A PROGRAMMABLE LOGIC CONTROLLER BASED LABORATORY -ANALYSIS OF CONVENTIONAL AND INTELLIGENT CONTROL SCHEMES FOR NON-LINEAR SYSTEMS

J.M. FERNANDES & T. VAN NIEKERK
NELSON MANDELA METROPOLITAN UNIVERSITY, SOUTH AFRICA

Abstract

Intelligent Neural Network (NN) based control schemes have surmounted many of the limitations found in the conventional control approaches such as Proportional Integral Derivative (PID) control. Nevertheless, these modern control techniques have only recently been introduced for use on industrial computational platforms such as the Programmable Logic Controller (PLC). Intelligent control on PLCs thus remains an area that is open to further research and development. In this paper, a strongly non-linear mechatronic-type system, namely the Ball-on-Wheel balancing system, is developed using a PLC as its control platform. The research details the implementation of an intelligent controller on a standard, medium specification PLC. The results from the intelligent controller are then compared to those produced by a variety of conventional controllers as physical parameters are varied. Finally, the system is presented as a stimulating educational tool that addresses the knowledge gap that exists in industry pertaining to the implementation of these intelligent control algorithms on PLCs.

Keywords: Process Control, Instrumentation, PID, PLC, Siemens, Sensors, Educational System, Laboratory, Neural Networks, Intelligent Control, Ball-on-Wheel

1. INTRODUCTION

Most industrial processes and all chemical processes are non-linear in nature (Nikolaou and Misra, 2001). As a matter of necessity, this often means the tedious process of system modelling and linearization cannot be ignored during the controller design phase. A negative aspect of system modelling and linearization is that costs can quickly escalate particularly in circumstances where the required measurement tools or skills are lacking. In industry today, various pre-programmed conventional control approaches such as Proportional-Integral-Derivative (PID), PD, PI, Lead and Lead-Lag schemes are used to control a variety of processes. In most cases, these controllers are tuned manually using rule of thumb methods.

However, because this manual approach leaves a lot up to guess-work, it could be both time consuming and unyielding and there would still be no guarantee of stability throughout the system's entire range of operation. Also, because conventional controllers are unable to adapt to changes in their immediate environment, they often have to be re-tuned to compensate for the parameter variation that may naturally occur in a system – most likely due to wear and tear of mechanical components. Failure to accurately re-tune these controllers could manifest in reduced plant efficiency, damage to equipment or worst of all, complete process and system failure. In the past decade, “intelligent” neural network (NN) based controllers have been used extensively in solving the more complex control problems. Unlike the conventional control approaches, integrated intelligent control with its parallel (non-linear) computational ability has the benefit of being able to solve any non-linear problem through learning and can therefore also adapt to an ever changing environment. This makes it desirable for use in the control of real dynamic non-linear systems (Ren and Rad, 2009). Although intelligent controllers have numerous advantages over their conventional counterparts, their use in industry on standard industrial computational platforms (such as motor drives and PLCs) has been significantly hampered because of their high computational demands and also owing to a general lack of knowledge in the field (Uddin et al., 2014).

In this paper, a non-linear mechatronic system namely the Ball-on-Wheel balancing system is developed. It may represent any complex Single-Input, Single-Output (SISO) system found in industry. The system makes use of a standard industrial automation PLC and servo motor drive provided by Siemens. To control the Ball-on-Wheel system in real-time, a number of conventional controllers, as well as an intelligent, online-learning Neural Network based PID controller, are developed and implemented on the PLC. The intelligent controller is designed to be user friendly in terms of its parameterization and configuration. It is also designed to use minimal computational resources. The Ball-on-Wheel system, on a broader scale, is intended to function as an educational laboratory for control systems classes. In this respect, it not only manages to capture the keen interest of students – but also aims to address the gap in knowledge that exists in the field of industrial automation with regard to traditional and intelligent controller design and implementation. The lab work follows a period of theoretical instruction that relates directly to the lab work but finds its wider application in real-world industrial problems. Simply put, the lab gives students a feel of what they will experience when solving real-life engineering problems in industry.

2. EDUCATIONAL SYSTEMS IN ENGINEERING

2.1 Importance of Educational systems in Engineering

An extensive body of research has established that technical content is most effectively learned through a process of deduction where learners are presented with the theoretical content followed by a section of practical application (Felder and Silverman, 1988). Active learning strategies that involve a significant degree of experimentation have thus been adopted by most engineering institutions around the world. Wikander et al. (2001), for instance, based on 15 years of teaching practice, reveal that mechatronic curriculums in particular should be problem-based, product design oriented, and project-team organized in a form similar to professional industrial product development and should include as many real situations as possible. They further suggest that engineering courses, as far as laboratory work goes, should focus on bridging the gap between individual course components such as software design, control and dynamic systems modelling and real-time implementation without overlooking the development of communication skills and teamwork experience among the students. Group projects that are structured so that they still allow individual assessment to be conducted are thus highly recommended. In addition, since 'design work' is considered to be the central or distinguishing activity of engineering, various educational works, including the work of Dym et al. (2005), express the need to develop design skills in students through project-based learning that entails a significant design aspect.

2.2 Motivation for PLCs in educational systems

Countless cost effective 'mechatronic' type labs have been developed by engineering institutions in light of the points already discussed. These labs include conveyor systems for material handling, robotic arms with grippers, stacking systems and steering applications just to name a few. The aim of most of these labs is to give students maximum exposure to modern electrical and computing technologies whilst taking up the least amount of lab space (Bassily et al., 2007; Craig, 1999). The widespread use of PLCs in industrial processes has motivated their use in many current mechatronic educational systems including some of those just mentioned. PLCs are usually performance graded and offer a rugged yet simple industrial solution to monitoring and control problems. The cost of PLCs generally increases with performance. Nevertheless, even lower performance PLCs are capable of performing relatively complex tasks including demanding maths functions, interrupt requests and data acquisition. They also usually come with a wide choice of graphical or text- based programming languages that relate closely to mainstream programming languages. Network connectivity for remote access and control is also a possibility with most standard PLCs (Saygin and Kahraman, 2004).

