Evolutionary Fuzzy Speed Regulation for a DC Motor*

M.-R. Akbarzadeh-T., Y. T. Kim & B. Feerouzbakhsh
NASA Center For Autonomous Control Engineering
Electrical Engineering and Computer Engineering Building
University of New Mexico, Albuquerque, NM 87131, USA
E-mail: akbazar@eece.unm.edu

Abstract

Fuzzy logic is a powerful tool in control of systems with ill-defined, inaccurate or unknown mathematical models. In classical applications of fuzzy logic, however, there is a great dependency on proper expert knowledge acquisition. In this paper, we will remove that dependency by using a Genetic Algorithm (GA) to automatically determine parameters of fuzzy rule sets such as membership functions. This approach differs from conventional applications of GA-fuzzy knowledge development in that expert knowledge is incorporated in creating an initial highly fit population while allowing for randomness among members of the population for diversity. This method is useful for search in GA-hard landscapes and is successfully applied to speed regulation of a DC motor. It is shown that the presented method improves upon the initial fuzzy knowledge-base and significantly outperforms classical PID response.
Keywords: DC Motor Speed Regulation, Genetic Algorithms, Fuzzy Control, GA-Fuzzy Systems.

1 Introduction

Due to its excellent speed control characteristic, the DC motor has been widely used in industry even though its maintenance costs are higher than the induction motor. As a result, speed control of the DC motor has attracted considerable research and several methods have evolved. Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers have been widely used for speed control of the DC motor. Kim et. al. [1] surveyed the current state of the PI, PID and command matching controllers for speed regulation of DC motors. To reduce the loading effect and minimize time delay, he added a feed-forward controller to the PID controller. In [2], the feedback gains of a PI controller are first nominally determined and thereafter tuned using fuzzy logic. Yousef [3] determined a fuzzy logic-based controller with superior performance over a DC motor's PI controller. Yousef controlled both speed and current variables. In [4], three different intelligent control architectures are considered. There, a feed-forward/feedback control strategy is used to ensure effective, high performance tracking of reference speed trajectories.

The above works indicate successful utilization of fuzzy logic over non-fuzzy PI and PID controllers in regulating DC motor drive systems. Yet, for *best* response, the above approaches have no capability to search for an *optimal* knowledge base. In particular, there are still two problems in conventional fuzzy reasoning. One is the lack of a definite method to determine the membership functions and the second is the lack of a learning ability. In this paper, an automatic way of searching for optimal knowledge is proposed and applied to the speed regulation problem.

This approach utilizes a Genetic Algorithm (GA) as a search tool in the multi-modal and complex landscape of knowledge domain. GAs are directed random search optimization routines modeled after nature's evolutionary process [5]. GAs have demonstrated the unique coding capability to represent fuzzy rule sets and parameters of membership functions. Genetic algorithms, therefore, have the capability to evolve fuzzy rule sets and adapt fuzzy controllers to changing operating conditions [6].

There are various approaches in design of evolutionary fuzzy controllers. Most of these approaches differ in their coding of fuzzy rule sets, membership functions and definition of fitness functions. In [7], Lee proposed a GA-based fuzzy controller design method which determines membership functions, number of fuzzy rules, and rule consequent parameters at the same time. In [8], the imprecision in available information is exploited and elements of guess work are

*This work was supported in part by NASA contract # NCCW-0087 and by Waste Education and Research Consortium grant under Award # WERC/NMSU/DOE Amd 35

employed to arrive at *acceptable* solutions. Cooper [9] evolved each fuzzy system in the form of a flexible rule base rather than as a list of parameters under a fixed set of membership functions. This algorithm was successfully applied to the boat rudder control problem. In [10, 11], off-line and online applications of genetic algorithms in knowledge optimization are explored. Although GA is in general, not suitable for real-time control, it is shown that its application is feasible within the structure of adaptive fuzzy control.

One of the important aspects in applying Genetic Algorithms to knowledge enhancement of fuzzy systems is how to incorporate existing expert knowledge into the GA-optimizing algorithm. And generally, how to take advantage of *several* experts' opinions in creation of initial population. Conventional applications of GA-Fuzzy suggest a random initial population. However, it is intuitively clear that any search routine could converge faster if starting points are good solutions. In this paper, a methodology is illustrated which incorporates expert knowledge in creating an initial population while allowing for randomness among members of the population for diversity. Furthermore, the methodology is applied to response optimization of a DC servo motor. Superior performance is demonstrated as the intelligent fuzzy controller enhances its knowledge base by utilizing the genetic algorithms optimization approach.

This paper is organized as follows. The following section formulates the system model of a a DC motor. Section 3 discusses the structure of the GA-Fuzzy controller and the method of incorporating initial knowledge. In Section 4, the simulation results of the corresponding system is then compared with non-fuzzy PID and non-optimized fuzzy PID controllers.

