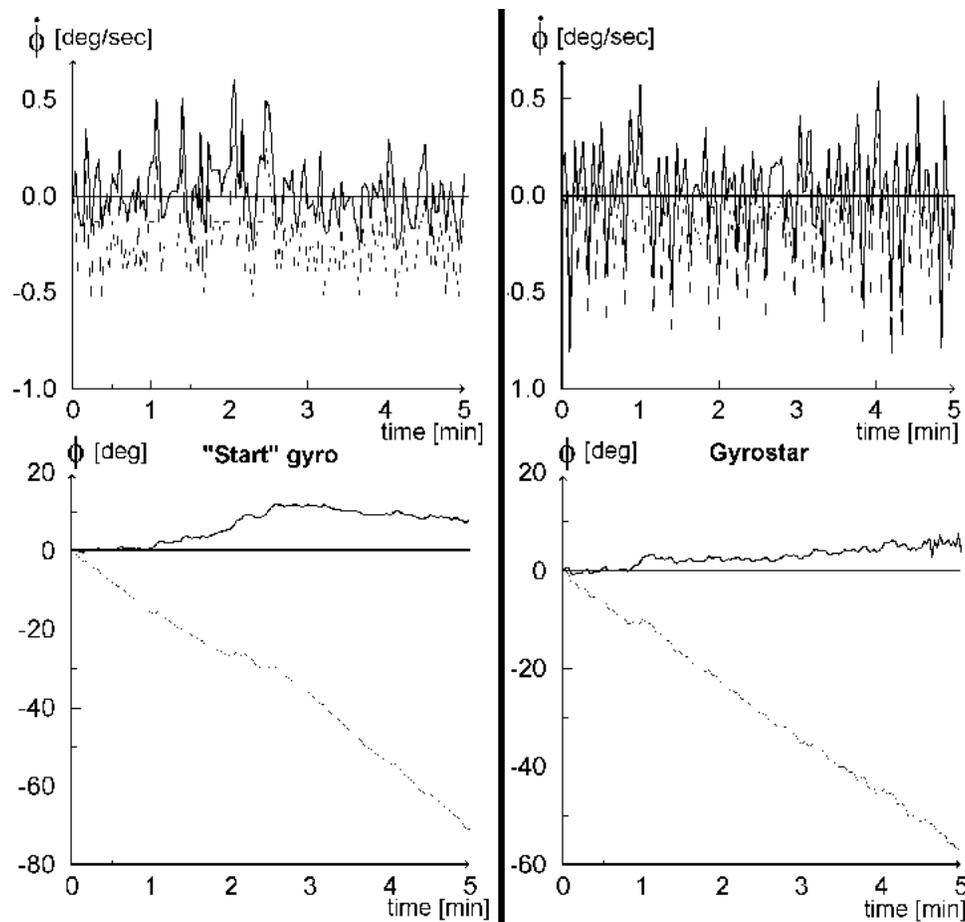
#### 5.4.2 Gyros

Gyros have long been used in robots to augment the sometimes erroneous dead-reckoning information of mobile robots. As we explained in Chapter 2, mechanical gyros are either prohibitively expensive for mobile robot applications, or they have too much drift. Recent work by Barshan and Durrant-Whyte [1993; 1994; 1995] aimed at developing an INS based on solid-state gyros, and a fiber-optic gyro was tested by Komoriya and Oyama [1994].

### 5.4.2.1 Barshan and Durrant-Whyte [1993; 1994; 1995]

Barshan and Durrant-Whyte developed a sophisticated INS using two solid-state gyros, a solid-state triaxial accelerometer, and a two-axis tilt sensor. The cost of the complete system was £5,000 (roughly \$8,000). Two different gyros were evaluated in this work. One was the ENV-O5S *Gyrostar* from [MURATA], and the other was the Solid State Angular Rate Transducer (*START*) gyroscope manufactured by [GEC]. Barshan and Durrant-Whyte evaluated the performance of these two gyros and found that they suffered relatively large drift, on the order of 5 to 15°/min. The Oxford researchers then developed a sophisticated error model for the gyros, which was subsequently used in an *Extended Kalman Filter* (EKF — see Appendix A). Figure 5.14 shows the results of the experiment for the *START* gyro (left-hand side) and the *Gyrostar* (right-hand side). The thin plotted lines represent the raw output from the gyros, while the thick plotted lines show the output after conditioning the raw data in the EKF.

