International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 10, October-2015

Robotics Controller: A Literature Survey

Prashant Badoni
Mechanical Engineering Department,
Graphic Era University,
Dehradun, India

***Abstract*— The controller is a vital subsystem of the robot that is designed to help the system in order to achieve stability, good disturbance rejection and minimum tracking error. This paper focuses on the effective control techniques for the robotic systems. It analyses the classical as well as intelligent controllers with study of the control system.**

***Index Terms— Controller, Control System, Control Techniques*.**

I. Introduction

The Early Years of Robotics was largely focused on manipulator arms and simple factory automation tasks; materials handling, welding, painting. Cost of computation, lack of good sensors, and lack of fundamental understanding of robot control were the primary barriers to progress. But Robotics today is a much richer field with far-ranging applications e.g. Robots are exploring Mars. The complexities of Robotic systems are increasing day by day as more and more intelligence is embedded into their controllers. Robotic controller is one of the most vital components which defines the accuracy and repeatability of a robot. It is used to modify the behavior of the physical system according to the input value through computations and actuations.

This paper is organized as follows: Section II presents the detail control system design. The various control techniques are given in Section III. Finally, conclusions are given in
Section IV.

II. Control System

A control system refers to a programmable hardware machine and its included program. Control system is one of the main subsystems of a robot. It consists of sensors, controllers and knowledge base etc. that provide convenient duty to robot.

Robots are often classified according to the type of control system used: non-servo and servo. The earliest type of robots was non-servo, considered as non-intelligent robots. The second type is the servo robots, are intelligent robots.

Generally, two types of control system are used: open-loop and closed-loop. The non-servo robots are open-loop system. No feedback mechanism is used in an open-loop system. The servo robots are closed-loop system. In a closed-loop system, the feedback signals are sent to the servo amplifier which affects the output of the system.

A generalized overview of a straightforward open-loop control system is shown in Figure 1. This simplified system has a cause and effect relationship described with the terms input and output. The input is the desired set point for which the controlled variable should reach and maintain. The process or plant is the component of the system driven by the controller. The output of the system is the effect of the process or plant with any disturbances applied. The open-loop configuration does not compensate for any disturbances added to the system; therefore, if disturbances arise, they become part of the output. Open-loop systems are not even able to detect disturbances as they occur.

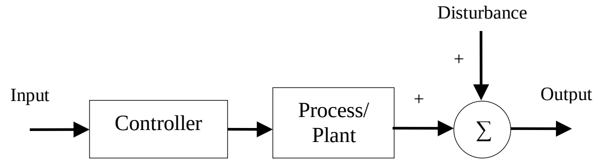


Fig. 1. Open Loop Control System

The advantage of an open-loop control system is the simple and straightforward input-output relationship. The disadvantages are found in the inability to detect and compensate for disturbances to the system. These disadvantages can have detrimental consequences depending on the nature and purpose of the system.

The closed loop system attempts to overcome the disadvantages experience by the open-loop configuration. A basic closed-loop system (Figure 2) compensates for disturbances by adding a feedback path. The input or set point of the system is set by the user to the desired value the manipulated variable should reach and maintain. The first summing junction connects the input with the output via the feedback path. Here the output value is subtracted from the input value to find the error.

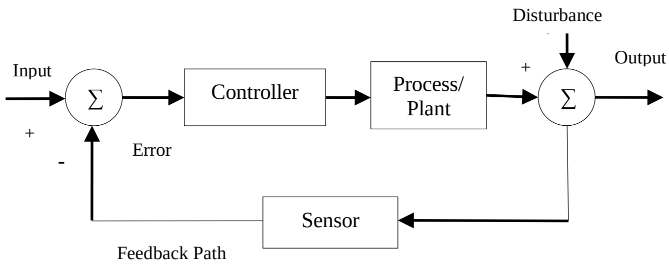


Fig. 2. Closed Loop Control System

The comparison of these values drives the process or plant to make the necessary corrections if needed. If there is no difference between the desired input and the output, the system is already producing the desired output, and no correction is needed at that time. The sensors utilized in the feedback path continuously supply feedback to the controller in order for the system to constantly monitor for

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

disturbances that could affect the desired output. The error of the system allows the controller to drive the process to continually reduce the difference between the set point and output.

