GUI based Control Scheme for Ball-on-Plate System using Computer Vision

Akanksh Kasula
Department of Computer Engineering
Rochester Institute of Technology
Rochester, NY, USA
Email:ak2755@rit.edu

Palash Thakur
Department of Electrical and Electronics Engineering
Manipal Institute of Technology
Manipal, India
Email: palash.thakur@learner.manipal.edu

Mukund Kumar Menon
Department of Instrumentation and Control Engineering
Manipal Institute of Technology
Manipal, India
Email: mukund.menon@manipal.edu

***Abstract*—Ball-on-Plate system, an example of nonprehensile robotic manipulation, constitutes a formidable control problem as well as an excellent testbench for complex control algorithms. The authors have developed a prototype of a linkage based Ball-on-Plate system with Computer Vision Sensing, and an accompanying GUI based controller implemented on MATLAB, with the capability to implement various control schemes. Position control was tested using a PD Controller, and a novel fuzzy inference control scheme has been developed.**

Keywords - GUI, control, computer vision, fuzzy control, ball on plate, position control, nonprehensile

I. Introduction

The ball-on-plate (BnP) constitutes a 4 Degree-of-Freedom system, consisting of a ball that can roll freely in two dimensions upon a rigid plate. The ball position is controlled by the manipulation of the plate inclination in two rectangular directions. The BnP is an example of a nonprehensile manipulator - manipulators capable of maneuvering objects without physically grasping them [1]. While these class of manipulators have the advantage of being able to physically move heavy/large objects with relative ease, they represent a difficult control problem as the state equations for such manipulators are based upon the geometry of the object.

Introductory courses to Control Theory focus on simple and linear systems. Many real world and industrial examples of systems to be controlled are non-linear, inherently unstable and can only be affected indirectly. The barrier thus noticed could only be bridged with the development of a multidimensional control testbench for educational purposes. The BnP system will provide users with a platform to test and compare various aspects of control systems without getting caught up in lower level hardware development. This will also add an intermediate step in the control design for high level systems, particularly in verifying the efficiency of the algorithm in practical conditions before risking expensive and delicate robots.

In a previous work, the authors built an obstacle clearing hexapod. However, the control schemes required for locomotion had to be tested directly on the system due to the lack of a hardware control testbench. This led to the search for a suitable testbench, leading to the BnP system.

II. Literature Review

While the ball-and-beam (BnB) platform (a problem centered on balancing a ball along a rigid beam) has seen substantial research interest, the more complex ball-on-plate (BnP) platform has not attracted as much attention. Nevertheless, a significant amount of research has focused on the BnP. Awtar, Bernard et al. [2] have built a BnP system using a novel spatial linkage mechanism for actuation and a resistive touchscreen for sensing. They have explained the merits and demerits of various ball sensing systems, including touch screen, infrared sensor grids, Computer Vision and an infrared/ultrasound transponder attachment to the ball itself, which serves as a useful guide for sensing system selection.

Another paper of interest is by Knuplez, Chowdhury et al. [3] who have developed a BnP system using a discrete lead compensator as the controller. They have used Euler-Lagranges equations to mathematically describe the dynamics of the system, and provided important assumptions to take into consideration for mathematical modelling and linearization of the system. The authors used these assumptions to linearly simplify their own BnP system. Moarref, Saadat, and Vossoughi [4] have provided a heuristic technique for sensing ball position using MATLAB, however the paper does not specify the ball sensing performance.

Espersson [5] has experimentally performed a performance comparison between different methods for ball position acquisition for a Ball-and-Beam system. The authors have used this paper for reference when experimentally evaluating processing performance. Djordjevich [6] compares a PD controller and a Fuzzy Inference Engine controller. The paper confirms the authors hypothesis on performance improvement with the addition of a soft technique.

Many publications implementing BnP systems have focused on BnP systems purely from the point of view of a challenging design problem, or as a benchmark for testing control systems [4] [7] [3]. However, although a few of these publications have suggested the use of BnP systems as educational tools [8], the overall implementation of BnP systems for this purpose has been limited. Hence an educational and demonstrable tool, for both linear and soft control is needed.

