Tuning and Optimisation of Membership functions of Fuzzy Logic Controllers by Genetic Algorithms

by

*M. Mohammadian**, *R J Stonier***
*Department of Computer Science
Edith Cowan University
Perth WA 6050 Australia
**Dept. of Mathematics and Computing
Central Queensland University
Rockhampton Qld 4702 Australia

Abstract

A method is developed to tune and optimise the membership functions of Fuzzy Logic Controllers (FLC) by using Genetic Algorithms (GAs). The set of fuzzy If-Then rules and their membership functions of the truck back-upper problem is considered.

1. Introduction

Fuzzy modelling or fuzzy identification, has numerous practical applications in control [1, 2], prediction and inference [3, 4]. The majority of FLC systems to date have been static and based upon knowledge derived from imprecise heuristic knowledge of experienced operators, and where applicable also upon physical laws that governs the dynamics of the process [11, 12, 13]. One basic aspect of fuzzy modelling which is in need of better understanding is the need for effective methods for tuning the membership functions (MF'S) of FLC so as to minimise the output error measure or maximise the performance index of the system.

Nguyen and Widrow [5] first investigated the application of artificial neural networks to the truck back-upper problem. They used two neural networks to build a self-learning controller to emulate and control the truck. The neural network controller and emulator required a significant amount of time to train, learning was time consuming, and in some cases the learning algorithm did not converge.

Kosko and Kong [6] studied the truck back-upper problem using FLC. They first trained the fuzzy logic controller (FLC) by encoding a common sense Fuzzy Associate Memory (FAM) bank. An adaptive fuzzy controller was then developed which generated FAM rules directly from training data using a product-space clustering algorithm. The extraction of driving knowledge was obtained by a comlicated off-line statistical approach.

Wang and Mendel [7] discussed the same problem using methodology similar to Kosko and Kong [6]. They also proposed a numerical-fuzzy controller. Extraction of knowledge again was obtained in an off-line manner.

Mohammadian and Yu and Smith [8, 9] also considered the truck back-upper problem using a knowledge acquisition architecture. This architecture enabled the extraction of driving knowledge to be automated in an on-line manner. A statistical learning approach was used which eliminated the experts knowledge required by Wang and Mendel's model [7].

In this paper we consider the same problem but develop a new architecture which tunes and "optimises" the fuzzy membership functions of the fuzzy truck back upper system.

2. Genetic Algorithms

GAs are algorithms for optimisation based on the principle of biological evolution. They simultaneously consider many points in the search space, and work not with the parameters themselves but with string of numbers representing the parameter set. Probabilistic rules are used to guide their search. By considering many points in the search space simultaneously the chance of converging to local optima is reduced [10].

GAs differ fundamentally from conventional search techniques, for instance:

1. *GAs consider a population of points, not a single point.*
2. *GAs work directly with strings of characters representing the parameter set, not the parameters themselves.*
3. *GAs use probabilistic rules to guide their search not deterministic rules.*

These properties that make GAs inviting as a technique for selecting high performance membership functions for FLC systems.

3. Implementation of GAs for tuning membership functions of FLC

The design and development of robust and optimal FLC systems can be achieved by establishing a fuzzy rule set and using GAs to find an 'optimal' membership function. These membership functions can be represented either by triangular

IEEE International Workshop on
Robot and Human Communication
0-7803-2002-6/94 $4.00 © 1994 IEEE

or trapezoidal shaped membership functions. In this study triangular fuzzy membership functions are used. These triangles can have variable based width and shift along the x-axis freely. Therefore each requires the definition of only one point to fix it.

There are two ways to achieve optimal membership functions:

1. Choose the entire membership functions as variables to be optimised.
2. Choose only the overlaps between membership functions as variables to be optimised.

In the first case the entire set of fuzzy membership functions for a FLC system must be represented as bit strings (of 0 and 1). This can become a lengthy and complicated procedure. To do this the best choice is to use a method called concatenated mapped, unsigned binary coding [10].

