Odometry, Dead Reckoning and Error Predictions

Anton Olsson, Harald Lilja
Halmstad University
(Dated: February 20, 2019)

This is a report covering the second exercise in the course Intelligent Vehicles (DT8020) at Halmstad University. The exercise gives the student at chance to experiment with odometry, deck reckoning and errors for two different robots. The datasets given to the student are comprised in two different ways, one in which the data is circular and the other in which the data isn't. Different equations presented gives the reader a good understanding of how to use the data to calculate not only the heading and distance for the robot but uncertainties aswell. Problems posed during the method is answered in the section covering the results. The results will also show the reader what effect different assumptions during the method will have on the outcome.

I. INTRODUCTION

A. Problem description

A technique for estimating how a vehicles position change in relation to time is odometry. This is the computation of data received from motion sensors to evaluate the position by intergrating velocity measurements over time. In this assignment our objective is to use provided sensor data from such measurements to estimate a position change and the errors it involves.

B. Overview of report

The assignment will be divided in two parts. In the first part two datasets from the robot khepera is provided for analysis. Khepera is a circular robot consisting of 2 wheels that uses different velocity on its wheels to perform turns. The dataset from this robot is provided in the form of velocity of each wheel(See figure 1). The second part of the assignment provides a dataset from a robot called snowhite. This is a three-wheeled robot that uses one wheel to steer and the other two for momentum. In this part, the data is given in the form of the velocity of the two rear wheels and the angle of the steering wheel(α in figure 2).

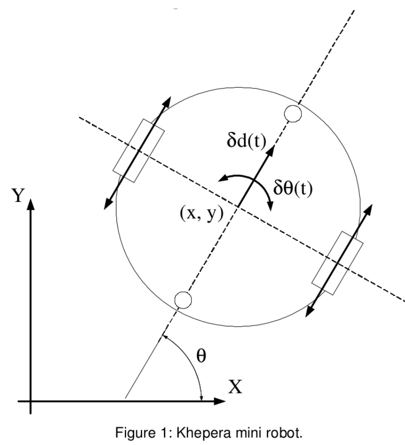

Figure 1. Model of the khepera robot

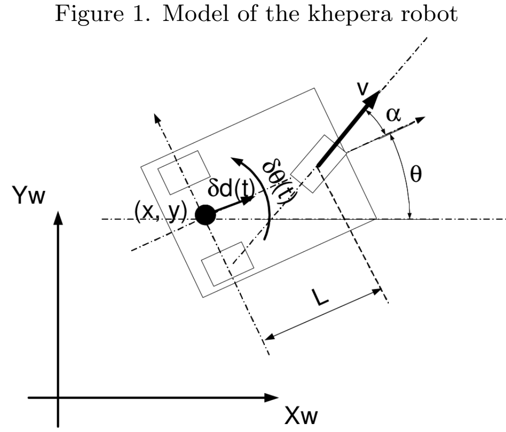

Figure 2. Model of the snowhite robot

1. Introduction to the Khepera robot

The task is to plot the path taken by the robot as well as the uncertainty. We are also to implement an error in the wheel diameter and wheel base and adjust our model after these errors.

- Wheel base (L) = 53 [mm]
- Wheel diameter = 15.3 [mm]
- Pulses per revolution = 600

For the first part we are handed two datasets, one in which the path taken is circular and another one where the path appears to be random.

2. Introduction to the Snowhite robot

The sensor data from this robot is given in velocity and angle of the steering wheel.

1. Wheel base = 680 [mm]
2. Sampling time = 50 [ms]

The data for Snowhite is given to us in the following format:

$v(1)\ \ \alpha(1)\ \ True_x(1)\ \ True_y(1)\ \ True_\theta(1)$

$v(2)\ \ \alpha(2)\ \ True_x(2)\ \ True_y(2)\ \ True_\theta(2)$
...
$v(N)\ \ \alpha(N)\ \ True_x(N)\ \ True_y(N)\ \ True_\theta(N)$

Where the first two values represent the velocity and angle of the front wheel while the three other values represent the true x, y and θ values.

