CONTROLLING THE INVERTED PENDULUM USING PULSE-TRAINS AND FUZZY LOGIC

Matee Serearuno Tony Holden

Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, United Kingdom

Abstract: This paper reports the results of an experiment simulating the use of pulse trains and a fuzzy logic controller to balance an inverted pendulum (or cart-pole) system. It differs from other implementations in that it applies a control-force only when the need for control is detected. Instead of applying a continuous force synchronously, the control-force is fired as discrete pulse-trains. It is found that the pendulum is successfully maintained at its equilibrium 'upright' position, with the use of fewer fuzzy rules. The Java applet simulation can be demonstrated interactively in real-time. *Copyright © 2000 IFAC*

Keywords: fuzzy logic, control, pulse trains, real-time, simulation, interactive

1. INTRODUCTION

An inverted pendulum (also known as cart-pole or "Broom" balancing) is a classic example of an inherently unstable dynamic system in the field of control engineering. Its dynamic behaviours employ several non-linear complex parameters that are difficult to solve directly with linear algebraic methods. As such, it is a good platform for demonstrating the performance of the control system proposed.

1.1 *Cart-Pole System*

The inverted pendulum system consists of a cart translating in one dimension along an x-axis track. The cart is equipped with a pole that is attached to the cart with a hinge at one end, so that the pole can freely swing to-and-fro from $-\pi/2$ to $\pi/2$. The initial state of the system is with the cart being at the centre of the track and the pole being at an upright position. Due to the imperfection of the apparatus set-up and the earth's gravity, the pole will fall either way and cause the cart to slide away from the centre. To regain the balance of the pole and the cart, the correct amount of force must be applied to the cart-pole through its centre of mass. The balancing control must be applied in a timely manner, before the pole position goes beyond the critical point from where it cannot be recovered and will fall down.

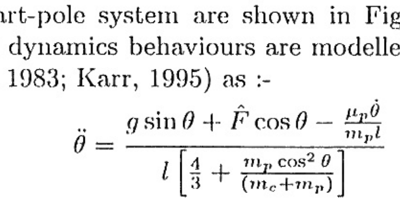


Fig. 1. Model of Inverted Pendulum (Cart-Pole)

The architecture and the mathematical model of the cart-pole system are shown in Figure 1, in which dynamics behaviours are modelled (Barto, *et al.*, 1983; Karr, 1995) as :-

$$\ddot{\theta} = \frac{g \sin\theta + \hat{F}\cos\theta - \frac{\mu_p \dot{\theta}}{m_p l}}{l\left[\frac{4}{3} + \frac{m_p \cos^2\theta}{(m_c+m_p)}\right]} \quad (1)$$

$$\hat{F} = \left[\frac{-F - m_p l \dot{\theta}^2 \sin\theta + \mu_c sign(\dot{x})}{(m_c + m_p)}\right] \quad (2)$$

$$\ddot{x} = \frac{F + m_p l\left[\dot{\theta}^2 sin\theta - \ddot{\theta}\cos\theta\right] - \mu_c sign(\dot{x})}{(m_c + m_p)} \quad (3)$$

From Euler's equation, then the following can be obtained.

$$x^{t+1} = x^t + \dot{x}^t \Delta t \quad (4)$$

$$\dot{x}^{t+1} = \dot{x}^t + \ddot{x}^t \Delta t \quad (5)$$

$$\theta^{t+1} = \theta^t + \dot{\theta}^t \Delta t \quad (6)$$

$$\dot{\theta}^{t+1} = \dot{\theta}^t + \ddot{\theta}^t \Delta t \quad (7)$$

where

x = cart position
$\dot{x}$ = cart velocity
$\ddot{x}$ = cart acceleration
θ = pole angular deviation
$\dot{\theta}$ = pole angular velocity
$\ddot{\theta}$ = pole angular acceleration

1.2 *Fuzzy Logic Controller*

Fuzzy logic controllers (FLCs) have been successfully demonstrated to be able to determine the right force to balance the cart-pole system (Cooper and Vidal, 1996). A number of system variables, including θ, $\dot{\theta}$, x and $\dot{x}$, are taken as inputs for which the process of "*fuzzification*" then determines the degree of membership of each variable in the subsequent fuzzy reasoning in order to obtain the final output. The "*defuzzification*" process then transforms the fuzzy output back into a real world value (as a control force) to compensate the cart-pole system.

