Accurate Odometry and Error Modelling for a Mobile Robot

Kok Seng **CHONG**
kok.seng.chong@eng.monash.edu.au

Lindsay **KLEEMAN**
lindsay.kleeman@eng.monash.edu.au

Intelligent Robotics Research Centre (IRRC)
Department of Electrical and Computer Systems Engineering
Monash University, Clayton Victoria 3168, Australia
http://calvin.eng.monash.edu.au/IRRC/

Abstract

This paper presents the key steps involved in the design, calibration and error modelling of a low cost odometry system capable of achieving high accuracy dead-reckoning. A consistent error model for estimating position and orientation errors has been developed. Previous work on propagating odometry error covariance relies on incrementally updating the covariance matrix in small time steps. The approach taken here sums the noise theoretically over the entire path length to produce simple closed form expressions, allowing efficient covariance matrix updating after the completion of path segments. Systematic errors due to wheel radius and wheel base measurement were first calibrated with UMBmark test [4]. Experimental results show that, despite its low cost, our system's performance, with regard to dead-reckoning accuracy, is comparable to some of the best reported odometry vehicle.

1 Introduction

One of the major tasks of autonomous robot navigation is localisation. In a typical indoor environment with a flat floorplan, localisation becomes a matter of determining the Cartesian coordinates (x,y) and the orientation θ of the robot on a two dimensional floorplan. For a wheeled robot, odometry (also known as *dead-reckoning*) is one of the most important means of achieving this task. In practice, optical encoders that are mounted on both drive wheels feed discretised wheel increment information to a processor, which continually updates the robot's state using geometric equations. However, with time, odometric localisation accumulates errors in an unbounded fashion due to wheel slippage, floor roughness and discretised sampling of wheel increments.

At the hardware level, gyroscopes and/or accelerometers [2] are used to determine the position of the robot based on inertial navigation, but this method is susceptible to drift. Blanche [6] uses a single steerable drive wheel with a pair of knife-edge unloaded bearing wheels solely for odometry. Dead-reckoning implementations at the University of Michigan include [3]: Cybermotion K2A utilises synchro-drive, making it insensitive to non-systematic errors; CLAPPER, essentially two TRC LabMates connected by a compliant linkage, uses two rotary encoders to measure the rotation of the labmates relative to the compliant linkage, and a linear encoder to measure the relative distance between their centrepoints, giving it the unique ability to measure and correct non-systematic dead-reckoning errors during motion; Andros, a tracked vehicle, has a two-wheeled encoder trailer attached to its back which allows free horizontal rotation. The rotations of the trailer wheels and the trailer with respect to Andros are measured with the attached optical encoders.

Theoretical research normally involves error modelling, so that a mathematical treatment is possible. For example, the Extended Kalman Filter requires that the odometry errors be expressed in the form of an *error covariance matrix*. Odometry errors can be classified as being systematic or non-systematic. In the work by Borenstein and Feng [3], a calibration technique called UMBmark test has been developed to calibrate for the systematic errors of a two wheel robot. The dominant systematic error sources are identified as the difference in wheel diameter and the uncertainty about the effective wheel base. In our work, this method has been used to calibrate systematic errors in our robot.

For dealing with non-systematic errors, all classical models [4,8,9,12,13] either suffer from a lack of physical basis or inconsistency. To illustrate, suppose that at stage k-1, the state of the robot is $\mathbf{S}_{k-1} = [x_{k-1} \quad y_{k-1} \quad \theta_{k-1}]^T$, which comprises Cartesian coordinates (x_{k-1}, y_{k-1}) and orientation θ_{k-1} with respect to a global reference frame. A rotation α_k followed by a translation D_k moves the robot to a new state $\mathbf{S}_k$..

$$\mathbf{S}_k = \mathbf{f}(\mathbf{S}_{k-1}, \mathbf{u}_k) = \mathbf{S}_{k-1} + \begin{bmatrix} D_k \cos(\theta_{k-1} + \alpha_k) \\ D_k \sin(\theta_{k-1} + \alpha_k) \\ \alpha_k \end{bmatrix} \tag{1}$$

