Mobile Robots
Kinematics and models

Ole Ravn
Automation and Control
DTU Elektro

Outline

- Kinematics
- Odometry model
- Motor model
- Line sensor model
- UMBmark
- Todays and tomorrows exercise

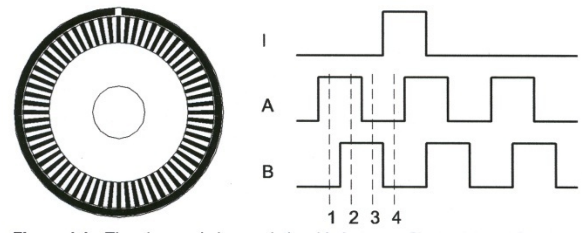
Kinematics

- Kinematics is the study of motion without regard to the forces which cause it
- Where are we and where are we going
- Within kinematics one studies the position, velocity and acceleration etc.

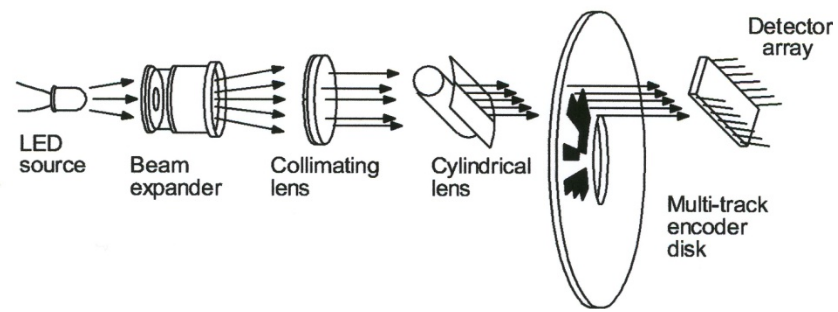
Encoders

- Encoders are used to measure the rotation of the wheels

- Using two channels enables detection of direction and four times as many pulses can be used.



State	Ch A	Ch B
S_1	High	Low
S_2	High	High
S_3	Low	High
S_4	Low	Low



Odometry model

From encoder signals to determine where we are.

The incremental displacement of each wheel:

$\Delta U_{L/R}(i) = c_m N_{L/R}(i)$

where

$c_m = \pi D / nC$

D=6.5 cm, n=1, C=2000

The displacement of the robot centre point

$\Delta U(i) = (\Delta U_R + \Delta U_L)/2$

Incremental change in orientation

$\Delta\theta(i) = (\Delta U_R - \Delta U_L)/b$

b is the wheel distance

New pose of the robot is

$$x(i) = x(i-1) + \Delta U(i)\cos(\theta(i))$$

$$y(i) = y(i-1) + \Delta U(i)\sin(\theta(i))$$

$$\theta(i) = \theta(i-1) + \Delta\theta(i)$$

UMBmark

Where am I: p. 14, pp. 19-20, and pp.130-142

Odometry errors

Systematic errors

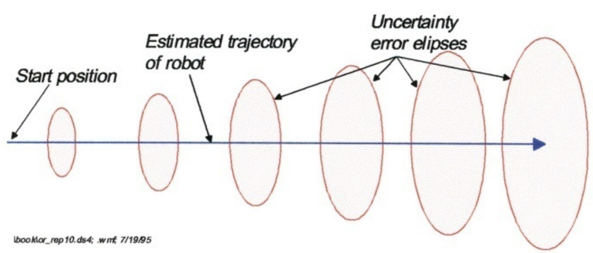
- Unequal wheel diameters
- Average of actual wheel diameters differ from nominal wheel diameter
- Actual wheelbase differs from nominal
- Misalignment of wheels
- Finite encoder resolution
- Finite encoder sampling rate