	$\dot{\theta}_L$	$\dot{\theta}_Z$	$\dot{\theta}_R$
θ_L	F_L	F_L	
θ_Z	F_L	F_Z	F_R
θ_R		F_R	F_R

Fig. 2. Fuzzy rule matrix (FAM) for pole control

The fuzzy rules, known as a *fuzzy association matrix* or *FAM* normally consist of a collection of domain knowledge rules as shown in Fig. 2. Typically, each rule takes two arguments to produce one fuzzy output. For example, the first rule (the first box in the left uppermost corner) states that

> If θ is *to-the-left* and $\dot{\theta}$ is *to-the-left* then *Force* is *to the left.*

Similarly, the other rules are applied in the same manner. It is noted that several *linguistic* terms have to be defined, such as what the *to-the-left* (denoted by a subscription $_L$) means in the context of this system; *i.e., x position is being negative.* The membership function and the fuzzy rules in the FAM matrix combined enable the FLC to determine the correction force effectively without directly solving the differential equations 1, 2 and 3 shown above. Further details of fuzzy theory and fuzzy logic controllers can be found in several good text books, such as Cox (1999), and Reznik (1997).

2. MOTIVATION

Although FLCs have seen marginal success in many engineering applications, a fundamental task is still that of producing an *analogue* output for the control process. As such, the control loop must be fast enough to reflect the changes in real-time. Otherwise, large time intervals between the control cycles can potentially ruin the overall performance due to resultant lack of control.

2.1 *Traditional Control Loop*

Since most closed-loop control systems are configured to perform synchronously, *i.e., performing regularly at a fixed time interval.*, FLCs have to determine the output at every time step. However, this is in fact unnecessary when it is certain that the system under control is performing well. In the case of the cart-pole system, for example, the applied force is not immediately effective, due the massive moment of inertia. As such, more forces are still applicable and often make the system *overshoot* the desired point. In such a case, an oscillation often occurs.

To address this undesirable behaviour, more 'knowledge' fuzzy rules have to be added to analyse the situation thoroughly before firing the control. However, these extra rules will demand more computational power since complexity will be increasing.

2.2 *Pulse Control Loop*

In response to the inherent overshoot situation in the traditional firing strategy, it is interesting to alter the way the FLC works by allowing ample time for the cart-pole to take action, *before* more control is applied. Unlike the FLC that fires at constant time intervals, the control is deferred, as long as it is obvious that the cart-pole is currently performing well or a positive sign of progress is apparent. That is to say, control rules will fire only when it is absolutely needed. From the viewpoint of the cart-pole system, this time-gap between the fired impulses allows ample time for the physical system to respond to the applied force, before any subsequent action is reconsidered. Simultaneously, from the viewpoint of the controller, this concept of an *'on-demand'* control loop brings up, more or less, an idea of a variable magnitude *impulse* of control force.

Pulse control techniques have been used widely in many engineering applications, including electronic signal communications and power controllers. The obvious benefits include the simplicity of the system because control is applied in an *on-off* fashion, as opposed to the conventional system controllers which tend to behave like a power amplifier; *i.e., a change at the input leads to a similar change at output with larger magnitude.*

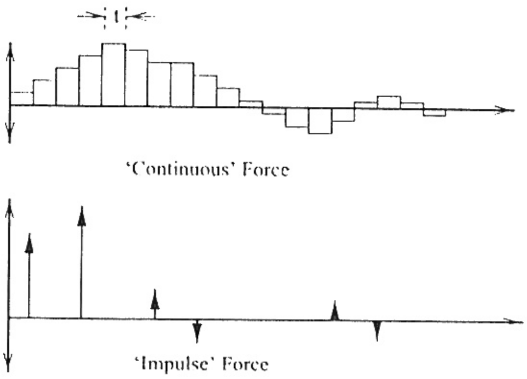


Fig. 3. Continuous and Pulse-trains Control

The pulse-controlled controllers allow the control signal (or power) to pass through as an output only during the *on* stage, whereas there is no output passing through at all during the *off* stage. A prospective benefit of the pulse-controlled method for the inverted pendulum problem is that of a simpler design for the fuzzy engine. To put it simply, it is expected that the FAM will be smaller, with a faster response time and also demands less computing power. It should also be less complicated to fine tune the system than a conventional FLC handling the same situation.