To propagate the error covariance matrix of $\mathbf{S}_{k-1}$ to $\mathbf{S}_k$, the error incurred is assumed to be small so that first order Taylor's expansion does not introduce significant higher order errors. By assuming that the error in stage *k-1* is not correlated with the error introduced by the input, the covariance matrix of stage k, $\mathbf{Cov}(\mathbf{S}_k)$, can be evaluated as follows,

$$\mathbf{Cov}(\mathbf{S}_k) = \nabla_{\mathbf{S}_k}\mathbf{f}\mathbf{Cov}(\mathbf{S}_{k-1})\nabla_{\mathbf{S}_k}\mathbf{f}^T + \nabla_{\mathbf{u}_k}\mathbf{f}\mathbf{Cov}(\mathbf{u}_k)\nabla_{\mathbf{u}_k}\mathbf{f}^T \qquad (2)$$

where $\nabla_{\mathbf{S}_k}\mathbf{f}, \nabla_{U_k}\mathbf{f}$ are the Jacobians of $\mathbf{f}$ with respect to $\mathbf{S}_k$ and $\mathbf{U}_k$, respectively, and

$$\mathbf{Cov}(\mathbf{u}_k) = \begin{bmatrix} \sigma_D^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix} \qquad (3)$$

In the authors' opinion, the major problem with this treatment is that there is no physical basis in assuming that the translation error is uncorrelated with the rotation error [8,4,12,13]. Model parameters do not reflect the physical characteristics of the system, hence they are difficult to obtain experimentally.

The model is also inconsistent. For the same path, if propagation of error is done in multiple parts, the model yields different solutions. To illustrate, suppose that $\mathbf{S}_{k-1} = [0 \quad 0 \quad 0]^T$ and $\mathbf{Cov}(\mathbf{S}_{k-1}) = \mathbf{0}$. Compare the following two scenarios :

(i) $\mathbf{u}_{k0} = [2D \quad 0]^T$ with $\mathbf{Cov}(\mathbf{u}_{k0}) = \begin{bmatrix} \sigma_{D0}^2 & 0 \\ 0 & \sigma_{\alpha 0}^2 \end{bmatrix}$.

From equation (2), $\mathbf{Cov}(\mathbf{S}_k) = \begin{bmatrix} \sigma_{D0}^2 & 0 & 0 \\ 0 & 4D^2\sigma_{\alpha 0}^2 & 2D\sigma_{\alpha 0}^2 \\ 0 & 2D\sigma_{\alpha 0}^2 & \sigma_{\alpha 0}^2 \end{bmatrix}$

(ii) $\mathbf{u}_{k1} = [D \quad 0]^T$ followed by $\mathbf{u}_{k1} = [D \quad 0]^T$, with $\mathbf{Cov}(\mathbf{u}_{k1}) = \begin{bmatrix} \sigma_{D1}^2 & 0 \\ 0 & \sigma_{\alpha 1}^2 \end{bmatrix}$. By applying equation (2) twice, $\mathbf{Cov}(\mathbf{S}_k) = \begin{bmatrix} 2\sigma_{D1}^2 & 0 & 0 \\ 0 & 5D^2\sigma_{\alpha 1} & 3D\sigma_{\alpha 1}^2 \\ 0 & 3D\sigma_{\alpha 1}^2 & 2\sigma_{\alpha 1}^2 \end{bmatrix}$

Even by setting $\mathbf{Cov}(\mathbf{u}_{k0})=2\cdot\mathbf{Cov}(\mathbf{u}_{k1})$, the two cases yield different final state error covariance, even though they lead to the same final state by following exactly the same path. This problem has been resolved [4] by performing error propagation for every time increment on the wheel encoders. This approach is conceptually similar to numerical integration but suffers a high computational cost. Also no physical justification for the error model has been offered.

The theoretical work by [14] has introduced a more realistic, physically-based error model for an arbitrary circular arc motion. The result is a model which is very accurate for large wheel turn variance, but limited in its applicability to a range of rotation angles. For large rotation angles, the robot path has to be divided into small segments in which the total turning angle is within the limitation of the model.

