Proceedings of the 2003 IEEE
International Conference on Robotics,Intelligent Systems and Signal Processing
Changsha, China - October 2003

Odometry Based Pose Determination and Errors Measurement for a Mobile Robot with Two Steerable Drive Wheels

Qinghao Meng

School of Electrical Engineering and Automation
Tianjin University (P. R. China)
qh_meng@yahoo.com

Rainer Bischoff

Institute of Measurement Science
Bundeswehr University Munich (Germany)
Rainer.Bischoff@unibw-muenchen.de

Abstract

A trigonometric method is proposed to calculate the pose (position and orientation) of a mobile robot, possessing a locomotion platform with a unique combination of two steerable and driven wheels and two caster wheels. The model of the locomotion platform is derived to deliver timely and accurate odometer information from the measured drive wheel revolutions and the steering angle measurements. Non-systematic errors, mainly slippage, are detected by a new computation method based on the ratios between the two drive wheels' incremental distances. Real-world experiments demonstrate the correctness of the proposed model and validate the ability of the algorithm to detect non-systematic errors.

1. Introduction

Dead reckoning based on odometric measurements is the most widely used navigation method for mobile robots. Although this integrative method norm-ally can only provide good short-term accuracy due to inevitably error accumulation, most researchers agree that odo-metry is an important part of mobile robot navigation [Borenstein et al. 1996]. Even with the advent of powerful range sensing and inertia-based methods relying on expensive laser scanners or gyros, odometers still constitute cheap means of calculating pose. Odometry-based pose estimation requires a model of the wheel configuration; different configurations require different calculation methods. One of the most used wheel configuration is the differential drive, in which incremental encoders are mounted onto two drive motors to count the wheel revolutions. The mobile robot's pose can be computed by using simple geometric equations [Crowley et al. 1992], [Klarer 1988].

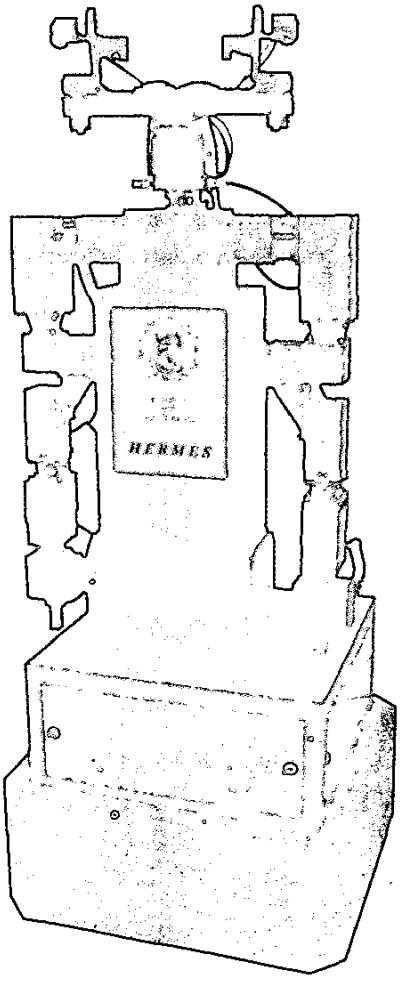


Figure 1: Mobile Service Robot HERMES with two steerable drive wheels

Another widely used mobility configuration is a tricycle drive with a single steered and driven front wheel and two passive rear wheels (or vice versa) [Everett 1995]. The pose calculation method for the tricycle drive is similar to the Ackerman steering [Byrne. et al. 1992], which is normally used for automobiles. Some other relatively complicated wheel mobility configurations include the synchro drive [Fisher et al. 1994], omnidirectional drives [Holland 1983] and multi-degrees-of-freedom (MDOF) vehicles [Borenstein 1995].