There are two methods used to control systems: linear and non-linear method. Linear control method is applicable only when the controlled system can be modeled mathematically [1] [2]. Most of the physical systems have non-linear characteristics. Non-linear methods considered as general case as compared to linear methods because it can be applied successfully on the linear methods, but linear method is not sufficient to solve and control nonlinear problems.

Controller is a device which can sense information from linear or nonlinear system (e.g. robot) to improve the systems performance [3] [4] [5] [6]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error [7] [8]. The controller helps to achieve these targets of the control system.

III. CONTROL TECHNIQUES

A robot is an advanced machine which consists of mechanical and electrical parts. The design of the mechanical structure of the robot involves with the design of robot links and gear boxes which requires stress-strain analysis. To analyze the motion of robot it is necessary to know the kinematics and dynamics of the constrained rigid bodies and inverse of these functions. There are defined methods for calculation of kinematics and inverse kinematics of the robot such as Denavit-Hartenberg for kinematics and Lagrange method for inverse kinematics of the robot. After deriving the inverse kinematics and dynamics of the robot a controller can be designed and be implemented with analogue or digital circuits. There are different methods that a robot can be controlled are as follows.

A. PID Controllers

PID controller is the popular technique in control applications due to its low cost, simplicity in design and implementation, and ability to be used in wide range of applications.

A PID controller is a three-term controller using a proportional term, integral term, and derivative term combined in a linear algorithm (Figure 3) to create a desired output response.

Mathematically,

$$PID\ Output = P_{out} + I_{out} + D_{out}$$

The proportional term calculates the gain based on present error.

$$P_{out} = K_p * e(t)$$

The integral term calculates the sum of all past errors.

$$I_{out} = K_i * \int_0^t e(t)dt$$

The derivative term uses the rate at which the error has been changing to predict future error.

$$D_{out} = K_d * \frac{d}{dt} e(t)$$

Therefore,

$$PID\ Output = K_p * e(t) + K_i * \int_0^t e(t)dt + K_d * \frac{d}{dt} e(t)$$

The terms K_p, K_i and K_d stands for the proportional, integral and derivative gains. The term $e(t)$ represents the error. Table 1 shows the effects of each controller gain K_p, K_i and K_d on the control system output response.

Parameters	Rise Time	Overshoot	Settling Time	Steady State Error
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Decrease significantly
K_d	Minor decrease	Minor decrease	Minor decrease	No effect in theory

Table1. PID Controller in a closed-loop system [9]

PID controller also uses a feedback loop to compensate for error. The error is described as the difference between the desired set point of the system and the measured variable calculated by the P, I, and D terms.

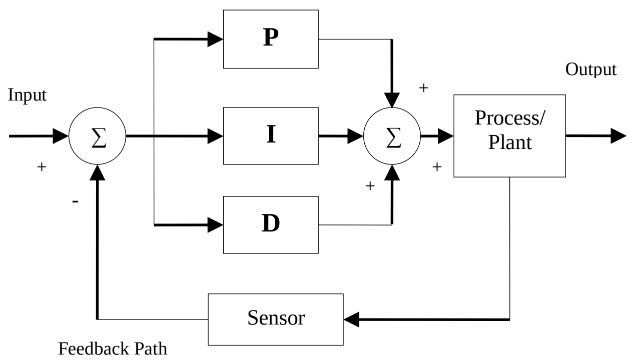


Fig. 3. PID Controller

Once a PID controller is designed, a *tuning* process must follow in order for the controller to meet the needs of a specific system. Stability is the basic need for all the control systems. If gains (PID parameters) are not chosen correctly, it will make system instable. There are two basic methods to tune a PID controller:

Manual Tuning Method

This method is used to determine the PID controller parameters. First Integral (K_i) and derivative (K_d) gains are set to zero. Then proportional gain (K_p) is tuned to give the desired response and neglect the steady state error. After that K_p is increased by small increment and adjustment of the K_d takes place to decrease the damping. Finally K_i is adjusting to remove the steady state error. Previous steps are repeated until the desired response is achieved.

