LOCATION ESTIMATION AND UNCERTAINTY ANALYSIS FOR MOBILE ROBOTS

C. Ming Wang

Computer Science Department
General Motors Research Laboratories
Warren, Michigan 48090

Abstract

A motion controller for the autonomous mobile vehicle commands the robot's drive mechanism to keep the robot near its desired path at all times. In order for the controller to behave properly, the controller must know the robot's position at any given time. The controller uses the information provided by the optical encoders attached to the wheels to determine vehicle position. This paper analyzes the effect of measurement errors, wheel slippage, and noise on the accuracy of the estimated vehicle position obtained in this manner. Specifically, the location estimator and its uncertainty covariance matrix are derived.

1. Introduction

Determining the location of a robot is an important problem in navigating an autonomous vehicle in an unstructured environment. In a two dimensional space, the location of a mobile robot can be represented by a triplet (x, y, θ) ([7]) where x, y and θ are, respectively, the position and orientation of the robot. A problem arises from the fact that there is always error associated with the robot's motion. For example, in a two-wheeled drive system, the robot controller uses the information provided by the optical encoders attached to the wheels to command the robot's drive mechanism to keep the robot near its desired path at all time. However, the robot drive mechanism and controller may or may not follow commands very well and the measurements from the optical encoders are not error-free. These imprecisions are assumed random and can be modeled by a parametric distribution. The parameters of the error distribution can be determined experimentally.

Several methods for quantifying and dealing with this uncertainty have been proposed in the literature. Chatila and Laumond[3] used a scalar error estimate as the uncertainty measure for position and were not concerned with angular error. The uncertainty measure is used as the weight in combining the redundant measurements of the same entity. Brooks[1], employing a min/max error bounds approach, developed the uncertainty manifold for each location estimator. Smith and Cheeseman[8] (also see Smith, Self and Cheeseman[9]) used the covariance matrix as the uncertainty measure for the location estimator. They also introduced two operations which can be used to manipulate the relationship between any coordinate frame, given the chain of uncertain relative transformations linking them. The first operation is called compounding and can be better explained by Figure 1 (taken from Smith and Cheeseman[8]).

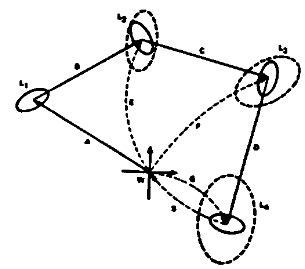

Figure 1. A Sequence of Transformations

In this example, the robot makes a number of moves and ends up near its initial position W. The solid ellipses represent the relative uncertainty of the robot with respect to its last position, while the dashed ellipses are the uncertainty of the robot with respect to W. The compounding process calculates the nominal location and associated error (dashed ellipse) of any object relative to any other object linked through a chain of transformations. The resulting compounded transformation has greater uncertainty than its components. That is, as the robot moves from one place to another, the uncertainty about its location with respect to the initial location grows.

The uncertainty could grow to a point that it becomes impossible for a robot to make any meaningful inference about its whereabouts. By using some auxiliary information, such as data from sensing landmarks, the uncertainty can be reduced. The reduction is done by employing a technique such as the Kalman filter to combine the existing uncertainty with new information. This is the merging process used by Smith and Cheeseman[8].

The calculation of the covariance matrix, in most cases, is not straightforward because of the nonlinear nature of the location estimator. Approximations are often needed. For example, Smith and Cheeseman[8] used the first-order Taylor expansion to evaluate the uncertainty covariance matrix. Wang[11] analytically verified the appropriateness of the first-order approximation, and described the limits of applicability.

In this article, the two-wheeled robot drive system based on the information from the optical encoders is examined. The estimation of robot states and the uncertainty analysis for such a system are presented. Specifically, the location estimator and its covariance matrix are derived. A comparison among existing

CH2555-1/88/0000/1230$01.00 © 1988 IEEE

methods for calculating the covariance matrix of the location estimator is also studied by simulation.

2. The Two-Wheeled Robot Drive System

A two-wheeled robot has two opposed drive wheels, mounted on the left and right sides of the robot, with their common axis passing through the center of the robot. The movement of the robot as a whole is indicated by the motion of the midpoint of the axis. In Figure 2 (also discussed by Tsumuru and Fujiwara[10]), the left and right wheel positions are denoted by A and C, and the midpoint of the axis, B, is the robot position reference point.