A MDOF platform has multiple driven and steered wheels, and different designs are possible. For example, the 4-degrees-of-freedom vehicle HERMIES-III [Reister et al. 1993] has two powered wheels that are also individually steered. The robot *HERMES*, which has been used for the studies described in this paper, has a mobility configuration similar to HERMIES-III, with the difference that the two steerable drive wheels of *HERMES* are situated in the front/back sides and not in left/right sides. In addition, the two drive wheels of *HERMES* are not aligned in the same line (the central line of the robot). To make the control easier and produce less slippage, the steering angles of both drive wheels of *HERMES* are always given the same value.

In this paper, we first introduce the locomotion system of the *HERMES* robot. Then the process of odometry based pose calculation is introduced in section 3. In section 4, experiments are presented to show the odometry based pose determination results. In section 5, the idea to measure the non-system errors mainly caused by slippage is discussed. Conclusions are placed at the end.

2. The locomotion system of *HERMES* robot

HERMES has two steerable drive wheels and two castor wheels. The propulsion system consists of two 500 W motor modules, speeding the 250 kg robot up to 2 m/s. Each motor can be controlled in position, velocity or pulse width, allowing for a wide range of different driving modes. For example, while the front wheel is used for propulsion, the back wheel could be used to measure traveled distances without the danger of slippage, or one wheel could be velocity-controlled while the other just supports propulsion with a pulse width that is small enough to prevent slippage. The two steering motors enable the robot to move in any direction, including diagonal or sideways motions. The steering angle of each drive wheel can be adjusted from -90° (clockwise direction) to +90° (anticlockwise direction). It should be mentioned that the circumferential central lines of two drive wheels are NOT aligned in the central line of the robot; there is a 4 cm distance (cf. Fig. 2).

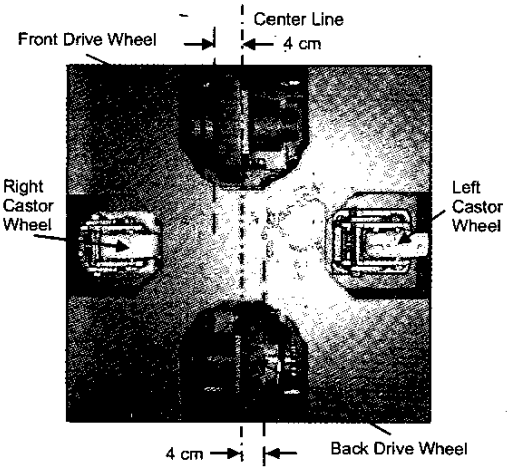


Figure 2: HERMES' undercarriage (bottom view)

Fig. 3 gives an overview over all possible motion directions and corresponding steering angles, where δ_f and δ_r stand for front and rear steering angles. In addition to the mentioned configurations, the robot could also independently steer the front or back wheel while keeping the other wheel straight, emulating car-like steering or fork lifter-like steering modes. Furthermore, the robot's body can be (manually) turned 90° atop the undercarriage for differential driving experiments.

3. The odometry based pose calculation model

The inputs of the model are traveled distances and steering angles of both steerable drive wheels, and the outputs are the pose, i.e., the *position* and *orientation* of the robot. Here the position means the x and y coordinates of the middle point CP in the central line of the robot in the global coordinate system X-O-Y; and the orientation is the angle θ between the central line and the positive direction of the horizontal axis X of the global coordinate system (cf. Fig. 4).

3.1 Calculation of global coordinates of two steerable drive wheels

The coordinates of front and rear drive wheels, represented by the ideal contact points between the circumferential central lines of the two drive wheels and the floor, and denoted by (x_f, y_f) and (x_r, y_r), respectively (cf. Fig. 4), can be calculated as follows:

$$\begin{cases} x_f(k) = x_f(k-1) + d_f(k/k-1)\cdot\cos[\theta(k-1)+\delta_f(k-1)] \\ y_f(k) = y_f(k-1) + d_f(k/k-1)\cdot\sin[\theta(k-1)+\delta_f(k-1)] \end{cases} \quad (1),$$

$$\begin{cases} x_r(k) = x_r(k-1) + d_r(k/k-1)\cdot\cos[\theta(k-1)+\delta_r(k-1)] \\ y_r(k) = y_r(k-1) + d_r(k/k-1)\cdot\sin[\theta(k-1)+\delta_r(k-1)] \end{cases} \quad (2),$$

where θ represents the orientation of the robot; k-1 and k are two successive sampling times; $d_f(k/k-1)$ and $d_r(k/k-1)$ are the incremental displacements from time (k-1) to k corresponding to the front and rear drive