It is time consuming method because it based on trial and error approach.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

Ziegler-Nichols Method

It is an analytical approach of tuning the PID controller. Ziegler and Nichols proposed this method based on their experience in industrial control. Initially the K_i and K_d are set to zero, and K_p is increased until the loop oscillates around the set points. At this point, the critical or ultimate gain (K_u) and oscillation or ultimate gain (P_u) are noted. Then thes values in Table 2 are used to tune the gain parameters.

Controller	K_p	K_i	K_d
P	0.50 K_u	-	-
PI	0.45 K_u	P_u /1.2	-
PID	0.60 K_u	0.50 P_u	P_u / 8

Table2. Ziegler-Nichols Method [10]

A third method of tuning a PID controller is by using PID tuning software. This method is popularized by industry to obtain consistency among systems. A person using either the manual or Ziegler-Nichols method takes time to obtain the optimal responses, and to industry, time equals money. The software provides a faster and more consistent method of tuning these controllers. Many software packages are available that tune according to certain performance criteria required by a specific system depending on its design use.

Table 3 presents the summary of these tuning methods for a PID controller with their respective advantages and disadvantages [11] [12].

Methods	Advantage	Disadvantage
Manual	Online method , No math expression	Requires experienced personnel
Ziegler-Nichols	Online method , Proven method	Some trial and error, process upset and very aggressive
Software tools	Online or offline method, consistent tuning, Support Non-Steady State tuning	Some cost and training involved

Table3. Summary of Tuning Method

Conventional PID controllers are generally not suitable for nonlinear systems, higher order, time-delayed systems. For that purposes, various modified conventional PID controllers such as auto-tuning and adaptive PID controllers are proposed for that purpose [13], [14], [15]. Also, during this period it was suggested that if the process was too complex to achieve a good physical description, conventional methods were not able to guarantee the final control aims, and the controller synthesis had to be based mainly on intuitions and heuristic knowledge. So, expert control strategies are favored since they are based on the process operator's experience and do not need accurate models [16], [17], [18], [19], [20], [21], [22].

One of the most successful expert system techniques applied to a wide range of control applications has been the Fuzzy Set Theory, which has made possible the establishment of *"intelligent control"*. The fuzzy approach provides a good support for translating the heuristic skilled operator's knowledge about the process and control procedures expressed in imprecise linguistic sentences into numerical algorithms, [16], [18], [19], [21], [23], [24], [25], [26], [27], [28].

B. Fuzzy Logic Controllers

Fuzzy Logic (FL) is based on the fuzzy set theory established by Lofti A. Zadeh in 1965 [29]. He showed that fuzzy logic could realize values between false and true. FL uses linguistic variables to represents a range of values. A linguistic variable represents the imprecise information, written in a natural language format. The basic idea of Fuzzy Logic Controller is that used to convert the linguistic variable based on the information. A conventional controller such as PID controller is efficient and offers powerful method to analysis linear systems. In case of nonlinear systems conventional controllers does not produces satisfactory results due to the nonlinearities of these systems [30]. Therefore, FLC may be an efficient tool to control these nonlinear systems [31].

The basic structure of a fuzzy logic controller is shown in Figure 4. Its fundamental components are fuzzification, control rule base, inference mechanism and defuzzzification.