2.3 *Pulse-Controlled FLC*

Pulse-controlled FLCs (PFLC) are in fact the modified FLC, in which the output is fired as a narrow impulse, whereas the conventional FLC fires the control for the whole period of given time interval (see Fig. 3). The period of 'on' state is pre-determined in such a way that it is effective in controlling the system, without much deteriorating the environment; *i.e. balancing the pole, without pushing the cart away.* As a result, the number of firing rules is considerably reduced. This results from the fact that it is unnecessary to monitor *every* possible behaviour of the inverted pendulum system. Indeed, the only task is the determination of *whether* the system is currently performing well *or not.*

In the case of the control of the pole angle, only *two* simple rules (instead of *seven* are sufficient to determine whether or not the control signals are about to be fired. The two firing rules are :-

	$\dot{\theta}_L$	$\dot{\theta}_Z$	$\dot{\theta}_R$
θ_L	F_L		
θ_Z			
θ_R			F_R

1) IF the pole (θ) is on the *left-hand* side AND it is *increasingly* ($\dot{\theta}$) falling to the *left* THEN *Force* is applicable to the *left* direction.

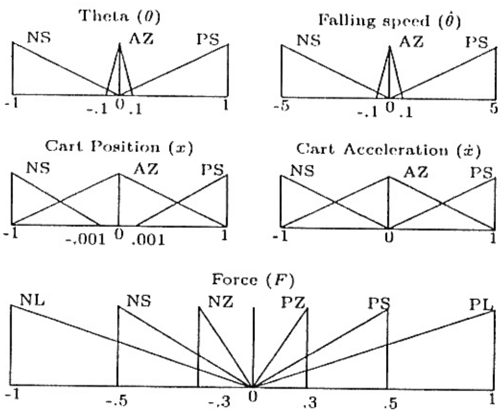
2) IF the pole (θ) is on the *right-hand* side AND it is *increasingly* ($\dot{\theta}$) falling to the *right* THEN *Force* is applicable to the *right* direction.

Otherwise, the system is *either* already at an upright position *or* currently making good progress towards it. Thus, no control is required.

3. THE EXPERIMENTAL SETTING

For comparison, the FLC and the PFLC systems are demonstrated concurrently. Both systems are constructed from the same fundamental components to ensure the compatibility of the result. The simulation is carried out using the following parameters.

Gravity	(g)	$= -9.81$ m/s^2
Mass of cart	(m_c)	$= 1.0$ kg
Mass of pole	(m_p)	$= 0.1$ kg
Half pole length	(l)	$= 0.5$ m
Cart-track friction	(μ_c)	$= 0.0005$
Cart-pole friction	(μ_p)	$= 0.000002$
Time Interval	(Δt)	$= 50$ ms



θ	$\dot{\theta}$	x	$\dot{x}$	Force
NS	NS	-	-	NZ
PS	PS	-	-	PZ
AZ	AZ	PS	PS	NL
AZ	AZ	NS	NS	PL
AZ	AZ	PS	AZ	NS
AZ	AZ	NS	AZ	PS

Fig. 4. FAM matrix and membership functions of the PFLC system (not drawn to scale)

To balance the pendulum, two inputs from the pole's deviation (θ) and pole's falling speed ($\dot{\theta}$) are required. Similarly, the cart's position (x) and acceleration ($\dot{x}$) are sufficient to centre the cart at the track's midway. The details of the fuzzy rule matrix and associated membership functions are shown in Fig. 4.

For the sake of simplicity, the *Center-of-Gravity* algorithm (Nauck, *et al.*, 1997) is used to defuzzify the magnitude of the output. In the case of the FLC system, this force is then applied to the cart-pole system (as an *average* force) for the whole period of the control-cycle time interval, whereas it is only applied for a pre-determined fraction of the time interval, in case of the PFLC system.

The dynamic changes of the conventional FLC are then simulated programmatically by :-

dynamics($appliedForce$, $deltaTime$)

where the *appliedForce* is the output from the controller and the *deltaTime* is equal to 50 milliseconds.

