Autonomous Pool Cleaning

Self Localization and Autonomous Navigation for Cleaning

M. SIMONCELLI, G. ZUNINO and H.I. CHRISTENSEN
CAS-Centre of Autonomous Systems
Royal Institute of Technology
SE-100 44 Stockholm, Sweden

K. LANGE
WEDA
Weda Poolcleaner AB
BOX621, 151 27 Södertälje, Sweden

Abstract. Cleaning is a major problem associated with pools. Since the manual cleaning is tedious and boring there is an interest in automating the task. This paper presents methods for autonomous localization and navigation for a pool cleaner to enable full coverage of pools. Path following cannot be ensured through use of internal position estimation methods alone; therefore sensing is needed. Sensor based estimation enable automatic correction of slippage. For this application we use ultrasonic sonars. Based on an analysis of the overall task and performance of the system a strategy for cleaning/navigation is developed. For the automatic localization a Kalman filtering technique is proposed: the Kalman filter uses sonar measurements and a dynamic model of the robot to provide estimates of the pose of the pool cleaner. Using this localization method we derive an optimal control strategy for traversal of a pool. The system has been implemented and successfully tested on the "WEDA B400" pool cleaner.

Keywords: mobile robot, underwater environment, sonar system, state estimation, uncertainty

1. Introduction

A major problem associated with swimming pools is cleaning. All commercial pools have to be cleaned daily. The water quality must be maintained within certain bounds. In addition various algae and other deposits attach to the walls and floor of the pool. These foreign products must be removed. In private pools this is often carried out by small cleaning 'robots' that traverse the pool at random. In a commercial pool cleaning must cover the entire floor and all walls. Random movements cannot ensure full coverage, thus a more structured approach must be deployed. In most places the cleaning is carried out manually. The cleaning is carried out using a brush attached to an underwater pump, that has an associated filter bag on the exhaust pipe, which catches dirt, and deposits. This is similar to an underwater vacuum cleaner.

Manual cleaning is tedious and boring, and there is thus an interest in automating the task.

An alternative to manual cleaning is automatic traversal of the pool. A number of different companies produce automatic pool cleaners. WEDA Poolcleaners in Södertälje, south of Stockholm is one of the major producers of such automatic pool cleaners. One of the products they produce is the B400 pool cleaner shown in Figure 1.

Figure 1 about here

The B400 pool cleaner has an on-board pump with an associated filter bag through which water is passed. A set of motorized brushes picks up algae and smut, which then are collected by the filter bag. To ensure cleaning of the entire swimming pool the vehicle performs a lawn mover type of sweep of the pool. The cleaner is driven by a pair of belts. At the front and the back of the cleaner there are bumpers that detect collisions with walls and other objects. Upon activation of a bumper the drive motors are reversed; for one of the bumpers a slight delay is introduced for one belt to allow generation of a zig-zag traversal pattern. Such a strategy is well suited for handling of rectangular pools.

The WEDA B400 has the following dimensions: $30 \times 48 \times 90[cm]$ $(H \times W \times L)$. Its weight in air is 35 kg, and about 12 kg in water. It can move at a speed of 0.3 m/s. The B400 is similar to a number of other commercial cleaners available.

A problem with the pool cleaner is that it might have drift due to slippage and poor contact with the floor or walls. The original B400 does not have any odometric or other external sensory reference. There is thus no mechanism for compensation for slippage. It is consequently difficult to guarantee full coverage of a pool. Another problem is that the B400 design only is suited for traversal of rectangular pools. For modern adventure type pools there are currently no method available for automatic cleaning.

Outline of the paper. In this paper we study methods for localization and navigation for a pool cleaner to enable full coverage of rectangular pools. Initially the overall task and performance of the system is studied; based on a task analysis a strategy for cleaning/navigation is developed (Section 2). The developed strategy exploits a sonar systems. The characteristics of the sonar is presented in Section 3. The strategy requires maintenance of a position estimate for the platform. A method for automatic localization using Kalman filtering is developed (Section 4). Using the localization method is is possible to derive an optimal control strategy for traversal of rectangular pools (Section 5). The system has been implemented in on the B400 (Section 6), a number of experiments are reported to illustrate the improvement in performance

(Section 7). We finally present the summary, our main results and we also discuss future developments for this project (Section 8).

2. Sensory perception

The purpose is to create an autonomous system capable of cleaning the bottom of a swimming pool without requiring either a guide to follow or tele-operator control. The objective is a machine which cleans the pool in the smallest possible time and in a satisfactory way (e.g. hygienic speaking), and with a limitation in price, as the final aim is a commercial product. The solution to this problem is through design of a control strategy that allow traversal of a pool using a “lawn-mover” style cleaning (for rectangular pools).