A major drawback is that the cost of PLCs compared to other programmable devices such as microcontrollers is significantly higher (up to 8 times for the lower performance range). Despite this, as industrial processes grow in size and complexity, the need for engineers with strong skills and knowledge in this area, who can propel industry in to the future, is great (Hsieh and Yeehsieh, 2004).

A definite skill gap exists when it comes to the digitization and implementation of complex control algorithms on PLCs. This is particularly true of 'intelligent' algorithms. Engineering courses that adequately prepare students, not only to fit in to industry but also to make meaningful contributions thereafter, are therefore of utmost importance. At a tertiary level this would imply exposing students to basic PLC concepts and subsequently to more complex applications that would direct them in thinking and solving problems just as engineers would in real-world applications. Throughout this process, strong emphasis would have to be placed on modern and futuristic trends in technology.

2.3 Non-linear balancing systems in education

Often, engineers are tasked with finding control solutions to complex non-linear problems. This includes the stabilization of naturally unstable systems. Good examples of such systems that are popular in laboratory setups include: the Ball-on-Beam balancing system, the inverted pendulum, the Ball-on-Plate system and the Hydraulically-Balanced-beam system. These systems are meant to be interesting in order to captivate the students, and when operational often fascinate onlookers (Craig, 1999). Although such systems often find no direct application, their underlying principles may be applied in understanding and solving real-world problems. Take for instance, the Ball-in-Hoop experiment in which a steel ball is free to roll on the inner surface of a rotating circular hoop. The system mimics the complex dynamics of the oscillations of liquid in a container when the container is moving and undergoing changes in velocity and direction. The significance of such a system in the real-world would be to limit the movement of large quantities of liquid which can strongly influence the movement of the container itself, leading to a potential disaster (Kanthalakshmi and Manikandan, 2010). Balancing systems, being complex in nature, also serve as reliable platforms for the investigation of intelligent control systems that can simplify the process of system modelling and controller design. Sufficient motivation thus exists for the development of laboratory-based balancing systems that incorporate current industrial technology.

3 EXPERIMENTAL SETUP

3.1 The Ball-on-Wheel balancing system

The Ball-on-Wheel system was selected for its complex non-linear dynamics (shown in the system equations below) and inherent instability. This system can thus represent any existing non-linear SISO system found in the real-world. The aim of the system is to balance various balls of different size, weight, and surface texture, on the top-centre of the wheel by controlling the applied torque (see Table 2). The apparatus consists of an aluminium wheel coupled to a servo motor via a tooth belt.

The servo motor is controlled by a Siemens servo drive which acts as a Profibus slave to a mid-performance S7-300 PLC. A picture of the setup is shown in Figure 1 and Figure 2. As shown in Figure 2, a laser distance sensor is used for feedback of the actual ball position. The wheel angle and applied torque are calculated within the drive and can be retrieved by the PLC over the Profibus network.

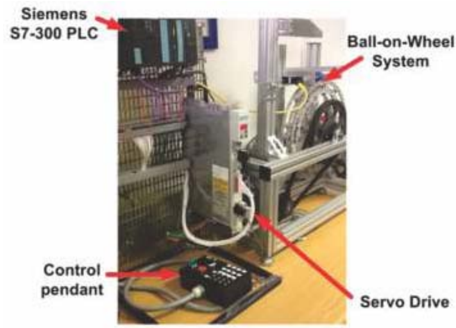

Figure 1: Ball-on-Wheel System

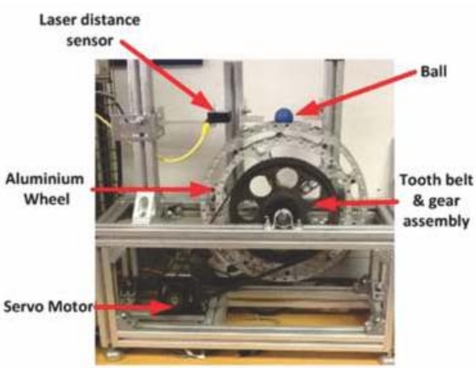

Figure 2: Ball-on-Wheel mechanical assembly

3.1 Preparing the Ball-on-Wheel system for use

The distance feedback from the laser sensor is calibrated to give the angular displacement of the ball on the periphery of the wheel. The wheel surface is rubberized to increase ball-wheel friction and slow down the natural response of the system to cater for processing delays in the drive and PLC. The laser distance sensor is wired to a high speed analogue module on the S7-300 PLC.

3.2 Control and data capture via Matlab & Simulink

In order to capture real-time experimental data from the plant for the purpose of analysis and comparison, the S7-300 PLC is linked wirelessly using Siemens wireless technology to a computer running Matlab/Simulink. OPC Server and Client software running on this computer then give Matlab exclusive access to all PLC inputs, outputs and memory areas. The entire system architecture including the network setup is shown in Figure 3.

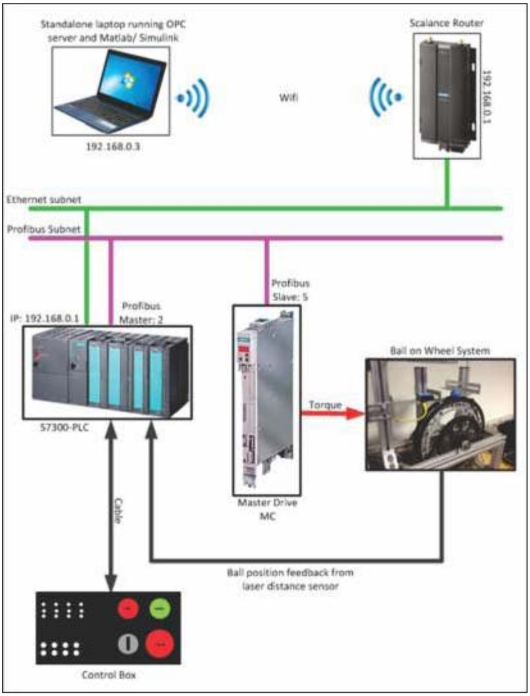

Figure 3: System architecture

Safety features on the system include an emergency stop button, automatic obstruction detection in the drive, visual alarms and warnings, and general user information.