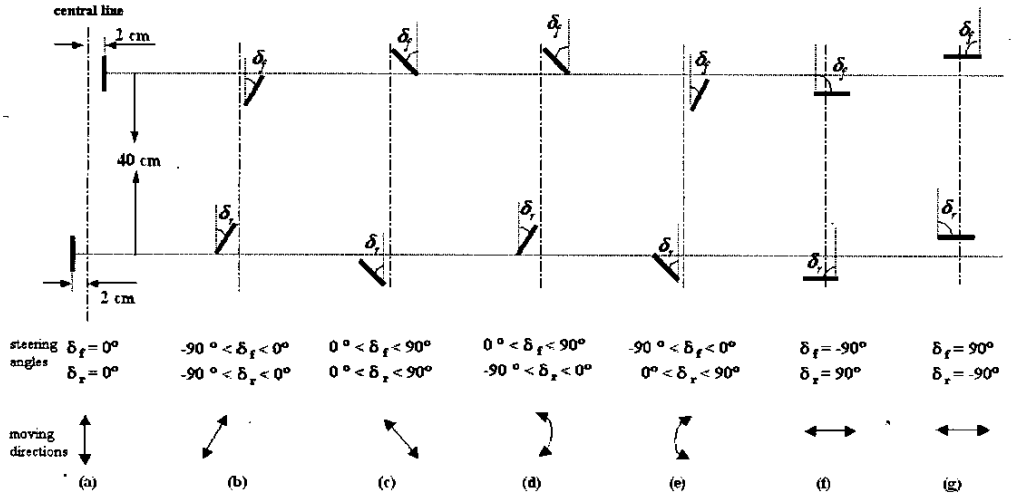


Figure 3: All possible motion directions and corresponding steering angles

wheels, respectively; the start values ($x_f(0), y_f(0)$), ($x_b(0), y_b(0)$) and $\theta(0)$ are supposed to be known based on the global coordinate system, or the pose computation will be relative to an arbitrary starting pose.

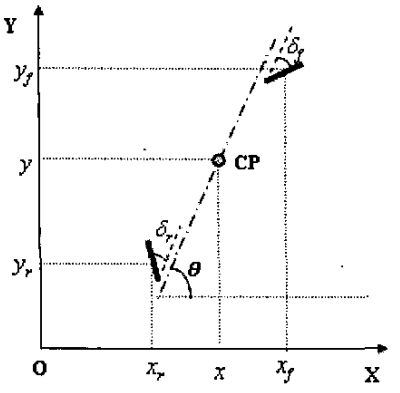


Fig.3 Global coordinate system

When the robot moves along straight lines (cf. Fig. 3 (a), (b), (c), (f) and (g)), the incremental displacements of the two drive wheels are calculated as follows:

$$d_f(k/k-1) = d_f(k) - d_f(k-1) \quad \text{(3-a)},$$

$$d_r(k/k-1) = d_r(k) - d_r(k-1) \quad \text{(3-b)},$$

where $d_f(k)$ and $d_f(k-1)$ are the displacements of the front wheel measured at the sampling time k and (k-1), respectively; in the same way, $d_r(k)$ and $d_r(k-1)$ are the displacements for the rear wheel.