Figure 2 about here

Figure 2 shows how the rectangular area should be traversed by the pool-cleaner. The robot will proceed initially in the upward $+x$ direction where $x(t + dt) > x(t)$. When the robot reaches the wall, it reverses its run (i.e., $-x$ movement) and follows an oblique path that increase its y position. This will continue until the entire pool is completely covered. This is the simplest algorithm, compatible with the robot structure for covering the bottom of a rectangular pool.

To compensate for eventual drift there is a need for sensing of position and orientation. With sensing it is in principle possible to perform automatic correction of slippage and it is possible to perform more advanced cleaning (Maksarov and Durrant-Whyte, 1995), (Leonard and Durrant-White, 1992).

A variety of different sensors can be used for pose estimation. The position of the vehicle along the major direction of the pool is of limited interest, while the position perpendicular to the direction of motion is essential. In addition the orientation of the vehicle must be recovered. Both entities can be recovered for sensory data related to the distance to the closest wall.

By virtue of their low cost and simplicity, ultrasonic sensors are widely used. Sonars are frequently used in marine applications and has been so for quite some time. A disadvantage of sonar based ranging is that it is a point estimate with no or little spatial context. An alternative might be use of gyroscopes and accelerometers, but the noise and vibrations from the pump and brushes do, however, introduce significant noise into the estimation process. We have thus decided to select sonar based ranging for this application.

Ultrasonic ranging is deployed to measure the y-distance[1] (see Figure 2) since the covered path length and the orientation of the robot can be derived for the robot motion equations, the robot speed, etc. The bumpers (electrical micro switches) allow detection of walls in the direction of motion. The system setup is shown in Figure 3.

Figure 3 about here

3. Sonar System

To measure the distance between the robot and the wall of the pool a NAVMAN 100 sonar is used. This particular system is a simple sonar normally used for marine applications to determine the water depth. It consists of an electronics unit and a transducer. The transducer is composed of a transmitter and a receiver.

The version of transducer used is the 2" In-Hull puck transducer (51 mm diameter); it has an opening angle of 9 degrees and a working frequency of 200 kHz.

The principal characteristics of the device are: estimates depth up to 150 meters, precision of one decimeter between 2-20 meters and 1 meter between 20-150 meters.

The sonar has a serial interface that uses a dedicated protocol (NMEA 0184). This is an uniform interface standard for digital data exchange between electronic devices. The National Marine Electronics Association has developed this standardized data protocol that permits high speed communications between devices.

To ensure safe operation, distances smaller than 3 meters are discarded. For non-parallel operation the data sheet specifies a beam-width of 9 degrees. In practice the sonar delivers robust reading at orientations (θ) up-to 15 degrees. This limits the allowed steering angle to 15 degrees.

Figure 4 about here

4. Localization

A precursor for autonomous control that ensures full coverage of a pool is localization for the vehicle. Localization can be relative or absolute. In this particular application relative positing is not enough and there is thus a need for estimation of the pose in metric coordinates, as shown the in Figure 5.

Figure 5 about here

[1] Distance to the wall parallel to the direction of travel

Our approach is based on the use of sonar sensors and micro-switches, as pointed out earlier. We utilize two sonars to measure the y distance. It is necessary to use two sonars because the minimum distance measured reliably by the sonar is between 2 and 3 meters. When the robot operates close to a wall the control system uses the distance estimate to the opposite wall.

We denote the state vector $(x\ y\ \vartheta)^T$ by $\mathbf{x}$, the control vector by $\mathbf{u}$ and the measurement vector by $\mathbf{y}$. The dynamic system, in the discrete form (refer to eq. (27) and (28)), can be described by:

$$\mathbf{x}_{i+1} = \mathbf{h}_i(\mathbf{x}_i, \mathbf{u}_i) + \mathbf{w}_i \quad i = 0, 1, ... \tag{1}$$

$$\mathbf{f}(\mathbf{y}_i, \mathbf{x}_i, \mathbf{n}_i) = \mathbf{0} \quad i = 0, 1, ... \tag{2}$$

where $\mathbf{w}$ is the vector of random disturbance of the dynamic system and it is assumed that the disturbance can be modeled as white noise. The system noise covariance $Q_i = E[\mathbf{w}_i \mathbf{w}_i^t]$ is determined empirically. The measurement vector $\mathbf{y_i}$ does not contain the complete state vector, but only the measurement from the sonar. We assume the measurements are contaminated by additive white noise n_i with zero mean and a variance $\sigma = .3$ (determined empirically).

A Kalman filter (Sorenson, 1985), (Bozic, 1994), (Chui and Chen, 1991), (Haykin, 1991) will use these measurements and the robot's dynamic model (linearized) to provide estimates of the state vector $\mathbf{x}$. The filter used in conjunction with external and internal position estimates will minimize the internal navigation output errors in a feedback configuration.

