

Design and Application of a Motion Platform in Three Degrees of Freedom

Webjørn Rekdalsbakken

Aalesund University College, 6025 Aalesund Norway

Abstract

Aalesund University College is building the first full-size high-speed craft simulator in Norway. In preparation of the construction of the simulator there has been built a small-scale motion simulator to test the possibilities to imitate the movements of the ship. The model is based on a Stewart platform in three degrees of freedom (roll, pitch and heave). The platform deck is moved by three DC-servomotors through three legs situated at the corners of an equal sided triangle with its center aligned with the mass center of the deck. The ship movements are generated by a simulator program on a PC, giving corresponding values for roll, pitch and heave at each instant. These values are transformed to equivalent values for the position of the three servo drives, which again are converted to analog voltages and applied as the reference signals to a Pulse Width Modulation card controlling each motor. Comprehensive testing have been done on the motion simulator, and it seems like the platform will imitate the movements of a ship in a satisfactory way. As part of the testing there has also been developed a control loop for manipulating the path of a moving object, e. g. a ball or a wheel, on the platform. For this purpose a camera was used to measure the position and velocity of the object in the x- and y-directions, and these state variables were used in a feedback loop to control the path of the moving object. Different regulator strategies have been tried, like pole placement, optimum control, fuzzy logic and neural networks. The results have been satisfactory and the object is controlled to move in a circle or an 8-curve or from point to point as decided.

1. Introduction

Aalesund University College is about to finish the construction of a full scale high-speed craft simulator. The simulator is an identical copy of the bridge of a modern high speed ferry. The bridge is mounted on a motion platform in three degrees of freedom. In this way the bridge is made to move like a ship in open sea and in synchronism with the visual simulation. In the preparation of the building of the full-scale ship motion simulator a model of the motion platform was constructed. The model is based on a Stewart platform with three degrees of freedom (roll, pitch, heave). The Stewart platform is a much used principle in the construction of motion platforms, both for ships, aircrafts and other types. A full Stewart platform has six degrees of freedom, and the principle describes the mathematical relations between the inputs to the actuators and the movement of the platform, see refs. [1] to [3]. Mechanically the platform consists of two circular rigid plates with a diameter of 60 cm. One of the plates functions as a stable bearing and the other as the moving deck. The deck is mounted on the bearing through an elastic spring at its centre so that the spring holds the weight of the deck. The deck is connected to the spring by a universal joint securing free movement in all directions. To move the deck there are three legs placed 120 degrees apart in an equal-sided triangle near the circumference of the deck. Each leg is attached to the shaft of a DC servomotor, and each motor is controlled by a pulse width modulated (PWM) signal. The primary signals (roll, pitch and heave) come from a simulator representing the movement of the ship. These signals are transformed to a corresponding set of control signals for the servo drives, giving the reference position for each motor. In this way the platform can be controlled to make all kinds of movements in three axes and thus imitate the motion of a ship. See ref. [4]¹.

¹ Ref. [4] is a summary of student projects performed at AaUC under the guidance of the writer and his colleagues.

2. Coordinates and Transformations

2.1 Principle

A Cartesian coordinate system is defined with the xy -plane along the platform deck and origo at the deck centre, the z -axis is pointing upwards from the deck. The primary signals from the ship simulator represent the angles roll (about the y -axis) and pitch (about the x -axis), in addition to the vertical movement (heave). To determine the control signals to the servo drives the vertical movements of the three legs have to be calculated. This is done by coordinate transformations of the points connecting the legs to the deck. The operation consists of three subsequent linear coordinate transformations each represented by a 3×3 -matrix. These operations represent the connection between the simulator signals and the control signals. The result of the total transformation is a 3×3 -matrix with the new coordinates for the three connection points. The motor positions are now calculated from the new z -values of the three points. A small problem arises in the positioning of the motors in that the angle of each leg changes a little with the movement. This introduces a small error in the resulting z -position. To reduce the error one can calculate the angle between the leg and the vertical line from the measured angle of the motor shaft, and thus get a corrected z -position. The nature of the problem is iterative. However, within the range of angles normally used in the operation of the deck, the error is small and within acceptable limits after the first calculation.