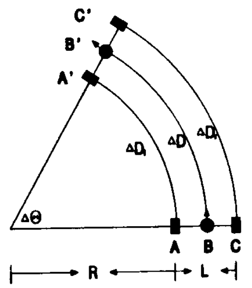

Figure 2. Illustration of the Drive System

In this example, the robot moves from B to B'. The length of the axis is $L = AC = A'C'$. The distance travelled, ΔD, and the angle changed, $\Delta\theta$, resulting from the movement can be calculated in terms of the incremental changes of the odometric measurements of the right and left wheel motions. Let ΔD_r and ΔD_l denote the covered distances of the right and left wheels respectively, then

$$\Delta D_r = (L + R)\,\Delta\theta, \quad \Delta D_l = R\,\Delta\theta.$$

Thus, we have

$$\begin{aligned}\Delta D &= (\Delta D_r + \Delta D_l)/2,\\ \Delta\theta &= (\Delta D_r - \Delta D_l)/L.\end{aligned} \tag{2.1}$$

That is, ΔD is the average of the outer and inner arcs and $\Delta\theta$ is proportional to the difference of the outer and inner arcs.

3. The Location of the Robot

In a two-dimensional space, the location (or state) of the robot at step (or time) n can be represented by

$$\mathbf{U}_n = (X_n\ Y_n\ \theta_n)^t.$$

Suppose we know that at time $n-1$, the robot is located at $O = (X_{n-1}, Y_{n-1})$ and is oriented in the direction of the point A as shown in Figure 3, i.e. $\theta_{n-1} = \angle AOF$. We wish to determine the location and orientation of the robot at time n, given that we know ΔD_n and $\Delta\theta_n$.

The new orientation, θ_n, is given by $\theta_{n-1} + \Delta\theta_n$. But the position of the robot is unknown. The robot can take any path that starts at O, has total arc length ΔD_n and turns by a total of $\Delta\theta_n$. To determine the position of the robot, we need to make some assumptions about the type of path it follows. If we assume a circular path, then (see Figure 3) $\Delta D_n = arc\ OP$ and $\Delta\theta_n = \angle ABP$. The location of the robot at time n, P, can be calculated as follows.

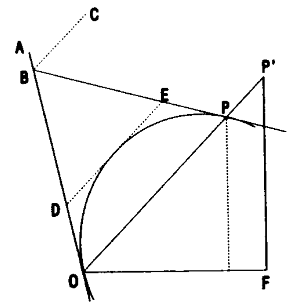

Figure 3. Illustration of Relationship of Positions

It can be shown that BC, which is parallel to DE and to OP, bisects the angle $\angle ABP$. That is,

$$\angle ABC = \angle ADE = \angle AOP = \Delta\theta_n/2.$$

Therefore, we have

$$\angle POF = \angle AOF - \angle AOP = \theta_{n-1} + \Delta\theta_n/2.$$

For the general case of arbitrary path, the length of OP is unknown, one commonly used method (e.g. Tsumura and Fujiwara[10], Julliere et al. [6], Iijima et al. [4]) is to approximate P by P' with the length of $OP' = arc\ OP = \Delta D_n$. With this approximation, we have

$$\begin{aligned}\Delta X_n &\approx \Delta D_n \cos(\theta_{n-1} + \Delta\theta_n/2)\\ \Delta Y_n &\approx \Delta D_n \sin(\theta_{n-1} + \Delta\theta_n/2),\end{aligned} \tag{3.1}$$

and the location of the robot at time n is approximated by

$$\begin{aligned}X_n &= X_{n-1} + \Delta X_n \approx X_{n-1} + \Delta D_n \cos(\theta_{n-1} + \frac{\Delta\theta_n}{2})\\ Y_n &= Y_{n-1} + \Delta Y_n \approx Y_{n-1} + \Delta D_n \sin(\theta_{n-1} + \frac{\Delta\theta_n}{2})\\ \theta_n &= \theta_{n-1} + \Delta\theta_n.\end{aligned} \tag{3.2}$$

However, with the added assumption of circular path, the relationship between the length of OP and arc OP can be obtained as (also see notations in Figure 2)

$$\frac{OP}{arc\ OP} = \frac{2(L/2 + R)\sin(\Delta\theta_n/2)}{(L/2 + R)\Delta\theta_n} = \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}.$$

That is,

$$OP = \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}\Delta D_n.$$

With this result, we have

$$\begin{aligned}\Delta X_n &= \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}\Delta D_n \cos(\theta_{n-1} + \frac{\Delta\theta_n}{2})\\ \Delta Y_n &= \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}\Delta D_n \sin(\theta_{n-1} + \frac{\Delta\theta_n}{2}),\end{aligned} \tag{3.3}$$

and the location of the robot at time n is given by

$$\begin{aligned}X_n &= X_{n-1} + \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}\Delta D_n \cos(\theta_{n-1} + \frac{\Delta\theta_n}{2})\\ Y_n &= Y_{n-1} + \frac{\sin(\Delta\theta_n/2)}{\Delta\theta_n/2}\Delta D_n \sin(\theta_{n-1} + \frac{\Delta\theta_n}{2})\\ \theta_n &= \theta_{n-1} + \Delta\theta_n.\end{aligned}$$

or in matrix form

$$\mathbf{U}_n = \mathbf{U}_{n-1} + \Delta\mathbf{U}_n. \tag{3.4}$$

Note that $\sin\phi/\phi \to 1$ as $\phi \to 0$, the term $\sin(\Delta\theta_n/2)/(\Delta\theta_n/2)$ can be viewed as an adjustment factor for the robot location due to the circular movement.