The two upper plots in Figure 5.14 show the measurement noise of the two gyros while they were stationary (i.e., the rotational rate input was zero, and the gyros should ideally show  $\dot{\phi} = 0$  °/s). Barshan and Durrant-Whyte determined that the standard deviation, here used as a measure for the



**Figure 5.14:** Angular rate (top) and orientation (bottom) for zero-input case (i.e., gyro remains stationary) of the *START* gyro (left) and the *Gyrostar* (right) when the bias error is negative. The erroneous observations (due mostly to drift) are shown as the thin line, while the EKF output, which compensates for the error, is shown as the heavy line. (Adapted from [Barshan and Durrant-Whyte, 1995] © IEEE 1995.)

amount of noise, was  $0.16^\circ/\text{s}$  for the *START* gyro and  $0.24^\circ/\text{s}$  for the *Gyrostar*. The drift in the rate output, 10 minutes after switching on, is rated at  $1.35^\circ/\text{s}$  for the *Gyrostar* (drift-rate data for the *START* was not given).

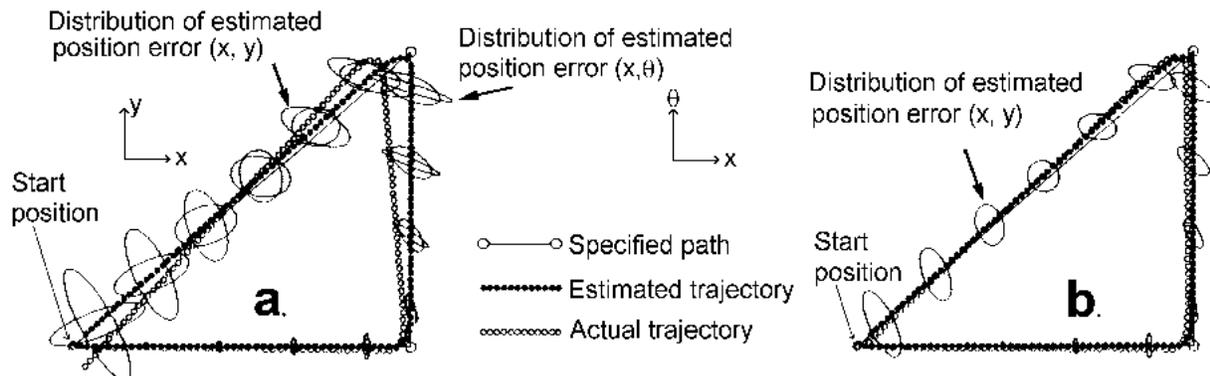
The more interesting result from the experiment in Figure 5.14 is the drift in the angular output, shown in the lower two plots. We recall that in most mobile robot applications one is interested in the heading of the robot, not the rate of change in the heading. The measured rate  $\dot{\phi}$  must thus be integrated to obtain  $\phi$ . After integration, any small constant bias in the rate measurement turns into a constant-slope, unbounded error, as shown clearly in the lower two plots of Figure 5.14. At the end of the five-minute experiment, the *START* had accumulated a heading error of  $-70.8$  degrees while that of the *Gyrostar* was  $-59$  degrees (see thin lines in Figure 5.14). However, with the EKF, the accumulated errors were much smaller: 12 degrees was the maximum heading error for the *START* gyro, while that of the *Gyrostar* was  $-3.8$  degrees.