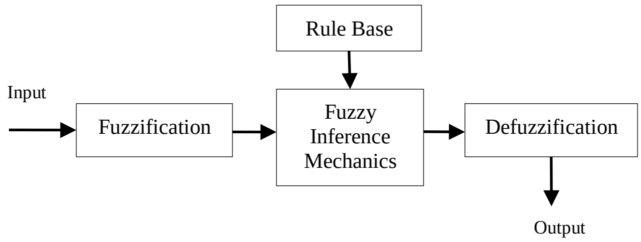


Fig. 4. Fuzzy Logic Controller

First step is identifying the linguistic input and output variables and definition of fuzzy sets. Fuzzification (or fuzzy classification) is the process of converting a set of crisp data into a set of fuzzy variables using the membership functions (fuzzy sets). For example in Figure 5, the degree of membership for a given crisp is 0.7. Shape of the membership functions depends on the input data can be triangular, piecewise linear, singleton, trapezoidal or Gaussian.

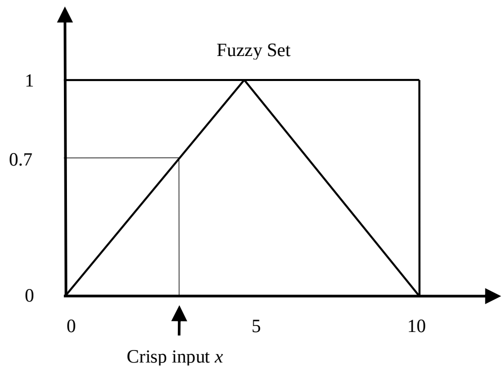


Fig. 5. Membership degree of a crisp input x in the fuzzy set

A rule base is obtained by a set of IF-THEN rules and inference evaluates the rules and combines the results of the rules. The final step is Defuzzification which is the process of converting fuzzy rules into a crisp output. An example of a simple fuzzy control system is shown in figure 6.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

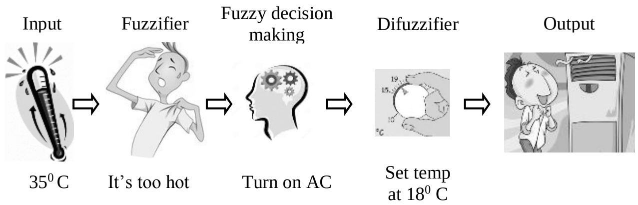


Fig. 6. Example of a fuzzy control system

Fuzzy systems have the advantage that the information which stores by fuzzy rules can be easily interpretable. Moreover they provide a simple interface for the extension of the system with new information (by adding new rules) or modifying the existing rules. The major problem with Fuzzy systems is that they totally depend on the experts who design them. It only uses the encoded information within the system and cannot learn itself and incapable of generalization. This nature of Fuzzy Systems indicates that merging with ANNs may possibly lead to a powerful computational model.

C. Artificial Neural Networks

ANN's are a form of artificial intelligence controllers and are the loose interpretation of biological neural networks. Neuron is the key factor which led to the development of the artificial neural network. A neuron is the basic unit of the biological nervous system designed to generate an electrical impulse. Using the impulse, neuron is able to transmit and process information.

There are four main components of a biological neuron cell that each carry out a specific function (Table 4).

Component	**Function**
Dendrites	Input
Cell body	Integration
Axon	Conduction
Pre-synaptic terminals	Output

Table 4. Main Components of a Biologic Neuron

An artificial neural network is made up of connections of the artificial neurons. The artificial neurons simulates the basic elements of is biological counterpart; accepting inputs, processing these inputs, turning the processed inputs into outputs and connecting to other neurons [32].

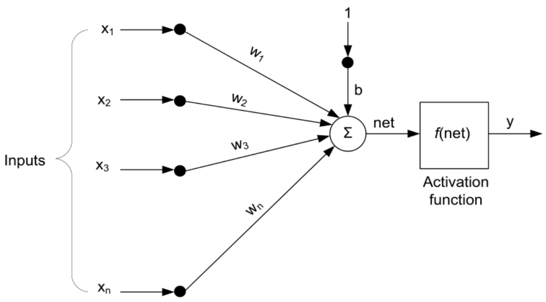


Fig.8. Structure of the artificial neuron

In Figure 8, inputs are represented by $x_1, x_2, x_3, \dots\dots\dots, x_n$ which are multiplied with corresponding weights $w_1, w_2, w_3, \dots\dots\dots, w_n$. Sometimes a threshold term b is added to the inputs. All inputs are multiplied by their corresponding weights and added together to form the net input to the neuron called net.