4. The Error of the Location Estimation

There is error in calculating the location of the robot at each time step. The error arises from the fact that the odometric measurements from both wheels are imperfect due to sensor noise and wheel slippage. These imprecisions are assumed random and can be modeled by a parametric distribution. In other words, we have

$$\Delta\hat{D}_r = \Delta D_r + \epsilon_r$$
$$\Delta\hat{D}_l = \Delta D_l + \epsilon_l,$$

where $\Delta\hat{D}_r$ and $\Delta\hat{D}_l$, respectively, are the observed distance that right and left wheels have moved; and ϵ_r and ϵ_l are random error. Furthermore, if we can assume that ϵ_r and ϵ_l are independently normally distributed with means 0 and variances σ_r^2 and σ_l^2, then

$$\Delta\hat{D}_r \sim N(\Delta D_r, \sigma_r^2)$$
$$\Delta\hat{D}_l \sim N(\Delta D_l, \sigma_l^2),$$

and the estimated incremental displacement $\Delta\hat{D} = (\Delta\hat{D}_r + \Delta\hat{D}_l)/2$ is distributed as normal with mean ΔD and variance $\sigma^2_{\Delta\hat{D}} = (\sigma_r^2 + \sigma_l^2)/4$. Similarly, the distribution of the estimated incremental orientation $\Delta\hat{\theta} = (\Delta\hat{D}_r - \Delta\hat{D}_l)/L$ is normal with mean $\Delta\theta$ and variance $\sigma^2_{\Delta\hat{\theta}} = (\sigma_r^2 + \sigma_l^2)/L^2$.

The covariance between $\Delta\hat{D}$ and $\Delta\hat{\theta}$ is given by

$$\begin{aligned} cov[\Delta\hat{D},\, \Delta\hat{\theta}] &= cov[(\Delta\hat{D}_r + \Delta\hat{D}_l)/2,\, (\Delta\hat{D}_r - \Delta\hat{D}_l)/L] \\ &= (\sigma_r^2 - \sigma_l^2)/2L. \end{aligned}$$

If $\sigma_r^2 = \sigma_l^2$; that is, the error variance of the right wheel is the same as that of the left wheel, which is a reasonable assumption in practice, then the covariance between $\Delta\hat{D}$ and $\Delta\hat{\theta}$ vanishes.

Under normality, $cov[X,Y] = 0$ implies that X and Y are independent. Thus, if the measurement errors of the optical decoders at both wheels are independently and identically normally distributed, then $\Delta\hat{D}$ and $\Delta\hat{\theta}$ are independent normal random variables. This result greatly simplifies the derivation of the covariance matrix of the location estimator of (3.4).

5. The Location Estimator

The trajectory-following controller commands right and left wheel velocities based on the information about the current desired location and current estimated location of the robot. The location of the robot at time n is estimated by

$$\hat{X}_n = \hat{X}_{n-1} + \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2}\Delta\hat{D}_n \cos(\hat{\theta}_{n-1} + \frac{\Delta\hat{\theta}_n}{2})$$
$$\hat{Y}_n = \hat{Y}_{n-1} + \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2}\Delta\hat{D}_n \sin(\hat{\theta}_{n-1} + \frac{\Delta\hat{\theta}_n}{2})$$
$$\hat{\theta}_n = \hat{\theta}_{n-1} + \Delta\hat{\theta}_n,$$

or

$$\hat{\mathbf{U}}_n = \hat{\mathbf{U}}_{n-1} + \Delta\hat{\mathbf{U}}_n. \tag{5.1}$$

Table 1. Biases of Adjustment Factor.

$\Delta\theta_n$ / $\sigma_{\Delta\hat{\theta}_n}$	0°	10°	30°	90°	180°
0°	.0	.0	.0	.0	.0
1°	.000012	.000012	.000012	.000010	.000004
2°	.000050	.000049	.000049	.000041	.000018
3°	.000114	.000112	.000111	.000093	.000041
4°	.000203	.000201	.000198	.000166	.000073
5°	.000317	.000315	.000310	.000260	.000114
6°	.000456	.000454	.000447	.000375	.000165
7°	.000621	.000619	.000608	.000510	.000224
8°	.000811	.000808	.000794	.000666	.000293
9°	.001027	.001023	.001005	.000843	.000371
10°	.001267	.001263	.001241	.001041	.000458
11°	.001533	.001529	.001501	.001260	.000554
12°	.001824	.001819	.001787	.001499	.000660

Knowledge of the statistical behavior of (5.1) is necessary in order to determine when the landmarks should be sensed and when such information is to be used to keep a guard on the robot's location information. We use the covariance matrix of (5.1) as the uncertainty measure for the location estimator.