978-1-7281-0255-9/18/$31.00

III. Overview and Mathematical Model

As given in figure 1, the ball-on-plate system developed by the authors consists of multiple subsystems. The mechatronic system at the core consists of a smooth plate whose orientation can be controlled. An overhead camera captures images of a free moving ball on the plate, and an associated computer vision algorithm extracts coordinates of the ball with respect to the plate. The controller can use the coordinates as an input, and upon comparision with the goal coordinates set by the user, generate corresponding motor angles. The subsequent motion of the ball, provides updated input into the vision system, closing the loop. This process continues till convergence.

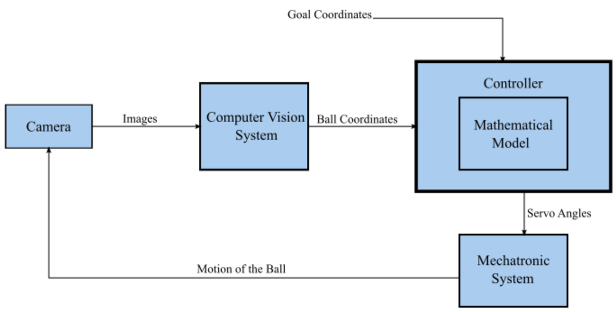

Fig. 1: System flow diagram

A. Dynamic Equations

The system can be modeled using Euler-Lagrangian equations, which are as following [2] [9]:

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{q_i}} - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i \quad (1)$$

Where q_i stands for i-direction coordinate, T is kinetic energy of the system, V is potential energy of system and Q_i is composite force. The system has 4 degrees of freedom; two in ball motion and two in inclination of plate. Here we assume the generalized coordinates of system to be x_b and y_b for ball's position on plate and α and β for the inclination of the plate. Subsequently, non-linear system equations can be obtained.

The system is linearized about its operating point. To linearize the system, we take the following simplifications [3]:

- Small angle of inclination for the plate: $\alpha \ll 1$ and $\beta \ll 1 \implies \sin\alpha \approx \alpha$ and $\implies \sin\beta \approx \beta$
- Slow rate of change for the plate: $\dot{\alpha} \approx 0$ and $\dot{\beta} \approx 0 \implies \dot{\alpha}\dot{\beta} \simeq 0$, $\implies \dot{\alpha}^2 \simeq 0, \dot{\beta}^2 \simeq 0$.
- Solid ball with moment of inertia = $\frac{2}{5}m_b r_b^2$, where m_b and r_b denote the mass and radius of the ball respectively

$$\frac{5}{7}m_b\ddot{x}_b + g\alpha = 0 \quad (2)$$

$$\frac{5}{7}m_b\ddot{y}_b + g\beta = 0 \quad (3)$$

IV. Implementation

A. Mechatronics

This section will deal with the construction of the Ball-on-Plate system. The plate is a square white acrylic sheet (aluminum reinforced) of side 1 ft (30.46 cm). At the center of the plate, a universal joint is attached. The universal joint, as well as the plate itself are detachable from the frame. The universal joint slots into an extender bar which is attached to the particle board base.

The linkage went through multiple iterations before being finalized. The authors initially went with a 3 bar linkage system. However, upon testing the motion of the system via single axis motor sweep tests, plate torsion (up to 20 degrees) was observed, therefore leading to a dependence of change in plate angle in one axis upon the other axis. This was solved by adding a passive joint to the crankshaft, thus the 3 bar linkage was converted into a 4 bar linkage. For actuation, RKI-1211 [10] digital servo motors were selected.

In Fig. 2, The Universal joint and extender bar that the plate rests on can be seen on the right, and crankshaft and bar linkage can be seen connected to the servo motor on the left.

Fig. 2: Improved Linkages

A camera stand with adjustable height was constructed. A Logitech HD C270 webcam [11] has been attached to the camera stand, directly over the plate. A Raspberry Pi 3B SBC has been used for motor control.