Mathematically,

$$net = \sum_{i=1}^{n} wi * xi + b$$
$$= w_1x_1 + w_2x_2 + w_3x_3 + \dots\dots\dots + w_nx_n + b$$

The neuron behaves as activation or mapping function f (net) to produce an output y which can be expressed as:

$$y = f\,(\text{net}) = f\left(\sum_{i=1}^{n} wi * xi\right)$$

Where, f is called the neuron activation function or the neuron transfer function. The most common activation functions used are the linear, threshold and sigmoid function (Table 5).

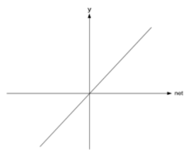
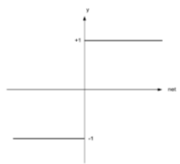
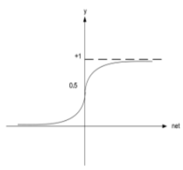
Activation Function	**Equation**
 **Linear**	In this case, $y = 1$ $y = f\,(\text{net}) = \sum_{i=1}^{n} wi * xi$ = net
 **Threshold**	The expression of the output y in this case can be written as: $y = \begin{cases} +1 & if\ net>0 \\ -1 & if\ net<0 \end{cases}$
 **Sigmoid**	The neuron transfer function is given by: $y = \frac{1}{1+exp(-\frac{net}{T})}$ where T is a constant

Table5. Activation Functions [33]

In 1943, when McCulloch and Pitts first implemented ANN, without the use of computers, it was formulated purely with mathematical models. The network developed by that time contained only two layers of neurons that could compute Boolean functions. These were known as a single layer perceptron. With the advancement of science the complexity of networks increases. Now ANNs are more accessible through computer simulation. Modern networks are much more complex, namely multi-layer perceptron (Figure 9). These networks consist of a series of parallel layers of nodes; the inputs, hidden and outputs.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

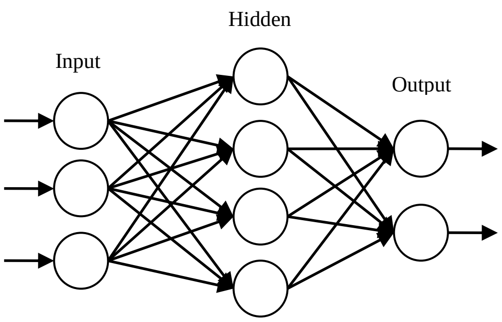


Fig.9. A feed forward multi-layer perceptron

The structure of a network is related closely to its function, a network where connections propagate from input to output nodes in the forward direction only (Figure 9) are feed forward networks. Alternatively, networks where nodes can connect to nodes in any direction including from output nodes to input nodes are called recurrent networks [34].

ANN controllers are not tuned like PID controllers. They incorporate learning. Learning means self-adjustment of connection weights between neurons till efficiency is achieved. Connection weights are learned/ changed through a training process. Training occurs in iterations of examples. A number of iterations are required to achieve a trained ANN which is based on complexity of system as well as performance efficiency needed.

ANN can be categorized on the basis of their learning strategy: Supervised, unsupervised or reinforcement learning and hybrid learning. Supervised and unsupervised learning are the popular strategies. However, supervised type of learning is frequently used in the majority of ANN applications.

Supervised Learning

In supervised method, the system is given a training set consisting of inputs with their corresponding output values, when the inputs are passed through the network the resultant output is compared to the desired output from the training set. The difference, known as the error is propagated back through the system and individual weights are adjusted to reduce the error. This process occurs over and over and each time small adjustments are made to the weights so that the output converges with the desired output.

Unsupervised Learning

In this learning, no desired or target value is available to the network and only the set of input is present. The system must learn itself what features it will use to group the inputs. This is often referred to as adaption or self-organization.

