BALANCING BALL ON PLATE USING FUZZY LOGIC

PROJECT REPORT

Aseem Rastogi
Rizwin Shooja
Cezar Chiru

3/31/2013
# Contents

Abstract.................................................................................................................................................. 2

I. Introduction ........................................................................................................................................... 3

II. Problem Formulation ............................................................................................................................. 3

III. System Overview ................................................................................................................................. 4

IV. Approach and method .......................................................................................................................... 5

V. Implementation ..................................................................................................................................... 6

1. Components ......................................................................................................................................... 6

2. Hardware Implementation ....................................................................................................................... 6
   A. Plate and Motors ................................................................................................................................. 6
   B. Setting up the camera ......................................................................................................................... 7

3. Image Processing Implementation .......................................................................................................... 8

4. Motor Control ....................................................................................................................................... 9

5. Implementation of Fuzzy Controller ..................................................................................................... 10

6. Serial Communication ........................................................................................................................... 13
   A. MATLAB Implementation .................................................................................................................. 13
   B. AVR32 Implementation .................................................................................................................... 13

VI. Result ............................................................................................................................................... 14

VII. Conclusions ..................................................................................................................................... 14

VIII. References ....................................................................................................................................... 14
Abstract

An intelligent system is designed that aims at moving a ball from wherever it is placed on a plate to the center. The system includes a ball, a plate, two servo motors, a web camera and EVK1100 microcontroller board. The plate has two degrees of freedom in order to achieve this objective and its movement is controlled by using two servo motors placed perpendicular to each other. A webcam is used to capture the frames and image processing is done in MATLAB to find the position of the ball. A fuzzy controller is used to bring the ball to a stop at the center of the plate. EVK1100 microcontroller board is programmed to give the required pulse width modulated signal to the motors which controls the movement of the plate. In this report, we have detailed the problem description, the approach adopted to do this project and the various phases of implementation.
I. Introduction
The balancing ball on plate is an intelligent control system capable of controlling the movement of ball so that it comes to a halt at the center of the plate. The initial position of the ball will be captured by the web camera which is recorded as the initial error and both the motors are moved such that the ball moves to the center of the plate. We assume that the ball is only rolling on the plate without slipping. The new position of ball on the plate is a feedback to the previous movement of motors and again motors are moved to rectify that. This process continues until the ball comes to a complete halt at the center of the plate.

Before implementing this project, we have referred quite a number of literatures to see how researchers have done previously. We have studied similar kind of projects like balancing ball on beam which is basically controlling the movement of ball in only one dimension. Also, we have studied various techniques adopted to control the motion of ball like the PID controller and fuzzy logic controllers and we found that fuzzy logic is more effective and the state of the art technique used for this kind of applications. One of the major challenges that we faced in this project is locating the ball on the plate. The literatures that we referred show that people have used various techniques such as web camera, resistive touchscreens, and sensors etc. for finding the position of the ball on the plate. In this project, we have used a web camera to find the position of the ball which is relatively a good method of feedback for the position of the ball on the plate.

The system has a reaction time of 20 milliseconds to respond to changes on position of the ball as it is the period of the pulse width modulated signal given to the micro servo motors used. A physical system has been designed and overshoots are minimal unless the ball is thrown at a higher velocity on to the plate.

II. Problem Formulation
The problem defined here is to balance a moving or static ball from any part of the plate to it center. The ball should try to become stable at the center of the plate if it is thrown from different directions at different speeds. One can either throw the ball to the plate or place it at some point on it. We broke down the system into three main problems:

- Calculating the position of the ball
- Applying the fuzzy rules
- Controlling the motors using PWM signals
III. System Overview

The working of the system starts with capturing the position of the ball by using a web camera and the frames are sent over to MATLAB. In MATLAB, the captured frames are processed such that the required area is cropped out of the captured frames. Then the pixel values above a certain threshold are treated as black and below that as white. So the processed image is black and white with ball as white and the plate (background) as black. Now the center of the white circular area is considered the current position of the ball. The error in position as well as speed is calculated and fed into the fuzzy controller. The fuzzy controller calculates the required pulse width modulated (PWM) signal for the motors and send it to the EVK1100 microcontroller board via a USART interface. The signals to the microcontroller board are decoded and the respective PWM signals are given to the servo motors which tilt the plate accordingly. The new error in position is recorded by the web camera and sent it to MATLAB. This process continues until the ball comes to a halt at the center of the plate.
IV. Approach and method

The various phases involved in this project are described below.

Phases of project:

| Phase 0 (Methodology) | Preliminary meeting  |
|-----------------------|----------------------|-------------------|
|                       | Master plan or strategy |
|                       | Feasibility study    |

<table>
<thead>
<tr>
<th>Phase 1 (Analysis and design methodology)</th>
<th>Data + Objectives = Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Simulations</td>
</tr>
<tr>
<td></td>
<td>Development</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phase 2 (Development)</th>
<th>Physical system</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Assembly</td>
</tr>
<tr>
<td></td>
<td>Tests</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phase 3 (Final)</th>
<th>Physical system</th>
</tr>
</thead>
</table>

Phase 0 (Methodology):

The first phase of the project includes a preliminary meeting with all the persons involved in the project in which we discussed and analyzed the objective and requirements of the project and then developed an appropriate plan for carrying out the project.