Control of the vehicle is in particular required when operating close to the wall. When the robot reverses its run, the heading is uncertain because of the stop of the aspiration pump and the brush. During this period the vehicle is floating, which introduces uncertainty. During the same period the sonar measurements are noisy due to readings combing from the close by wall and the far away parallel wall. I.e., a multi-path disturbance is experienced. During this period it is necessary to proceed without an good estimate of the state variable.

We can characterize the initial state of the system with a Gaussian probability estimate:

$$p(\mathbf{x}_0) = \frac{1}{2\pi\|Q\|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}_0-\alpha)^T Q^{-1}(\mathbf{x}_0-\alpha)} \tag{3}$$

where α is the mean value of the state vector and Q is the covariance matrix.

For the position the mean value is centered on the reference value, and the variance is 0.2. Experimentally the angle shows a negative mean value because of the pump, which revolves anti-clockwise. The probability density function can be approximated by a Gaussian which has mean value of $-1.74 \cdot 10^{-2}$ radians and a variance of $3.48 \cdot 10^{-2}$.

With this information it is possible to use the Kalman filtering model (Sorenson, 1985) to obtain a state estimate. Since the process noise $\mathbf{w}_i$ and the measurement noise n_i are white and uncorrelated, a first order linear Kalman filter for the robot pose can be formulated, i.e.:

$$\hat{\mathbf{x}}_{i+1|i} = \hat{\mathbf{x}}_{i|i} + \mathbf{u}_i \tag{4}$$

$$P_{i+1|i} = P_{i|i} + Q_i \tag{5}$$

with the standard method for updating.

As the covariance matrix computation is independent of the state-variables variables, it is possible to calculate it off-line.

5. Control of vehicle

The physical system we are considering, the pool-cleaner, is a tracked vehicle capable of moving on the bottom of pools; hence it is a mobile base capable of motion on a planar or near-planar surface, a rolling machine that can turn and move forward or backward. I.e., it uses skid steering. This robot can be characterized as non-holonomic. The robot is constrained to move in the direction that it is pointing in (this is the non-holonomic constraint).

If W is the robot width, and if v_r and v_l is the speed of the right and left band, respectively, the x and y variables give us the evolution over time of the vehicle position, i.e.:

$$\dot{x} = v\cos\vartheta \quad \dot{y} = v\sin\vartheta \quad \dot{\vartheta} = \tfrac{v_r - v_l}{W}$$

where:

$$v = \frac{v_r + v_l}{2}$$

Given a representation of the physical system the control strategy and the corresponding control strategy can be formulated (MartinsDe-Carvalho, 1993), (Kanjilal, 1995), (Marro, 1992). The control strategy has already been discussed and is shown in Figure 2. Each of the path segments requires a reference value (e.g. the desired path as defined by the lawn-mower pattern) that the control system will use as a servo target. To eliminate minor corrections, a hysteresis band ($\pm e$) is introduced around the reference value.

The target of the control system is to provide a feedback law necessary and sufficient to satisfy the following conditions:

$$\lim_{t\to\infty} (y - y_{ref}) = 0$$

$$\lim_{t\to\infty}(\vartheta - \vartheta_{ref}) = 0$$

We will prove the sufficiency condition for $\exists \lim_{t\to\infty}(y - y_{ref}) = 0$ and for $\exists \lim_{t\to\infty}(\vartheta - \vartheta_{ref}) = 0$. To reach the target a Liapunov function is proposed:

$$V = \frac{1}{2}\left((\mathbf{x} - \mathbf{x}_{ref})^t(\mathbf{x} - \mathbf{x}_{ref})\right) \tag{6}$$

This function is defined positive and $V = 0 \Leftrightarrow (\mathbf{x} - \mathbf{x}_{ref}) = 0$ We consider this function as a candidate Liapunov function. It is possible to write the function (6) on the form:

$$V = \frac{1}{2}\left((y - y_{ref})^2 + (\vartheta - \vartheta_{ref})^2\right) \tag{7}$$

and consider the state variables separately:

$$V_y = \frac{1}{2}(y - y_{ref})^2 \tag{8}$$

$$V_\vartheta = \frac{1}{2}(\vartheta - \vartheta_{ref})^2 \tag{9}$$

Now that we have to investigate $\dot{V}$ we should render:

$$\dot{V} = \begin{cases} 0 & if(\mathbf{x} - \mathbf{x}_{ref}) = \mathbf{0} \\ < 0 & otherwise \end{cases} \tag{10}$$

Initially $\dot{V}_y$ is considered:

$$\dot{V}_y = \dot{y}(y - y_{ref}) \tag{11}$$

We should render:

$$\dot{V}_y = \begin{cases} 0 & if(y - y_{ref}) = 0 \\ < 0 & otherwise \end{cases} \tag{12}$$