2 System Model

Let us now consider a separately excited DC motor as is shown in figure 1. The equations describing the dynamic behavior of the DC motor are given by

$$V_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + k\omega(t) \qquad (1)$$

$$T(t) = J\frac{d\omega(t)}{dt} + \beta\omega(t) + T_1(t) = ki_a(t) \qquad (2)$$

Where $\omega(t)$ rotational speed, $i_a(t)$ armature circuit current, $T_1(t)$ constant torque-type load, $R_a(t)$ armature circuit resistance, β coefficient of viscous-friction, k torque coefficient, J moment of inertia, and L_a armature circuit inductance. In state space form, if we

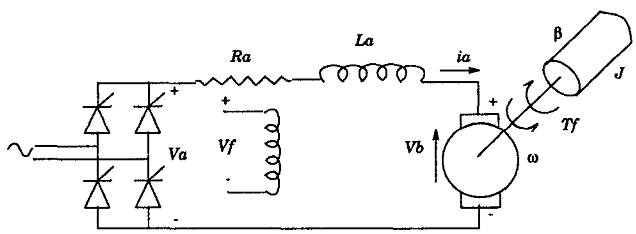

Figure 1: The drive system of the separately excited DC motor

let

$$\begin{aligned} x_1(t) &= i_a(t), \quad x_2(t) = \omega(t) \\ u(t) &= V_a(t), \quad d(t) = T_1(t) \end{aligned} \qquad (3)$$

be our choice of state and control variables, then the state space model of the system can be represented by the following

$$\dot{x}(t) = Ax(t) + Bu(t) + Ed(t) \qquad (4)$$

$$y(t) = Cx(t) \qquad (5)$$

Where $x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$, $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$, $A = \begin{bmatrix} \frac{-R_a}{L_a} & \frac{-k}{L_a} \\ \frac{k}{J} & \frac{-\beta}{J} \end{bmatrix}$, $B = \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix}$, and $E = \begin{bmatrix} 0 \\ \frac{-1}{J} \end{bmatrix}$. The load torque is considered as disturbance input.

2.0.1 Numerical Values

The DC motor under study has the following specifications and parameters:
a) Specifications:
1 hp, 220 volts, 4.8 ampers, 1500 rpm
b) Parameters:
$R_a = 2.25\Omega, L_a = 46.5mH, J = 0.07kg{\cdot}m^2, \beta = 0.002N{\cdot}m \cdot \frac{sec.}{rad.}, k = 1.1V\frac{sec.}{/rad}$

3 The Control Architectures

The control architecture used here is the standard application of genetic algorithms on fuzzy controllers with Proportional, Integral, and Derivative variable inputs. Through the GA, various combinations of candidate solutions are evaluated and the best, *fittest*, solution is chosen to control the actual system. The GA has the capability to alter the scaling factor or shape of the membership functions of individual inputs and outputs, and to modify the rule set fuzzy associative memory. Figure 2 illustrates a block diagram of the closed loop control system.

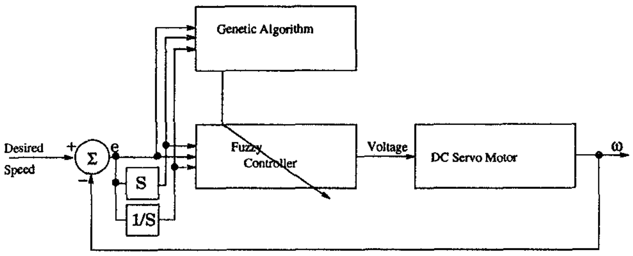

Figure 2: GA-optimized fuzzy PID control architecture.

3.1 Incorporating Initial Knowledge

The conventional GA applications generate a random initial population without using any expert knowledge. This, in general, will provide a more diverse population while sacrificing convergence time. However, it is intuitively clear that any search routine could converge faster if starting points are good solutions. Considering this and the availability of some form of expert knowledge in most applications of fuzzy logic raises the following question. How can one incorporate initial knowledge in the process of genetic evolution to achieve better results faster.

In [12] the process of *seeding* the initial population with one or more experts' knowledge is discussed. The few seeded chromosomes then had the chance of reproducing through mutation and crossover with other randomly generated chromosomes in the population. In this paper, a different method is applied and is based on a *grandparent* scheme [13], where all individuals in the initial population are created based on a mutation from the "knowledgeable" grandparent. This scheme takes advantage of expert knowledge while maintaining the necessary diversity for effective search. An analogy to this approach is the biblical theory of creation where all men were created from Adam & Eve.