Because of the presence of the adjustment factor, there is no exact method for calculating the covariance matrix of the location estimator in (5.1). Approximation methods are available. Commonly used approaches range from linearization to dropping off $\sin(\Delta\hat{\theta}_n/2)/(\Delta\hat{\theta}_n/2)$ in the calculation of the covariance matrix. Before we assess the virtues of various approximations, we shall examine this adjustment factor more closely (under the normality assumption).

Since $\Delta\theta_n$ lies between $-\pi$ and π, and $\sin\phi/\phi$ is an even function of ϕ, we will only study the behavior of $\sin\hat{\phi}/\hat{\phi}$ with $\phi \in [0, \pi/2]$ and $\hat{\phi}$ the estimate of ϕ.

First, we investigate the bias of the adjustment factor, i.e. see how close on average $\sin\hat{\phi}/\hat{\phi}$ is to $\sin\phi/\phi$. It is shown (Wang[12]) that

$$E\left[\frac{\sin\hat{\phi}}{\hat{\phi}}\right] - \frac{\sin\phi}{\phi} = -s_1 + s_2 - s_3 + \cdots + (-1)^n s_n + \cdots,$$

where

$$s_n = \frac{\sigma^{2n}}{n!\,2^n}\left(\frac{1}{2n+1} - \frac{1}{2n+3}\frac{\phi^2}{2!} + \cdots + \frac{(-1)^k}{2n+2k+1}\frac{\phi^{2k}}{(2k)!} + \cdots\right)$$

with σ the error variance of $\hat{\phi}$, or $\sigma = \sigma_{\Delta\hat{\theta}_n}/2$.

It can be shown that $s_{2n-1} - s_{2n} \geq 0$ and is a decreasing function of ϕ. Thus, for fixed σ^2, the bias

$$\left|E\left[\frac{\sin\hat{\phi}}{\hat{\phi}}\right] - \frac{\sin\phi}{\phi}\right| = (s_1 - s_2) + (s_3 - s_4) + \cdots + (s_{2n-1} - s_{2n}) + \cdots$$

attains its maximum at $\phi = 0$. Furthermore, it can be shown that

$$\left|E\left[\frac{\sin\hat{\phi}}{\hat{\phi}}\right] - \frac{\sin\phi}{\phi}\right| = \frac{\sin\phi}{\phi} - \int_0^1 e^{-\sigma^2 x^2/2}\cos(\phi x)dx. \tag{5.2}$$

The values of (5.2) for various combinations of $\sigma_{\Delta\hat{\theta}_n}$ and $\Delta\theta_n$ are given in Table 1. It is seen that the biases are quite small even for moderate values of angular error.

Next, we evaluate the variance of the adjustment factor, i.e. to see the spread of the distribution of $\sin\hat{\phi}/\hat{\phi}$. It is shown that

$$E\left[\frac{\sin^2\hat{\phi}}{\hat{\phi}^2}\right] = \int_0^1\int_0^{2x} e^{-\sigma^2y^2/2}\cos(\phi y)dy\,dx.$$

Thus, we have

$$var\left[\frac{\sin\hat{\phi}}{\hat{\phi}}\right] = \int_0^1\int_0^{2x} e^{-\sigma^2y^2/2}\cos(\phi y)dy\,dx - \left[\int_0^1 e^{-\sigma^2y^2/2}\cos(\phi y)dy\right]^2. \tag{5.3}$$

Although we can not show, analytically, any monotonicity properties of (5.3) as a function of ϕ (for fixed σ), the numerical results (from numerical integrations by routines DBLIN and DCADRE of IMSL[5]) do indicate that the variance of $\sin\hat{\phi}/\hat{\phi}$ is an increasing function of ϕ. Table 2 reports the variances of $\sin\hat{\phi}/\hat{\phi}$ for some selected values of $\sigma_{\Delta\hat{\theta}_n}$ and $\Delta\theta_n$.

Table 2. Variances of Adjustment Factor.