2.2 Reference system

The coordinate transformations start with the Cartesian coordinate system as described above with the deck in horizontal position at a given vertical height. First the coordinates of the three connection points between the deck and the legs are derived in this reference system. The platform and the initial coordinate system are shown in figure 2.1 below:

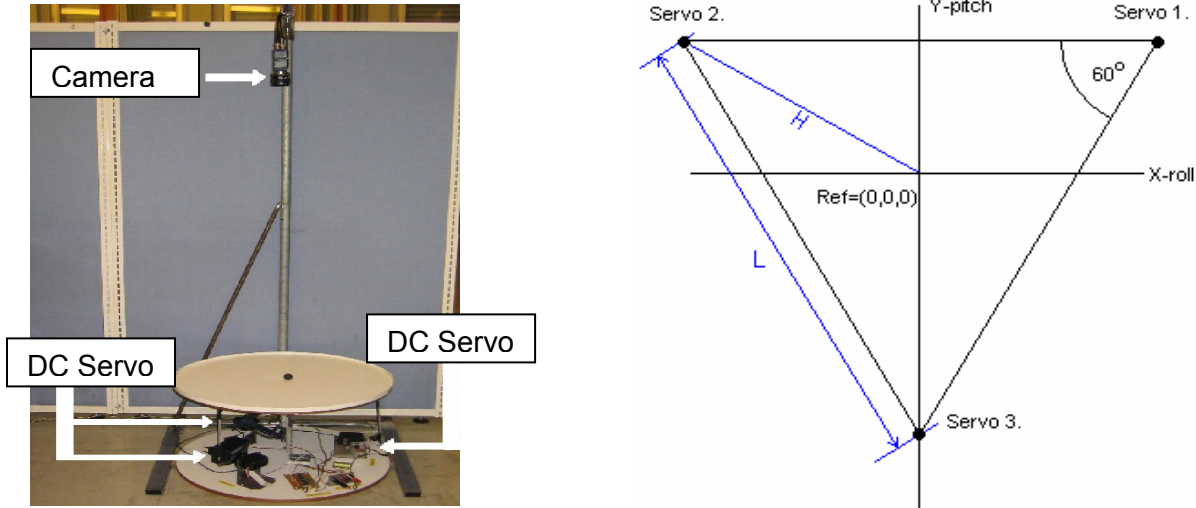


Figure 2.1: The platform and the reference system.

The initial positions of the connection points are thus given by the following coordinates:

$$P_1 = \left(\frac{L}{2}, \frac{L}{2\sqrt{3}}, 0 \right) \quad P_2 = \left(-\frac{L}{2}, \frac{L}{2\sqrt{3}}, 0 \right) \quad P_3 = \left(0, -\frac{L}{\sqrt{3}}, 0 \right)$$

These points are represented by the columns of the position matrix. On a ship the bridge (wheel house) is normally not situated at the centre alongside the ship, but located at some distance d from

the centre. In the reference coordinate system this means along the y -axis. This results in an initial position matrix as follows:

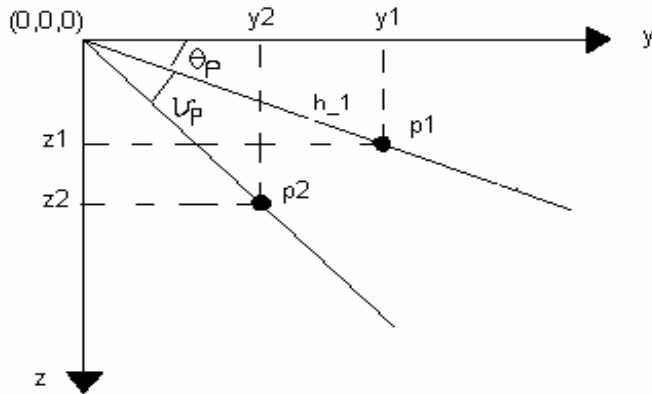
$$P_{123} = \begin{bmatrix} \frac{L}{2} & -\frac{L}{2} & 0 \\ \frac{L}{2\sqrt{3}} + d & \frac{L}{2\sqrt{3}} + d & -\frac{L}{2\sqrt{3}} + d \\ 0 & 0 & 0 \end{bmatrix} \quad 1)$$

The platform deck is now turned about the two coordinate axes one at a time. These operations are represented by linear transformations given by matrices. The position matrix are premultiplied by these matrices to give the resulting position of the platform deck.

2.3 Matrix transformations

2.3.1 Pitch motion

The pitch movement is a rotation about the x -axis. In figure 2.2 below the x -axis is pointing normally into the plane and is shown as a point. The rotation is from the point P_1 at an angle θ_p to the point P_2 at an additional angle ν_p .



The figure shows the relations:

$$x_1 = x_2 \text{ (axis of rotation)}$$

$$y_1 = h_1 \cos \theta_p$$

$$z_1 = h_1 \sin \theta_p$$

Figure 2.2: Rotation about the x -axis (pitch motion).