In the second case, the overlaps between different fuzzy memberships function are considered as the parameters to be optimised. Unlike the first case all the fuzzy membership functions are not needed to be coded into bit strings. This makes utilisation of GAs for fuzzy membership optimisation much easier and more efficient.

The only constraint placed on the individual triangles of membership functions is that the triangles bordering the extreme limits of the action or control must not be changed. This is because for almost all applications of FLC systems, the membership functions have two extreme limits (ie. upper bound and lower bound of the fuzzy membership functions).

Figure 1 shows the overlap between fuzzy membership functions of a FLC system.

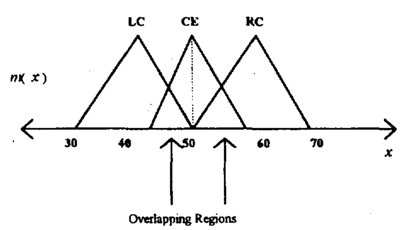


Figure 1 Overlapping regions of fuzzy membership functions of a FLC

Therefore modification of the bordering triangle of the membership functions can not exceed these two values. Figure 2 shows a set of membership functions of a FLC system and indicates the bordering triangle of the membership functions. As shown in Figure 2, the two bordering triangles **LC** and **RC** have the upper and lower bounds of (30,50) and (50,70) respectively. Now let us assume that the membership functions in Figure 2, are the membership functions of a output variable of a FLC. Then the values of the output variables of this FLC can only be between 30 and 70.

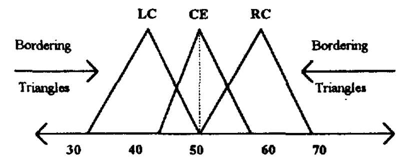


Figure 2 A set of membership functions of a FLC with bordering triangle of the membership functions

Triangles not bordering the extreme limits of the control variables can change their variable base and shift along the x-axis. Hence only one point or base of such a triangle can shift along the x-axis, whereas other triangles not bordering the extreme limits can shift any or both their bases along the x-axis.
We call the extreme bordering triangles, corner triangles and the triangles between the two extreme bordering triangles, inner triangles. Each corner triangle needs the definition of only one point to determine it, while each inner triangle needs the definition of two points to determine it.

For optimisation of membership functions of a FLC we need only to code the overlaps between fuzzy membership functions into bit strings. The bit strings representing the overlapping parameters then must be judged and assigned a fitness value, which is a score representing the degree to which they accomplished the goal of defining high performance. The squared-error term can be evaluated to determine the fitness of the strings in a population. This error could be the distance between the set point and the state of the system. However the definition of fitness function that enables the GAs to locate high performance and efficient membership functions is application dependent.

4. Application to truck back upper problem

Backing a truck to dock is difficult for all but the most skilled truck drivers. Normal driving instincts lead to erroneous movements, and much practice is needed to get it right. The truck back upper problem is a typical control problem about the determination of a global strategy for guiding a truck backing into a parking dock. The only available information is a knowledge of local conditions.

We first specify the docking work-space and the truck. Figure 3 shows a simulated truck and loading zone [6]. The FLC of truck back upper system has only two inputs, the x position and azimuth ϕ. The output is the steering angle θ. We assume that there exists enough clearance between truck and the loading dock for the y coordinate to be ignored., The variables x and ϕ determine the truck position and angle of the truck with the horizontal. The goal is to make the truck arrive at the loading dock at a right angle ($\phi_d = 90^o$) and to align the position (x,y) of the truck with the desired loading dock (x_d, y_d). In this study only backing up is considered, that is the truck moves backwards by some fixed distance at every stage.

The linguistic fuzzy subsets for x, ϕ and θ, and the membership functions associated with the subsets for x, ϕ, are the same as used in [6]. But the membership functions associated with subsets for θ are set to have no overlaps.