In accordance with the equation (11) if we put:

$$\dot{y} = -\gamma_y(y - y_{ref}) \tag{13}$$

the condition (12) is satisfied. Analogously as before for $\dot{V}_\vartheta$ we obtain:

$$\dot{\vartheta} = -\gamma_\vartheta(\vartheta - \vartheta_{ref}) \tag{14}$$

With these condition it is guaranteed that the (6) is decreasing, and converging asymptotically to $V = 0$. Our control law will assume the following shape:

$$correction = -\gamma_y \left[(y - y_{ref}) + \frac{\gamma_\vartheta}{\gamma_y}(\vartheta - \vartheta_{ref}) \right] \quad (15)$$

The variable γ_y is the variable gain given by:

$$\gamma_y = 200|MaxErr - |y - y_{ref}|| + 1 \quad (16)$$

where $MaxErr$ is the maximum permissible error (fixed at 2 meters).

In accordance with simulation experiments for this type of control we have chosen $\gamma_\vartheta/\gamma_y = 7$.

Figure 6 about here

The convergence behaviour of the system has been analysed using simulations. The simulations run over 160 iterations. The model used is the theoretical full state model rather than the simplified/reduced model. The vehicle is moving at a speed of $0.3\frac{m}{s}$ and data are collected every 500 ms. The example in Figure 6 is for a simulation with an initial error of 0.3 meters, after about 30 iterations (4.5 meters) the error becomes negligible. This result is acceptable for a real system. Similar experiments have been carried out for a range of different error values.

Having a dynamic model for the vehicle and a methodology for position estimation it is possible to formulate and implement the control strategy. The formulation will be carried out in the discrete domain.

In each iteration, information about the pose of the robot is given by the triple $(x, y, \vartheta)^T$. If now we consider two iterations (i) and (i+1), and a steering motion first and a linear one later on, the new pose of the machine will be described by the triple $(x', y', \vartheta')^T$.

The changes in orientation is assumed to depend on steering alone, while for changes in position both regular and steering motion must be considered. The straight path motion is termed Δs while the steering angle is denoted $\Delta\vartheta$. The two different movements give rise to the following variation in position:

- steering motion:

$$dx = \left[\frac{W \sin \Delta\vartheta}{2}\right] \cos\vartheta - \left[\frac{W(1 - \cos\Delta\vartheta)}{2}\right] \sin\vartheta \quad (17)$$

$$dy = \left[\frac{W \sin \Delta\vartheta}{2}\right] \sin\vartheta + \left[\frac{W(1 - \cos\Delta\vartheta)}{2}\right] \cos\vartheta \quad (18)$$

Figure 7 about here

– straight motion:

$$\Delta x = \Delta s \cos(\vartheta + \Delta\vartheta) \quad (19)$$
$$\Delta y = \Delta s \sin(\vartheta + \Delta\vartheta) \quad (20)$$

This allow computation of the total change in state between two iterations. Initially the orientation variation is considered and the later the translational change is introduced. Considering small angles and referring to equations (17) and (18), we can write:

$$dx \simeq 0 \quad (21)$$
$$dy \simeq 0 \quad (22)$$

since, for $\varphi \simeq 0$, we have:

$$\sin\varphi \simeq 0 \quad \cos\varphi \simeq 1$$

In practise this is considered acceptable as the steering angle is limited to 12 degrees. Consequently, dx and dy are always small (i.e., they are less than 0.04m and 0.005m, respectively).

The straight line motion is consider constant. Control actions are not introduced in each steps, and even if they were, the steering time is small compared to ΔT. For this reason we can consider:

$$\Delta s \simeq v\Delta T$$

and we will use it in the control strategy.

Now we can linearize the system around the working point of control (i.e., the variations of ϑ around 0). We use a first order Taylor expansion for the linearization of the system. Given that the x position is not observable, a reduced state vector and state estimation system can be formulated.

$$\mathbf{x}_{i+1} = A\mathbf{x}_i + B\mathbf{u}_i \quad i = 0, 1, ... \quad (23)$$

The state vector was defined as $(y\ \vartheta)^t$, the control vector $\mathbf{u}$ is:

$$\mathbf{u} \stackrel{\text{def}}{=} \begin{bmatrix} u_r \\ u_l \end{bmatrix} \quad (24)$$

and the A and B matrices are:

$$A = \begin{bmatrix} 1 & \Delta s \\ 0 & 1 \end{bmatrix} \quad (25)$$

$$B = \begin{bmatrix} \Delta s/W & -\Delta s/W \\ 1/W & -1/W \end{bmatrix} \quad (26)$$

Considering the whole system, with both process disturbance and measurement noise, we have:

$$\mathbf{x}_{i+1} = A\mathbf{x}_i + B\mathbf{u}_i + \mathbf{w}_i \quad i = 0, 1, ... \qquad (27)$$

$$\mathbf{y}_i = H\mathbf{x}_i + \mathbf{n}_i \quad i = 0, 1, ... \qquad (28)$$

where $\mathbf{w}$ and $\mathbf{n}$ are described in Section 6 and H, the measure matrix, is:

$$H = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad (29)$$

Figure 8 shows the final block diagramme of the system.