Other methods of representing position error includes the 'circular-error probable' (CPE) by [11] which is questionable because it is well known that the position error is usually not equal in all directions. [10] has proposed the use of equal-error probability *isoline* and has outlined some ways of growing the isoline as the vehicle moves. It remains uncertain whether new and sound methods could be developed to make use of them.

The new non-systematic error model developed by the authors has a strong physical basis which is closely related to the odometry design of our robot, Werrimbi. The model also generates error representation in the form of a error covariance matrix, which is the standard operating block for a multitude of noise filtering tools. Unlike the model illustrated earlier, the new model is consistent in a multiple path segments scenario. The computational load in incrementally updating the covariance matrix in small time steps, as done in [4] has been removed because simple close form formulae have been derived for three simple path types: (I) circular arc motion (II) straight line (III) rotation about the centre of the axle. Complex paths can be divided into a small number of sections which can be approximated by the aforementioned cases, hence the model can be applied on a section by section basis. Unlike the model in [14], it is valid for arbitrary distance and rotation angle. Even though the model by [14] is more accurate when the errors are large, compared to the first order accuracy of the new model, in the authors' opinion, should the robot be operating in conditions likely to incur large errors, an accurate representation of these large errors is inconsequential. Instead, preventive measures should be adopted to avoid such errors, such as employing external referencing. Our odometry model can then be used to optimally combine other sensors with odometry.

The remainder of this paper is organised into six sections: Section 2 presents the odometry system, and states, with justification, the key assumptions being incorporated into the model. Section 3 describes the UMBmark test used for reducing systematic errors and lists the key equations used. This is followed by section 4 on the derivation of the proposed non-systematic error model and its properties. The details and results of calibration of systematic errors and validation of the proposed error model constitute section 5. Lastly, section 6 is the conclusion which suggests possible future extensions of the model.

2 STEP ONE: Robot Design and Assumptions

Werrimbi has two pairs of wheels: drive wheels and encoder wheels. The encoder wheels are as sharp-edged as practically possible to reduce the wheel base (*B*) uncertainty, and are unloaded because they are independently mounted on linear bearings which allow vertical motion, hence the problem of wheel distortion and slippage is minimised. The idea of unloaded wheels can also be found in [7]. Based on the design, it is reasonable to assume that for a short unit of travel, the error incurred on both wheels are uncorrelated. because the two drive wheels are driven by two different motors, and two separate optical shaft encoders are used to gather odometry information. This assumption is also adopted in [3].

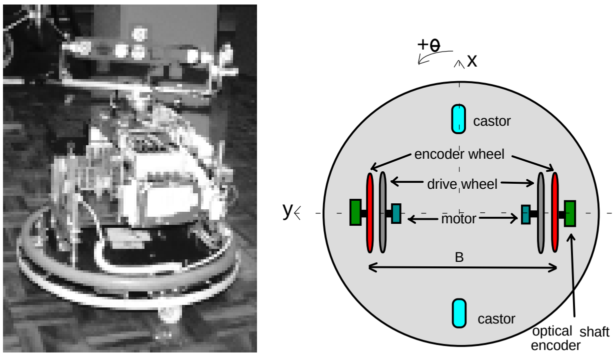


Figure 1 : (Left) Werrimbi, the sonar sensing robot (Right) Design of the accurate odometry system

Our work takes the assumption one step further. For a short unit of travel, the error is assumed to be zero mean, and white, that is, uncorrelated with the previous or next unit of travel. The variance of the cumulative error is then the sum of the variance of each statistically independent unit. This leads to a reasonable assumption that the variance of each unit of travel is proportional to the distance travelled

$$\sigma_L^2 = k_L^2 |d_L| \qquad \sigma_R^2 = k_R^2 |d_R| \tag{4}$$

where d_L and d_R are the distances travelled by each wheel, and k_L^2 and k_R^2 are constants with unit m$^{1/2}$.