3.4 Modelling the Ball-on-Wheel System

The Lagrangian energies technique is used to derive the Ball-on-Wheel system model. Figure 4 shows the free body diagram for the system.

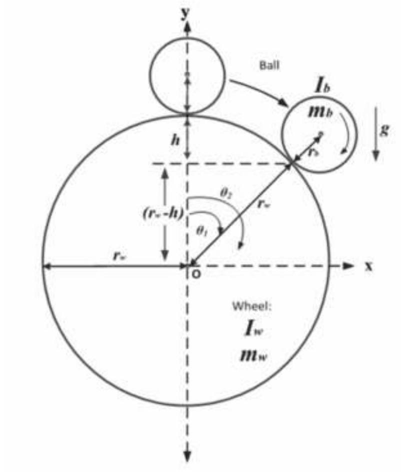

Figure 4: Free body diagram of Ball-on-Wheel system

The Ball-on-Wheel system has 2 degrees of freedom represented by Θ_1 and Θ_2 see Figure 4). For the dynamic model to be accurate, the assumption taken is that the ball is rolling on the surface of the wheel and not sliding (Ming-Tzu et al., 2009).

According to the Lagrangian equation:

$$\frac{d}{dt}\left[\frac{dL}{dq}\right] - \frac{dL}{dq} = Q$$

Where *L* = Lagrangian function, *Q* = Generalized forces of system and *q* = Generalized coordinates of system

Then:

$$L = T - V$$

Where *T* = the kinetic energy of the system and *V* = the potential energy of the system. Total kinetic energy possessed by the ball due to translational motion plus rolling motion is given as:

$$T_b = \frac{1}{2} m_b (r_w + r_b)^2 \dot{\theta}_1{}^2 + \frac{1}{2} I_b \dot{\theta}_3{}^2,$$

Where, the ball's moment of inertia is given by:

$$I_b = \frac{2}{5} m_b r_b{}^2$$

The kinetic energy possessed by the wheel due to rotation is given by:

$$T_w = \frac{1}{2} I_w \dot{\theta}_2{}^2$$

Where the wheels moment of inertia is given by:

$$I_w = \frac{1}{2} m_w r_w{}^2$$

Therefore the total kinetic energy possessed by the system is given by:

$$T_t = \frac{1}{2} m_b (r_w + r_b)^2 \dot{\theta}_1{}^2 + \frac{1}{5} m_b r_b{}^2 \dot{\theta}_3{}^2 + \frac{1}{4} m_w r_w{}^2 \dot{\theta}_2{}^2$$

Now since Θ_3 (the rolling angle of the ball) is not measurable, it must be expressed in terms of Θ_1 and Θ_2 giving

According to the Lagrangian equation:

$$T_t = \frac{1}{2} m_b (r_w + r_b)^2 \dot{\theta}_1{}^2 + \frac{1}{5} m_b (r_w \dot{\theta}_2 - r_w \dot{\theta}_1 - r_b \dot{\theta}_1)^2 + \frac{1}{4} m_w r_w{}^2 \dot{\theta}_2{}^2$$

The potential energy possessed by the system is given by:

$$V = m_b g (r_w + r_b) cos\theta_1$$

Therefore, according to (2),

$$L = \frac{1}{2} m_b (r_w + r_b)^2 \dot{\theta}_1{}^2 + \frac{1}{5} m_b (r_w \dot{\theta}_2 - r_w \dot{\theta}_1 - r_b \dot{\theta}_1)^2 + \frac{1}{4} m_w r_w{}^2 \dot{\theta}_2{}^2 - g (r_w + r_b) cos\theta_1$$

And according to (1), the simplified system dynamic equations are given as:

$$(7r_b + 7r_w)\ddot{\theta}_1 - 2r_w \ddot{\theta}_2 - 5g sin\theta_1 = 0$$

$$(-\frac{2}{5} r_w{}^2 m_b - \frac{2}{5} r_w r_b m_b)\ddot{\theta}_1 + (\frac{1}{2} m_w r_w{}^2 + \frac{2}{5} r_w{}^2 m_b)\,\ddot{\theta}_2 = \tau$$

These two equations are only true as long as the centripetal force is large enough to maintain the circular motion of the ball on the wheel. After defining the state variables, the plant is then linearized about its equilibrium point (i.e. when Θ_1 is equal to zero) using the Jacobian linearization technique. The system model is thus determined in state-space form as:

$$\dot{X} = J_A X + J_B U$$
$$Y = J_C X + J_D U$$

$$J_A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ N & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ M & 0 & 0 & 0 \end{bmatrix} \qquad J_B = \begin{bmatrix} 0 \\ R \\ 0 \\ P \end{bmatrix}$$

$$J_C = [1 \quad 0 \quad 0 \quad 0] \qquad J_D = 0$$

Where:

$$R = \frac{H}{EH-FG}$$
$$N = \frac{IF}{EH-FG}$$
$$P = \frac{G}{FG-EH}$$
$$M = \frac{IE}{FG-EH}.$$

And:

$$E = -\frac{2}{5}m_b r_w{}^2 - \frac{2}{5}m_b r_w r_b$$
$$F = \frac{1}{2}m_w r_w{}^2 + \frac{2}{5}m_b r_w{}^2$$
$$G = -7r_b - 7r_w$$
$$H = 2r_w$$
$$I = 5g$$

3.5 Ball-on-Wheel system analysis in MATLAB

With a default set of plant parameters selected as highlighted in Table 1, the system transfer function is given as:

$$G(s) = \frac{4.888e^{-15}s + 28.31}{s^2 - 5.329e^{-15}s - 32.74}$$

A step command to the input of the system yields an infinitely increasing response as the ball rolls off the surface of the wheel (see Figure 5) away from the zero degree equilibrium point.