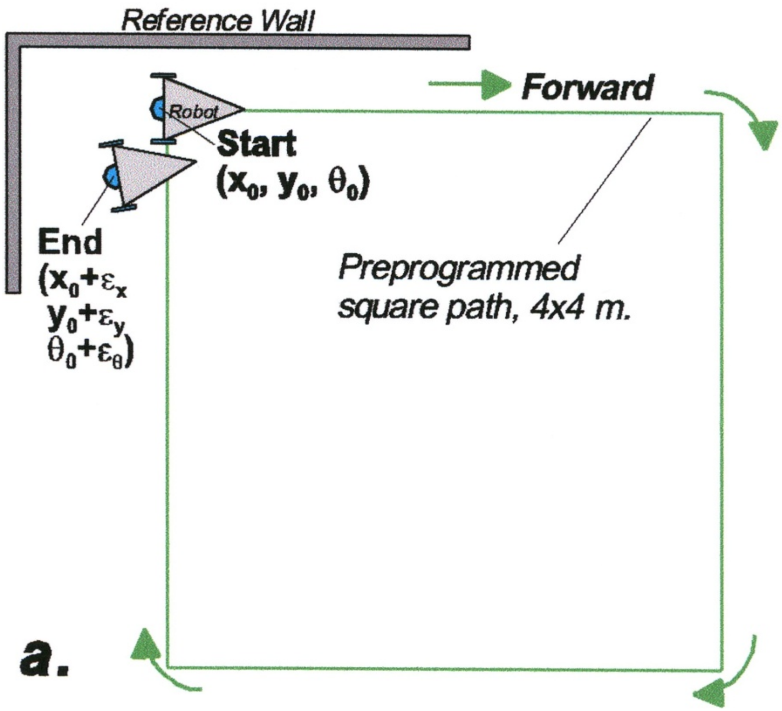
Odometry errors

Non-systematic

- Uneven floors
- Unexpected object on the floor
- Wheel slippage
- slippery floors
- overacceleration
- skidding
- interaction with external bodies
- castor wheels
- non-point wheel contact with floor

UMBMark


Start position
Estimated trajectory of robot
Uncertainty error elipses


Reference Wall
Robot
Forward
Start
(x_0, y_0, θ_0)
End
($x_0+\varepsilon_x$
$y_0+\varepsilon_y$
$\theta_0+\varepsilon_\theta$)
Preprogrammed square path, 4x4 m.
a.