3 STEP TWO: Calibration of Systematic Error Using UMBmark Test

UMBmark test [3] is used to calibrate wheel base error and unequal wheel diameter error. Briefly, the robot is programmed to travel a square path of side D in the clockwise sense (CW) n times, and the offsets of the final Cartesian coordinates from the initial Cartesian coordinates, e$x_{i,CW}$, e$y_{i,CW}$ are recorded. The experiment is repeated for the counterclockwise sense (CCW) and e$x_{i,CCW}$, e$y_{i,CCW}$ are recorded. The 'tuning factors' required to be incorporated into the software to counteract the effect of the systematic errors are calculated from the weighted Cartesian offsets in both senses. In summary, the centres of gravity of the offsets can be computed from their averages

$$x_{c.g.,CW/CCW} = \tfrac{1}{n}\sum_{i=1}^{n} ex_{i,CW/CCW}$$

$$y_{c.g.,CW/CCW} = \tfrac{1}{n}\sum_{i=1}^{n} ey_{i,CW/CCW} \tag{5}$$

With the two pairs of centres of gravity, the tuning factors for the wheel base, the radius of left wheel and the radius of right wheel, c_b, c_l and c_r, can be found by following this sequence of computations:

$$\alpha = average\left(\frac{x_{c.g.,CW} + x_{c.g.,CCW}}{-4D}, \frac{y_{c.g.,CW} - y_{c.g.,CCW}}{-4D}\right) \tag{6}$$

$$\beta = average\left(\frac{x_{c.g.,CW} - x_{c.g.,CCW}}{-4D}, \frac{y_{c.g.,CW} + y_{c.g.,CCW}}{-4D}\right) \tag{7}$$

$$E_d = \frac{D + B\sin(\beta/2)}{D - B\sin(\beta/2)} \tag{8}$$

$$c_b = \pi / (\pi - \alpha) \tag{9}$$

$$c_l = 2/(E_d + 1) \tag{10}$$

$$c_r = E_d c_l \tag{11}$$

and finally, the measure of dead-reckoning accuracy for systematic errors is defined in [3] as

$$E_{\max,syst} = \max\left(\sqrt{x_{c.g.,CW}^2 + y_{c.g.,CW}^2}, \sqrt{x_{c.g.,CCW}^2 + y_{c.g.,CCW}^2}\right) \tag{12}$$

4 STEP THREE: The New Non-systematic Error Model

With the new non-systematic error model, the entire path travelled by the robot is treated as consisting of k small segments. Propagation of error covariance is required to be done k times to obtain the error covariance of the final state. This section shows that it is possible to obtain a closed form solution for this model, as k approaches infinity. The solution for a general circular arc motion is first developed. The solutions for two special cases, straight line motion and on-spot turn are then obtained by suitably taking limits.

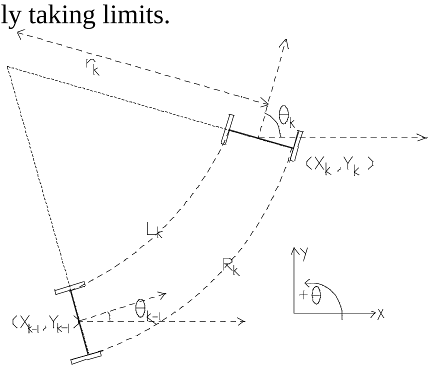


Figure 2 : Initial and final state of the robot after following a circular arc trajectory

Suppose that at segment k-1, the state of the robot is $\mathbf{S}_{k-1}$. It then propels its left wheel by L_k metre and its right wheel by R_k metre, to bring the robot to a new state $\mathbf{S}_k$. Over an infinitesimal time increment, the speed of the wheels can be assumed constant, hence the path takes on a circular arc trajectory with constant radius of curvature

r_k. Refer to Figure 2.