Table 1: Default System parameters

Parameter	Value	Comment
m_w	2.5 Kg	Mass of wheel
r_w	0.195 m	Radius of wheel
m_b	0.1 Kg	Mass of ball (hard rubber ball)
r_b	0.025 m	Radius of ball
g	9.81 m/s^2	Acceleration due to gravity

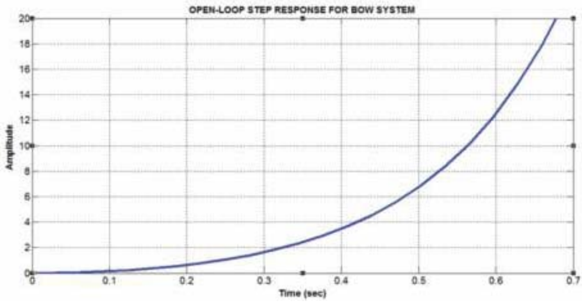

Figure 5: Uncompensated system step response

A closer examination of the system's dynamics reveals a pole in the right hand s-plane of the root-locus plot (see Figure 6). Consequently, this pole yields an unstable system.

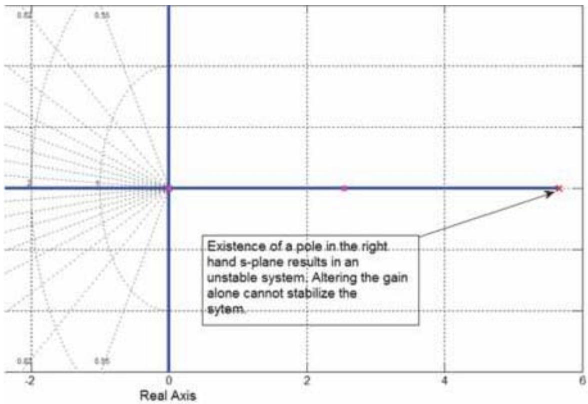

Figure 6: Uncompensated system root locus

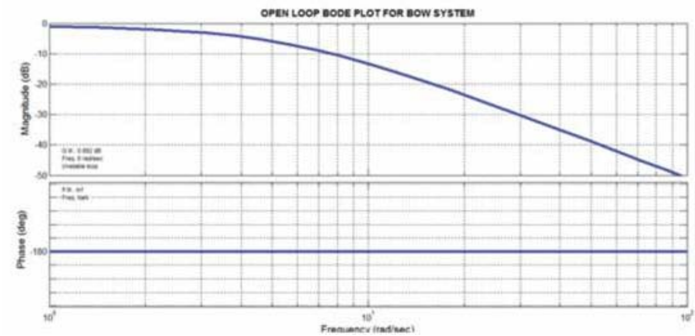

Figure 7: Uncompensated system Bode plots

The Bode plots for the uncompensated open-loop Ball-on-Wheel system (shown in Figure 7) reveal a very small gain margin of 0.892 dB and an infinite phase margin due to the phase never crossing -180 degrees.

3.6 Conventional Controller Design & Implementation

Using standard design principles such as frequency domain controller design, root locus controller design or a combination of these design techniques, various conventional controllers were developed in order to control the Ball-on-Wheel system. Controllers included a PID controller, a Lead controller and a Lead-Lag controller. For simplicity, of the three conventional controllers, only the PID controller is shown below (see Figure 8 and Figure 9). The PID controller is implemented on the PLC within a Function Block (FB) coded in Structured Control Language (SCL).

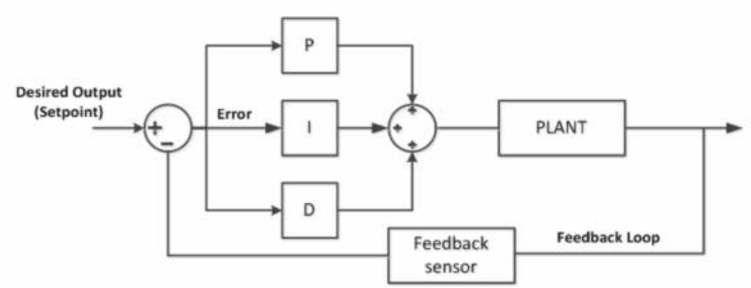

Figure 8: PID controller block diagram

$$PID = K_p . e(t) + K_i . \int_0^t e \, . dt + K_d . \frac{de(t)}{dt}$$

Equation 27 shows the standard PID algorithm in the parallel form which needs to be digitized in order to be implemented on a sampled system such as a PLC. The code is ideally implemented in SCL because of the ease with which algorithms can be programmed in this language. Equation 28 shows the PID controller in its digitized form.

$$u(kT) = \left(\frac{1}{b_2}\right) e(kT) - \left(\frac{b_1}{b_2}\right) u(k-1)T - \left(\frac{b_0}{b_2}\right) u(k-2)T$$

Where:

$$b_0 = \frac{T_d}{T}$$
$$b_1 = \left(\frac{T}{2T_i} - \frac{2T_d}{T} - 1\right)$$

$$b_2 = \left(\frac{T}{2T_i} + \frac{T_d}{T} + 1\right)$$

Figure 9 shows the PID controller implemented in the PLC within a parameterizable function block for easy access and tuning.