II. THEORY AND METHOD

A. Theory Khepera

To calculate the direction the following equation is used:

$$\Delta d = \frac{\Delta r + \Delta l}{2} \tag{1}$$

The heading of the robot can be calculated in the following way.

$$\Delta\theta = \frac{\Delta r - \Delta l}{L} \tag{2}$$

Where Δr, Δl is the distance travelled by the right and left wheel, while L is the wheel base mentioned in section I B 1.

We calculate the uncertainty in the direction and heading using the following equations:

$$\sigma^2_{\Delta d} = \frac{\sigma^2 r + \sigma^2 l}{4} \tag{3}$$

$$\sigma^2_{\Delta\theta} = \frac{\sigma^2 r - \sigma^2 l}{L^2} \tag{4}$$

Here the $\sigma^2 r$ and $\sigma^2 l$ represent uncertainty in right and left wheel.

When assuming a circular path a compensation term is introduced, this compensation term is from [1]:

$$C = \frac{sin(\frac{\Delta\theta}{2})}{\frac{\Delta\theta}{2}} \tag{5}$$

To calculate the errors we use the error propagation law:

$$\Sigma_{X_k} = J_{X_{k-1}}\Sigma_{X_{k-1}}J^T_{X_{k-1}} + J_{U_k}\Sigma_{U_k}J^T_{U_k} \tag{6}$$

The update for the state variables is calculated in the following way, where U_k will represent the different displacement vectors based on the measured data. The vector represents the previous value calculated.

$$X_k = f(X_{k-1}, U_k) = \begin{bmatrix} X_{k-1} \\ Y_{k-1} \\ \theta_{k-1} \end{bmatrix} + U_k \tag{7}$$

Displacement vector based on Δd and $\Delta\theta$:

$$U_{k_{\Delta d \Delta\theta}} = \begin{bmatrix} \Delta d cos(\theta_{k-1} + \frac{\Delta\theta}{2}) \\ \Delta d sin(\theta_{k-1} + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{bmatrix} \tag{8}$$

Displacement vector based on Δr, Δl and b:

$$U_{k_{\Delta l \Delta r b}} = \begin{bmatrix} \frac{\Delta r + \Delta l}{2} cos(\theta_{k-1} + \frac{\Delta r - \Delta l}{2b}) \\ \frac{\Delta r + \Delta l}{2} sin(\theta_{k-1} + \frac{\Delta r - \Delta l}{2b}) \\ \frac{\Delta r - \Delta l}{b} \end{bmatrix} \tag{9}$$

Displacement vector based on v, a and T:

$$U_{k_{v\alpha T}} = \begin{bmatrix} vcos(\alpha)Tcos(\theta_{k-1} + \frac{vsin(\alpha)T}{2L}) \\ vcos(\alpha)Tsin(\theta_{k-1} + \frac{vsin(\alpha)T}{2L}) \\ \frac{vsin(\alpha)T}{L} \end{bmatrix} \tag{10}$$

Problems posed for Khepera

a) What uncertainties do we assume?

b) Are the estimated state variables and co-variance matrices the same?

c) If the encoder values where not read as often as they are, i.e. if they where read 2, 5 or maybe 10 times as seldom, what would then happen to the state variables?

d) When do the state variables start to differ – and why?

e) How are the covariance matrices evolving during the run? Is the uncertainties realistic?

f) How does a new wheel base and wheel diameter affect the estimated state variables?

g) Can we also include the contributions of the new uncertain parameters to the overall uncertainty in our state variables?

h) What assumptions on the errors are we making?

Problems posed for Snowhite

i) What errors do we assume in the steering angle and in the speed?

j) How long is the path in distance and time?