The rotation gives the following relations which are derived from the figure:

$$y_2 = h_1 \cos(\theta_p + \nu_p) = h_1 \cos \theta_p \cos \nu_p - h_1 \sin \theta_p \sin \nu_p$$

$$z_2 = h_1 \sin(\theta_p + \nu_p) = h_1 \sin \theta_p \cos \nu_p + h_1 \cos \theta_p \sin \nu_p$$

The relations between the old and the new coordinates are then expressed like this:

$$x_2 = x_1$$

$$y_2 = y_1 \cos \nu_p - z_1 \sin \nu_p$$

$$z_2 = y_1 \sin \nu_p + z_1 \cos \nu_p$$

which give the transformation matrix:

$$T_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \nu_p & -\sin \nu_p \\ 0 & \sin \nu_p & \cos \nu_p \end{bmatrix} \quad 2)$$

Multiplying this matrix with the initial position matrix gives the new positions of the points P_1 , P_2 and P_3 :

$$P_p = T_p \cdot P_{123} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \nu_p & -\sin \nu_p \\ 0 & \sin \nu_p & \cos \nu_p \end{bmatrix} \cdot \begin{bmatrix} \frac{L}{2} & -\frac{L}{2} & 0 \\ \frac{L}{2\sqrt{3}} + d & \frac{L}{2\sqrt{3}} + d & -\frac{L}{2\sqrt{3}} + d \\ 0 & 0 & 0 \end{bmatrix}$$

which gives:

$$P_p = \begin{bmatrix} \frac{L}{2} & -\frac{L}{2} & 0 \\ \left(\frac{\sqrt{3}L}{6} + d\right)\cos \nu_p & \left(\frac{\sqrt{3}L}{6} + d\right)\cos \nu_p & \left(-\frac{\sqrt{3}L}{3} + d\right)\cos \nu_p \\ \left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p & \left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p & \left(-\frac{\sqrt{3}L}{3} + d\right)\sin \nu_p \end{bmatrix} \quad 3)$$

2.3.2 Roll motion

The roll movement is a rotation about the y -axis. The movement starts from the positions given by matrix 3), and the calculations are similar to those for the pitch. The resulting transformation matrix will be:

$$T_r = \begin{bmatrix} \cos \nu_r & 0 & -\sin \nu_r \\ 0 & 1 & 0 \\ \sin \nu_r & 0 & \cos \nu_r \end{bmatrix} \quad 4)$$

The new positions of the connection points are found by premultiplying this matrix with matrix 3). The result is the following matrix:

$$P_{pr} = \begin{bmatrix} -\left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p \sin \nu_r + \frac{L}{2}\cos \nu_r & -\left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p \sin \nu_r - \frac{L}{2}\cos \nu_r & -\left(-\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p \sin \nu_r \\ \left(\frac{\sqrt{3}L}{6} + d\right)\cos \nu_p & \left(\frac{\sqrt{3}L}{6} + d\right)\cos \nu_p & \left(-\frac{\sqrt{3}L}{3} + d\right)\cos \nu_p \\ \left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p \cos \nu_r + \frac{L}{2}\sin \nu_r & \left(\frac{\sqrt{3}L}{6} + d\right)\sin \nu_p \cos \nu_r - \frac{L}{2}\sin \nu_r & \left(-\frac{\sqrt{3}L}{3} + d\right)\sin \nu_p \cos \nu_r \end{bmatrix} \quad 5)$$

2.3.3 Heave motion

The last transformation is a translation along the z -axis (heave). This is a pure addition of a constant to the z -coordinates of the three points given by matrix 5) above. The resulting coordinate matrix of the connection points becomes as follows:

$$P_{prz} = \begin{bmatrix} -\left(\frac{\sqrt{3}L}{6}+d\right)\sin\psi_p\sin\psi_r+\frac{L}{2}\cos\psi_r & -\left(\frac{\sqrt{3}L}{6}+d\right)\sin\psi_p\sin\psi_r-\frac{L}{2}\cos\psi_r & -\left(\frac{\sqrt{3}L}{6}+d\right)\sin\psi_p\sin\psi_r \\ \left(\frac{\sqrt{3}L}{6}+d\right)\cos\psi_p & \left(\frac{\sqrt{3}L}{6}+d\right)\cos\psi_p & \left(\frac{\sqrt{3}L}{3}+d\right)\cos\psi_p \\ \left(\frac{\sqrt{3}L}{6}+d\right)\sin\psi_p\cos\psi_r+\frac{L}{2}\sin\psi_r+z & \left(\frac{\sqrt{3}L}{6}+d\right)\sin\psi_p\cos\psi_r-\frac{L}{2}\sin\psi_r+z & \left(-\frac{\sqrt{3}L}{3}+d\right)\sin\psi_p\cos\psi_r+z \end{bmatrix} \quad 6)$$