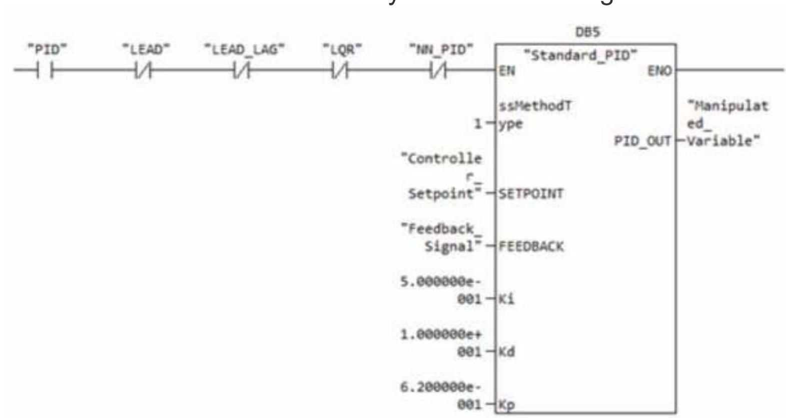

Figure 9: Conventional PID controller

3.7 Intelligent Neural Network (NN) controller design & implementation

In the control environment, NNs have received widespread attention, especially for their ability to learn non-linear characteristics through experimental data even without prior knowledge of the plant (Rahmat et al., 2010).

Research has proven that NNs can estimate every non-linear function with at least one hidden layer. NNs are therefore extensively used in simulation and control of non-linear processes (Nikolaou and Hanagandi, 1993; Mohamed et al., 2011; Liao et al., 2009). The cumbersome process of system modelling found with conventional controllers is thus eliminated provided that suitable operational data can be obtained from the plant for the purpose of training the network (Levin and Narendra, 1993). Numerous studies, including the research carried out by Hagan and Demuth (2002) and Han et al. (1999) highlight the effectiveness of NN based control schemes. It has also been shown that by combining aspects of conventional and intelligent control strategies, superior controllers are born that maintain the best of both worlds. One such controller is the NN-PID controller that automatically finds the most suitable PID controller gains for a particular system. The hidden to output layer weights w_{10}, w_{20} and w_{30} shown in Figure 10 essentially form the PID controller gains K_p, K_d and K_i respectively.

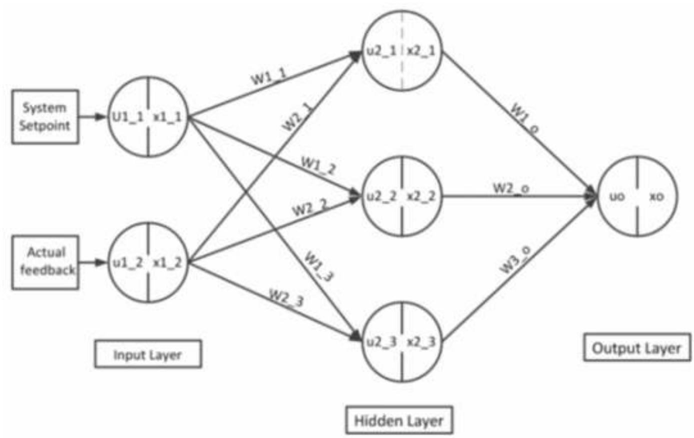

Figure 10: NN-PID structure

The network uses Back Propagation (BP) and utilizes the gradient descent learning algorithm to update its weights and thus minimize the system error. The sigmoid activation function is utilized in the BP computations. Based on this structure, an NN-PID controller was designed and simulated in Matlab and then implemented on the PLC for real-time control of the Ball-on-Wheel system. The NN was coded within an FB in SCL. The most important parameters to be set by the user are the system set-point, the actual feedback signal from the sensor, the learning-rate (Eta), the training momentum (Alpha) and the manipulated variable as shown in Figure 11.

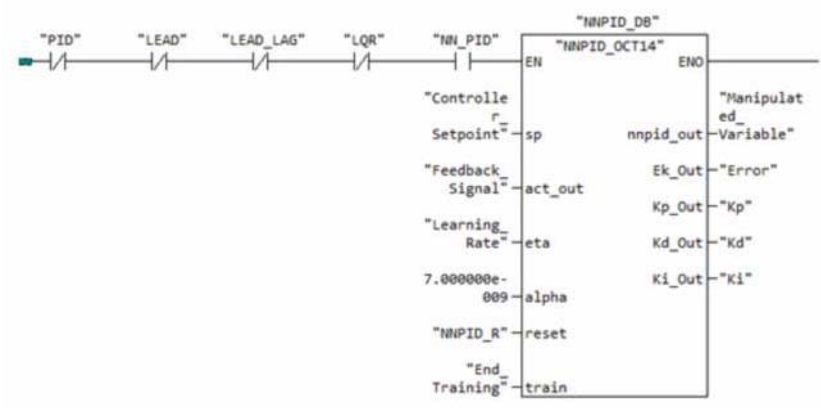

Figure 11: Intelligent NN-PID controller FB

3.8 Human Machine Interface (HMI) development

In order to enter controller parameters, an HMI was developed that runs on the remote (wireless) user PC (see Figure 3) and features a variety of monitoring and control screens as shown in Figure 12 and Figure 13. Other features of the HMI include pop-up warnings and alarms, plant status information, user instructions and disturbance injection settings.