Similarly, the dynamic changes of the PFLC are simulated by

dynamics($appliedForce$, $pwmTime$)
dynamics(0, $deltaTime$ - $pwmTime$)

where the *appliedForce* is the output from the controller and the *pwmTime* and *deltaTime* are equal to 10 and 50 milliseconds, respectively.

3.1 *Implementation*

Both simulated fuzzy controllers, as well as the cart-pole system, are implemented in *Java* programming language as a Java applet[1] As a result of object-oriented programming style in Java, both systems share the same common features by inheriting properties from the parent FLC class. However, the defuzzification method of the PFLC class is altered in such a way that the determined output is fired as an impulse, instead of a continuous force as in the FLC system.

The applets are also designed to allow the two systems to operate graphically in real-time. By using an internet Java-capable browser, all important system parameters, including x, $\dot{x}$, $\ddot{x}$, θ, $\dot{\theta}$, and $\ddot{\theta}$, are displayed simultaneously on the same time scale for easy comparison.

To test the robustness of the two systems, an intentional but unexpected *shot* can be fired. This is done by purposefully interfering with the pendulum's angle and then observing its dynamic reactions. There are *four* possible shots, including BigLeft, Left, Right and BigRight. The cart-pole can also be reset to its initial state.

[1] Java applets and corresponding class files are available to interested readers via the internet, at the homepage "http://www-dsg.eng.cam.ac.uk/java/pwm".

4. EXPERIMENTAL RESULTS AND DISCUSSION

The dynamics of the cart-pole system controlled by the FLC and PFLC when the pole was initially set at -0.8 radian (approximately 45.8 degrees to the left-hand side) and 0.2 radian (approximately 11.4 degrees to the right-hand side) are illustrated in Figures 5, 6, 7 and 8, respectively.

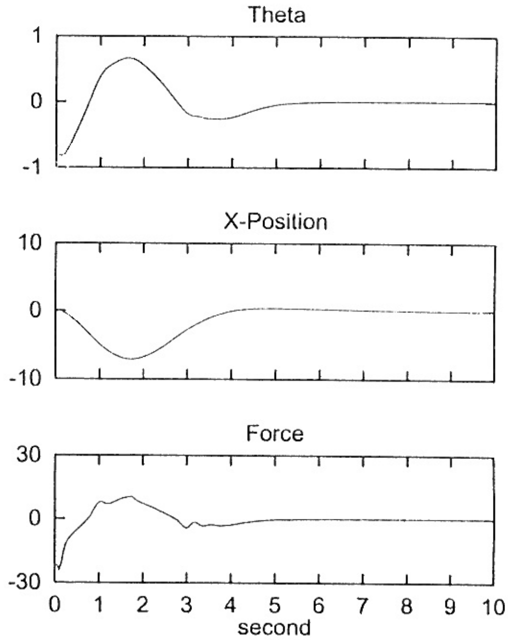


Fig. 5. Dynamics of the cart-pole with conventional FLC when pushed by -0.8 radian

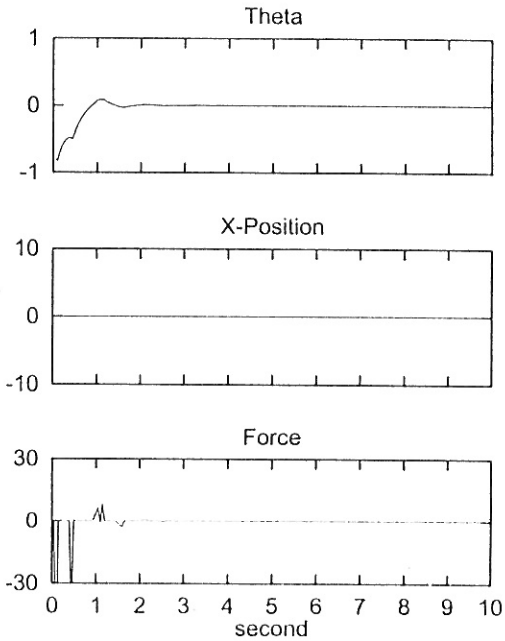


Fig. 6. Dynamics of the cart-pole with PFLC when pushed by -0.8 radian

It can be seen that both systems are able to regain equilibrium successfully. As shown in Figures 5 and 6, the FLC system takes almost *five* seconds, while the PFLC system achieves the same result in less than *two* seconds. The graph also shows that the FLC controller tries to balance the pole in the first instance by firing a large negative force. As a result, the cart is pushed away to the right, so that the controller has to react by firing the opposite force to stop the cart's movement. This subsequently results in the pole swinging to the right by its moment of inertia. Then, the system gradually adjusts the control force to bring the cart-pole back to its initial state. It is noted that a small overshoot of the pole is also observed at about the third second.