Figure 8 about here

Simulation of the system provides the following results. The simulation is initiated with an error of 0.3 meters. The following figures show the sonar output, the estimation of the state vector (Figure 9), the control vector and the real position of the vehicle (Figure 10), respectively.

Figure 9 about here

Figure 10 about here

The sonar output data has a resolution of 10 cm. The control values are fed through a sample hold circuit that has a period of 50 ms corresponding to the control periods for the tracks. Here, for the sake of simplicity, we used both negative and positive values of time. Negative values of time refer to control of the right track of the vehicle. Similar simulations have been carried out for a range of different error values.

6. Implementation

The control system was implemented on a regular PC, the software was written in C++. It is composed of two main components, the pool cleaner part, which contains all the routines for driving the robot and planning the path, and the sensor part, which contains the routines for acquiring and filtering data from the sensors. The computer communicates with the pool cleaner using a serial (sonar interface) and a parallel port (control of the tracks and input from the bumpers).

7. Experiments

The performance of the system is illustrated by a test in a rectangular 25×8 meters pool.

The test is a complete cleaning sequence for the pool.

The pool cleaner starts from one corner of the pool, after it has taken the first reference data readings, e.g. the width of the pool, it proceeds according to the pre-defined control sequence until the distance to the wall becomes less than 3.5 meters. At this point the control switches sonar and the machine continues until the end of the pool using the other sonar. The raw sonar data are shown in Figure 11.

Figure 11 about here

Figure 12 about here

At the start of a session the sonar generates noisy data. As the sonar reaches working temperature the noise is reduced. To compensate for this the speed of traversal is reduced during startup and while switching between the two sonars. Switch of sonars during a cleaning session in general implies that a few laps are repeated to ensure full coverage. The trajectory of vehicle is shown in Figure 12.

The machine completed the cleaning section of the entire swimming pool in about 3600 iterations, and since each iteration in the graphic is 1 sec., the total time was 1 hour. At the start of the session an estimated cleaning time of 55 minutes was computed by the system; considering that the machine spend some time for the change of sonar, and other time for the correction of a path error, the time estimate is considered acceptable.

8. Summary and Conclusions

The objective was to create an autonomous pool cleaning machine that can be introduced on the world market at a competitive price. The first step of this project was to evaluate the use of sonar as a sensor for this type of application. In addition a theoretical model for control has been developed. Subsequently the developed model has been evaluated in an realistic setting. A theoretical study of the whole system, considering possible solutions for the integration between the robot and the sensors was reported. The experimental part provided interesting results; it was divided into several tests, a few of which are reported here. The tests included several different pools to ensure generality.

The autonomous pool cleaner is in principle ready for commercialization; many of the achieved results in our tests are very promising, especially considering that the bottom of the swimming pool is completely covered in a realistic and acceptable time. Current work is directed at packaging of the system for commercial use.

The sonar system is well suited for the present application. Current research is investigating use of the same sonar system for mapping and traversal of adventure type pools.

References

Bozic, S. M. 1994. *Digital and Kalman filtering: an introduction to discrete-time filtering and optimum linear estimation.* London : Arnold.

Catlin, Donald E. 1989. *Estimation, control, and the discrete Kalman filter.* New York, Berlin: Springer-Verlag.

Chui, Charles K. and Chen, Guanrong 1991. *Kalman filtering: with real-time applications .* Berlin, New York: Springer-Verlag.

Haykin, Simon S. 1991. *Adaptive filter theory.* Englewood Cliffs, N.J.: Prentice Hall.

Kanjilal, Partha Pratim 1995. *Adaptive prediction and predictive control.* Stevenage: P. Peregrinus on behalf of Institution of Electrical Engineers.

Leonard, J.J. and Durrant-White, Hugh F. 1992. *Directed sonar sensing for mobile robot navigation.* Kluwer Academic Publishers.

Leroy, C.C. 1992. Development of simple equations for accurate and more realistic calculation of the speed of sound in sea water. *J.Acoust.Soc.Am..* 46:216.

Maksarov, D. and Durrant-Whyte, H. 1995. Mobile vehicle navigation in unknown environments: a multiple hypothesis approach. *Control Theory and Applications, IEE Proceedings.* Vol:142. Pages: 385-400.

Marro, G. 1992. *Controlli Automatici.* Zanichelli.

Martins De Carvalho, J.L. 1993. *Dynamical systems and automatic control.* New York: Prentice-Hall.

Sorenson, Harold W. 1985. *Kalman filtering: theory and application.* New York: IEEE Press.

Urick, Robert J. 1993. *Principles of underwater sound.* McGraw-Hill.