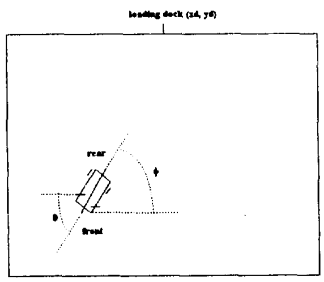


Figure 3 Diagram of simulated truck and loading zone

They are defined as follows:

NB ↔ [-30, -20] NM ↔ [-20, -7.5]
NS ↔ [-7.5, -2.5] ZE ↔ [-2.5, 2.5]
PS ↔ [2.5, 7.5] PM ↔ [7.5, 20]
PB ↔ [20, 30]

The reason for setting overlap to zero is that we do not know in advance what overlap will give good results. The amount of overlaps will be used as the parameter for optimisation by GAs.

Suppose we have the following fuzzy rules that are obtained by an analysis of the truck system and using the operators' knowledge. It is possible that we can not obtain all the fuzzy rules for the system correctly. There is also a lack of knowledge of what is the best setting of the membership functions of fuzzy regions and their corresponding overlap. The following fuzzy sets in Figure 4 and fuzzy rule in Table 1 have been chosen for this problem. We have used the fuzzy rules of Table 1, and a GA to optimise the underlying fuzzy membership functions of the fuzzy rules. We are interested to finding an optimised backing strategy that guides the truck to the dock, and which minimises the docking error.

A number of fuzzy rules in Table 1 do not give good performance. For example consider the two trajectories obtained by the FL controller in Figures 5 (a) and (b).

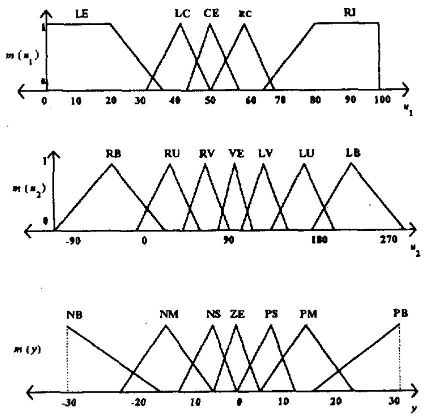


Figure 4 Fuzzy regions and the corresponding membership functions of u_1, u_2 and y

		u_1				
		LE	**LC**	**CE**	**RC**	**RI**
	RB	PS	PM	PM	PB	PB
	RU	NM	NB	NS	NS	PM
u_2	**RV**	ZE	PS	NB	ZE	PB
	VE	NS	NS	NB	PS	PM
	LV	NS	NB	NS	PM	PB
	LU	NM	NS	NS	NM	PS
	LB	NB	NS	NM	NB	NS

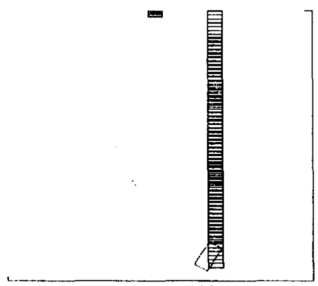
Table 1 Fuzzy rules of the truck back upper problem

This is because some of the fuzzy rules obtained by an analysis of the truck system and using the operators' knowledge are not correct. The fuzzy membership functions for θ are also not tuned, and the overlap between fuzzy membership functions that was arbitrarily set, is not good.

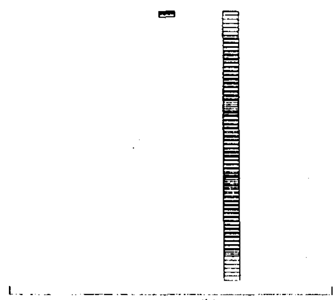
Let the docking error of the truck back upper system be defined by the following formula:

$$DE = \sqrt{(x_f - x_d)^2 + (y_f - y_d)^2 + (\phi_f - \phi_d)^2}$$

where DE stands for the Docking Error of the truck. x_f, y_f and ϕ_f correspond to the final position and final angle of the truck respectively. x_d, y_d and ϕ_d correspond to desired position and angle of the truck respectively. x_d, y_d and ϕ_d are fixed numbers taken to be given as: $x_d = 50,\ y_d = 100,\ \phi_d = 90^\circ$



(a)



(b)

Figure 5 Trajectory obtained by using fuzzy rules in Table 1 from initial condition (a) $x = 70, y = 10, \phi = 30^\circ$ and (b) $x = 70, y = 10, \phi = 90^\circ$ with 20% overlap

The fuzzy membership functions can be divided into fuzzy membership functions of input variables and fuzzy membership functions of the output variable. To code the fuzzy membership functions, three different approaches are suggested:

1. To code the entire input/output set of membership functions necessary to drive the truck back upper system as a bit string.
2. To code the entire overlap of the input/output membership functions necessary to drive the truck back upper system as a bit string.
3. To code the overlap between of the output membership functions of the truck back upper system, not the output membership functions parameters.