The PFLC system is also able to balance the cart-pole successfully by first firing a large negative force and then seeing the cart-pole to respond (see Fig. 6). Only when it is found that the pole is starting to fall to the left again (about the first half a second), the PFLC fires another shot to force the pole to move towards the upright position again. Although this time, the pole overshoots slightly, a series of small opposite shots are then fired to bring the pole back to the centre-line. Interestingly, the whole balancing process takes less than two seconds to complete.

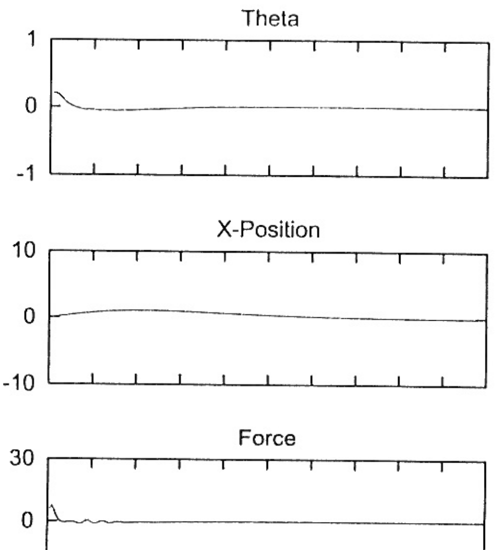


Fig. 7. Dynamics of the cart-pole with conventional FLC when pushed by 0.2 radian

Figures 7 and 8 also confirm similar results when the pole is suddenly set to 0.2 radian to the right. Again, the FLC responds this by firing a series of small positive forces, in which the cart is pushed to move away to the right. Then, the FLC controller fires negative and positive forces alternatively to bring back the equilibrium. However, the PFLC achieves the same result more gracefully with smaller fluctuations, and also within a shorter period of time.

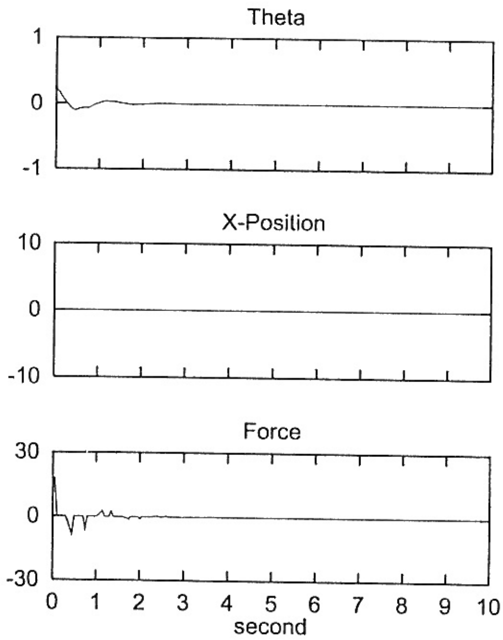


Fig. 8. Dynamics of the cart-pole with PFLC when pushed by 0.2 radian

4.1 *Discussion*

It is evident from the results illustrated above that the PFLC has shown a potential benefit over the conventional FLC. It is also found that in achieving the same performance, the PFLC has much simpler fuzzy rules, due to the reduced needs to monitor the behaviour of the system under control. Experimentally, the results show that the conventional fuzzy controller requires up to *seven* fuzzy rules in order to reliably balance the pole and another *four* more rules to keep the cart close at the centre of the track. On the other hand, the PFLC requires only *two* fuzzy rules to balance the pendulum, while the same *four* rules for x-positional rectification are required. With less fuzzy rules, thus, the PFLC is at least theoretically easier to build.