Training an ANN can be done in simulation, implementation, or may be needed in both.

ANNs are influential computational models for solving complex estimation and classification problems as they are robust and are capable of high level generalization, moreover they can handle incomplete data, too [35]. However no information can be extracted from a trained neural network about the connections between the parameters, e.g. a generic ANN model can only approximate the output parameters but cannot tell what kind of connections exist between the input and output parameters. This is a key disadvantage of the Neural Network model which led to the creation of Neuro-Fuzzy Systems.

D. Neuro-Fuzzy Controllers

The main purpose of fuzzy logic control (FLC) is to design a mathematical model of a human control expert which is capable of controlling the plant without thinking. The control expert specifies its control actions in the term of linguistic rules. These control rules are converted into the fuzzy set theory framework to simulate the behavior of the control expert. The parameters of good linguistic rules depends on the control expert’s knowledge, but the translation of these rules into framework of fuzzy set theory is not formalized and arbitrary choices concerning, for e.g., the shape of membership functions have to be made. The attributes of fuzzy logic controller can be awfully affected by its choice of membership functions. Hence methods for tuning fuzzy logic controllers are significant.

Neural networks offer the possibility of solving the tuning problem. Although a neural network is well known for its ability to learn and adapt to unknown/changing environment to acquire better performance. The trained network can be inferred as a black box. Neither it is feasible to extract information from the trained neural network nor can we incorporate information into the neural network in order to facilitate the learning procedure. On the other hand, a fuzzy logic controller is designed in the form of rules to work with the structured knowledge and nearly everything in the fuzzy system remains profoundly transparent as well as easily interpretable. However, no proper framework exists for the choice of various design parameters and generally these parameters are optimized through trial and error.

A brief comparison between fuzzy logic and neural networks from the point of knowledge acquisition, uncertainty, reasoning, adaptation and natural language processing is shown in Table 6. The merging of these two fields results in a paradigm called "*neuro-fuzzy networks*" or "*neuro-fuzzy systems*".

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

	Skill	Fuzzy Systems	Neural Networks
Knowledge acquisition	Input Tools	Human experts Interaction	Sample sets Algorithms
Uncertainty	Information	Quantitative and qualitative	Quantitative
	Cognition	Decision making	Perception
Reasoning	Mechanism	Heuristic search	Parallel computations
	Speed	Low	High
	Fault-tolerance	Low	Very High
Adaptation	Learning	Induction	Adjusting synaptic weights
Natural language	Implementation	Explicit	Implicit
	Flexibility	High	Low

Table6. A comparative study between fuzzy systems and neural networks [38]

The neuro-fuzzy controller uses the neural network learning techniques to tune the membership functions while keeping the connotation of the fuzzy logic controller intact [37] [38]. This new approach combines the well established advantages of both the methods and avoids the drawbacks of both.

IV. Fuzzy Supervised PID Controllers

A conventional controller is not sufficient for controlling a non-linear process to obtain a desired performance. To ensure better performance and stability for all the operational set point in nonlinear process, the controller gains should change to adapt the variation of physical parameters. Fuzzy inference can be used to tune the PID controller gains for improvement in performance of system. It serves a nonlinear mapping from the error signal $e(t)$ and change in error $\Delta e(t)$, to the PID gain parameters K_p , K_i , and K_d [40]. Figure 10 shows the basic structure of the controller.

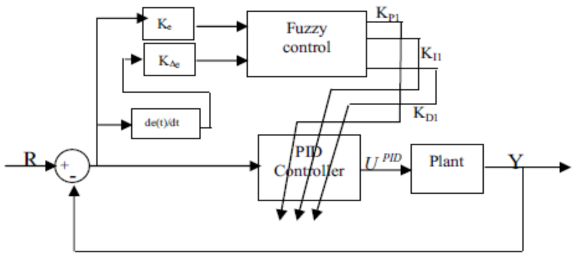


Fig. 10. Fuzzy supervised PID Controller [39]

The fuzzy supervised PID controller has two inputs error "e" and rate of change-in-error "Δe" and the output of the controller generates K_p, K_i, and K_d values for tuning PID gains. The function of the fuzzy supervisory is to generate a desired value for each one of the three parameters. According to the principle of fuzzy control the three parameters are used to modify in order to meet different requirements for control parameters when "e" and "Δe" are different and making the control object to produce a good dynamic and static performance.