Based on the constant radius of curvature assumption, the following transition equation can be derived:

$$\mathbf{S}_k = \mathbf{f}(\mathbf{S}_{k-1}, \mathbf{u}_k) = \mathbf{S}_{k-1} + \begin{bmatrix} r_k\left[\sin\theta_{k-1} - \sin\left(\theta_{k-1} + \frac{R_k - L_k}{B}\right)\right] \\ r_k\left[\cos\left(\theta_{k-1} + \frac{R_k - L_k}{B}\right) - \cos\theta_{k-1}\right] \\ \frac{R_k - L_k}{B} \end{bmatrix} \qquad (13)$$

where r_k is the instantaneous radius of curvature,

$$r_k = \frac{B}{2}\left(\frac{L_k + R_k}{L_k - R_k}\right) \qquad (14)$$

where $\mathbf{u}_k = [L_k \quad R_k]^T$. The next step is to propagate the error covariance matrix associated with (*k-1*)*th* stage to the *kth* stage using equation (2). Now suppose that the arc segment is infinitesimally small, and the full path actually comprises k such segments being concatenated from end to end. The initial state of the robot is $\mathbf{S}_0$ which is at the starting end of the first segment and the last segment is $\mathbf{S}_k$ which is the destination of the last segment.

The expression for covariance propagation can be recursively expanded (like a Markov process),

$$\begin{aligned}\mathbf{Cov}(\mathbf{S}_k) &= \nabla_{\mathbf{S}_k}\mathbf{f}\,\mathbf{Cov}(\mathbf{S}_{k-1})\nabla_{\mathbf{S}_k}\mathbf{f}^T + \nabla_{\mathbf{u}_k}\mathbf{f}\,\mathbf{Cov}(\mathbf{u}_k)\nabla_{\mathbf{u}_k}\mathbf{f}^T \\ &= \left(\prod_{i=1}^{k}\nabla_{\mathbf{S}_i}\mathbf{f}\right)\mathbf{Cov}(\mathbf{S}_0)\left(\prod_{i=1}^{k}\nabla_{\mathbf{S}_i}\mathbf{f}\right)^T \\ &\quad + \sum_{i=1}^{k}\left\{\left(\prod_{j=i+1}^{k}\nabla_{\mathbf{S}_j}\mathbf{f}\right)\mathbf{Cov}(\mathbf{u}_k)\left(\prod_{j=i+1}^{k}\nabla_{\mathbf{S}_j}\mathbf{f}\right)^T\right\}\end{aligned} \qquad (15)$$

Let L_i , R_i denote the small increments in wheel turn at for the ith segment, and for circular arc motion, $L = kL_i$, $R = kR_i$. The $\mathbf{Cov}(\mathbf{S}_k)$ can be further evaluated to

$$\begin{aligned}&\mathbf{Cov}(\mathbf{S}_k) \\ &= \begin{bmatrix} 1 & 0 & r(\cos\theta_0 - \cos\theta_k) \\ 0 & 1 & r(\sin\theta_0 - \sin\theta_k) \\ 0 & 0 & 1 \end{bmatrix}\mathbf{Cov}(\mathbf{S}_0)\begin{bmatrix} 1 & 0 & r(\cos\theta_0 - \cos\theta_k) \\ 0 & 1 & r(\sin\theta_0 - \sin\theta_k) \\ 0 & 0 & 1 \end{bmatrix}^T \\ &+ \sum_{i=1}^{k}\left\{\left(\prod_{j=i+1}^{k}\nabla_{\mathbf{S}_j}\mathbf{f}\right)\begin{bmatrix} k_L^2|L_i| & 0 \\ 0 & k_R^2|R_i| \end{bmatrix}\left(\prod_{j=i+1}^{k}\nabla_{\mathbf{S}_j}\mathbf{f}\right)^T\right\}\end{aligned} \qquad (16)$$