To get more accurate result, when the robot rotates (cf. Fig. 3 (d) and (e)) and the interval between two sampling times is relatively long, the two incremental displacements along arcs, i.e., $d_f(k/k-1)$ and $d_r(k/k-1)$ in equations (1) and (2), should be replaced by two incremental displacements along chords denoted by $d_{f\_chord}(k/k-1)$ and $d_{r\_chord}(k/k-1)$ (cf. Fig. 5). $d_{f\_chord}(k/k-1)$ and $d_{r\_chord}(k/k-1)$ can be calculated as follows:

1) Coordinates of momentary rotary center of two drive wheels:

The global coordinates of the momentary rotary center (see point MP in Fig. 5) of the two drive wheels, denoted by (x_{mp}, y_{mp}), can be calculated as follows:

$$\begin{cases} x_{mp}(k) = \dfrac{\tan(\delta_{r\_a}(k))\tan(\delta_{f\_a}(k))(y_f(k)-y_r(k)) + x_f(k)\tan(\delta_{r\_a}(k)) - x_r(k)\tan(\delta_{f\_a}(k))}{\tan(\delta_{r\_a}(k)) - \tan(\delta_{f\_a}(k))} \\ y_{mp}(k) = \dfrac{x_f(k) - x_r(k) + \tan(\delta_{f\_a}(k))y_f(k) - \tan(\delta_{r\_a}(k))y_r(k)}{\tan(\delta_{f\_a}(k)) - \tan(\delta_{r\_a}(k))} \end{cases}$$

where $\delta_{f\_a}(k) = \theta(k) + \delta_f(k)$ and $\delta_{r\_a}(k) = \theta(k) + \delta_r(k)$ are the absolute steering angles of the front and rear drive wheels, respectively.

2) Rotary radii of two drive wheels:

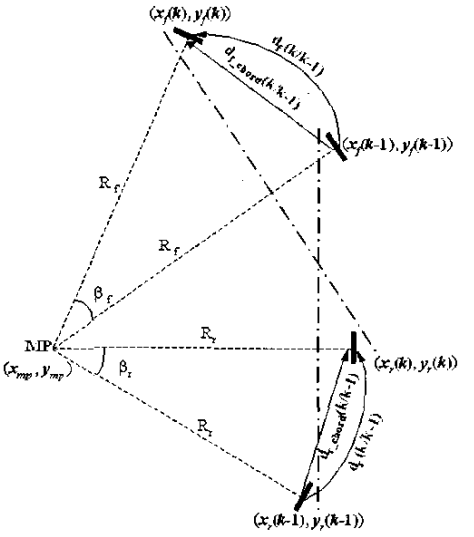


Figure 5: Calculation of incremental displacements along chords

The two rotary radii, denoted by $R_f(k)$ and $R_r(k)$, respectively, can be expressed as follows:

$$R_f(k) = \sqrt{(x_f(k) - x_{mp}(k))^2 + (y_f(k) - y_{mp}(k))^2} \ ,$$

$$R_r(k) = \sqrt{(x_r(k) - x_{mp}(k))^2 + (y_r(k) - y_{mp}(k))^2} \ .$$

3) Rotary angles of two drive wheels:

The two rotary angles, denoted by $\beta_f(k)$ and $\beta_r(k)$ (unit is radian), respectively, can be calculated as follows:

$$\beta_f(k) = \frac{d_f(k/k-1)}{R_f(k)} \ , \qquad \beta_r(k) = \frac{d_r(k/k-1)}{R_r(k)} \ .$$

4) Length of two chords:

The lengths of the two chords can be derived:

$$d_{f\_chord}(k/k-1) = R_f(k)\sqrt{2 - 2\cos(\beta_f(k))} \ ,$$

$$d_{r\_chord}(k/k-1) = R_r(k)\sqrt{2 - 2\cos(\beta_r(k))} \ .$$

3.2 Calculation of the global pose of the robot

The coordinates of the front and rear drive wheels derived by equations (1) and (2) cannot be directly used to calculate the position and orientation of the robot. The reason is that the position of the ideal contact point between the drive wheel and floor is changing relatively to the robot when each drive wheel changes its steering angle (see the different positions of ideal contact points expressed by small circle dots and their corresponding steering angles in Fig. 6). The positions of both drive wheels' ideal contact points are changing along two arcs (the radii of the two arcs are 2 cm, and the centers of the two arcs are denoted by FP and RP, respectively).