$\Delta\theta_n$	0°	10°	30°	90°	180°
$\sigma_{\Delta\hat{\theta}_n}$					
0°	.0	.0	.0	.0	.0
1°	.000000	.000000	.000000	.000004	.000012
2°	.000000	.000000	.000002	.000018	.000050
3°	.000000	.000000	.000005	.000041	.000112
4°	.000000	.000001	.000009	.000073	.000200
5°	.000000	.000001	.000014	.000115	.000312
6°	.000000	.000002	.000020	.000165	.000449
7°	.000000	.000003	.000028	.000225	.000611
8°	.000001	.000005	.000037	.000294	.000798
9°	.000002	.000007	.000048	.000373	.001010
10°	.000003	.000009	.000059	.000460	.001245
11°	.000004	.000012	.000073	.000557	.001506
12°	.000006	.000015	.000088	.000663	.001791

The above results (bias and variance) indicate that the distribution of the adjustment factor $\sin(\Delta\hat{\theta}_n/2)/(\Delta\hat{\theta}_n/2)$ is highly concentrated around $\sin(\Delta\theta_n/2)/(\Delta\theta_n/2)$ for a wide range of angular errors and incremental orientations. This simple fact enables us to find a good approximate covariance matrix for the location estimator in (5.1)

6. The Covariance Matrix of the Location Estimator

The covariance matrix of $\hat{\mathbf{U}}_n$ of (5.1) is given by

$$cov[\hat{\mathbf{U}}_n] = cov[\hat{\mathbf{U}}_{n-1}] + cov[\Delta\hat{\mathbf{U}}_n] + cov[\hat{\mathbf{U}}_{n-1},\ \Delta\hat{\mathbf{U}}_n] + cov[\Delta\hat{\mathbf{U}}_n,\ \hat{\mathbf{U}}_{n-1}],$$

where $cov[\mathbf{Z}]$ represents the *variance/covariance* matrix of $\mathbf{Z}$, while $cov[\mathbf{V},\ \mathbf{W}]$ is the *cross-covariance* matrix between $\mathbf{V}$ and $\mathbf{W}$. As we mentioned in the previous section, the presence of the adjustment factor makes the calculation of the covariance matrix of (5.1) difficult. To circumvent this problem and since for small $\Delta\hat{\theta}_n$, the adjustment factor is close to 1, an approximate method one would consider first is to discard $\sin(\Delta\hat{\theta}_n/2)/(\Delta\hat{\theta}_n/2)$ in the calculation of the covariance matrix. With this approximation, the derivation of $cov[\hat{\mathbf{U}}_n]$ is straightforward (see Wang[12]). For example, the $(1,1)$ diagonal element of matrix $cov[\Delta\hat{\mathbf{U}}_n]$ is given by

$$var_1(\Delta\hat{X}_n) = (1-a_n^2)^2\Delta X_n^2/2 + (1-a_n^4)\Delta Y_n^2/2 + (1+a_n^4\cos 2\phi_n)\sigma^2_{\Delta\hat{D}_n}/2 \tag{6.1}$$

and the $(1,2)$ off-diagonal element is

$$cov_1(\Delta\hat{X}_n,\ \Delta\hat{Y}_n) = a_n^2(a_n^2-1)\Delta X_n\Delta Y_n + a_n^4\sin\phi_n\cos\phi_n\sigma^2_{\Delta\hat{D}_n} \tag{6.2}$$

where

$$\phi_n = \theta_{n-1} + \Delta\theta_n/2$$
$$\hat{\phi}_n = \hat{\theta}_{n-1} + \Delta\hat{\theta}_n/2$$
$$a_n = exp[-var(\hat{\phi}_n)/2] = exp[-(4\sigma^2_{\hat{\theta}_{n-1}} + \sigma^2_{\Delta\hat{\theta}_n})/8].$$

The $(1,1)$ element of matrix $cov[\hat{\mathbf{U}}_{n-1},\ \Delta\hat{\mathbf{U}}_n]$ is given by

$$cov_1(\hat{X}_{n-1},\ \Delta\hat{X}_n) = b_n(1-c_n)^2\Delta X_{n-1}\Delta X_n/2 + b_n(1-c_n^2)\Delta Y_{n-1}\Delta Y_n/2$$

where

$$b_n = exp[-var(\hat{\phi}_{n-1}-\hat{\phi}_n)/2] = exp[-(\sigma^2_{\Delta\hat{\theta}_{n-1}} + \sigma^2_{\Delta\hat{\theta}_n})/8]$$
$$c_n = exp[-cov(\hat{\phi}_{n-1},\ \hat{\phi}_n)] = exp[-(2\sigma^2_{\hat{\theta}_{n-1}} - \sigma^2_{\Delta\hat{\theta}_{n-1}})/2].$$

The second approach uses the Taylor series to approximate the adjustment factor. Specifically, since

$$\sin(\hat{\theta}_{n-1}+\Delta\hat{\theta}_n) \approx \sin\hat{\theta}_{n-1} + \Delta\hat{\theta}_n\cos\hat{\theta}_{n-1} - \frac{(\Delta\hat{\theta}_n)^2}{2}\sin\hat{\theta}_{n-1},$$

thus,

$$\begin{aligned}\Delta\hat{X}_n &= \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2}\Delta\hat{D}_n\cos(\hat{\theta}_{n-1}+\frac{\Delta\hat{\theta}_n}{2})\\ &= \Delta\hat{D}_n\frac{\sin(\hat{\theta}_{n-1}+\Delta\hat{\theta}_n)-\sin\hat{\theta}_{n-1}}{\Delta\hat{\theta}_n}\\ &\approx \Delta\hat{D}_n(\cos\hat{\theta}_{n-1} - \frac{\Delta\hat{\theta}_n}{2}\sin\hat{\theta}_{n-1}).\end{aligned}$$

Similarly,

$$\Delta\hat{Y}_n \approx \Delta\hat{D}_n(\sin\hat{\theta}_{n-1} + \frac{\Delta\hat{\theta}_n}{2}\cos\hat{\theta}_{n-1}).$$

These results are then used in the calculation of $cov[\hat{\mathbf{U}}_n]$.