We have chosen the third case, namely to code the overlap of the output membership functions. This is more efficient than the first and the second choice. We need only to code and optimise the overlap between different output membership functions. The output membership functions of the truck back upper system in fact are the consequents of the fuzzy rules of the FLC. They determine the signal or control actions that must be taken, given the conditions of the rules that correspond to them, are satisfied. As mentioned, it is assumed that initially the overlap between the output membership functions of FLC of the truck back upper system is zero. In fact it is very difficult and unrealistic to determine the overlap of the membership functions in many real world applications, such as the truck back upper system.

We did not have to encode the output membership function parameters. Instead we need only encode the overlap between the membership functions. In this way the complex task of encoding and decoding of all output membership functions are eliminated. By optimising the underlying output (that is, control), membership functions of the truck back upper system, we reduce the docking error. A GA is used to optimise the overlapping area of the output membership functions. This method is similar to choosing all the points of the triangles of the output membership functions and move these points around on the x-axis until efficient and effective membership functions are found; without having to decode and encode all the base points of the fuzzy membership triangles. Figure 6 shows the output fuzzy membership functions of the truck back upper system.

Choosing the correct membership functions for a FLC system can produce effective and efficient systems. FLC systems have most of their powers because of the overlapping of their membership functions. That is FLC system invoke few rules at each instant to different degrees. By changing the overlapping regions between the fuzzy membership functions we are changing the underlying mathematical models of the fuzzy rules of Table 1 (ie. we are moving the base points of the fuzzy triangular membership functions).

The encoding and decoding of the overlapping regions is much easier then the encoding the decoding of all the points of all triangles. That is by choosing overlapping regions instead of the membership functions itself we make the convergence of GAs faster and eliminate the unnecessary computations, that must be performed each time to encode and decode the fuzzy membership functions.

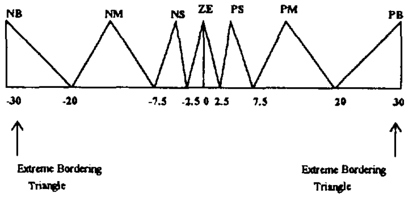


Figure 6 The output fuzzy membership functions of the truck back upper system before optimisation

It is assumed that the output namely θ, can only assume values in the range (-30,30) degrees. So modification of the bordering triangle of the membership functions can not exceed these two values. Triangles other than the bordering triangles can freely shift both their bases along the x-axis as long as they do not exceed the upper and lower limits of the fuzzy membership functions. They can change their variable base and shift along the x-axis. Therefore for the bordering triangles only one point or one base of the triangle can shift along the x-axis. Next we use a GA to optimise the output fuzzy membership functions of the truck back upper system shown in Figure 6.

The GA is run for a number of generations until we find a overlap value which gives satisfactory performance. We stopped after 10 generations were completed as the overlap between the fuzzy membership functions of the FL controller gave a small docking error. The fuzzy membership functions for θ with overlap 20% suggested by the GA after 10 generations are shown below.

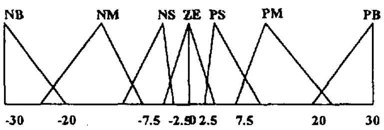


Figure 7 The output fuzzy membership functions for θ with 20% overlap

Figures 8 (a) and (b) show the control performance of the FL controller using the fuzzy rules with 20% overlap between fuzzy membership functions for θ of Table 1.