4 Simulation

In this paper, three different controllers are simulated and compared. The first simulation involves a model-based PID controller as discussed by kim et. al. [1]. The corresponding control law is as follows,

$$u(t) = -P\omega(t) + Q\int_{t_0}^{t_f}(\omega_r - \omega)dt - R\frac{d\omega(t)}{dt} \quad (6)$$

where $\omega_r = 10.0\frac{rad}{sec}$ is the reference input, $P = 1.1712, Q = 13.236, R = 0.03$.

The second simulation is a fuzzy-PID control law $u(t)$ based on a crude expert knowledge.

$$u(t) = f(e, \dot{e}, \int e) \quad (7)$$

where f is a nonlinear function determined by the fuzzy associative memory and parameters of input and output membership functions, $e = \omega_r - \omega$, $\dot{e} = \dot{\omega}_r - \dot{\omega}$, and $\int e = \int^t{}_0(\omega_r - \omega)dt$.

In the third simulation, genetic algorithm is used to optimize parameters of the above fuzzy controller.In order to minimize the parameter set, GA is applied to optimize *only* the input membership parameters of the fuzzy controller as is shown in Figure 2. Other parameters in the knowledge-base are not allowed to vary. This will reduce simulation's processing time and will still demonstrate the potential utility of GA. The following fitness function was used to evaluate various individuals within a population of potential solutions,

$$fitness = \frac{1}{t_f - t_i}\int_{t_i}^{t_f}\frac{1}{k_1e^2 + k_2\dot{e}^2 + k_3\gamma^2 + 1}dt, \quad (8)$$

where e and $\dot{e}$ represent the errors in angular position and velocities, γ represents overshoot, and k_1, k_2, k_3 are design parameters. Consequently, a fitter individual is an individual with a lower overshoot and a lower overall error (shorter rise time) in its time response. The above fitness function is normalized such that a fitness of 1 represents a perfectly fit individual with zero error and overshoot. Similarly, a divergent response receives a fitness of zero. The design parameters k_i are extremely important since a perfect solution, i.e. zero error and overshoot, does not exist in real systems. If a particular attribute is more important than the others, its corresponding multiplier k_i is increased. In this simulation, the following values were used for k_i.

$$k_1 = 25, \quad k_2 = 150, \quad k_3 = 1 \quad (9)$$

Figure 3 shows the maximum and average fitness of each GA's generation. A total of 40 generations were simulated while each generation included 100 individuals. The performance measure never reaches steady state since it is constantly trying out new directions of search through mutation. As is shown in figure 3, the curve for the maximum fitness converges very quickly, i.e. within the first two generations. However, the fitness of the whole population converges within 20 generations. The mutation rate for creating the initial population was set at 0.1. Thereafter, the mutation rate was set at 0.033. The probability of crossover was set to 0.6.

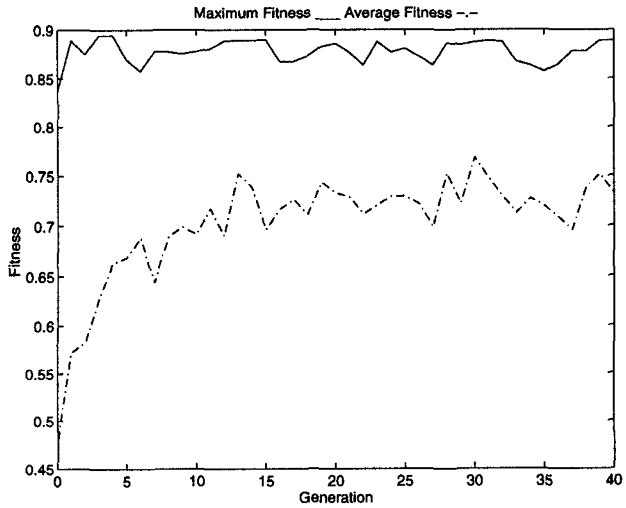

Figure 3: Plot of maximum and average fitness values

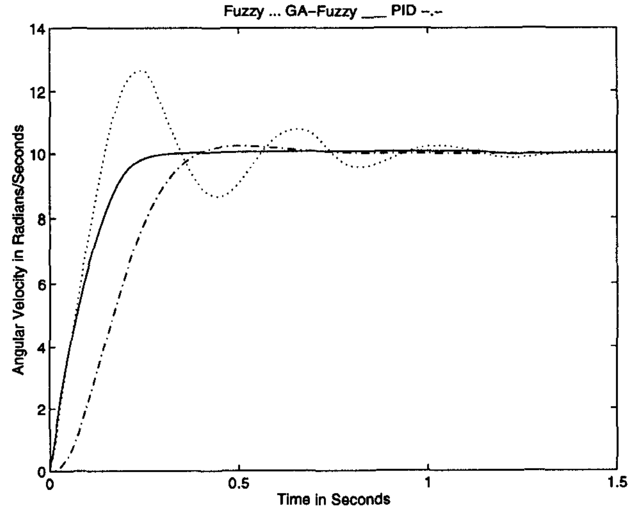

Figure 4: Comparison of model-based PID controller with GA-Optimized Fuzzy Controler