Both approaches discussed so far require that $\Delta\hat{\theta}_n$, the incremental orientation at time n, be small. However, the observer of the robot controller will update the location of the robot, but not calculate the covariance matrix, at each step because of the real-time consideration. Thus, by the time the observer evaluates the covariance matrix of the location estimate, the orientation change accumulated through steps may not be small enough to make the above two approximations valid. A better approximation which can tolerate a wider range of $\Delta\hat{\theta}_n$ is needed.

Instead of treating $\sin(\Delta\hat{\theta}_n/2)/(\Delta\hat{\theta}_n/2)$ as unity and dropping it from the calculation of the covariance matrix, the results in the previous section suggest that we can treat the adjustment factor as a constant in the calculation of the covariance matrix. That is, the diagonal elements of $cov[\Delta\hat{\mathbf{U}}_n]$ are given by

$$
\begin{aligned}
var(\Delta\hat{X}_n) &= var\left[\frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2}\Delta\hat{D}_n\cos(\hat{\theta}_{n-1}+\frac{\Delta\hat{\theta}_n}{2})\right] \\
&\approx \frac{\sin^2(\Delta\hat{\theta}_n/2)}{(\Delta\hat{\theta}_n/2)^2} var\left[\Delta\hat{D}_n\cos(\hat{\theta}_{n-1}+\frac{\Delta\hat{\theta}_n}{2})\right] \\
&= \frac{\sin^2(\Delta\hat{\theta}_n/2)}{(\Delta\hat{\theta}_n/2)^2} var_1(\Delta\hat{X}_n) \\
var(\Delta\hat{Y}_n) &\approx \frac{\sin^2(\Delta\hat{\theta}_n/2)}{(\Delta\hat{\theta}_n/2)^2} var_1(\Delta\hat{Y}_n) \\
var(\Delta\hat{\theta}_n) &= var_1(\Delta\hat{\theta}_n),
\end{aligned}
\tag{6.3}
$$

and the off-diagonal elements are

$$
\begin{aligned}
cov(\Delta\hat{X}_n,\ \Delta\hat{Y}_n) &\approx \frac{\sin^2(\Delta\hat{\theta}_n/2)}{(\Delta\hat{\theta}_n/2)^2} cov_1(\Delta\hat{X}_n,\ \Delta\hat{Y}_n) \\
cov(\Delta\hat{X}_n,\ \Delta\hat{\theta}_n) &\approx \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2} cov_1(\Delta\hat{X}_n,\ \Delta\hat{\theta}_n) \\
cov(\Delta\hat{Y}_n,\ \Delta\hat{\theta}_n) &\approx \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2} cov_1(\Delta\hat{Y}_n,\ \Delta\hat{\theta}_n),
\end{aligned}
\tag{6.4}
$$

where $var_1(\cdot)$ and $cov_1(\cdot)$ are given in (6.1) and (6.2). For the elements of the cross-covariance matrix $cov[\hat{\mathbf{U}}_{n-1},\ \Delta\hat{\mathbf{U}}_n]$, we have

$$
\begin{aligned}
cov(\hat{X}_{n-1},\ \Delta\hat{X}_n) &= cov(\Delta\hat{X}_{n-1},\ \Delta\hat{X}_n) \\
&\approx \frac{\sin(\Delta\hat{\theta}_{n-1}/2)}{\Delta\hat{\theta}_{n-1}/2}\,\frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2} cov_1(\hat{X}_{n-1},\ \Delta\hat{X}_n) \\
cov(\hat{\theta}_{n-1},\ \Delta\hat{X}_n) &\approx \frac{\sin(\Delta\hat{\theta}_n/2)}{\Delta\hat{\theta}_n/2} cov_1(\hat{\theta}_{n-1},\ \Delta\hat{X}_n).
\end{aligned}
$$

Since $0 \le \sin\phi/\phi \le 1$, the elements of $cov[\Delta\hat{\mathbf{U}}_n]$ in (6.3) and (6.4) are smaller than the elements in (6.1) and (6.2). When $\Delta\hat{\theta}_n$ is small, the differences are negligible. However, for larger $\Delta\hat{\theta}_n$, the differences can be significant. For example, $\sin^2(\Delta\hat{\theta}_n/2)/(\Delta\hat{\theta}_n/2)^2 = 0.912$ when $\Delta\hat{\theta}_n = 60°$, i.e. $var(\Delta\hat{X}_n)$ and $var(\Delta\hat{Y}_n)$ in (6.3) are almost 10% smaller than their counterparts in (6.1). For $\Delta\hat{\theta}_n = 90°$, it would result in a nearly 20% reduction. Thus, the question of interest is whether (6.1) and (6.2) overestimate or (6.3) and (6.4) underestimate the covariance matrix $cov[\Delta\hat{\mathbf{U}}_n]$. We will try to answer this question in the next section.