The actuation of the servo is done directly from MATLAB R2017b, linking to the Raspberry Pi. Two versions of the control code were written and implemented in MATLAB R2017b. The first code, a non-GUI based implementation, was primarily used to test the sensing and corresponding motor control. The second code is the GUI based implementation.

B. Computer Vision

For initial testing of the Computer Vision system, images of a ball placed on the plate were taken by an external camera and the algorithm was run, with satisfactory results. This was extended to a video with 25 frames, and the centroid of the ball in each frame was calculated, thus extracting the trajectory of the ball.

10 trials were obtained, and video was processed for the ball centroid using the CV algorithm. A timer was run through software to evaluate processing time.

The initial design of the computer vision algorithm was inspired by the work of Moarref [4]. The region of interest of the camera frame, consisting of the plate and the ball placed on it, are binarized. This is followed by the summation of pixel values across each row (column) and finding the maxima, through which the y - coordinate (x - coordinate) of the centroid can be acquired. Summation is localized to maximize performance. Upon testing using the above method, the average processing time was found to be 0.328 seconds per 25 frames. However, this method was found unsuitable, as improper cropping, along with a darker plate background, led to incorrect centroid detection.

To circumvent these disadvantages of the previous algorithm, the authors switched to an implementation based on Circular Hough Transform (CHT). This approach is used because of its robustness in the presence of noise, occlusion and varying illumination. Since the CHT scans for circles in the image, any incorrect cropping of the image will not have a detrimental effect upon accuracy. In a similar manner as for the previous algorithm, the performance was calculated for 10 iterations, achieving an average processing time of 0.434 seconds per 25 frames. Though the speed decreases slightly, the authors chose to use CHT as it eliminates the disadvantages listed above.

C. Control

The mathematical model presented in section III is obtained by the linearization of nonlinear equations by the use of several standard assumptions [2] [9]. It is of interest to note the fact that the above equations represent the nonlinear system in x,y into two decoupled linear systems in x and y separately. Each of these decoupled equations is similar to that of a 2 Degree-of-Freedom Ball-and-Beam system, which is a classical control problem [12] [13].

With two decoupled systems, the controller implementation can also be split to form two independent controllers. The authors chose to implement a separate controller to control the position of the ball on the X and Y axes of the plate respectively, working in parallel.

As seen from Malini [12] and Ortiz's prior work [13] , the implementation of a classical PID controller is feasible on a Ball-and-Beam system. The authors chose not to add an Integral term, as the addition of error with a fast moving ball adds up very fast, contributing to high overshoot.

Preliminary simulation of the proposed PD controller was done on MATLAB. The plant model as derived in Section III was implemented in the laplace domain.

In the first iteration of the controller design, the proportional and derivative constants obtained from the MATLAB PID Tuner tool were used. The system response was appropriate with respect to the direction of actuation, but was completely unstable. The constants were henceforth iteratively modified until the best performance was achieved.

D. GUI

A GUI was developed with the controller in MATLAB. The GUI is used as follows:

1) The user enters in their values of K_p and K_d and then initializes the system
2) The user can select the plate boundary and a goal point from a snapshot.
3) The system then connects wirelessly to the Raspberry Pi and sets up the motors. Control operation is initiated with a Run button press. Pause and Reset Goal functionality is also implemented.
4) The system automatically pauses when the ball slides off the plate, and the user is given the option to place the ball back on the plate and re-initiate system operation.

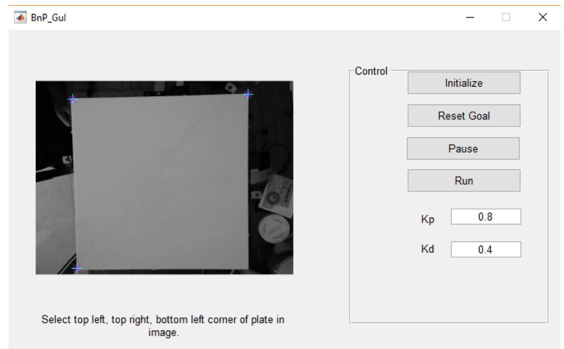

Fig. 3: GUI: Selection of region of interest

V. Result Analysis and Further Work

A. PD Control

PD controller was built using the MATLAB control system toolbox. The developed system model was made closed-loop and tuned using the PID tuner tool, with acceptable design criteria being set at a settling time of less than 5 second and an overshoot of less than 5 %. The simulation achieved stability with parameters as follows:

- Rise Time - 2.09 Sec
- Settling Time - 3.36 Sec
- Overshoot - 1.56 %

Fig. 4: Ball balanced on plate

Due to the inherent nonlinear characteristics of the Ball-on-Plate system, it was to be expected that the actual implementation of the controller would not provide such ideal results. However, the authors implemented and displayed the PD controller, not to control the position of the ball but to demonstrate various control parameters and phenomena like overshoot and oscillations. This was aimed towards the stated application of the system as an educational and demonstrational tool.

At this point in the design process, the system was closed-loop stable, but goal position tracking was not achieved. Disturbance rejection was poor, with even minor disturbances like wind and external vibrations destabilizing the system.

B. Fuzzy Control

To achieve better tracking and disturbance rejection a controller with Non-Linear characteristics was needed, such as by using nonlinear compensation, H-Infinity Control, Sliding Mode control etc. However, the differences between the ideal mathematical models and the implemented system as well as the complexities in design and implementation make these techniques prohibitive.

Due to their probabilistic nature, the use of stochastic techniques were of particular interest to the authors. The use of Artificial Neural Networks requires extensive training, which is quite tedious , and Genetic Algorithms have a slow convergence rate, which can be counter productive. The development of Fuzzy algorithms is done through human knowledge, and is intuitive, and hence it was chosen.

Conventional fuzzy systems would involve direct control of the plate angle, making use of the nonlinear mathematical model [6]. However, due to the pre-implemented PD system, The authors chose to develop an adaptive Fuzzy PID system.

The governing equation of PID control is modified as follows [14]:

$$U(t) = (K_p + \Delta K_p)e(t) + (K_i + \Delta K_i)\int_0^t e(t) + (K_d + \Delta K_d)\frac{de(t)}{dt} \tag{4}$$

Where the delta terms are the change in Tuning constants, which are the outputs of three fuzzy inference systems. The inputs to each fuzzy inference system are chosen to be the error between the current position and the goal (e) and the difference between current and previous errors ($\dot{e}$). The outputs are the change in K_p, K_i, K_d respectively.

$\Delta\ K_p$, $\Delta\ K_d$, was designed to have both inputs and outputs with seven fuzzy membership functions while $\Delta\ K_i$ has five membership functions on the input side, and three on the output. The membership functions are chosen to be overlapping and gaussian in shape for generating a smooth surface. The design is done in the intuitive Mamdani model, and then converted to the computationally efficient Sugeno model.

The fuzzy rulebase for all three systems was designed as follows:

- e * $\dot{e} > 0$ implies that the ball is moving away from the goal. Inversely, e * $\dot{e} < 0$ implies that the ball is moving towards the goal
- For large $|e|$, K_p needs to be large, in order to push the ball towards the goal, but for large $\dot{e}$ K_p can be slightly decreased, as the ball is already moving fast.
- Slow moving ball with larger $|e|$ needs a higher K_d ,and vice versa
- K_i should only act as the ball nears the goal position, and it can be slightly higher if the ball slows down

Based on the salient points above fuzzy rulebases were designed and implemented in MATLAB. An isometric view of the surface hence obtained for ΔK_p can be seen in Fig. 4.