Authors' Vitae

M. Simoncelli

Maurizio Simoncelli was born in Savona, Italy in 1973. He earned his M.Sc. in 1999 from the University of Genoa, Italy. picture maury.ps

G. Zunino

Guido Zunino received the "laurea" degree in Electrical engineering from the University of Genoa, Italy, in 1999. He then joined the Centre for Autonomous Systems at the Royal Institute of Technology, Sweden, as a Ph.D. student. He is currently working on robust fusion of sensory information for navigation in a domestic environment. This project investigates methods for landmark detection and navigation for domestic robotics. picture guido1.ps

H.I. Christensen

Henrik Christensen received M.Sc. EE and Ph.D degrees in 1987 and 1989, respectively, from Aalborg University. He was a researcher and lecturer at Aalborg University 1990-1996. He has held visiting positions at Oak Ride National Laboratory and University of Pennsylvania. Dr.

Christensen is the director of the Centre for Autonomous System, Royal Institute of Technology, Sweden and has a full chair in computer science at the department of Numerical Analysis and Computer Science. He has published more than 120 papers on robotics and computer vision. picture hic.ps

K. Lange
Klas Lange earned an M.Sc. in Mechanical Design from the Royal Institute of Technology, Sweden. He is now the CEO of WEDA Poolcleaners and WEDA Water Cleaning. No picture available

Figure Captions

1. The WEDA B400 pool cleaner
2. Control algorithm: the ideal cleaning path.
3. Block-diagram for pool cleaner control system
4. Tests with the sonar system: a) and c) regard the problem with the sonar when the robot is not aligned to the wall; b) illustrates the problem of short distances.
5. The pose of the robot: x, y and ϑ
6. Control system simulation. Initial error 0.3 meters. Control system with noise and measurement noise.
7. Steering motion: variation in position in the global coordinate system.
8. Control system with noise and measurement noise
9. Simulated sonar output (initial error 0.3 meters) with an additive noise, and data received from the Kalman filter
10. Control output sequence and real vehicle position (y)
11. Sonar data stored by the control software in the test. The y distance (in meters) decreases over time (navigation iterations). This graphic shows sonar data with a solid line and the tracked measurements with a dashed line.
12. Robot path. The actual robot position has been reconstructed from the computed position estimate.

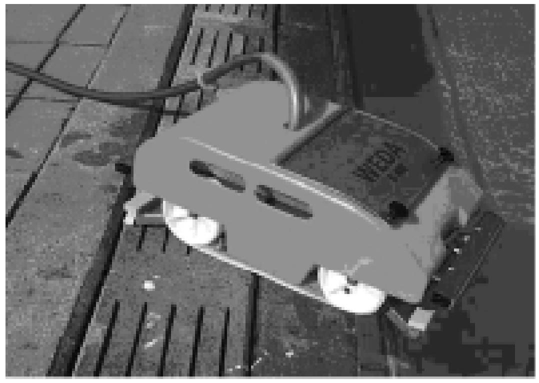

Figure 1. The WEDA B400 pool cleaner

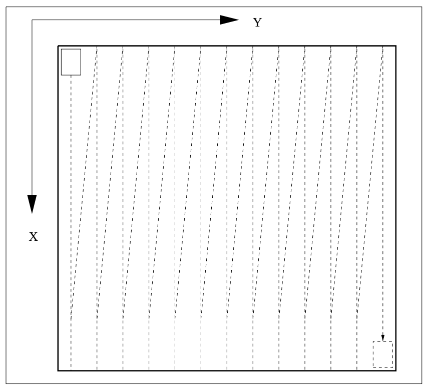

Figure 2. Control algorithm: the ideal cleaning path.

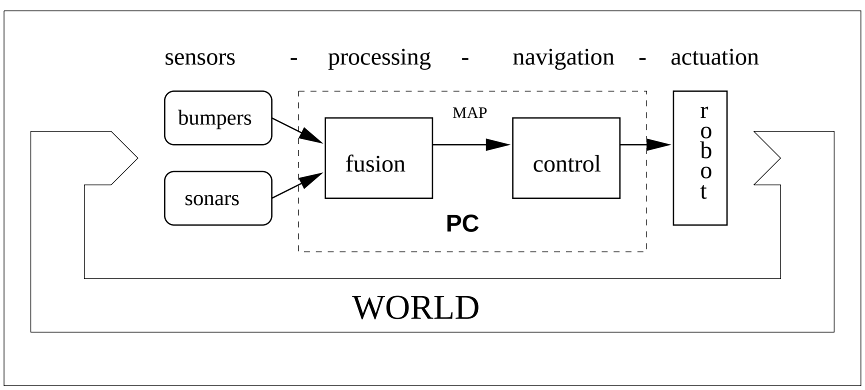

Figure 3. Block-diagram for pool cleaner control system.