2.4 Actuator signals

The matrix P_{prz} gives the coordinates of the connection points between the deck and the three actuator legs when the ship is in a given position relative to the reference system. The actuator legs are connected to the shafts of the three servo motors as shown in figure 2.3 below:

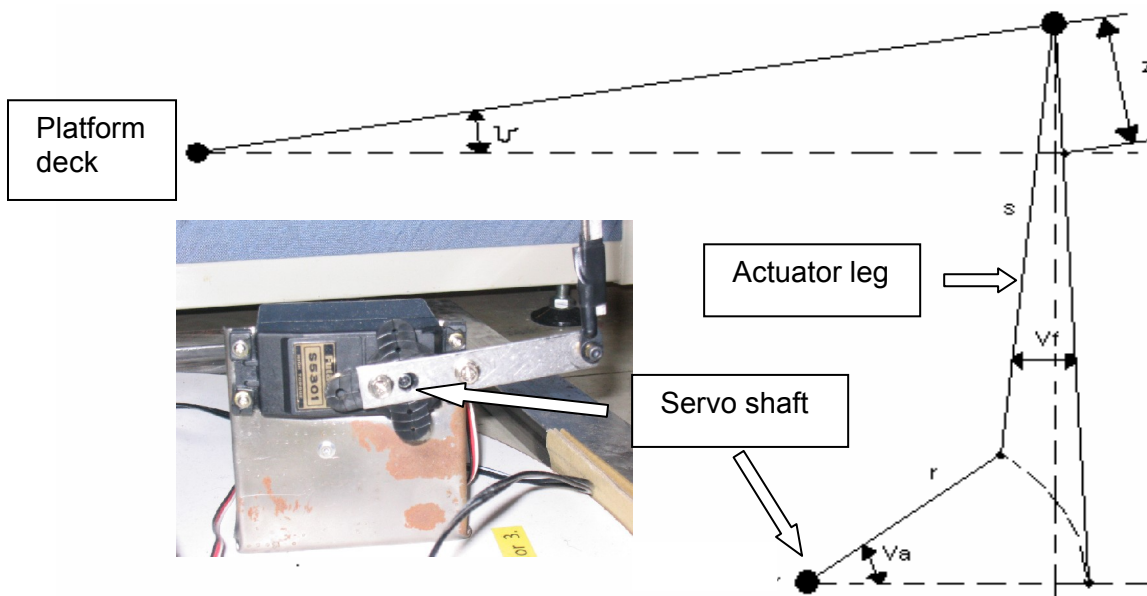


Figure 2.3: Relation between shaft angle V_a and platform angle ψ .

To decide the necessary rotation of the motor shafts one needs the z -values of the three connection points represented by the last row of matrix 6). The angle of rotation for each of the shafts is calculated from the following relation (see fig. 2.3), using the corresponding z -value:

$$V_a = \arcsin(z/r) \quad 7)$$

This angle is converted to an analog voltage representing the reference signal to the servo drive. As indicated in fig. 2.3 the vertical movement of the actuator leg will differ from that of the motor shaft. This is because the actuator leg changes its angle with the vertical line during the motion. Shaft rotation within $\pm 45^\circ$, which gives a tilting of the deck of about $\pm 15^\circ$, is enough for realistic movements of a ship. Within these limits the resulting deviation in actual deck position from the reference position will be restricted to about 5%, which is acceptable.

3. Ship Motion Simulator

In the preparation of the building of a full-scale high-speed craft simulator, a ship motion simulator was designed for the small platform model. The motion simulator includes a wave generator and the necessary communication and control software for the platform. The wave generator consists of three independent signal generators (roll, pitch and heave) with adjustable amplitude, frequency and phase shift between the signals. The simulator also has a mode for independent manual operation of the three axes and direct manipulation of each of the servo drives. The program is developed in LabView and runs on a PC. Through a user-friendly graphical interface the user can control the movements of the platform in an easy and understandable way. The operator can choose the wave type and set all attributes related to the signals. He can also choose the position of the bridge relative to rotation centre of the ship and thus simulate the movements at different locations alongside the ship. The simulator program is an important tool in the testing and maintenance procedures of the full-scale simulator, see ref. [4].