The sum of products part of equation (16) gives rise to an error covariance matrix henceforth known as $\mathbf{Cov}(\mathbf{U}_k)$. Let $Cov(U_k)_{i,j}$ be the ith row, jth column component of $\mathbf{Cov}(\mathbf{U}_k)$, after taking k→∞ (and considerable effort!), then

$$\begin{aligned}Cov(U_k)_{1,1} &= \frac{-2rc_1}{(L-R)^2}(\sin\theta_0 - \sin\theta_k)\cos\theta_k + \left(\frac{r}{B}\cos\theta_k\right)^2 c_2 \\ &\quad - \frac{Bc_3}{4(L-R)^3}\left[2(\theta_k - \theta_0) - \sin(2\theta_0) + \sin(2\theta_k)\right] \\ Cov(U_k)_{2,2} &= \frac{-2rc_1}{(L-R)^2}(\cos\theta_k - \cos\theta_0)\sin\theta_k + \left(\frac{r}{B}\sin\theta_k\right)^2 c_2 \\ &\quad - \frac{Bc_3}{4(L-R)^3}\left[2(\theta_k - \theta_0) + \sin(2\theta_0) - \sin(2\theta_k)\right] \\ Cov(U_k)_{3,3} &= \frac{c_2}{B^2} \\ Cov(U_k)_{1,2} &= \frac{-rc_1}{(L-R)^2}(\cos(2\theta_k) - \cos(\theta_0 + \theta_k)) + \frac{c_2}{2}\left(\frac{r}{B}\right)^2\sin(2\theta_k) \\ &\quad - \frac{Bc_3}{4(L-R)^3}\left[\cos(2\theta_0) - \cos(2\theta_k)\right] \\ Cov(U_k)_{1,3} &= \frac{c_1}{(L-R)^2}(\sin\theta_0 - \sin\theta_k) - \frac{rc_2}{B^2}\cos\theta_k \\ Cov(U_k)_{2,3} &= \frac{c_1}{(L-R)^2}(\cos\theta_k - \cos\theta_0) - \frac{rc_2}{B^2}\sin\theta_k\end{aligned} \qquad (17)$$

where

$$\begin{aligned}c_1 &= k_R^2|R|L + k_L^2 R|L| \\ c_2 &= k_R^2|R| + k_L^2|L| \\ c_3 &= k_R^2 L^2|R| + k_L^2 R^2|L|\end{aligned} \qquad (18)$$

These equations remove the need to incrementally update the covariance matrix in small time steps. As closed form expressions, they are applicable to any circular arc motion with constant radius of curvature. For simple paths, the matrix terms can be further simplified by suitably taking limits.

4.1 Special Cases : Straight Line Translation and Rotation about the Centre of Wheel Axle

For a straight line path of length *D*, both wheels rotate by the same amount and the initial angle is approximately the same as the finally angle, the limits to be taken are *L,R*→*D* and $\theta_0 \rightarrow \theta_k$. Contrary to popular assumptions [7,12], our model predicts that variance in the direction perpendicular to the direction of motion is proportional to the cube of the distance travelled, whereas the variance in the direction of motion is only proportional to the distance travelled. For rotation about the centre of wheel axle, both wheels still rotate by the same amount, but in opposite direction. The limits to be taken are $L \rightarrow \frac{B}{2}(\theta_k - \theta_0)$ and $R \rightarrow \frac{B}{2}(\theta_0 - \theta_k)$. All simplified expressions can also be found in [5].

This model is **consistent** unlike the models presented in the introduction of this paper [4,8,9,12,13]. That is, if propagation is done in multiple parts, the model generates exactly the same result. This is because the model itself is founded upon the concept of incrementally propagating error covariance from one infinitesimal section to the next.

5 Implementation and Results

5.1 Calibration for Systematic Error

The experiments were conducted in a lab with a parquetry floor. The wheel encoder measurements were used to calculate the perceived final state of the robot, whereas a sonar array [15] mounted on top of the robot was to used to estimate its actual state by sensing some reference walls placed close to the initial position. Two reference walls were used to establish the robot's coordinates and orientation.

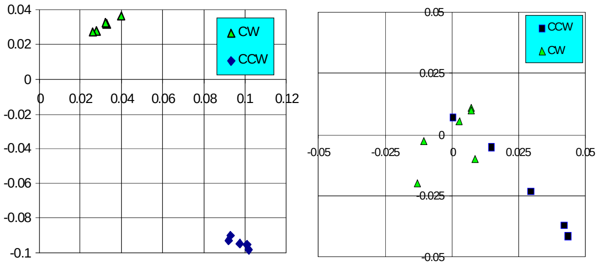


Figure 3 : Result of UMBmark test, before and after calibration. (metre units)

The distribution of Cartesian offsets after the completion of *D*=4m square path for 5 runs in each sense (clockwise and counterclockwise), before and after calibration, are shown in Figure 3. The value of D has been chosen as such in order to make benchmark comparison with the results presented in [3].