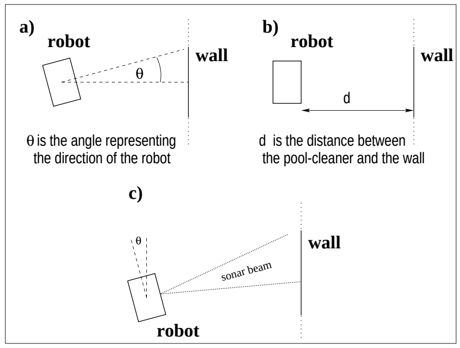

Figure 4. Tests with the sonar system: a) and c) regard the problem with the sonar when the robot is not aligned to the wall; b) illustrates the problem of short distances.

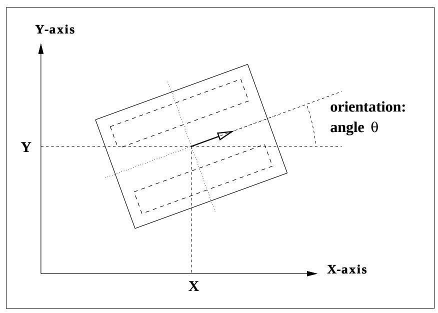

Figure 5. The pose of the robot: x, y and ϑ.

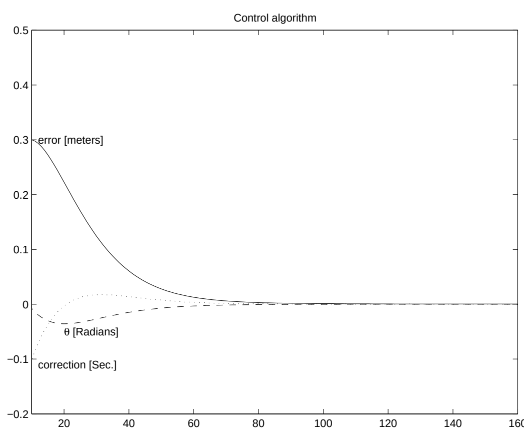

Figure 6. Control system simulation. Initial error 0.3 meters. Control system with noise and measurement noise.

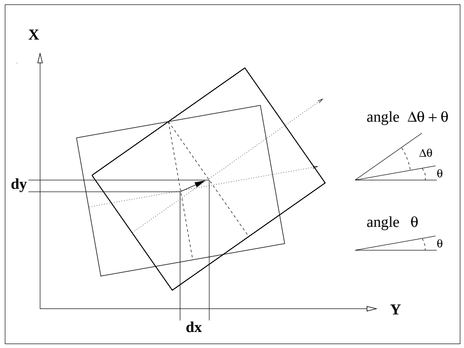

Figure 7. Steering motion: variation in position in the global coordinate system.

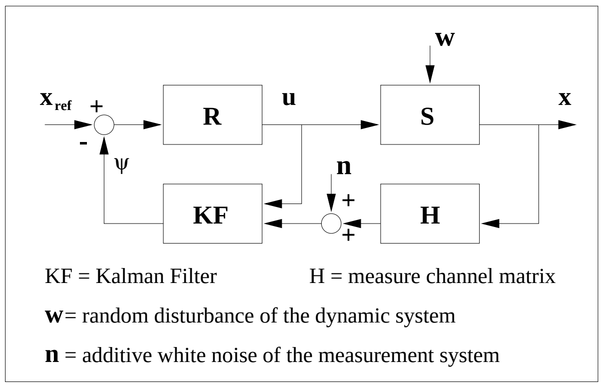

Figure 8. Control system with noise and measurement noise.

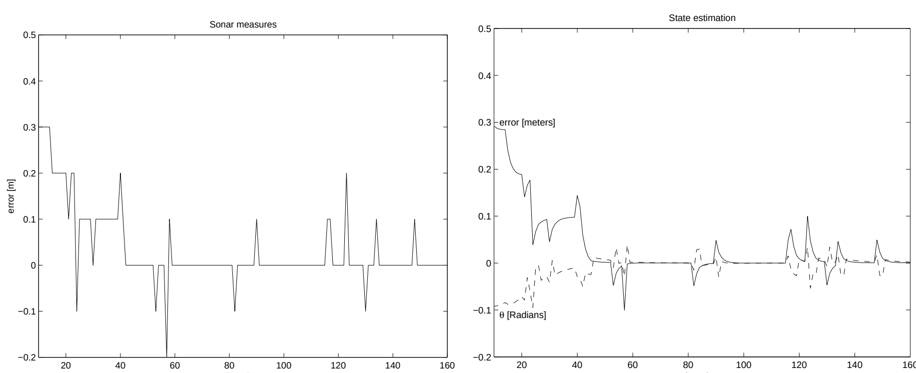

Figure 9. Simulated sonar output (initial error 0.3 meters) with an additive noise, and data received from the Kalman filter.