Figure 4 illustrates the three controller's time responses. The GA-optimized fuzzy controller is a significant improvement over the initial fuzzy controller based on crude expert knowledge. While keeping the same rise time, the GA-optimized controller has no oscillations and almost no overshoot. When compared with the model-based PID controller, the GA-optimized fuzzy controller also shows significant improvement. In this respect, the overshoot and rise time is reduced by over half.

5 Conclusion

As we enter the era of more complex systems, the need for more intelligent and ultimately autonomous controllers arises. This need is currently being addressed to by applying fuzzy logic to bridge the gap between machine's number processing capability and human's thinking. Even though the power of original thinking and innovation is what we look for in autonomous systems, fuzzy logic, in its conventional form, does not provide that power. That is why we equip fuzzy logic with nature-based evolutionary algorithms in search of machine self-innovation. Through genetic operations such as mutation and crossover, GA is able to invent and re-combine new search paths.

This paper is an example of how a fuzzy controller can be optimized by genetic algorithms. In this search of an *optimal* solution, we have equipped the optimization algorithm with a new method to take advantage of any a-priori knowledge. As a result, the optimization algorithm converges much faster than is reported by most other application s of genetic algorithm. It is shown that the resulting optimal solution performs significantly better than both the initial crude fuzzy controller and the model-based PID controller.

The future of this research is continued by real-time experimentation of the proposed algorithms. Genetic algorithms depend heavily on a large population, whereas in real time systems, the population often consists of only one individual!

References

[1] Y. T. Kim & S. H. Baek, "The Speed Regulation of a DC Motor Drive System with a PI, PID, and Command Matching Controllers," *Dongguk Journal*, (in Korean), Vol. 29, 1990, pp.525-541.

[2] D. P. Iracleous & A. T. Alexandridis, "Fuzzy Tuned PI controllers for Series Connected DC Motor Drives," *Proceedings of the IEEE International Symposium on Industrial Electrocincs*, Athens, Greece, pp. 495-499, 1995

[3] H. Yousef & H. M. Khalil, "Fuzzy Logic-based Control of Series DC Motor Drives," *Proceedings of the IEEE International Symposium on Industrial Electronics*, Athens, Greece, 1995.

[4] M. E. Fisher, A. Ghosh, & A. M. Sharaf, "Intelligent Control Strategies for Permanent Magnet DC Motor Drives," *Proceedings of the IEEE International Conference on Power Electronics*, New Delhi, India, 1996.

[5] D. E. Goldberg,"Genetic Algorithms in Search, Optimization and Machine Learning," *Addison-Wesley*, MA, NY, 1989.

[6] A. Homaifar & E. McCormick, "Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms," *Transactions on Fuzzy Systems*, Vol. 3, No. 2, pp.129, 1995.

[7] M. A. Lee and H. Takagi, "Integrating Design Stages of Fuzzy Systems Using Genetic Algorithms," *Proceedings of the 1993 IEEE International Conference on Fuzzy Systems*, San Francisco, CA, pp. 612-617.

[8] T. Williams, "Fuzzy, Neural, and Genetic Methods Train to Overcome Complexity," *Computer Design*, Vol. 34. pp. 59, May, 1995.

[9] M. G. Cooper,"Evolving a Rule-Based Fuzzy Controller," *Simulation*, Vol. 65 pp. 67-72, July, 1995.

[10] D. A. Linkens & H. O. Nyongesa, "Genetic Algorithms For Fuzzy Control. Part 1. Offline System Development and Application," *IEEE Proceedings, Control Theory and Applications* Vol. 142, pp. 177-185, May, 1995.

[11] D. A. Linkens & H. O. Nyongesa, "Genetic Algorithms For Fuzzy Control. Part 2. Online System Development and Application," *IEEE Proceedings, Control Theory and Applications* Vol. 142, pp. 161-176, May, 1995.

[12] M. A. Lee, and H. Takagi, "Embedding Apriori Knowledge into an Integrated Fuzzy System Design Method Based on Genetic Algorithms," *Proceed ings of the 5th IFSA World Congress*, 1993.

[13] M.-R. Akbarzadeh-T. & M. Jamshidi, "Evolutionary Fuzzy Control of a Flexible-Link," *To appear in the Journal of Intelligent Automation and Soft Computing, 1996.*