III. RESULTS

A. Results from part 1

The forward direction Δd and heading $\Delta\theta$ was computed using equations (1) and (2) respectively. When calculating the covariances an uncertainty of $\sigma_r^2 = \sigma_l^2 = 0.5/12$ mm was assumed. This assumption was made as a suggestion provided in the assignment and no further evaluation was done for it. This answers question a) in II. The variance of the wheels was then computed as proposed in equations (3) and (4). The jacobian matrices with respect to X, Y, θ and ΔD, $\Delta\theta$ was computed respectively using equation (7) and (8) as U_k. Finally the state uncertainty was computed using the error propogation law described in equation (6)

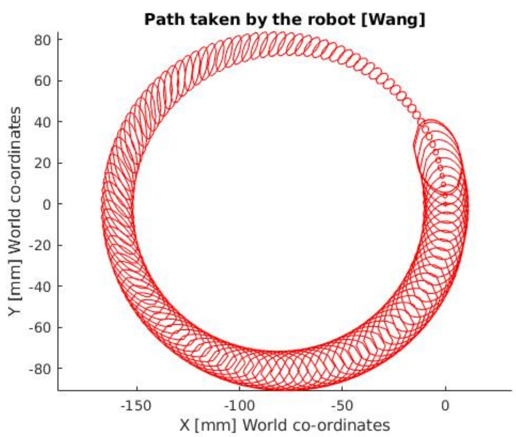

Figure 3. The robots path with the uncertainty represented as an ellipse around the estimated position

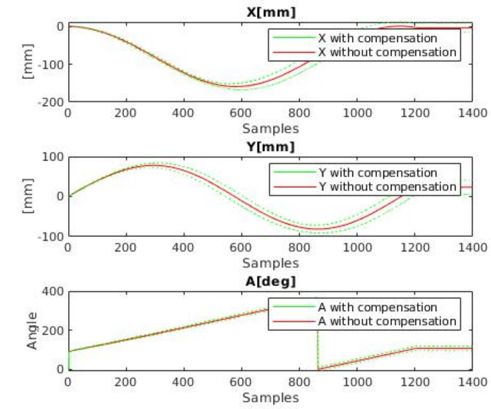

Figure 4. State variables with and without compensation term. Sampling period of 1s

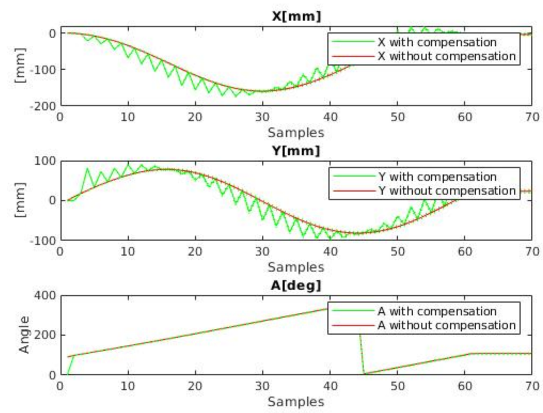

Figure 5. State variables using sampling period of 20s

In figure 3 the circular path taken by the robot is visualized along with its uncertainty. Here, it is easy to see how the uncertainty grows over time as expected due to the error being integrated over time. This answers question e) in II A.

After implementing the compensation term (5) we plotted this result in relation to the previous setup to view changes in the state variables. These plots were made using two sampling periods to view how this would effect our results. The sampling periods were 1s and 20s.

With the 1s sampling time, the compensation term is not influencing the result very much. This is due to the small $\Delta\theta$ which does not actually need any bigger adjustments to stay on its track. When using a sampling time of 20s, $\Delta\theta$ is greater which results in a larger influence from the compensation term. This can be seen in the difference between figure 4 and figure 5. As the figures shows, the compensation term has a greater influence when the change in orientation (larger sampling period in this case) is big. This answers c) in *Problems posed for Khepera* II A. To answer question b), no they are not the same. The introduced compensation term is only applied to the state variables and not to the uncertainties. To answer question d) the state variables start to differ as soon as the change in orientation is large. So when the sampling period is large, the robot will have moved a larger distance than when the sampling period is small, therefor the change in orientation will be larger. This will lead to the compensation term having a larger impact.