4. Controlling a moving object on the platform

4.1 Introduction

The second challenge was to manage the control of a moving object on the platform. For this purpose an industrial camera with a frame grabber card was provided. The first goal was to make a small ball move from an arbitrary position to the centre of the deck. A mathematical model of the dynamics of the rolling object on the platform was designed, and a modal control algorithm was constructed. The camera is used to measure the position of the ball in real-time. The velocity of the ball is estimated from the position measurements. Position and speed are the states of the dynamic system, and these states are weighted through a K-matrix and summed to form the control signal that is fed back to the actuators. With some tuning and testing the ball was made to follow all kinds of continuous paths. With this success the experiments were extended to use a neural network and a fuzzy regulator. After a while these regulators also functioned very well. The control systems were developed in Visual C++ and in LabView, see ref. [4].

4.2 Modal control

4.2.1 Dynamic model

The aim is to control the path of a ball rolling on the deck. First a simple model was derived for the dynamics of the ball. Figure 4.1 shows the situation:

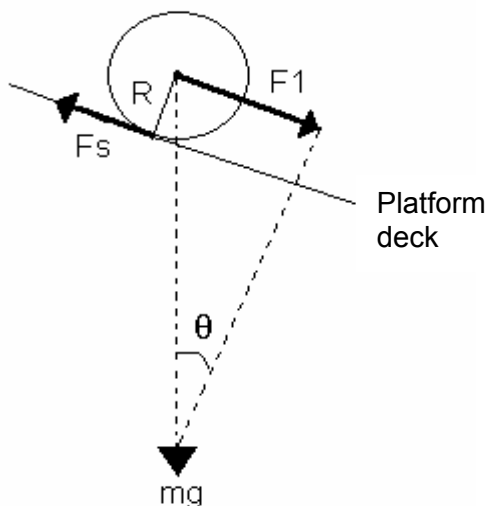


Figure 4.1: The dynamics of a rolling ball on an inclined plane.

The torque-balance about the centre of the ball gives:

$$F_s \cdot R = I \cdot \varepsilon = \frac{2}{5} m R^2 \cdot \frac{a}{R} \Rightarrow F_s = \frac{2}{5} m a \quad 8)$$

Here a is the acceleration of the ball along the deck and ε is the angular acceleration.

The force balance along the deck gives:

$$F_1 - F_s = m a \Rightarrow m g \sin \theta - m \dot{\theta}^2 l - \frac{2}{5} m a = m a$$

where l is the radial distance of the ball from the deck centre. This gives:

$$a = \frac{5}{7} (g \sin \theta - \dot{\theta}^2 l) \quad 9)$$

This shows a nonlinear relation between the deck movement and the acceleration of the ball. In the control loop it is assumed that the deck movement is limited to small angles from the horizontal position and at a low speed, so that the model can be linearized. This gives:

$$a \approx \frac{5}{7} g \theta \approx 7.0 \theta \quad 10)$$

This is quite a rough approach to the dynamical situation. The centrifugal force on the ball has been neglected because in most situations this force on the ball will be small compared to gravity. There are also several other comparable sources of error and noise in this model, e.g. dead bands in the servos and transmissions and friction between the ball and the deck. The model is crude, but compared to other sources of error it may be acceptable. The control system, however, has to be robust.

4.2.2 Control loop

The states of the system are chosen to be position and speed in both the x - and y -directions. Since the situation is symmetric in these directions the system is treated as two independent and equal control loops. The feedback loop for controlling the movement of the ball along one axis is shown in figure 4.2 below:

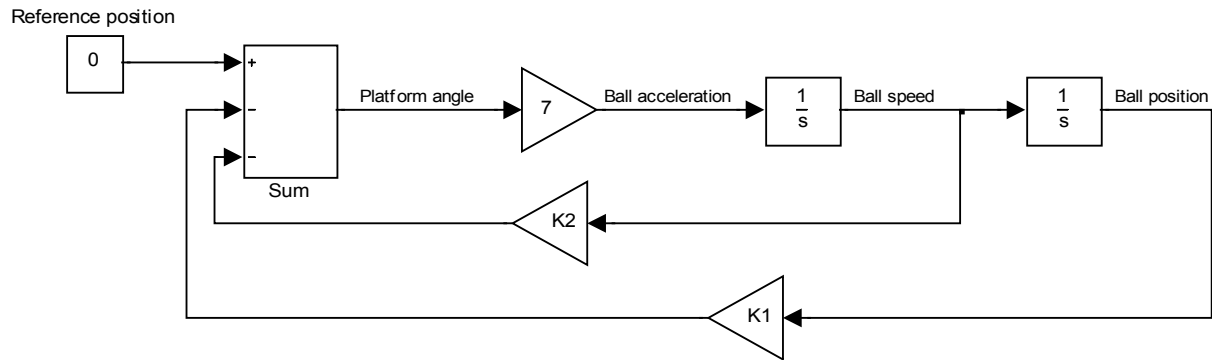


Figure 4.2: The control loop for the ball in one dimension.