The calibration results are presented in Table 1. Comparison with other robot vehicles are made in Table 2. All except Cybermotion K2A are calibrated. It can be seen that the measure of dead-reckoning accuracy for systematic errors, $E_{max,syst}$, of Werrimbi is comparable to those achievable by many advanced odometry systems. Further calibration has been carried out with the compensated parameters but no significant improvement has been made.

Table 1 : Key results before and after calibration

	Before	After
$x_{c.g.,CCW}$ (mm)	97	-26
$y_{c.g.,CCW}$ (mm)	-94	16
$x_{c.g.,CW}$ (mm)	32	1.5
$y_{c.g.,CW}$ (mm)	31	11
$E_{max,syst}$ (mm)	135	30

Table 2 : Comparison of dead-reckoning accuracy and approximate cost of our design with four vehicles. The first four sets of figures are obtained from [3]

Vehicle	$E_{max,syst}$	Cost (US$)
TRC LabMate	average 27	10K
Cybermotion K2A	63	<50K
CLAPPER	22	30K
Andros with Trailer	74	?
Werrimbi	30	~4K

5.2 Computation of Non-systematic Error Parameters

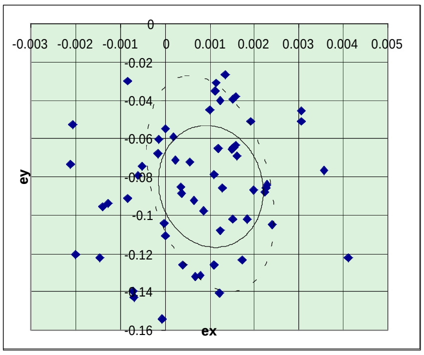
After calibration, Werrimbi was programmed such that it scanned two reference walls, moved forwards 10 metres, moved backward 10 metres, re-scanned the two reference walls and compared the position estimation from sonar sensing and odometry reading. The process was repeated 60 times (5 hours consumed) and the difference in the Cartesian coordinates, (*ex*, *ey*) and the difference in orientation *et* for all 60 runs are plotted against each other in Figure 4(a)-(c). The values of k_L and k_R have been obtained by fitting error ellipses to the data, and they are k_L=0.00040m$^{1/2}$, k_R=0.00058m$^{1/2}$. In Figure 4, both the 95% confidence error ellipses of the actual data and the 95% confidence error ellipses generated with k_L and k_R are overlayed on the plots for comparison.

6 *Conclusion and Future Work*

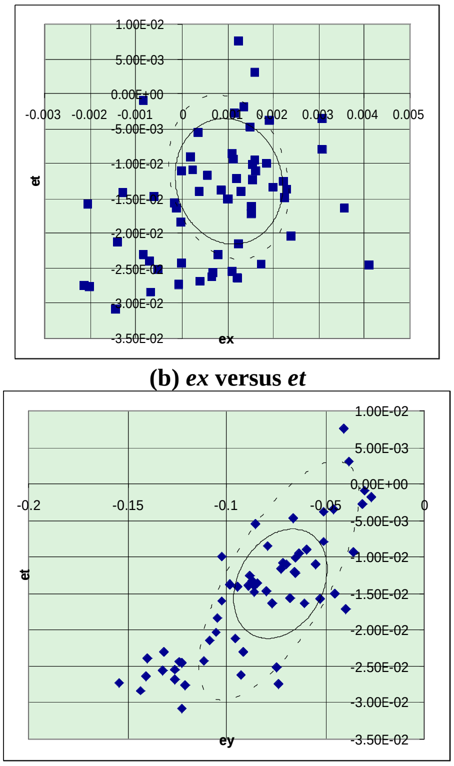
This work is the preliminary stage of a robotics mapping project [5]. It draws together the key considerations and procedures involved in the calibration and error modelling for an odometry design. An accurate odometry system has been presented which is comparable to the best reported system but can be fabricated at low cost. A new first order odometry error model has been derived to propagate the state error covariance following a motion. The error model takes the form of a covariance matrix which is prevalent in statistics and filtering theory, hence could easily fit into many powerful tools such as the Kalman Filter. This model cannot account for unexpected errors such as hitting a bump on the floor, which violates the flat floorplan assumption. For certain applications such as mapping, external referencing should be deployed to detect such errors.