As a next step we changed the settings of the robot to the following setup and plotted the both cases in a single graph:

- Wheel base (L) = 45 [mm]
- Wheel diameter = 14 [mm]
- Pulses per revolution = 600

In figure 6 we can see the relation between the different settings. The case with the smaller diameter and wheel base generates a circle with a smaller radius. This is mainly due to the smaller wheel base.

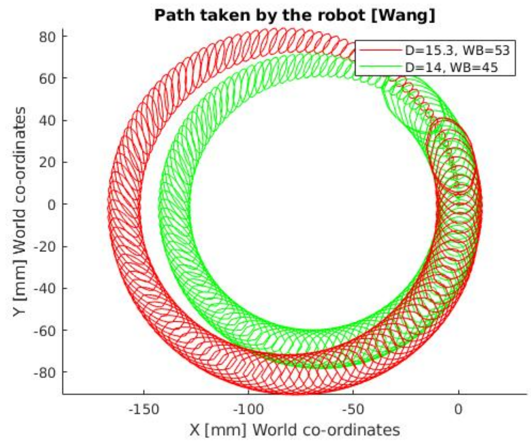

Figure 6. Relation between the old and new setup of the robot

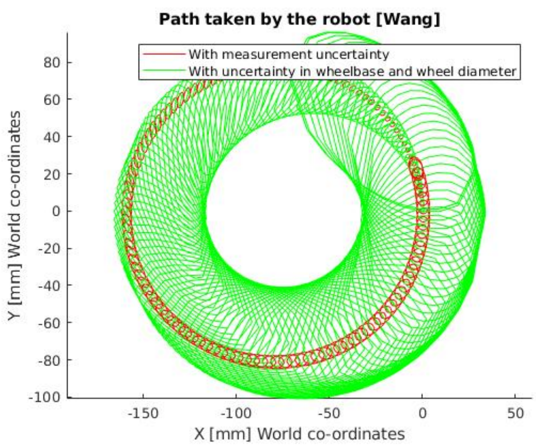

Figure 7. new setup with adjustment to uncertainty in wheel base and wheel diameter

This answers question f) in II A. In figure 7 the same setups are shown after adjustment to the uncertainty in wheel diameter and wheel base. Since we used a wheel diameter that is 1.3 mm smaller than the original setup we use this value as our variance. Similarly in the wheel base there is a difference of 13 mm which is used as the uncertainty in this parameter. This answers question h). The adjustment for this factor was done in the same manner as in the previous part with the exception of (9) as U_k and the corresponding jacobian matrix. This answers question g).

We then redid the entire process using the second dataset. Figure 8 shows the result in form of the different state variables.

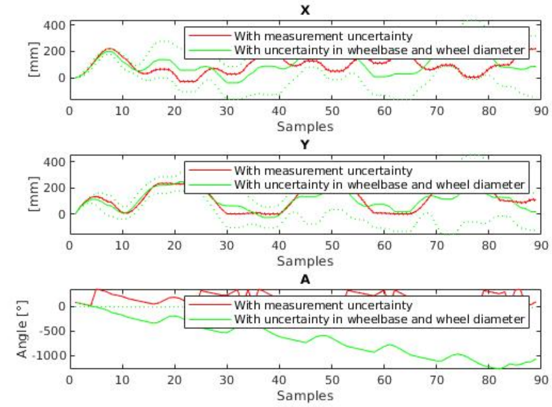

Figure 8. State variables when using a dataset where the robot is moving in a non-circular pattern