Also a feasibility study has been carried out to decide if based on our resources the project can be completed successfully.

Phase 1 (Analysis and design methodology):

We analyzed the objectives of the project more thoroughly and we gathered the necessary data that we used to achieve the goals of the project.

Based on the gathered data we did some simulations. For example for the web camera we simulated the capture of frames in MATLAB. For the servo motors we simulated a program that would give a PWM signal that would move the motor axis from one end to the other (180 degrees).

Phase 2 (Development):

After carrying out the simulation and obtained satisfying results we started to build the physical system. We started with the mechanical part. We built the 2 boards (base and moving board) and the assembly to connect them. After that we fitted the motors and their connections to the
plate. We built the frame that holds the web camera and connected with the existing system. After all parts have been fitted we calibrated the system. We wrote the necessary code for image processing and control of the motors.

We then did some test on the assembly to see if it meets the requirements. We tested the parts individually motors, web camera, serial connection. After that we started to test them together and everything came out fine.

Next we designed the fuzzy rules and we implemented them in MATLAB.

**Phase 3 (Final):**

We then tested the whole system. We ran multiple tests with ball moving from different positions and at different speed. The test had as outcome that the system reacted as in the requirements.

**V. Implementation**

1. **Components**
   2. 30cm X 30 cm square plate.
   3. 2 X 9g servo motors
   4. Web camera
   5. EVK 1100 board
   6. USB to serial cable
   7. MATLAB
   8. AVR32 Studio

2. **Hardware Implementation**
   The various steps involved in hardware implementation are as follows:

   A. **Plate and Motors**
      
      **Step 1:** The square plate is drilled with a hole at the centre and is mounted on a screw which is attached to the base. The plate should be able to move freely in all directions.
      
      **Step 2:** The motors are fixed to the base such that the two motors are perpendicular to each other. Also the arm of the motor should be directly in line with the plate’s axis.
      
      **Step 3:** The joints which connect the arm from motors to the plate are made. These joints provide two degrees of freedom in movement. This is required to avoid rotatory motion of the plate about the centre when only one motor is moving.
Step 4: The joints are attached to the plate using glue such that the joints are at the same distance from the plate centre as the motors are from the centre of screw at the base. The joints should also be on the x and y axes of the plate.
Step 5: The joints are attached to the arm of the motor using the screw.
Step 6: The movement of the plate is tested.

B. Setting up the camera
Step 1: First we turn on the web camera and measure the height at which it should be mounted so that whole of the plate is in frame.
Step 2: An arch structure above the plate is created where the camera can be mounted. The arch should be directly over the centre of the plate.
Step 3: Next we find the point on the arch that is directly overhead the centre of the plate.
Step 4: The camera is mounted on the arch using the glue such that the camera is directly over the centre of the plate. Also the camera should be mounted such that the plate is not rotated with respect to the camera frame.
Below is an image of the setup:
3. **Image Processing Implementation**

This requires getting the image from the camera and extracting the co-ordinates of the center of the ball from the image. The image acquisition and processing is done in MATLAB.

**Step 1:** For the camera to distinguish between the ball and the plate the plate is covered with a black paper and the ball is painted white to provide a good contrast.

**Step 2:** Configure the camera to capture a grey scale image of size 640x480 pixels. Also set the frame rate and trigger type as manual for each frame.

**Step 3:** Each frame that is captured is got into an array along with the time at which it was captured. The captured image is shown below:

![Captured Image](image-url)

The image shows a blue square of 400x400 pixels which encloses the plate and the centre of the square is at the centre of the plate.

**Step 4:** The frame is cropped into 400x400 pixel image so that the image only contains the plate and the ball.

**Step 5:** The cropped image is then applied with a grey threshold such that all values above that threshold are treated as white i.e. 255 while all below that are black i.e. 0.

**Step 6:** Next all white spots smaller than the size of the ball in pixels are removed so that we have only one white blob in the image. The final processed image is shown below:
In the image we can see the ball as a white spot.

**Step 7:** The centroid of the blob is then calculated in x and y co-ordinates. The centre of the plate is given co-ordinate as (0, 0). The x and y ranges are then in the range of +200 to -200 pixels

**Step 8:** The speed of the ball in each of x and y direction is calculated by finding out how much the centre of the ball has moved since last frame and dividing it by the time elapsed between the frames.

### 4. Motor Control

The motor is controlled by giving PWM (Pulse width Modulated) signals from the EVK1100 board. The steps in implementing control for motors are:

**Step 1:** Initialize the PWM module in AVR32.

**Step 2:** The clock used is 12 MHz clock with a prescaler of 32. Hence the effective clock rate is 375 kHz.

**Step 3:** The servo motors require a periodic signal with period of 20ms. Hence the period for the PWM signal is set to 375000 * 20 / 1000 = 7500.