The state space model for the system is very simple:

$$\dot{\vec{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \vec{x} + \begin{bmatrix} 0 \\ 7 \end{bmatrix} u, \quad \vec{y} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \vec{x} \quad 11)$$

where u is the angle of the platform deck in the respective direction.

The control law $u = -K \cdot \vec{x}$ is now used to control the system. This gives:

$$\dot{\vec{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \vec{x} - \begin{bmatrix} 0 \\ 7 \end{bmatrix} [K_1 \quad K_2] \vec{x} = \begin{bmatrix} 0 & 1 \\ -7K_1 & -7K_2 \end{bmatrix} \vec{x}$$

The characteristic equation of the system is:

$$s^2 + 7K_2s + 7K_1 = 0 \quad 12)$$

Now the requirements set to the system, will decide the values of the feedback constants. Several possibilities have been tried out, such as a Butterworth polynomial, a critical damped system and even

a Linear Quadric Regulator. The best result was obtained by approximating a critical damped system with a bandwidth of the system equal to about $\omega_0 \approx 2.4 \text{ rad/s}$. This gave the values $K_1 = 0.82$ and $K_2 = 0.69$. Because of trembling in the platform a low pass-filter with a cutoff frequency of 10 Hz had to be implemented in the estimation of the speed. With these values the closed system is stable with two equal and real poles at $s \approx -2.4$.

4.3 Localization of the ball

To localize the ball and thus measure the states of the system, a camera was used. The camera was analog and monochromatic with a resolution of 8 bits, from black (0) to white (255). The camera was placed vertically above the deck with the left/right and up/down directions of the image aligned with the x - and y -axis of the deck, respectively. A frame grabber card captured the image 25 times each second. To localize the black ball on the white surface a search for pixels below a certain value was performed. The search path was made up of increasing circles starting from the centre of the deck. The spatial resolution of the camera is 768×576 pixels and the search algorithm used a spacing of 10 pixels. When a low-value pixel is found, a more closely search is done in that area to verify that it is the ball. The search algorithm localizes the centre of the ball and the x - and y -coordinates of the ball centre are calculated. The values of the speed in the x - and y -directions are estimated from succeeding measurements of the coordinates.

4.4 Neural network control

4.4.1 A neural network replacing the modal regulator.

A neural network was designed to control the path of the rolling ball, see ref. [5] on neural networks. Two identical networks were used for the two independent axes. The neuron function was of type *sigmoid*. Since the dynamical system is nonlinear and the movements of the platform also tend to have some discontinuities, a network with two hidden layers was explored. The network has two inputs representing the states of the system and one output, the next angular position of the deck. The first hidden layer has five neurons, the second has four neurons. For the training of the network the back-propagation method was used. In the first experiment the network was trained from data acquired during a control session with the modal regulator already designed. Corresponding data for position, speed and the angle of the platform deck in both the x - and y -directions were collected while the modal control was running. The data were sorted in tables and written to files. Several series of data with different kinds of reference paths for the ball were acquired. With these data the network was trained to adapt the control of the modal regulator. The modal regulator was then substituted with the resulting trained network, and this procedure worked very well. See ref. [4].

4.4.2 Imitating the mathematical model

So far the neural network represents no substantial improvement from the modal control. The next experiment therefore consisted in a continuous training of the network while the loop was running. The network was trained against a mathematical model derived from eq. 9), given by:

$$a = \frac{5}{7} g \sin \theta \approx 7.0 \sin \theta \quad 13)$$

Here the centrifugal contribution is neglected. The optimization criterion is the acceleration derived from the modal algorithm:

$$a_d = 7.0(K_1 \cdot \text{position} + K_2 \cdot \text{speed}) \quad 14)$$

The error in the acceleration is: $e = a_d - a = a_d - 7.0 \sin \theta$, where θ is the angle calculated by the neural network. The angle θ represents the new position of the deck and is the source of the control signals to the servos. The estimated error is used to calculate the error gradient of the back-propagation algorithm. In this way the nonlinearity of the model is approximated by the neural network. This network has the benefit that it can be trained in stand-alone against the theoretical model without logged data. When the result of this training has reach a certain error limit, the network can be implemented in the control program and adapt itself further while running. The crucial point is the optimization criterion used in the training. The network will not be better than the insight the designer has got into the system. In this case modal control is the leading principle and thus sets the constraints on the quality of the regulator.