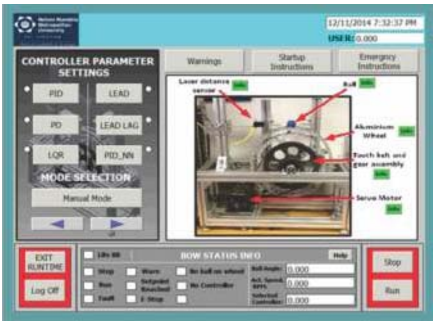

Figure 12: HMI for system information

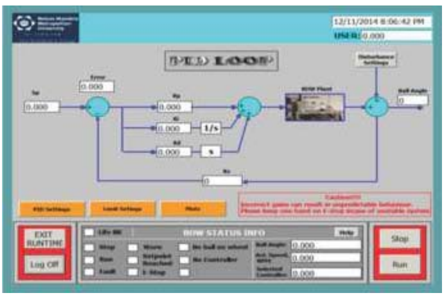

Figure 13: HMI screen for parameter selection

3.9 Experimental procedure

One of the goals of the research was to compare the performance of conventional control to that of intelligent control when implemented on a standard PLC. Attention was given to the achieved system settling time and steady state error as system parameters were altered and the plant was disturbed. In order to achieve parametric variation in the plant, the Ball-on-Wheel system was subjected to a variety of balls as listed in Table 2. As the balls are altered, plant parameters including ball weight, radius and moment of inertia are also changed. According to the system model, this changes the plant dynamics which in turn influence the extent of control that is possible. Furthermore, since the balls used in the experiment have varying physical properties such as different colour, surface texture and composition, unpredictable non-linear disturbances are also introduced (such as feedback sensor distortion, surface contact friction, uneven ball-to-wheel contact and ball bounce that disrupts the motion etc.). These uncertainties could not be accounted for in the system model for there being no way to precisely measure them. However, they serve a good purpose in creating unpredictable disturbances which assist in further interrogating the performance of each controller. Each of the conventional controllers was tuned to balance one specific ball only. Ball E from Table 2 was picked for this purpose because of its average specifications. The intelligent controller used in the experiment did not require any manual tuning, except for the selection of the learning rate. For each implemented controller, an automated disturbance signal was injected in to the system (in the absence of controller action) by momentarily applying a torque to the wheel for a short duration. The magnitude and duration of disturbance signal was kept constant throughout the entire experiment.

Table 2: Ball specifications

Ball	Mass	Radius
A	7.2 g	20.75 mm
B	3.6 g	20.25 mm
C	23 g	21.15mm
D	99.5 g	25.5 mm
E	100 g	24.65 mm
F	181 g	27.2 mm
G	52.3 g	30 mm

4. LABORATORY IMPLEMENTATION

4.1 Main objectives of the labratory

As an educational tool, the main objectives of the Ball-on-Wheel laboratory are to:

- Give students a clear practical understanding of the limitations found with conventional controllers;
- Expose students to modern intelligent methods of control – especially in solving complex non-linear control problems;
- Bridge the gap between individual course components such as system modelling, software design and real-time implementation;
- Familiarize students with current and futuristic industrial technologies such as drive systems, HMI's and PLCs;
- Give students a real practical engineering problem to solve with a significant design aspect;
- Develop social and organizational skills within students through team based problem solving tasks.

4.2 Outcomes of the labratory

On completion of this laboratory project, students must be able to:

- Model, linearize and simulate real systems using mathematical software tools such as Matlab and Simulink;
- Analyse and present systems in the state-space format;
- Analyse non-linear systems in the s-plane to determine the location of poles and zeros, and in the frequency domain to determine other important frequency related specifications;
- Determine characteristic curves of sensors and actuators using data recording and analysis techniques;
- Perform static and dynamic data analysis;
- Determine the overall process uncertainty;
- Design suitable conventional control strategies to meet given design criteria in Matlab;
- Design intelligent control strategies (such as NN-PID) in Matlab;
- Digitize designed control laws;
- Implement control laws on to digital platforms such as PLCs;
- Understand the limitations found with conventional control methods and the benefits of intelligent control;
- Work within a team environment.

The engineering problem, as discussed in the next section, is presented to the students in the form of a design problem similar to real-world product development tasks. Throughout the experimentation, team sizes remain limited to a maximum of three members only.

As a final step, each group must submit one written report detailing how they went about solving the presented problem. Students must be ready to answer questions regarding their designs during the assessment. Five consecutive weeks are given for the students to complete the theoretical and practical components of the task.

4.3 Controller design specifications

Sample design problem (as presented to students):

"Referring to Figure 14, the equilibrium point of the Ball-on-Wheel system is located at the top centre of the wheel (i.e. at zero degrees). This point is therefore the system set-point. An over-damped system is undesirable especially for systems that use tooth belt couplings; hence an overshoot of about 25% (7.5 degrees to either side of the equilibrium point) is acceptable for the Ball-on-Wheel system. The rise time and settling time must be kept within 500 ms. For feedback control, it may be considered that the laser distance sensor has a unity gain. A steady state error of less than +/- 5 degrees is acceptable."

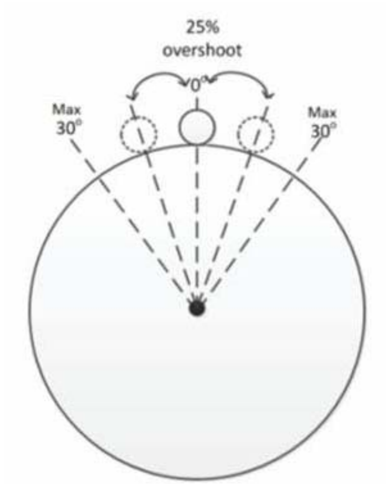

Figure 14: Design specifications

To meet the design requirements, each student team is required to design and simulate the controllers listed below in Matlab/ Simulink before actual implementation on the PLC. Controllers include:

a) A Phase-lead compensator,
b) A Lead-lag compensator,
c) A PID controller,
d) A PD controller,
e) A NN-PID controller.

Only one controller may be executed at a time. As part of the learning outcomes, students are expected to statistically analyse data captured from the Ball-on-Wheel plant over the wireless connection in Matlab.

5. RESULTS

5.1 Performance Specification analysis

The NN-PID controller must undergo a period of training in order to balance whichever ball is subjected to the system. For simplicity, only the results for Ball A are shown in Figure 15 and Figure 16. Figure 15 shows the reduction in the NNs root mean square error as it 'learns' to balance Ball A. During the training period, the NN determines the most suitable PID gains. By the operator randomly disturbing the plant during the training phase, the NN-PID controller can be made more robust. However, caution must be taken to avoid over-training which could result in an over-responsive or unstable system. By adjusting the learning rate of the NN, the system either tunes itself too quickly, yielding poor control performance, or too slowly, which results in time wastage, or optimally which results in good control performance.