7. Comparisons of Different Approximations

We conducted simulation studies to compare the three approximations for various combinations of parameters and angular errors. One thousand independent random deviates $\Delta\hat{D}_n$, $\hat{\theta}_{n-1}$ and $\Delta\hat{\theta}_n$ were generated from normal with means ΔD_n, θ_{n-1} and $\Delta\theta_n$, and variances $\sigma^2_{\Delta\hat{D}_n} = (a + b\,\Delta D_n)^2$, $\sigma^2_{\hat{\theta}_{n-1}}$ and $\sigma^2_{\Delta\hat{\theta}_n}$, respectively. Covariance matrices of $(\Delta\hat{X}_n,\ \Delta\hat{Y}_n,\ \Delta\hat{\theta}_n)^t$ for different approximations are then constructed to compare with the nominal one. Since there is no approximation involved in the calculation of $var(\Delta\hat{\theta}_n)$, we compare only the covariance matrices of $(\Delta\hat{X}_n,\ \Delta\hat{Y}_n)^t$, $cov(\Delta\hat{X}_n,\ \Delta\hat{\theta}_n)$ and $cov(\Delta\hat{Y}_n,\ \Delta\hat{\theta}_n)$.

A total of 84 cases were run, but only 2 cases will be reported and discussed here. For detailed results, see Wang[12].

Figures 4 and 5 display the 95% confidence regions corresponding to the covariance matrices of $(\Delta\hat{X}_n,\ \Delta\hat{Y}_n)^t$. The parameters used were $\Delta D_n = 50$, $\sigma_{\Delta\hat{D}_n} = 4.25$ for both figures, $\theta_{n-1} = -45°$, $\sigma_{\hat{\theta}_{n-1}} = 10°$, $\Delta\theta_n = -10°$, $\sigma_{\Delta\hat{\theta}_n} = 4°$, for Figure 4, and $\theta_{n-1} = -90°$, $\sigma_{\hat{\theta}_{n-1}} = 15°$, $\Delta\theta_n = 120°$, $\sigma_{\Delta\hat{\theta}_n} = 5°$, for Figure 5. Ellipses labeled 0, 1, 2 and 3 correspond to nominal, method 1, method 2 and method 3 respectively. Method 1 evaluates the covariance matrix by using the Taylor expansion approximation; method 2 drops the adjustment factor from (5.1); and method 3 treats the adjustment factor as a constant in the calculation of the covariance matrix. It is seen that for small $\Delta\theta_n$ (Figure 4), methods 1, 2 and 3 give almost identical results which agree well to the nominal covariance matrix. Specifically, in Figure 4, the covariance matrices of $(\Delta\hat{X}_n,\ \Delta\hat{Y}_n)^t$ for nominal and methods 1 to 3 are, respectively,

$$
\begin{pmatrix} 86.6017 & \\ -5.2264 & 18.2567 \end{pmatrix},
\begin{pmatrix} 87.6494 & \\ -4.9043 & 18.3491 \end{pmatrix},
\begin{pmatrix} 86.8769 & \\ -5.1544 & 18.2732 \end{pmatrix}, \text{ and }
\begin{pmatrix} 86.6564 & \\ -5.1413 & 18.2268 \end{pmatrix}.
$$

It is evident that method 3 is by far the best approximation method when $\Delta\theta_n$ is large (Figure 5). The other 2 methods, especially method 1, are too conservative. The covariance matrices in Figure 5 are

$$
\begin{pmatrix} 40.745 & \\ 47.032 & 102.119 \end{pmatrix},
\begin{pmatrix} 205.030 & \\ 167.263 & 228.001 \end{pmatrix},
\begin{pmatrix} 60.668 & \\ 69.740 & 147.878 \end{pmatrix}, \text{ and }
\begin{pmatrix} 41.492 & \\ 47.696 & 101.137 \end{pmatrix}.
$$

Similar results are also obtained when comparing the covariance of $\Delta\hat{Y}_n$, $\Delta\hat{\theta}_n$ and the covariance of $\Delta\hat{X}_n$, $\Delta\hat{\theta}_n$.