4.5 Fuzzy logic control

Regarding all the uncertainties in this dynamic system a Fuzzy Logic (FL) system should be a relevant control method to explore, see ref [5] on FL-systems. For the implementation of a fuzzy logic controller the Free Fuzzy Logic Library (FFLL)² was used, see ref. [6]. This is an "open source" class library and API, optimized for time critical applications. FFLL is able to read files written in Fuzzy Control Language (FCL). FCL files contain the membership functions and the rules used in the fuzzy logic system. FFLL employs the Mamdani method and for defuzzification uses the Center of Gravity (COG) method. The platform is, as in the other experiments, controlled by two independent and identical control loops. The state variables are position and speed. The output from the controller is the change in platform angle, which is used to calculate the control signals to the servos. For position and deck angle the membership functions are chosen to be:

-Big, -Medium, -Small, None, Small, Medium, Big.

For the speed they are: *-Big, -Small, None, Small, Big.*

Combinations of the state variables will give altogether 31 fuzzy rules, assuming that with either of the states in the membership function *None*, the result is *None*. To train the FL-system it was chosen to use the ANFIS³ editor in the Fuzzy Logic Toolbox in Matlab, see ref. [7]. The system was trained against data obtained from runs with the modal control regulator, first with the ball reference at the center of the deck, and then with a circle path as reference. The ball's position and speed in addition to the control signal, $\Delta\theta$, were logged to a file. After training of the membership functions, Matlab's ANFIS generates a Fuzzy Inference System (FIS), which is a proposed fuzzy system. The only user input to this operation is the number of membership functions and their form. The error tolerance was set to 0.01 and 30 époques of data were used. The resulting FIS was then implemented in the application code. To compare the performance of the modal and fuzzy regulators, a step response was applied to the control loop. The centre of the deck is the reference point, and at the start the ball is placed 25 cm from the centre. The results are shown in figure 4.3 below:

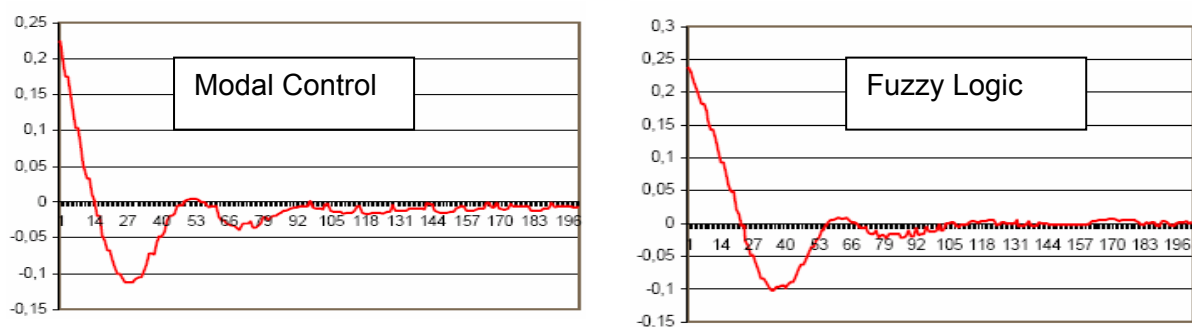


Figure 4.3: Step response of ball on the platform. The time scale is 25 frames/sec. The curves show little difference in the responses, but the modal control is a little bit faster.

² Copyright(c) 2001, Louder Than A Bomb! Software.

³ ANFIS - Adaptive Neuro-Fuzzy Inference System

A reference signal equal to a circle path was also tested. The actual deviation from the circle reference was logged during the control session, see fig 4.4 below:

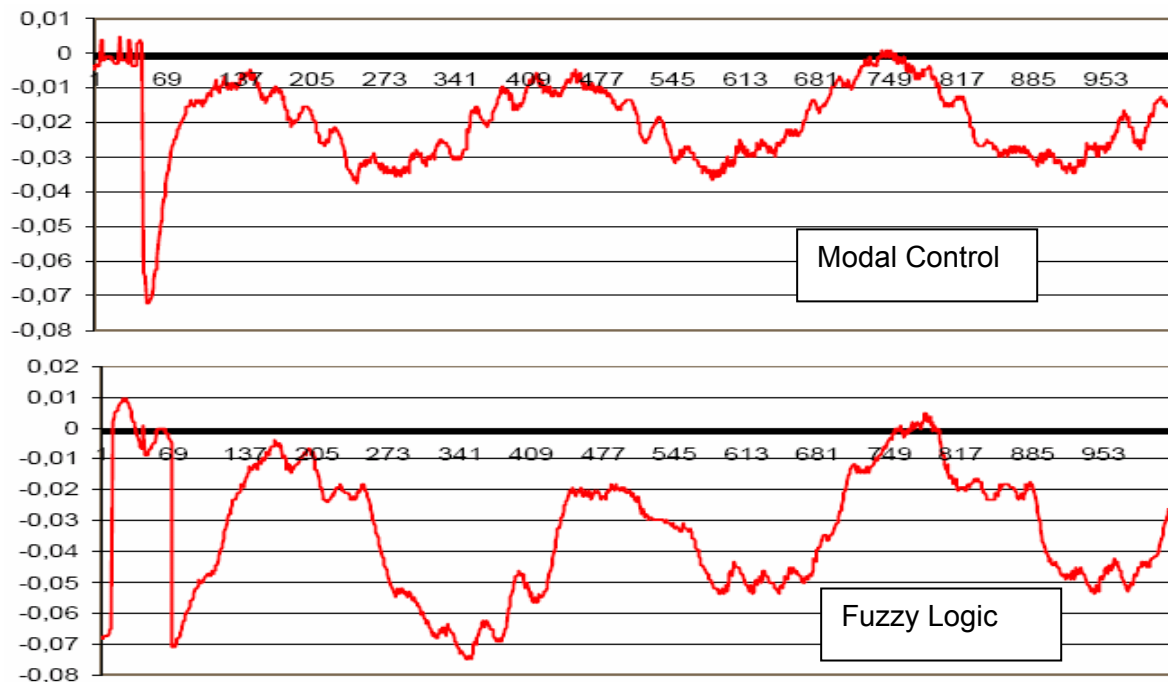


Figure 4.4: Deviation from circle reference in one axis for the ball. The time scale is 25 frames/sec, and the deviation is in fractions of a meter.

4.6 Comments on stability

A linear model is derived for the rolling ball on the inclined deck, and this open unstable system is made stable through a modal feedback control. This model is, however, only a part of the dynamics constituting the control system. The platform itself has its own dynamics, which is not accounted for in the control loop. It is taken for granted that the platform behaves in a static way giving a proportional response to a certain input signal. In reality the platform has both nonlinear characteristics and time delays. This is especially serious in relation to the time constants of the rolling ball and the sampling times required to localize it, approx. 40 msec. To a certain degree a neural network can be trained to cope with this. The network used, was simple in the way that it only used momentary values of states and control signal, it did not utilize earlier values. A recursive network could have made better predictions of the next platform angle, a more correct control signal and thus a more stable system. The fact that the different control strategies are functioning is, however, a proof that the theoretical assumptions are not too far from reality.

5. Conclusion

The purpose of building this model platform was to investigate the possibilities to design a full-scale ship motion simulator. It was important to get acquainted with the movements of such a platform and to test possible ways to control it. At first the building and mathematics was at focus, but soon the different control methods and strategies became the challenge. Much effort was laid down to make the modal control work properly. After this success, control strategies became the most interesting subject, and much work has been done to design and implement the best possible neural network and fuzzy regulator. A problem has been the physical condition of the model with its nonlinearities and non-ideal behavior, but this is also the reason the model has been such a challenge to control. This work has been an important activity in our control laboratory, and it has been accomplished as student projects under the guidance of the teachers. Through these projects the students have got a possibility

to test the theories they learn in their courses in cybernetics an real-time programming. The scientific staff too has earned much experience and insight from these projects. This has been valuable in the construction of the full-scale simulator.

6. References

- [1] Stewart D.: A Platform with Six Degrees of Freedom. The Institution of Mechanical Engineers, Proceedings 1965-66, 180 Part 1, No. 15, pages 371-386.
- [2] Merkle Ralph C.: A New Family of Six Degree Of Freedom Positional Devices. Nanotechnology. Vol 8. No 2. June 1997, pages 47-52.
Web site: <http://www.merkle.com>
- [3] Servos & Simulation inc.: Six Axis Motion System.
Web site: <http://www.servos.com/domframe.htm>
- [4] Rekdalsbakken W.: Student projects at Aalesund University College:
Building a Ship motion simulator (2001).
Controlling a motion platform in three degrees of freedom with National Instruments LabView (2003).
Controlling a rolling object on the motion platform with Visual C++ and Euresys eVision (2003).
A High-speed Craft Simulator (2004).
Controlling a rolling object on the motion platform with National Instruments LabView and IMAQ Vision (2004).
Controlling a rolling object on the motion platform with a neural network (2004).
Controlling a rolling object on the motion platform with fuzzy logic (2004).
- [5] Michael Negnevitsky: Artificial Intelligence. Pearson Education Limited (2002). Addison-Wesley.
- [6] sourceForge.net: Free Fuzzy Logic Library. Web site: <http://ffll.sourceforge.net/index.html>
- [7] The MathWorks: Fuzzy Logic Toolbox.
Web site: <http://www.mathworks.com/products/fuzzylogic>