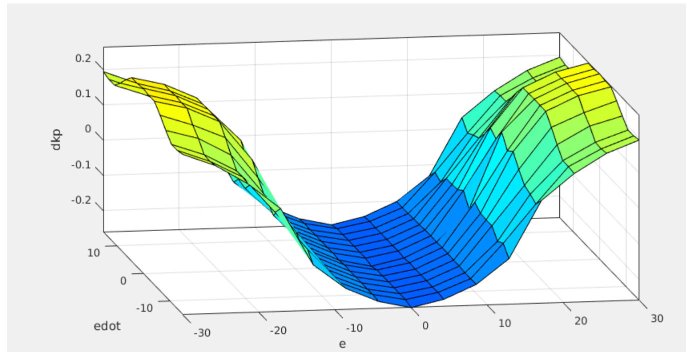

Fig. 5: Surface plot for ΔK_p

For ideal performance of any fuzzy controller, smooth surfaces are desired [6]. Any crispness in the surface can cause the output to vary rapidly when the system input as at the edge bounday, adding unwanted variations and jitter. It can be noted that in the crispness can be seen in the surface obtained due to the small number of fuzzy sets. For better performance, a much more exhaustive set of fuzzy inputs and tables is being designed. Real-life performance testing and comparison will be done in the future.

VI. Conclusion

A mechatronic Ball-on-Plate system was developed and constructed over several iterations. The choice of Computer Vision as the sensing modality provides acceptable ball position detection at an affordable price compared to touch screen sensing. The mathematical model derived is a linearized simplification of a complex non linear and unstable system. While it provides advantages in the form of simplicity and the capability of testing classical control algorithms, the accuracy of the model is affected due to various assumptions made during the linearization process.

An over-wrapping GUI was developed and implemented to enable the system to be used as a desktop evaluation and comparison tool, with an initial implementation of a PD controller for each axis. Closed loop stability was achieved. To further demonstrate the use of the BnP system as a testbench, an Adaptive Fuzzy PID algorithm was developed.

Despite the PD controllers poor performance in terms of disturbance rejection and tracking, the research is successful insofar as it serves to prove the viability of the GUI-based system as a control system testbench, and as an educational tool. As such, in the future, multiple control algorithms can be implemented on the device, and integrated with the GUI.

REFERENCES

[1] K. M. Lynch and M. T. Mason, "Dynamic nonprehensile manipulation: Controllability, planning, and experiments," *The International Journal of Robotics Research*, vol. 18, 1999.

[2] S. Awtar, C. Bernard, N. Boklund, A. Master, D. Ueda, and K. Craig, "Mechatronic design of a ball-on-plate balancing system," *Mechatronics*, vol. 12, pp. 217–228, 2002.

[3] A. Knuplez, A. Chowdhury, and R. Svecko, "Modeling and control design for the ball and plate system," in *IEEE International Conference on Industrial Technology*, vol. 2. IEEE, 2003, pp. 1064–1067.

[4] M. Moarref, M. Saadat, and G. Vossoughi, "Mechatronic design and position control of a novel ball and plate system," in *Mediterranean Conference on Control and Automation*, vol. 2. IEEE, 2008, pp. 1071–1076.

[5] M. Espersson, "Vision algorithms for ball on beam and plate," student Paper.

[6] A. Djordjevich, "Fuzzy motion control of the ball on beam," 2002.

[7] S. Kinoshita, T. Hirogaki, E. Aoyama, and W. Wu, "Robust visual servoing for motion control of the ball on a plate," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 2. American Society of Mechanical Engineers, 2013, pp. 653–659.

[8] A. Jadlovska and S. Jajcisin, "Modelling and pid control design of nonlinear educational model ball plate," in *17th International Conference on Process Control*, vol. 2, 2009, pp. 871–874.

[9] M. Nokhbeh and D. Khashabi, "Modeling and control of ball-plate system," Master's thesis, Amirkabir University of Technology, 2011.

[10] "Rki-1211 digital servo datasheet," https://robokits.co.in/motors/high-torque-metal-gear-standard-digital-servo, accessed: 2018-02-15.

[11] "Logitech c270 hd webcam details," https://www.logitech.com/enin/product/hd-webcam-c270, accessed: 2018-04-15.

[12] A. Malini and M. Prabhakar, "Modelling and control of ball and beam system using pid controller," in *International Conference on Advanced Communication Control and Computing Technologies*. IEEE, 2016.

[13] W. Yu and F. Ortiz, "Stability analysis of pd regulation for ball and beam system," in *IEEE Conference on Control Applications*. IEEE, 2005.

[14] V. Kumar and S. Ali, "Adaptive fuzzy pid for the control of ball and beam system," *International Journal of Engineering Research and Technology*, vol. 2, 2013.