To calculate the global pose of the robot, first the coordinates of FP and RP (i.e., the two fixed points relatively

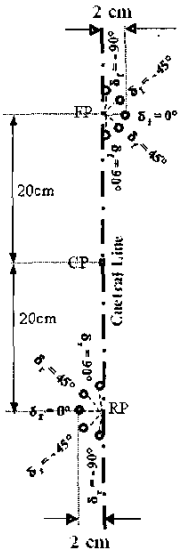


Figure 6: Pose calculation

to the robot) at k^{th} sampling time, denoted by $(x_{FP}(k), y_{FP}(k))$ and $(x_{RP}(k), y_{RP}(k))$, respectively, should be determined as follows:

$$\begin{cases} x_{FP}(k) = x_f(k) - r_{FP} \cdot \sin \delta_{f\_a}(k) \\ y_{FP}(k) = y_f(k) + r_{FP} \cdot \cos \delta_{f\_a}(k) \end{cases} \quad (4),$$

$$\begin{cases} x_{RP}(k) = x_r(k) + r_{RP} \cdot \sin \delta_{r\_a}(k) \\ y_{RP}(k) = y_r(k) - r_{RP} \cdot \cos \delta_{r\_a}(k) \end{cases} \quad (5),$$

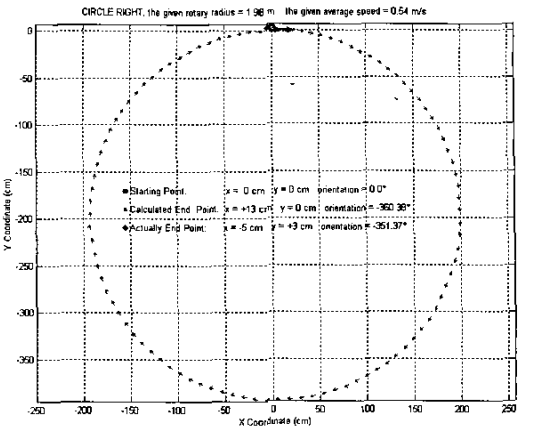
where $r_{FP} = r_{RP} = 2$ cm are the radii.

Eventually, the global position and orientation of *HERMES* is calculated as follows:

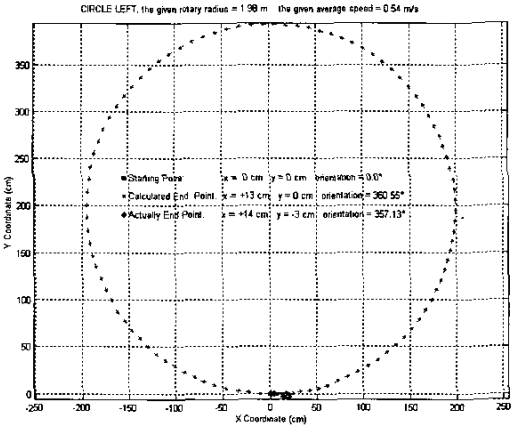
$$\begin{cases} x(k) = \dfrac{x_{FP}(k) + x_{RP}(k)}{2} \\ y(k) = \dfrac{y_{FP}(k) + y_{RP}(k)}{2} \\ \theta(k) = \arctan\left(\dfrac{y_{FP}(k) - y_{RP}(k)}{x_{FP}(k) - x_{RP}(k)}\right) \end{cases} \quad (6)$$

4. Experiments

To verify the correctness of pose calculation model derived in section 3, experiments have been carried out in an indoor open area of the Institute of Measurement Science, Bundeswehr University Munich. The material of the floorboard is PVC. In total, 72 motion experiments have been carried out. The robot moved along different trajectories (circle and rectangle), in different directions (clockwise and anticlockwise), at different speeds (0.26 m/s, 0.40 m/s, 0.54 m/s and 0.66 m/s) and rotary radii (for the circle trajectories, the radii included 0.86 m, 1.19 m, 1.49 m, 1.75 m and 1.98 m; for the rectangle motion; the trajectories at the four corner are circular arcs and the radii include 0.46 m, 0.86 m, 1.19 m and 1.49 m). The absolute position and orientation errors