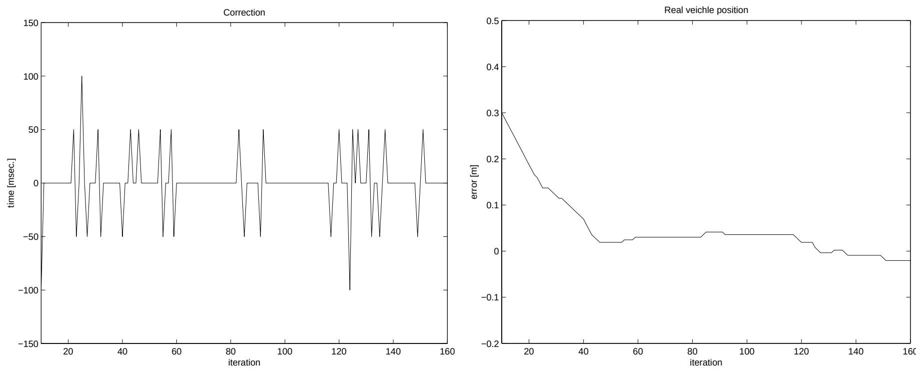

Figure 10. Control output sequence and real vehicle position (y).

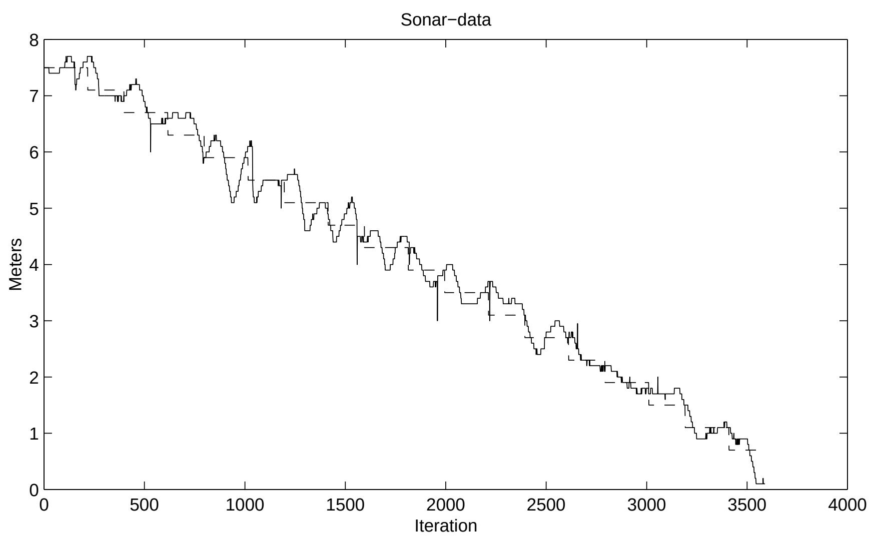

Figure 11. Sonar data stored by the control software in the test. The y distance (in meters) decreases over time (navigation iterations). This graphic shows sonar data with a solid line and the tracked measurements with a dashed line.

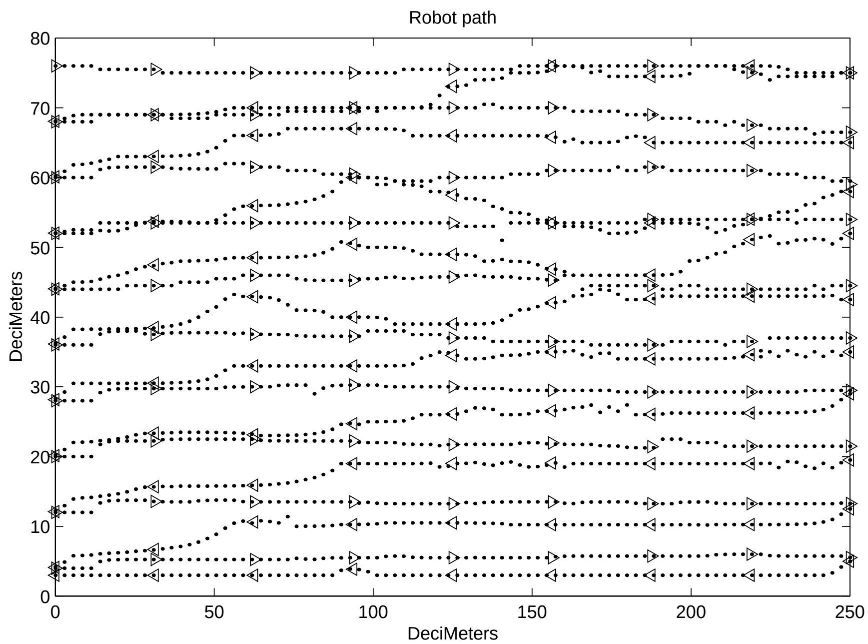

Figure 12. Robot path. The actual robot position has been reconstructed from the computed position estimate.