Overall, the results from applying the EKF show a five- to six-fold reduction in the angular measurement after a five-minute test period. However, even with the EKF, a drift rate of 1 to  $3^\circ/\text{min}$  can still be expected.

#### 5.4.2.2 Komoriya and Oyama [1994]

Komoriya and Oyama [1994] conducted a study of a system that uses an optical fiber gyroscope, in conjunction with odometry information, to improve the overall accuracy of position estimation. This fusion of information from two different sensor systems is realized through a Kalman filter (see Appendix A).

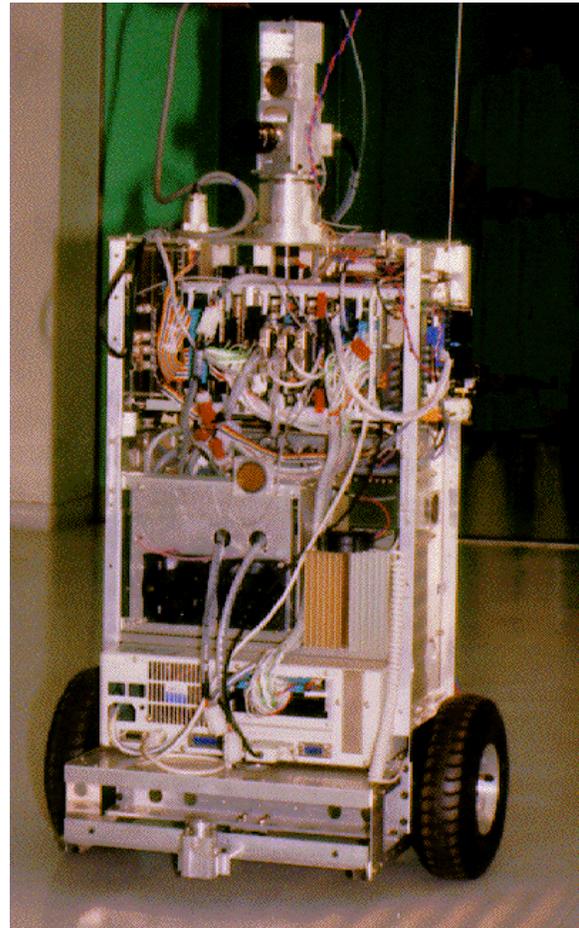
Figure 5.15 shows a computer simulation of a path-following study without (Figure 5.15a) and with (Figure 5.15b) the fusion of gyro information. The ellipses show the reliability of position estimates (the probability that the robot stays within the ellipses at each estimated position is 90 percent in this simulation).



**Figure 5.15:** Computer simulation of a mobile robot run.. (Adapted from [Komoriya and Oyama, 1994].)  
 a. Only odometry, without gyro information.      b. Odometry and gyro information fused.

In order to test the effectiveness of their method, Komoriya and Oyama also conducted actual experiments with *Melboy*, the mobile robot shown in Figure 5.16. In one set of experiments *Melboy* was instructed to follow the path shown in Figure 5.17a. *Melboy*'s maximum speed was 0.14 m/s (0.5 ft/s) and that speed was further reduced at the corners of the path in Figure 5.17a. The final position errors without and with gyro information are compared and shown in Figure 5.17b for 20 runs. Figure 5.17b shows that the deviation of the position estimation errors from the mean value is smaller in the case where the gyro data was used (note that a large average deviation from the mean value indicates larger non-systematic errors, as explained in Sec. 5.1). Komoriya and Oyama explain that the noticeable deviation of the mean values from the origin in both cases could be reduced by careful calibration of the systematic errors (see Sec. 5.3) of the mobile robot.