Comparing Figures 8 (a) and (b) with Figures 5 (a) and (b) it is obvious that 20% overlap between the membership functions for θ improves the FLC system performance significantly. The docking errors of the trajectories in Figures 5 (a) and (b) are 20.05, 20.0 respectively, whereas the docking errors of the trajectories in Figures 8 (a) and (b) are 0.1 and 0.08 respectively. As we can see the GA has found the overlap that reduces the docking error significantly.

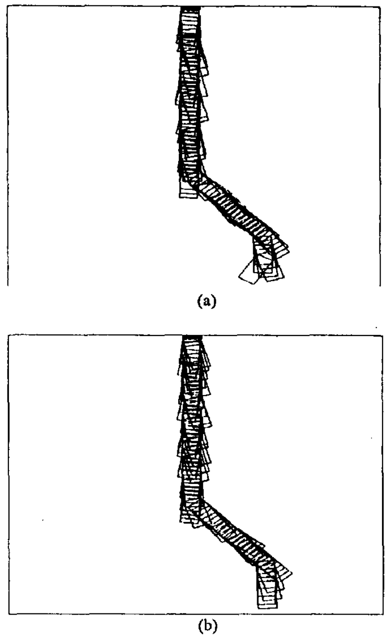


Figure 8 Trajectory obtained by using fuzzy rules in Table 1 from initial condition (a) $x = 70, y = 10, \phi = 30°$ and (b) $x = 70, y = 10, \phi = 90°$ with 20% overlap

5. Conclusion

In this paper we have shown that GAs can serve as a basis for tuning of fuzzy membership functions for nonlinear and dynamic systems.

In particular a GA has been used for optimisation of single overlap of fuzzy membership functions for a FLC system. It is also possible to use GAs to tune more than one overlap parameter and optimise the performance index of FLC systems.

The main features and advantages of using GAs for tuning fuzzy membership functions of FLC system can be summarised as:

1. This architecture provides us with a general method for tuning and optimisation of FLC systems
2. It can be used not only to reduce the error but also minimising or maximising other constraint of the system, such as reduction of overshoot etc.

References

[1] M. Sugeno, editor, *"Industrial applications of fuzzy control"*, Elsevier Science Pub. Co., 1985.

[2] W. Pedrycz, *"Fuzzy control and fuzzy systems"*, Wiley, New York, 1989.

[3] A. Kandel, *"Fuzzy expert systems"*, Addison-Wesley, 1988.

[4] A. Kandel, *"Fuzzy expert systems"*, CRC Press, Boca Raton, FL, 1992.

[5] H. D. Nguyen and B. Widrow, "Neural network for self-learning control systems", *IEEE Control Systems Magazine*, April 1990.

[6] B. Kosko, *"Neural networks and fuzzy systems, a dynamic system"*, Prentice-Hall, Englewood Cliff, 1992.

[7] L, X, Wang and J, M, Mendel, "Generating fuzzy rules by learning from example", *IEE Trans. Sys., Man and Cybernetics*, vol. 22, pp, 1414-1427, 1992.

[8] M. Mohammadian, X. H. Yu, and J. D. Smith, "A self-learning control system, docking a truck", *29th Australian Applied Mathematics Conference*, Adelaide, Australia, February 9-11 1993.

[9] Mohammadian, X. H. Yu and J. D. Smith, "A case study of knowledge acquisition: from connectionist learning to an optimised fuzzy knowledge base", *IEEE International Workshop on Emerging Technologies and Factory Automation*, Australia, Cairns , 27-29 September 1993.

[10] Goldberg D., *"Genetic Algorithms in search, optimisation and machine learning"*, Addison Wesley, 1989.

[11] Evans[G W., and Karwowski W., and Wilhelm M R., *"Application of the fuzzy set methodolgies in industrial engineering"*, Elsevier Netherlands, 1989.

[12] Takagi T., and Sugeno M., "Derivation of fuzzy control rules from human operators control actions", *Proceedings of the IFAC Symposium on Fuzzy Information Representation and Decision Analysis*, pp 55-60, July 1983.

[13] Mohammadian M., "Fuzzy linguistic approach for decision making and estimation", *Proceedings of the 1993 Annual Postgraduate Research Symposium, University of Central Queensland*, Australia, Rockhampton, pp 16-21 August 1993.