(a) clockwise



(b) anticlockwise

Figure 7: Calculated circle motion trajectories and actual end point

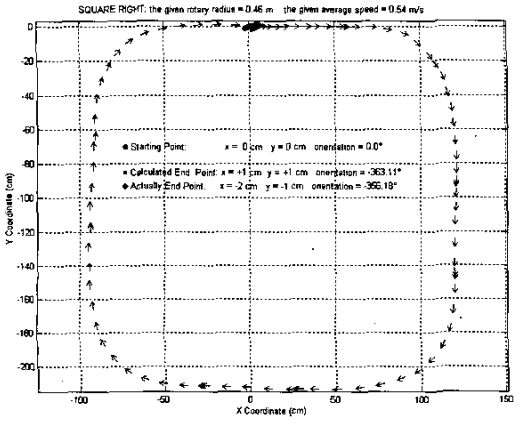
Motion trajectory type (total test number)	Maximal absolute position error (cm)	Minimal absolute position error (cm)	Maximal absolute orientation error (°)	Minimal absolute orientation error (°)
Clockwise circle (20)	25	3.6	9.8	6.0
Anticlockwise circle (20)	24.2	2.2	4.6	0.7
Clockwise rectangle (16)	23.2	3.6	10.8	4.3
Anticlockwise rectangle (16)	18.0	1.0	4.1	0.2

Table 1: Absolute position and orientation errors

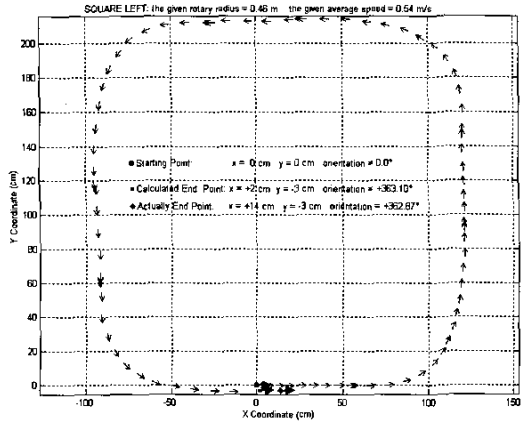
are shown in Table 1. Four examples from the test set are also illustrated in Figures 7 and 8. The model has also been tested in relatively long distance test runs where the robot had been controlled by a wireless keyboard to move along four corridors (the four corridors constitute a 30-meter-long and 20-meter-wide rectangle) in the lab area. The starting pose was set to be (0 cm, 0 cm, 0°). To turn around corners (i.e., junctions between

two corridors), the steering angles of two drive wheels were changed manually by hitting dedicated cursor keys which modified the angles from 0° to approximate 10°. It should be noted that, although the steering angles had been reset to 0° in the beginning and after each turn, the actual motion trajectory of the robot was not a straight line. Instead the robot tended to move to one side or the other, making it necessary to adjust the heading direction manually by hitting the cursor keys once in a while ("human controller" for having the robot follow the corridor).

These experiments show that odometry alone is not sufficient to determine a robot's pose, e.g., for map building purposes; complementary sensor information is required to correct the systematic and non-systematic errors of the odometry based pose estimation before it can accumulate intolerable values. Odometry data could be improved, e.g., by other methods with external sensors information, such as landmark-based [Marsland et al. 2001], model-based [Reina et al. 2000] or cooperative localization methods [Kurazume et al. 2000].



(a) clockwise



(b) anticlockwise

Figure 8: Calculated rectangle motion trajectories and actual end point

5. Error measurement

From the experiments described in section 4 we found that the odometry results are generally acceptable, except for cases where obvious wheel slippage occurred. Wheel slippage is the main non-systematic error source that let drastically decrease the accuracy of odometry calculation. This section is thus devoted to wheel slippage measurement. The idea of the proposed method is to track the ratio of the two drive wheels' displacement increment, denoted by $\Delta d_b/\Delta d_f$, after a suitable period of time. $\Delta d_b/\Delta d_f$ is expressed as follows:

$$\Delta d_b/\Delta d_f = (d_b(t+\Delta t) - db(t))/(d_f(t+\Delta t) - d_f(t)),$$

where Δt is the sampling time.