**Step 4:** The motor is controlled by adjusting the pulse width of the signal. This is shown below:
As we can see that the full movement of the motor is from 1ms to 2ms which is the range 375 to 750 when the period is 7500. But full movement of the motor is not required for controlling the ball on the plate. So we first make the plate horizontal and note down the PWM value. Then by experimentation we found the range of PWM signal required for plate movement. It is found to range from centre (PWM value – 130) to (centre PWM value+130).

5. Implementation of Fuzzy Controller
The fuzzy controller is implemented using fuzzy toolbox in MATLAB. The steps in designing the fuzzy rules are as follows:

**Step1:** Since the plate is symmetric about x and y axis hence we just need one fuzzy design. The output values for x and y can be calculated by passing the respective input values.

**Step 2:** To stop the ball at the centre of the plate we need to know the position and speed of the ball. Hence, the inputs to the fuzzy controller are decided as:

**a) Error in position (e):** This is the distance of the ball from the centre position. Since the centre of the plate is denoted as (0, 0) hence the co-ordinate of the ball centre is the error in position. It can vary from -200 to +200. The membership functions of error are shown below:
The error has 5 membership functions namely NL (Negative Large), NS (Negative Small), Z (Zero), PS (Positive Small) and PL (Positive Large). Here we see that Z is a trapezium with flat area from -10 to +10. This is to take into account the errors in the calculation of the centre of the ball.

b) Change in error (de/dt): This tells us how fast the ball is moving. For x axis, if the ball is moving in +x direction de/dt is positive else it is negative. The range of de/dt is -650 to +650. Here 650 pixels/second is the maximum speed that the ball can attain at the centre of the plate and the ball will just stop at the edge if the plate is tilted maximally in the opposite direction. The membership functions for de/dt are shown below:

The change in error has 5 membership functions namely NL (Negative Large), NS (Negative Small), Z (Zero), PS (Positive Small) and PL (Positive Large). Here we see that Z is a trapezium
with flat area from -50 to +50. This is to take into account the errors in the calculation of the speed of the ball.

**Step 3:** The output of the fuzzy controller is the PWM signal which should be given to the motors. Hence the output has the range of -130 to +130. This is added to the PWM value of the horizontal plate to get the PWM value of the signal. The membership functions of the output are shown below:

![Membership Functions](image)

The change in error has 7 membership functions namely NL (Negative Large), NM (Negative Medium), NS (Negative Small), Z (Zero), PS (Positive Small), PM (Positive Medium) and PL (Positive Large).

**Step 4:** Next the rules mapping input and output are defined. The rules are shown below:

<table>
<thead>
<tr>
<th>e</th>
<th>NL</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PL</th>
</tr>
</thead>
<tbody>
<tr>
<td>de/dt</td>
<td>NL</td>
<td>NL</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
</tr>
<tr>
<td>NL</td>
<td>NL</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
</tr>
<tr>
<td>NS</td>
<td>NM</td>
<td>NM</td>
<td>Z</td>
<td>PM</td>
<td>PM</td>
</tr>
<tr>
<td>Z</td>
<td>NM</td>
<td>NM</td>
<td>Z</td>
<td>PM</td>
<td>PL</td>
</tr>
<tr>
<td>PS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PL</td>
</tr>
<tr>
<td>PL</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PL</td>
<td>PL</td>
</tr>
</tbody>
</table>
The first rule states that if the error is NL and the change in error is NL then the output is NL. The graph below shows the surface of output with inputs as 2 axes.

6. **Serial Communication**

The PWM values output from the fuzzy controller are sent over serial cable to the EVK1100 board. The implementation requires sending data from MATLAB and receiving data in AVR32.

**A. MATLAB Implementation**

**Step1:** Initialize the serial port with baud rate = 9600, Data bits = 8. Open the serial port for communication.

**Step2:** Convert the output values from fuzzy controller from range -130 to +130 to 0 to 260.

**Step3:** Send a character which has the ASCII value of the output over the serial port. The x output value is sent followed by the y output value.

**B. AVR32 Implementation**

**Step 1:** Initialize the USART on AVR32 with same parameters as on MATLAB.

**Step 2:** The interrupt is enabled for USART and the interrupt function reads from the USART whenever the data is there. We receive the output values for x and y alternatively which is
then added to the PWM value for the horizontal plate for respective axes and then sent to the respective PWM channel for output to motor.

VI. Result
We were able to bring the ball to a halt at the center of the plate. The ball was thrown from different directions and different speeds and in each case the plate reacted and made the ball stop at the centre.

VII. Conclusions
Our hypothesis is that the system would have reacted much better if the plate that we have used would have been flat. Also in the middle of the plate there was a raised point from the assembly that kept the plate still, which made the system react in some situation for longer than what we expected.

Other than that the system reacted very well to the tests and each and every time the ball stopped in the middle of the plate, even if it took it longer in some situations because of the mechanical assembly inequalities.

An interesting future study would be to use better mechanical parts and try to make the plate tilt even more so it can react to some higher speeds of the ball.

VIII. References