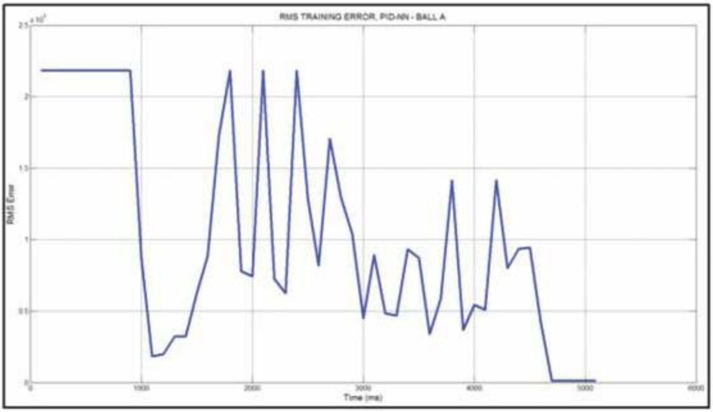

Figure 15: NN training error – Ball A

Figure 16 shows the system responses produced by the conventional controllers superimposed upon the response from the intelligent controller when Ball A is used. A disturbance is injected in to the system at T = 1000 ms in each instance. With this particular ball, the intelligent NN-PID controller outshines the conventional Lead, Lead-Lag and PID controllers in that it is able to stabilize the system in a much shorter space of time.

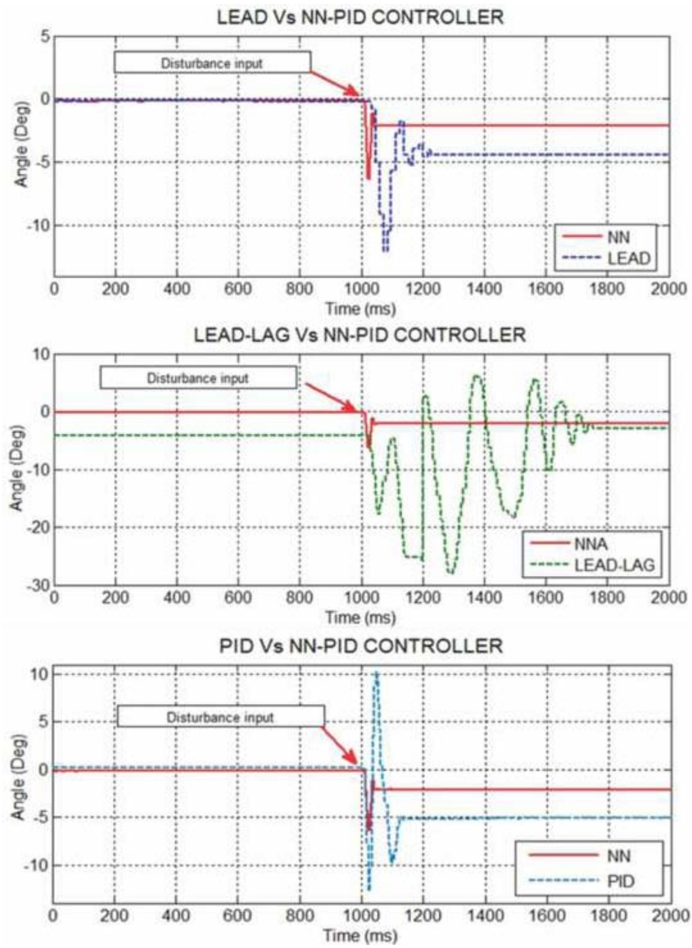

Figure 16: Ball-on-Wheel system response

Figure 17 shows a comparison between the average steady state errors produced by each implemented controller as different balls are introduced to the system. Where no result appears for a particular ball-controller combination, this simply means that the controller was not able to stabilize the system when that specific ball was used because the system was now outside that controller's linear range of operation. It can clearly be seen that the Lead controller for instance could not stabilize Balls C and F and the PID and Lead-Lag controllers could not stabilize Ball G. The NN-PID controller, however, was able to repeatedly stabilize every ball subjected to it.

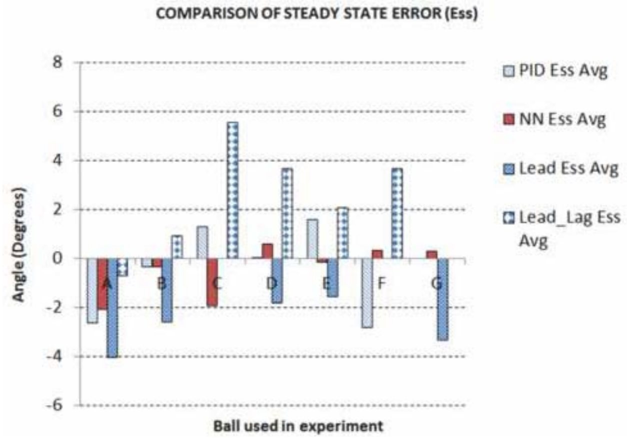

Figure 17: Comparison of steady state errors

It can be seen from Figure 17 that the NN-PID controller was the only controller able to produce a steady state error that remained consistently less than +/- 5 degrees as required in the design problem for each ball used in the experiment.

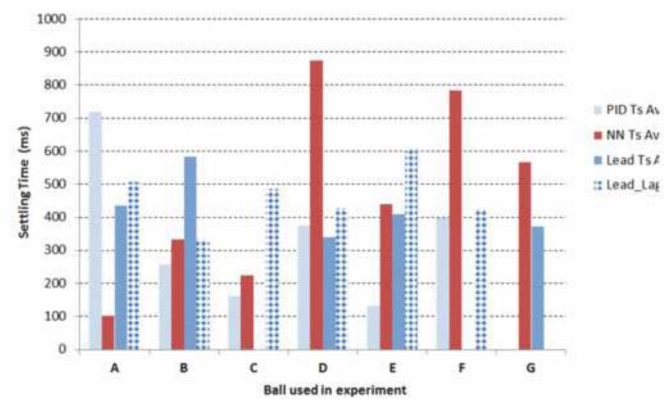

Figure 18: Comparison of settling time (Ts)

From Figure 18, it can be seen that the NN-PID controller has the highest average settling time. This however falls within the 500 ms settling time design constraint laid out in section 4.3. The conventional PID controller has the lowest average settling time of all implemented controllers.