First, several tests were carried out to determine if the moving speed of the robot has an obvious impact on the ratio of the two drive wheels' displacement (denoted by d_b/d_f). We also tested d_b/d_f by other steering angles, the results are that speed (from 0.26m/s to 0.66m/s) has no obvious effect on the value of d_b/d_f. So, in the following tests only a speed of 0.54m/s is used.

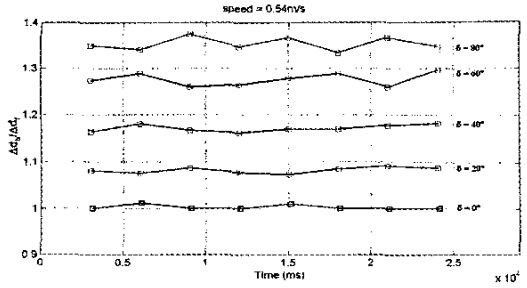
	0°	20°	40°	60°	80°
Position error (cm)	8.6	19.4	17.9	27.7	28.9
Orientation error (°)	14	8.9	11.2	14.1	15.8

Table 2: Absolute position and orientation errors of different steering angles (given speed = 0.54m/s)

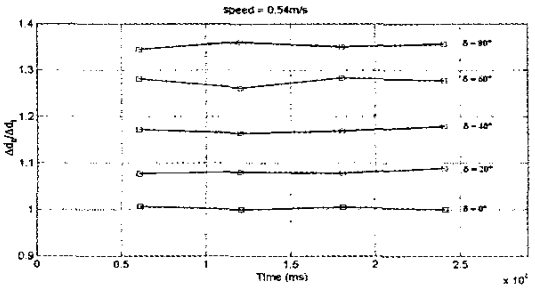
Different steering angles have different ratio values. Through experiments we got the values of $\Delta d_b/\Delta d_f$ corresponding to the steering angles 0°, 20°, 40°, 60° and 80°, which are depicted in Fig. 9. The associated position and orientation errors are presented in Table 2. From Fig. 9 (a) to (c) it can be derived that the bigger the sampling time, the smaller the fluctuation of $\Delta d_b/\Delta d_f$.

Here, we want to point out that there are no obvious (macroscopic) slippages found during the experiments depicted in Fig. 9, but actually from Fig. 9 (a) and (b) we can find that the value of $\Delta d_b/\Delta d_f$ changes somewhat. Such experiments have been carried out several times, but the results are very similar. [**Note**: We suppose that such phenomenon probably comes from dynamics aspects, which exceeds the discussion range of this paper. Here only kinematics aspects are considered].

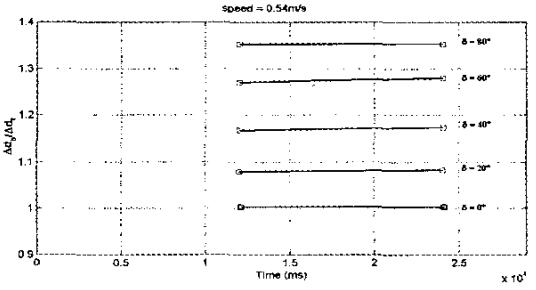
One of the experiments results with obvious slippages are illustrated in Fig. 10. The experiments are implemented in a slightly uneven area. Based on the results depicted in Fig. 10, if the absolute value of the difference between the calculated $\Delta d_b/\Delta d_f$ (calculated once every 12 seconds) and the nominal value (different steering angles have different nominal $\Delta d_b/\Delta d_f$, cf. Fig. 9



(a) The result of $\Delta d_b / \Delta d_f$ (time interval Δt =3s)



(b) The result of $\Delta d_b / \Delta d_f$ (time interval Δt =6s)



(c) The result of $\Delta d_b / \Delta d_f$ (time interval Δt =12s)

Figure 9: Experimental results of $\Delta d_b / \Delta d_f$ of different steering angles

(c)) is bigger than 0.05, the system thinks obvious slippage happened.