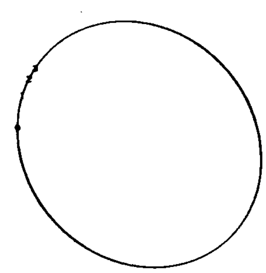

Figure 4. Comparisons with small $\Delta\theta_n$

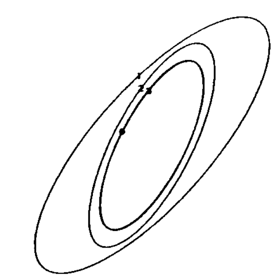

Figure 5. Comparisons with large $\Delta\theta_n$

From the above discussions, we can conclude that method 3 is a good approximation method for calculating the covariance matrix of the location estimator in (5.1). It is simpler than method 2, yet produces much more adequate results. It can tolerate a much wider range of $\Delta\theta_n$ than can methods 1 and 2.

8. Concluding Remarks

The role of the observer of a mobile robot controller is to maintain and update the information about the vehicle position and associated uncertainties, so the vehicle can behave properly. In this paper, we have presented a method for estimating the robot's location for the two-wheeled robot drive system based on the information from the optical encoders. This result differs from the earlier work by introducing a circular adjustment factor into the estimator. Since the variance of the adjustment factor is rather small under regular conditions, the proposed method gives a more accurate estimate of the robot's location without increasing the uncertainty of the estimate.

We have also proposed an approximate method for calculating the covariance matrix of the location estimator. It is shown that the covariance matrix so obtained approximates extremely well to the nominal one. The covariance matrix enables us to continuously maintain an approximation to uncertainty in location estimation and helps us to make decisions about the docking strategies and the need of sensing.

The real-time implementation of the controller makes the calculation of the covariance matrix an important issue. The first-order approximation used by Smith and Cheeseman[8] is very easy to compute and generalize. In addition, by precomputing and storing a set of Jacobian matrices that represent useful and frequently encountered spatial transformations, not only can an even greater computational efficiency be achieved, but also various types of transformations can be handled by a *common* general procedure. Although the appropriateness of the first-order approximation has been analytically verified by Wang[11] for uncertain transformations described by Smith and Cheeseman[8], the goodness of the first-order approximation in calculating the covariance matrix for (5.1) is generally unknown. Further study is planned to investigate the behavior of the observer obtained by the first-order approximation and compare it with that of the observer proposed in this article. The motivation is that if the first-order approximation can compare favorably enough, we would prefer the first-order approximation because of its simplicity, efficiency and generality.

References

1. Brooks, R.A. (1985), "Visual Map Making for a Mobile Robot", *Proc. IEEE Int. Conf. on Robotics and Automation*, 824-829, St. Louis, Missouri.
2. Chapman, J.W. (1970), "Moments, Variances, and Covariances of Sines and Cosines of Arguments which are Subject to Random Error", *Technometrics*, 12, 693-694.
3. Chatila, R. and Laumond, L-P. (1985), "Position Referencing and Consistent World Modeling for Mobile Robots", *Proc. IEEE Int. Conf. on Robotics and Automation*, 138-145, St. Louis, Missouri.
4. Iijima, J., Yuta, S. and Kanayama, Y. (1981), "Elementary Functions of A Self-Contained Robot - YAMABICO 3.1", *Proceedings of the 11th International Symposium on Industrial Robots*, 211-218, Tokyo.
5. IMSL Inc. (1980), *The IMSL Library*, Houston, Texas.
6. Julliere, M., Marce, L. and Place, H. (1983), "A Guidance System for a Mobile Robot", *Proceedings of the 13th International Symposium on Industrial Robots*, 58-68, Chicago, Illinois.
7. Lozano-Perez, T. (1983), "Spatial Planning: A Configuration Space Approach", *IEEE Transactions on Computers*, C-32, 108-120.
8. Smith, R.C. and Cheeseman, P. (1987), "On the Representation and Estimation of Spatial Uncertainty", *Int. Journal of Robotics Research*, 5, 56-68.
9. Smith, R.C., Self, M. and Cheeseman, P. (1986), "Estimating Uncertain Spatial Relationships in Robotics", *Proc. Second Workshop on Uncertainty in Artificial intell.*, Philadelphia, AAAI.
10. Tsumura, T. and Fujiwara, N. (1978), "An Experimental System for Processing Movement Information of Vehicle", *28th IEEE Vehicle Technology Conference Proceedings*, 163-168, Denver, Colorado.
11. Wang, C.M. (1986), "Error Analysis of Spatial Representation and Estimation of Mobile Robots", *GM Research Publication GMR-5573*, General Motors Research Laboratories, Warren, Michigan.
12. Wang, C.M. (1987), "Location Estimation and Error Analysis for an Autonomous Mobile Robot", *GM Research Publication GMR-5897*, General Motors Research Laboratories, Warren, Michigan.