7 *Acknowledgment*

The authors are grateful to Mr. Greg Curmi who took part in the design of the robot.



(a) *ex* versus *ey*



(b) ***ex*** **versus** ***et***

(c) ***ey*** **versus** ***et***

Figure 4 : Errors in the along-track direction (*ex*), the off-track direction (*ey*) and orientation (*et*), for 60 runs, are plotted against each other. The solid ellipses belong to the real data whereas the dashed ellipses are generated with the k_R and k_L obtained by trial and error.

8 Reference

[1] Bar-Shalom, Y. and Li, X.R. "Estimation and Tracking: Principles, Techniques and software", Boston, London: Artech House Inc., 1993

[2] Barshan, B. and Durrant-Whyte, H. “An Inertial Navigation System for a Mobile Robot”, IEEE Transactions on Robotics and Automation, Vol 11, No. 3, June 1995, pp. 328-342.

[3] Borenstein, J. and Feng, L. “UMBmark - A method for Measuring, Comparing, and Correcting Dead-reckoning Errors in Mobile Robots.”, Technical Report UM-MEAM-94-22, University of Michigan

[4] Crowley, J.L. and Chenavier, F. "Position Estimation for a Mobile Robot Using Vision and Odometry", Proceedings 1992 IEEE International Conference on Robotics and Automation, pp.2588-2593.

[5] Chong, K.S. and Kleeman, L, “Accurate Odometry and Error Modelling for a Mobile Robot”, Technical Report MECSE-1996-6, 1996, Intelligent Robotics Research Centre, Department of Electrical and Computer Systems Engineering, Monash University

[6] Chong, K.S. and Kleeman, L. “Sonar Based Map Building for a Mobile Robot”, 1997 International Conference on Robotics and Automation (this issue).

[7] Cox, I.J. "Blanche - An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle", IEEE Transaction on Robotics and Automation, 1991, Vol 3, pp.193-204.

[8] Durrant-Whyte, H.F. et al "Position Estimation And Tracking Using Optical Range Data", Proceedings 1993 IEEE International Conference On Robotics and Automation, Vol 3, pp.2172 - 2177.

[9] Jenkin, M. et al "Global Navigation for ARK", Proceedings 1993 IEEE International Conference On Robotics and Automation, Vol 3, pp.2165-2171.

[10] Krantz, D. “Non-Uniform Dead-Reckoning Position Estimate Updates”, IEEE International Conferencve on Robotics and Automation, April 1996, Vol 2, pp 2061-2066.

[11] Leenhouts, P. “On the Computation of Bi-Normal Radial Error”, Navigation: Journal of The Institute of Navigation, Vol 32, No 1, Spring 1985, pp. 16-28.

[12] Lu, F. and Milios, E.E. "Optimal Global Pose Estimation for Consistent Sensor Data Registration", IEEE International Conference on Robotics and Automation, 1995, pp.93-100.

[13] Nishizawa, T. et al "An Implementation of On-board Position Estimation for a Mobile Robot", IEEE International Conference on Robotics and Automation, 1995, pp.395-400.

[14] Wang, C.M., “Location Estimation and Uncertainty Analysis for Mobile Robots”, in IEEE Conference on Robotics and Automation, 1988, pp. 1230-1235

[15] Kleeman, L. and Kuc, R. "Mobile Robot Sonar for Target Localization and Classification", The International Journal of Robotics Research, Vol. 14, No 4, August 1995, pp.295-318.