V. Conclusion

A survey of various control techniques for robotic systems was carried out in this work. This overview of various information about classical PID Controller, Fuzzy Logic Controllers, Artificial Neural Network as well as Advanced controllers, focuses on its usability and challenges. It also gives conceptual overview of methodology.

References

[1] M.W. Spong, S. Hutchinson and M. Vidyasagar, *Robot Modeling and Control*, 1st Edition, Jon Wiley & Sons Inc, 2005.
[2] J.J. Crage, Introduction to Robotics Mechanics and Control, 3rd Edition, Prentice Hall, 2005.
[3] T. R. Kurfess, Robotics and automation handbook, CRC, 2005.
[4] J. J. E. Slotine and W. Li, Applied nonlinear control, *Prentice hall Englewood Cliffs, NJ*, vol. 461, 1991.
[5] K. Ogata, *Modern control engineering*, Prentice Hall, 2009.
[6] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties, pp.189-194, 2008.
[7] J. J. D'Azzo, C. H. Houpis, and S. N. Sheldon, Linear control system analysis and design with MATLAB, *CRC* 2003.
[8] B. Siciliano and O. Khatib, Springer handbook of robotics, *Springer-Verlag New York Inc*, 2008.
[9] A. T. Kambiz and M. Augustin, *Introduction to PID Controllers - Theory, Tuning and Application to Frontier Areas*, 2012.
[10] National Instruments, *PID theory explained*, 2006. Retrieved from http://zone.ni.com/devzone/cda/tut/p/id/3782.
[11] R. S. Barbosa, J. A. T. Machado and I. M. Ferreira, "Tuning of PID controllers based on Bode's ideal transfer function", *Nonlinear dynamics*, pp.38 305-321, 2004.
[12] R. S. Barbosa, J. A. T. Machado and I. M. Ferreira, "A fractional calculus perspective of PID tuning", *In Proceedings of ASME 2003 design engineering technical conferences and Computers and information in engineering conference*. Chicago: ASME.
[13] K. J. Åström and T. Hägglund, "Benchmark systems for PID control," in *Proceedings of IFAC Workshop on Digital Control: Past, Present and Future of PID Control*, Terrassa, Spain, pp. 165–166, 2000.
[14] K. J. Åström, T. Hägglund, C. C. Hang and W. K. Ho, "Automatic tuning and adaptation for PID controllers - a survey", *Control Engineering Practice*, vol. 1, no. 4, pp. 699-714, August 1993.
[15] K. M. Passino, "Bridging the gap between conventional and intelligent control," Special Issue on Intelligent Control, *IEEE Control Systems Magazine*, vol. 13, no. 3, pp. 12–18, 1993.
[16] K. J. Åström, "Towards intelligent control," *IEEE Control System Magazine*, vol. 9, no.3, pp. 60-64, 1989.
[17] K. J. Åström, "Where is the intelligence in intelligent control?", *IEEE Control System Magazine*, vol. 11, no.1, pp. 37-39, 1991.
[18] K. J. Åström, C. C. Hang, P. Persson and W. K. Ho, "Towards intelligent PID control", *Automatica*, vol. 28, no. 1, pp. 1-9, 1992.
[19] K. J. Åström, J. J. Anton and K. E. Årzén, "Expert Control", *Automatica*, vol. 22, no. 3, pp. 277-286, 1986.
[20] M. Santos, J. M. de la Cruz, S. Dormido and A. P.de Madrid, "Between Fuzzy-PID and PID-Conventional Controllers: a Good Choice*," in Proceedings of Biennial Conference of the North American Fuzzy Information Processing Society, NAFIPS*, Berkeley, CA, USA, pp. 123-127, 1996.
[21] R. E. King, "*Computational Intelligence in Control Engineering"*, Marcel Decker, New York, 1999.
[22] G. Stephanopoulos and C. Han, "Intelligent systems in process engineering: a review," *Computers & Chemical Engineering*, vol.20, no. 6-7, pp 743-791, 1996.
[23] J. Zumberge and K. M. Passino, "A Case Study in Intelligent vs. Conventional Control for a Process Control Experiment," *Journal of Control Engineering Practice*, vol. 6, no. 9, pp. 1055–1075, 1998.
[24] K. J. Åström, "Intelligent tuning," in *Adaptive Systems in Control and Signal Processing*, L. Dugard, M. M'Saad and I.D. Landau, eds., Pergamon Press, Oxford, pp. 360–370, 1992.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