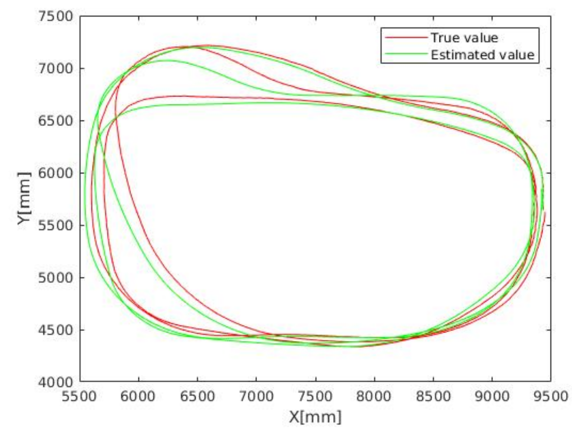

Figure 9. Estimated path and true path

B. Results from part 2

In this part of the assignment we are analysing the data from the snowhite robot. Using the velocity and steering angle we computed the position update according to (7) with (10) as our U_k. This gave us the position estimate showed in figure 9.

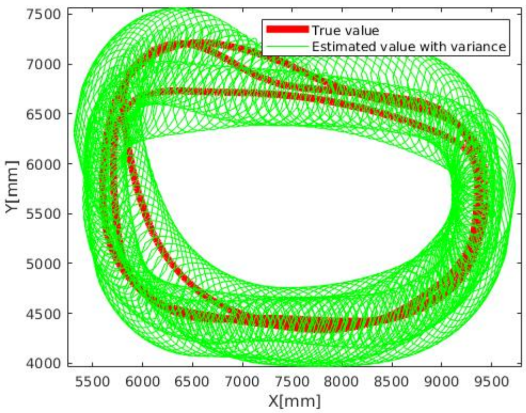

Figure 10. Estimated path and uncertainty

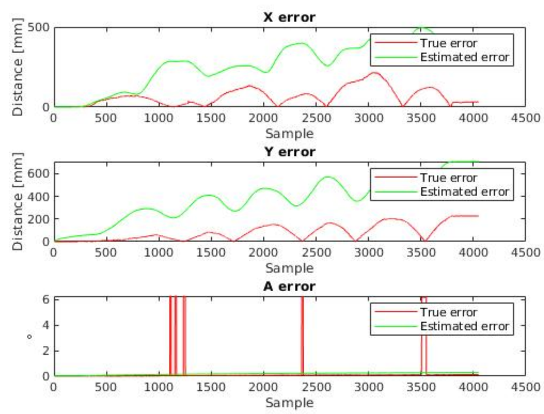

Figure 11. Estimated error and true error

To estimate the uncertainty of v and a we computed their variance. These gave us the values $\sigma_v^2 = 4.4 * 10^3$ and $\sigma_a^2 = 0.1$. For the sample period T we used an arbitrary value of 0.01 ms as the uncertainty. This answers question i) in II A. The position and the uncertainty can be viewed in figure 10. We then computed the true error and estimated error to see how well our uncertainty estimation was. This is shown in figure 11. The total distance traveled was estmimated to be 32 meters during a time period of 202.5 seconds. This answers question j).

IV. DISCUSSION

This exercise was a good introduction to odometry and dead reckoning for different kinds of robots with introduced errors. All of the questions posed for the exercise has been answered in the result section. Parts we think can be calibrated are how precise we should've been with the uncertainties in the Snowhite part. A small remark is the choice of color in our plots could've been chosen more carefully, as we know green and red can be hard to read for people with vision disabilities.
Our method of calculating σ_v^2 gave us an answer we think is quite big. A reason for this could be that the robot starts collecting data in a stand still and also ends in a stand still. That means that the first and last couple of samples will be zero. Therefor the variance between moving and standing still will be quite large and that creates a large σ_v^2.

As for future installements of this exercise we think that the exercise could somehow have been more focused, in short we think that atleast the Khepera part could've been shortened a bit.

[1] C Ming Wang. Location estimation and uncertainty analysis for mobile robots. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pages 1231–1235. IEEE, 1988.