Experimentally, the simulation also shows that the PFLC has some advantages in balancing the cart-pole system. A large swing of $\pm$ 0.8 radian can be recovered as reliably as a small swing of $\pm$ 0.2 radian, by using only two fuzzy rules. Moreover, it is also robust enough to recover from a series of pole-shots, whereas an ordinary FLC is out of control. It is worth noting that there is a very small displacement along the x axis when the control force is fired as an impulse, so that the required balancing time is considerably shortened.

However, conventional FLCs seem to perform better, when the pole is *slightly* out of centre. In this case, a single large pulse often worsens the situation and a few more cycles are required to re-stabilise the system. Furthermore, the pulse-control FLC has performed no better than the conventional FLC in controlling the cart's position. Indeed, both FLC systems need *four* fuzzy rules to successfully centre the cart-position, while balancing the pole. It may be inferred that the cart-position goal is much harder to address, since every cart's movement always makes the pole swaying. To prevent the pole from falling down (which is obviously unacceptable), the pole angle thus needs more attention than the position of the cart. In that case, the cart is usually left unattended until the pole has regained its balanced position.

5. EPILOGUE

The experimental simulation has shown that the pulse-control FLC system can reliably balance the cart-pole system. It is interesting that the PFLC requires fewer and simpler fuzzy rules than the conventional FLC to achieve similar (if not better) performances. The results show that by firing the control force as an impulse with large magnitude but narrow pulse-width, the pole's angle is efficiently recovered from falling down.

In summary, the key behaviours of the PFLC system are listed as follows :-

- The control force is fired as an impulse or *"on-off"* fashion to allow a time-gap for the cart-pole system to respond.
- The control actions are taken only when they are *absolutely* needed; *i.e., not at a fixed time interval.* That is to say, there is *no* control at all, provided the system is currently performing well. Thus, less power consumption in firing controls can be expected.
- Smaller set of fuzzy rules demands less computational power, in which shorter control-loop cycles are very likely. It is also easier and cheaper to implement too.
- The PFLCs are able to cope with large disturbances, as well as repetitive interference.

However, the potential limitations of the PFLCs, worth noting are :-

- Since the impulse has a very narrow *active* period but with large magnitude, a single pulse may itself destabilise the system under control and cause an overshoot.
- An impulse also has too an strong effect when a small change is required. In that case, an oscillation often occurs.
- The switching gate which turns the pulses *on* and *off* may produce electromagnetic interference. An additional noise filter circuit may be needed to suppress these noisy harmonics.

In conclusion, both systems are able to balance the cart-pole as they are expected to. Although the PFLCs have shown some advantage in effective balancing, it may not have been fair to compare them directly because not all fuzzy rules were thoroughly optimised. Despite that, this experimental simulation has successfully demonstrated the potential use of employing the pulse control technique in addressing a second-order differential equation.

This study is still in progress. The pulse-controlled concept cannot be completed without proving on other control systems. Further work, thus, involves investigation of the viability of the pulse-controlled fuzzy logic technique in other engineering applications. In addition, the enhancement of fuzzy rule optimising techniques, pulse-width variation control, and automatic discovery of the fuzzy rules, to name but a few, are also very promising lines of investigation.

ACKNOWLEDGEMENT

The authors would like to thank his colleagues for their willingness to proof-read the manuscript and give their constructive comments.

REFERENCES

A. G. Barto, R. S. Sutton, C. W. Anderson (1983). *"Neural-like adaptive elements that can solve difficult learning control problems"*, IEEE Transactions on Systems, Man and Cybernetics, SMC-13(5).

Charles L. Karr (1995). *"Genetic Algorithms and Fuzzy Logic for Adaptive Process Control"*, Intelligent Hybrid Systems, edited by Suran Goonatilake and Sukhdev Khebbal, John Wiley & Sons.

Mark G. Cooper and Jacques J. Vidal (1996). *"Genetic Design of Fuzzy Controllers"*, Genetic Algorithms for Pattern Recognition, edited by Sankar K. Pal and Paul P. Wang, CRC Press, Inc.

Detlef Nauck, Frank Klawonn, and Rudolf Kruse (1997). *"Foundations of Neuro-Fuzzy Systems"*, John Wiley & Sons.

Reznik, Leonid (1997). *"Fuzzy Controllers"*, Oxford, Newnes.

Cox, Earl (1999). *"The fuzzy systems handbook: a practioner's guide to building, using, and maintaining fuzzy systems"*, 2nd Edition, London, AP Professional.