[25] K. M. Passino, "Bridging the gap between conventional and intelligent control," Special Issue on Intelligent Control, *IEEE Control Systems Magazine*, vol. 13, no. 3, pp. 12–18, 1993.
[26] K. M. Passino and S. Yurkovich, *Fuzzy Control*. Menlo Park, CA: Addison Wesley Longman, 1998.
[27] K. M. Passino, "Intelligent control", in *The Control Handbook*, W. Levine, ed., Boca Raton: CRC Press, pp. 999–1001, 1996.
[28] R. J. P. deFigueiredo and G. Chen, *Nonlinear Feedback Control Systems*. Academic Press, New York, 2003.
[29] L. A. Zadeh, "Fuzzy Sets", *Information and Control*, vol. 8, pp. 338-353, 1965.
[30] M. Dotoli, B. Maione and B. Turchiano, "Fuzzy-Supervised PID Control: Experimental Results," EUNITE the 1st European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems, pp. 31–35, 2001.
[31] C. C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I", IEEE Transactions on Systems, Man and Cybernetics, Vol. 20, No.2, pp. 404–418, 1990.
[32] X. Yao, "Evolving Artificial Neural Networks". In Proceedings of the IEEE, Vol. 87, no. 9, Pages 1423-1447, 1999.
[33] D. Skapura, *Building neural networks*. New York, New York: ACM Press, 1996.
[34] S. Ge Shuzhi, T. H. Lee and C. J. Harris, "Adaptive Neural Network Control of Robotic Manipulators". World Scientific Series in Robotics and Intelligent Systems – Vol. 19, 1998.
[35] Zs. J. Viharos and K. B. Kis, "Diagnostics of wind turbines based on incomplete sensor data", IMEKO World Congress Technical Diagnostics, Republic of Korea, 2012.
[36] D. H. Rao and M. M. Gupta, "Neuro-Fuzzy Controller for Control and Robotics Applications", Engng Applic. Artif. Intell. Vol 7, No.5, pp. 479-491, 1994.
[37] P. J. Werbos, "Neurocontrol and fuzzy logic: connections and design," Int. J. Approximate Reasoning, Vol. 6, pp. 185-220, 1992.
[38] D. Nauck, F. Klawonn and R. Kruse, "Combining neural networks and fuzzy controllers," In E. P. Klement and W. Slany, editors, Fuzzy Logic in Artificial Intelligence, Springer-Verlag, Berlin, pp. 35-46, 1993.
[39] K. D. Sharma, M. Ayyub, S. Saroha and Faras Ahmad, "Advanced Controllers Using Fuzzy Logic Controller (FLC) for Performance Improvement", International Electrical Engineering Journal (IEEJ) Vol. 5 No.6, pp. 1452-1458, 2014.
[40] K. K. Ahn and B. K. Nguyen, "Position Control of Shape Memory Alloy Actuators Using Self Tuning Fuzzy PID Controller", International Journal of Control, Automation, and Systems Vol. 4, No. 6, pp. 756-762, 2006.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)