Errors

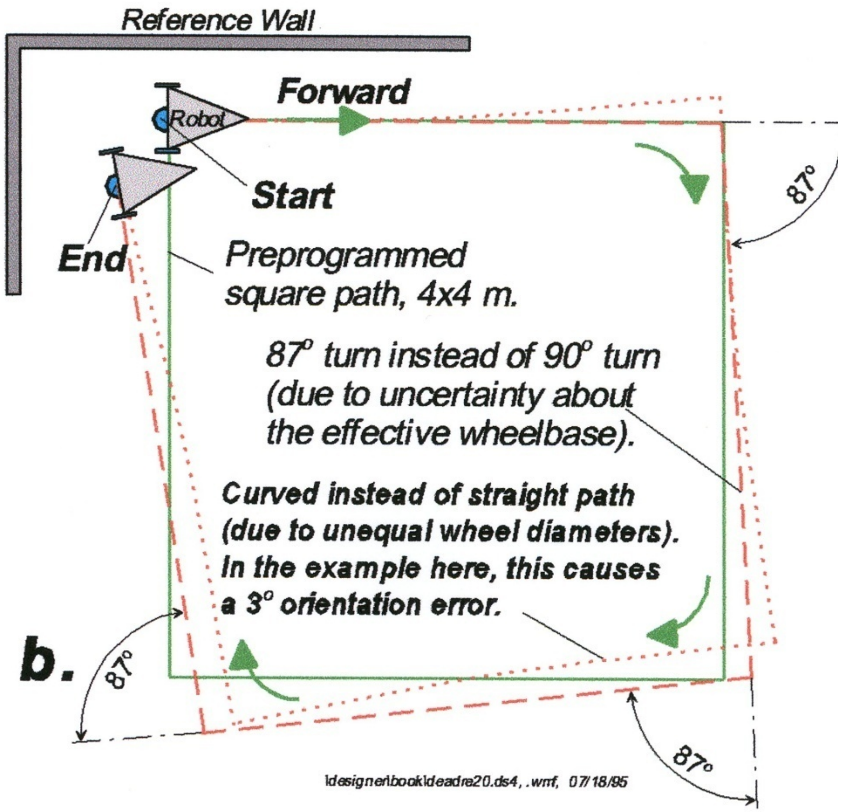
- Wheel base

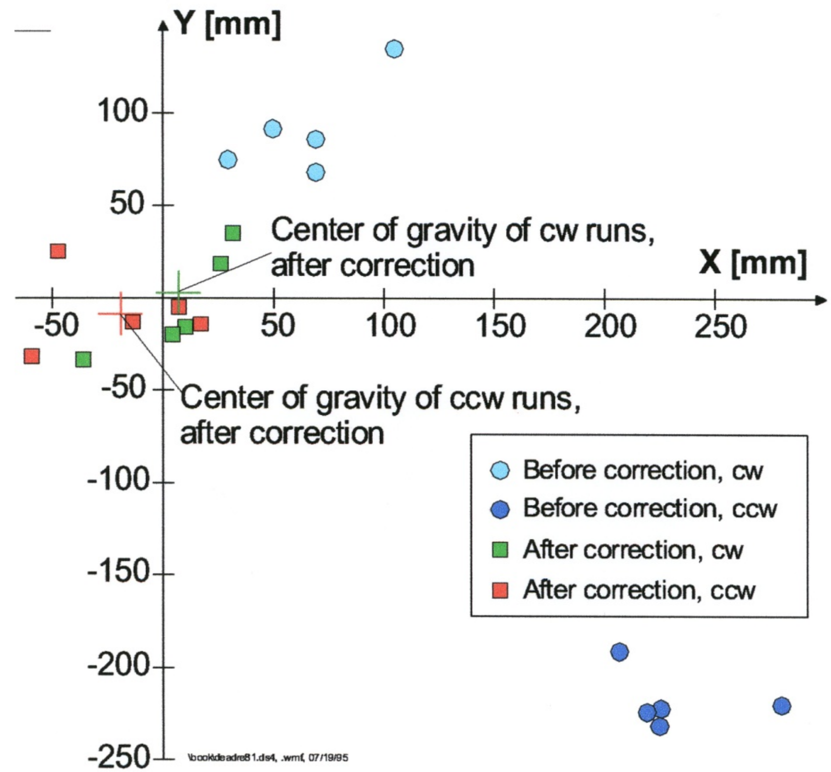
$$E_b = \frac{b_{actual}}{b_{nomi}}$$

- Wheel diameter

$$E_d = \frac{D_R}{D_L}$$

UMBMark


Reference Wall
Forward
Robot
Start
End
Preprogrammed square path, 4x4 m.
87° turn instead of 90° turn (due to uncertainty about the effective wheelbase).
Curved instead of straight path (due to unequal wheel diameters). In the example here, this causes a 3° orientation error.
b.
87°
87°
87°
\designer\book\deadre20.ds4, .wmf, 07/18/95


Y [mm]
X [mm]
100
50
-50
-100
-150
-200
-250
-50
50
100
150
200
250
Center of gravity of cw runs, after correction
Center of gravity of ccw runs, after correction
Before correction, cw
Before correction, ccw
After correction, cw
After correction, ccw

Calculating the Corrections

$$\alpha = \frac{x_{c,g,cw} + x_{c,g,ccw}}{-4L}\frac{180^o}{\pi}$$

$$\beta = \frac{x_{c,g,cw} - x_{c,g,ccw}}{-4L}\frac{180^o}{\pi}$$

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

$$E_d = \frac{R + b/2}{R - b/2}$$

$$E_b = \frac{90^o - \alpha}{90^o}$$

Todays exercise

1. Make a Matlab script for the UMBmark
2. Test the script
3. Use the uncalibrated simulator - rhdconfig.odo.xml
4. Do UMBMark
5. Work out calibration constants using (Excel or) Matlab
6. Check when we put in the calibration constants in the simulator

Documentation

- MATLAB figures on the web
- If you use jpg make sure that the ‘quality’ is 100 % or you get bad plots.
- Generally png (or gif) works well.

```
print -rxxx -dpng <name>
```

- Use the following commands the produce a (better) jpg file:

```
print -rxxx -djpg100 <name>
```

- For best results the size of the plot should be adjusted using the -rxxx (72) option to print, NOT later in the html code.