The PID and NN-PID controllers show outstanding performance in their ability to remain within the design criteria while the balls are changed.

5.2 Laboratory implementation and feedback

Following a compulsory theoretical class on the Ball-on-Wheel system, the design problem was issued to a group of 3rd year Control Systems students in 2014 as an optional component with no formal mark allocation. Although only a handful of students actually attempted the challenge, almost all who did showed a keen interest in intelligent methods of control and the use of PLCs in solving complex control problems. The practical will be included as a compulsory component of the Control Systems curriculum at NMMU in 2016.

6. CONCLUSION

The research successfully shows that various simplified intelligent controllers that adopt features of both the intelligent and traditional approaches such as the NN-PID controller may be implemented with ease on standard industrial computational platforms. Furthermore, unlike their traditional counterparts, with intelligent controllers the entire process of system modelling and controller design can be ignored since this type of controller is able to adjust or re-tune itself through a process of learning to cater for parametric changes or non-linear disturbances introduced to the plant. This makes intelligent controllers suitable to all processes that experience natural change over time or that are susceptible to non-linear variation.

The Ball-on-Wheel system when applied as a teaching tool succeeds in linking individual course components such as system modelling, software design and real-time implementation. It also succeeds in giving students first-hand experience in solving real-world control problems while making use of standard industrial equipment – thus familiarizing students with currently used technologies. By investigating intelligent methods of control and their benefits when implemented on standard industrial computational platforms (i.e. PLCs), the curriculum is better equipped to produce students who have the necessary insight and skill to overcome the existing knowledge gap in industry and who can promote the use of computational intelligence in solving complex problems.

8. ACKNOWLEDGEMENTS

All the equipment and resources used to carry out this research were provided by the AMTC (Advanced Mechatronic Training Centre) NMMU, in partnership with Siemens.

9. REFERENCES

BASSILY, H., SEKHON, R., BUTTS, D. E. & WAGNER, J. 2007. A mechatronics educational laboratory–Programmable logic controllers and material handling experiments. Mechatronics, 17, 480-488.

CRAIG, K. Mechatronics at rensselaer: a two-course senior-elective sequence in mechanical engineering. Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on, 1999. IEEE, 452-458.

DYM, C.L., AGOGINO, A.M., ERIS, O., FREY, D.D. & LEIFER, L.J. 2005. Engineering design thinking, teaching, and learning. Journal of Engineering Education, 94, 103-120.

FELDER, R.M. & SILVERMAN, L.K. 1988. Learning and teaching styles in engineering education. Engineering education, 78, 674-681.

HAGAN, M.T. & DEMUTH, B.H. 2002. Neural Networks for Control .

HAN, W.-Y., HAN, J.-W. & LEE, C.-G. Development of a self-tuning PID controller based on neural network for nonlinear systems. Proceeding of the 7th Mediterranean Conference on Control and Automation (MED99), Haifa, Israel, 1999.

HSIEH, S.J.T. & YEEHSIEH, P. 2004. Integrated Virtual Learning System for Programmable Logic Controller*. Journal of Engineering Education, 93, 169-178.

KANTHALAKSHMI, S. & MANIKANDAN, V. 2010. Genetic Algorithm Based Self Tuning Regulator. International Journal of Engineering Science and Technology, 2, 7719-7728.

LEVIN, U.A. & NARENDRA, S.K. 1993. Control of Nonlinear Dynamical Systems Using Neural Networks: Controllability and Stabilization. Transactions on Neural Networks, 4, 192-206.

LIAO, F., LUM, K.Y., WANG, J.L. & BENOSMAN, M. 2009. Adaptive control allocation for non-linear systems with internal dynamics. IET Control Theory Application, 4, 909-922.

MING-TZU, H., YI-WEI, T. & HAO-SHUAN, L. 2009. Controlling a Ball and Wheel System Using Full State Feedback Linearization: A Testbed for Nonlinear Control Design. Michigan, USA.

MOHAMED, R.B., NASR, H. B. & M'SAHLI, F. 2011. A multimodel approach for a nonlinear system based on neural network validity. Intelligent Computing and Cybernetics, 4, 331-352.

NIKOLAOU, M. & HANAGANDI, V. 1993. Control of nonlinear dynamical systems modeled by recurrent neural networks. AIChE journal, 39, 1890-1894.

NIKOLAOU, M. & MISRA, P. 2001. Linear and Nonlinear Processes: Recent Developments and Future Direction.

RAHMAT, M.F., WAHID, H. & ABDUL-WAHAB, N. 2010. Application of Intelligent Controller in Ball and Beam Controller. Smart Sensing and Intelligent Systems, 3.

REN, X.M. & RAD, A.B. 2009. Adaptive non-linear compensation control based on neural networks for non-linear systems with time delay. International Journal of Systems Science, 40, 1283-1292.

SAYGIN, C. & KAHRAMAN, F. 2004. A web-based programmable logic controller laboratory for manufacturing engineering education. The International Journal of Advanced Manufacturing Technology, 24, 590-598.

UDDIN, M.N., HUANG, Z.R. & HOSSAIN, A.S. 2014. Development and Implementation of a Simplified Self-Tuned Neuro–Fuzzy-Based IM Drive. Industry Applications, IEEE Transactions on, 50, 51-59.

WIKANDER, J., TORNGREN, M. & HANSON, M. 2001. The science and education of mechatronics engineering. Robotics & Automation Magazine, IEEE, 8, 20-26.