6. Conclusions

How to increase the accuracy of an odometry-based pose estimation of a mobile robot with two steerable drive wheels has been carefully studied from a kinematics point of view. A detailed trigonometry-based model has been derived and its accuracy has been verified by numerous experiments on relatively even ground. The idea using the ratio of two drive wheels' incremental displacements is proved to be valid to detect non-systematic errors mainly caused by slippage, which in turn can be used to evaluate the accuracy of the odometry results and to take appropriate counter-measures, e.g., to re-initiate the odometer based on other sensor modalities.

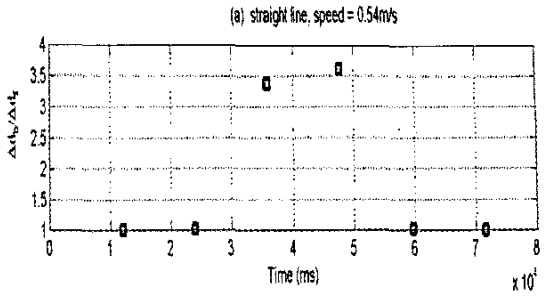


Figure 10: Detect slippages

Acknowledgement

The work for the paper was performed at the Institute of Measurement Science, Bundeswehr University Munich. The authors would like to thank Prof. Dr. Volker Graefe, the head of this institute. Prof. Graefe has provided all the necessary support to the research work of this paper. He also made constructive suggestion about this paper.

References

[1] Borenstein, J.: Control and kinematic design for multi-degree-of-freedom mobile robots with compliant linkage, *IEEE Transactions on Robotics and Automation*, Vol. 11, No. 1, Feb., pp. 21-35, 1995.

[2] Borenstein J. (eds.), Everett H. R., and Feng L.: Where am I? – Sensors and methods for mobile robot positioning, April 1996.

[3] Byrne, R.H., Klarer, P.R., and Pletta, J.B.: Techniques for autonomous navigation, *Sandia Report SAND92-0457*, Sandia National Lab:, Albuquerque, NM, 1992.

[4] Crowley, J.L. and Reignier, P.: Asynchronous control of rotation and translation for a robot vehicle, *Robotics and Autonomous Systems*, Vol. 10, 1992, pp. 243-251.

[5] Everett, H. R.: Sensors for mobile robots: theory and application, A K Peters, Ltd., Wellesley, MA, 1995.

[6] Fisher, D., Holland, J.M., and Kennedy, K.F.: K3A marks third generation synchro-drive, *American Nuclear Society Winter Meeting, Proceedings of Robotics and Remote Systems*, New Orleans, LA, June, 1994.

[7] Holland, J.M.: Basic robotics concepts, Howard W. Sams, Macmillan, Inc., Indianapolis, IN, 1983.

[8] Klarer, P.R.: Simple 2-D navigation for wheeled vehicles, *Sandia Report SAND88-0540, Sandia National Laboratories*, Albuquerque, NM, April, 1988.

[9] Kurazume Ryo and Hirose Shigeo: An experimental study of a cooperative positioning system, *Autonomous Robot*, Vol. 8, No. 1, Jan.2000, pp43-52.

[10] Marsland S., Nehmzow U. and Duckett T.: Learning to select distinctive landmarks for mobile robot navigation, *Robotics and Autonomous Systems*, Vol.37, No.4, Dec. 2001, pp.241-260.

[11] Reina A., Gonzalez J.: A two-stage mobile robot localization method by overlapping segment-based maps, *Robotics and Autonomous Systems*, Vol.31, No.4, Jun. 2000, pp.213-226.

[12] Reister, D.B. and Unseren, M.A.: Position and constraint force control of a vehicle with two or more steerable drive wheels, *IEEE Trans. on Robotics and Automation*. Vol. 9, No. 6, December, pp. 723-731, 1993.