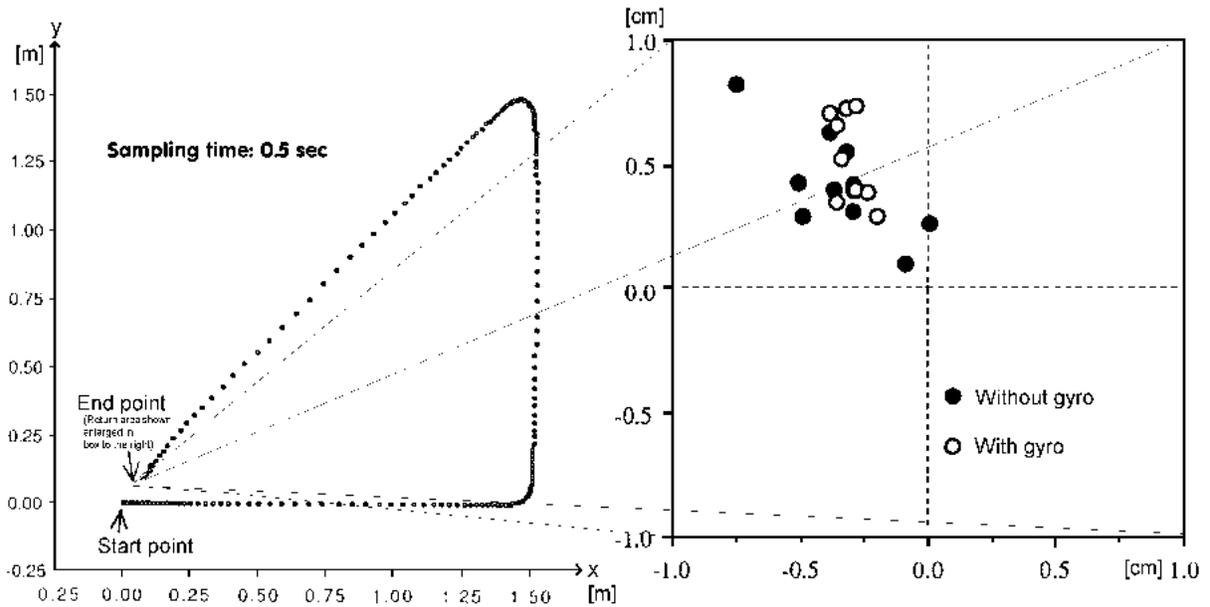
We should note that from the description of this experiment in [Komoriya and Oyama, 1994] it is not immediately evident how the “position estimation error” (i.e., the circles) in Figure 5.17b was found. In our opinion, these points should have been measured by marking the return position of the robot on the floor (or by any equivalent method that records the absolute position of the robot and compares it with the internally computed position estimation). The results of the plot in Figure 5.17b, however, appear to be too accurate for the absolute position error of the robot. In our experience an error on the order of several centimeters, not millimeters, should be expected after completing the path of Figure 5.17a (see, for example, [Borenstein and Koren, 1987; Borenstein and Feng, 1995a; Russel, 1995].) Therefore, we interpret the data in Figure 5.17b as showing a position error that was *computed* by the onboard computer, but not measured absolutely.



**Figure 5.16:** *Melboy*, the mobile robot used by Komoriya and Oyama for fusing odometry and gyro data. (Courtesy of [Komoriya and Oyama, 1994].)

## 5.5 Summary

- Odometry is a central part of almost all mobile robot navigation systems.
- Improvements in odometry techniques will not change their incremental nature, i.e., even for improved odometry, periodic absolute position updates are necessary.



**Figure 5.17:** Experimental results from *Melboy* using odometry with and without a fiber-optic gyro.

- Actual trajectory of the robot for a triangular path.
- Position estimation errors of the robot after completing the path of a. Black circles show the errors without gyro; white circles show the errors with the gyro.

(Adapted from [Komoriya and Oyama, 1994].)

- More accurate odometry will reduce the requirements on absolute position updates and will facilitate the solution of landmark and map-based positioning.
- Inertial navigation systems alone are generally inadequate for periods of time that exceed a few minutes. However, inertial navigation can provide accurate short-term information, for example orientation changes during a robot maneuver. Software compensation, usually by means of a Kalman filter, can significantly improve heading measurement accuracy.