

Genetic Algorithm Based Fuzzy Control
of Spacecraft Autonomous Rendezvous

C. L. Karr¹, L. M. Freeman², and D. L. Meredith³

¹U.S. Bureau of Mines, Tuscaloosa Research Center,
Tuscaloosa, AL 35487-9777

²The University of Alabama, Department of Aerospace Engineering,
Tuscaloosa, AL 35487-0280

³The University of Alabama, Department of Computer Science,
Tuscaloosa, AL 35487-0290

ABSTRACT

The U.S. Bureau of Mines is currently investigating ways to combine the control capabilities of fuzzy logic with the learning capabilities of genetic algorithms. Fuzzy logic allows for the uncertainty inherent in most control problems to be incorporated into conventional expert systems. Although fuzzy logic based expert systems have been used successfully for controlling a number of physical systems, the selection of acceptable fuzzy membership functions has generally been a subjective decision. In this paper, high-performance fuzzy membership functions for a fuzzy logic controller that manipulates a mathematical model simulating the autonomous rendezvous of spacecraft are learned using a genetic algorithm, a search technique based on the mechanics of natural genetics. The membership functions learned by the genetic algorithm provide for a more efficient fuzzy logic controller than membership functions selected by the authors for the rendezvous problem. Thus, genetic algorithms are potentially an effective and structured approach for learning fuzzy membership functions.

INTRODUCTION

In recent years, rule-based systems have become increasingly popular as practical applications of artificial intelligence. These expert systems have performed as well as humans in several problem domains¹⁷, however their lack of flexibility in representing the subjective nature of human decision-making limits their performance in control problems. The uncertainty inherent in human decision-making can be incorporated into expert systems via fuzzy set theory¹⁸. In fuzzy set theory, abstract or subjective concepts can be represented with linguistic variables. Linguistic variables have been used in expert systems in the form of fuzzy logic controllers (FLCs)^{1,12,13}.

FLCs are rule-based systems that use fuzzy linguistic variables to model a human's "rule-of-thumb" approach to problem solving. FLCs have been used in a number of control problems¹⁶. These "fuzzy expert systems" include rules to direct the decision process, and membership functions for categorizing the precise numeric variable values as linguistic variables and vice versa. The rule set is gleaned from a human expert's experience and the membership functions are chosen by the FLC developer to represent the human expert's interpretation of the linguistic variables. A change in the membership functions alters the performance of the controller because it is the membership functions that determine when a given rule is eligible to be put into effect. Thus, the performance of the FLC is restricted by the choice of membership functions (given a set of rules).

Procyk and Mamdani¹⁵ introduced an iterative procedure for altering membership functions to improve the performance of an FLC, but in general, little has been done to develop a method for choosing membership functions that optimize the performance of an FLC. A standard method for determining the membership functions that produce maximum FLC performance is needed, yet selecting such a method poses a substantial problem due to the nonlinearity present in the search.

A search technique that is finding increasing popularity in the field of optimization is the genetic algorithm (GA)⁶. GAs are search algorithms based on the mechanics of natural genetics; they use operations found in natural genetics to guide their trek through the search space. GAs search through large spaces quickly, requiring only objective function value information to guide their search, an inviting characteristic since the majority of commonly used search techniques require derivative information, continuity of the search space, or complete knowledge of the objective function to guide their search. Occasionally these restrictions prove to be inconvenient if not insurmountable. Furthermore, because of the processing leverage associated with GAs, they take a more global view of the search space than many methods encountered in engineering optimization practice⁵. These favorable characteristics of GAs have been theoretically investigated by Holland's⁷ monograph. Empirical investigations by Hollstien⁸ and De Jong⁴ have demonstrated the technique's efficiency in function optimization. De Jong, in particular, establishes the GA as a robust search technique--one that is efficient across a broad spectrum of problems--as compared to several traditional schemes. Subsequent application of GAs to the search problems of pipeline engineering, very large scale integration (VLSI) microchip layout, structural optimization, job shop scheduling, medical image processing, and machine learning⁹ adds considerable evidence to the claim that GAs are broadly based.

The robust nature of GAs make them inviting tools for learning fuzzy membership functions. A GA has in fact been successfully used to learn high-performance fuzzy membership functions employed by a liquid level FLC¹⁰. In this application, an FLC was developed to control a mathematical model of a vessel containing liquid. The simple system is governed by a first order ordinary differential equation. All control decisions performed on the system involved altering one control variable based on the state or condition of two decision variables.

In this paper, an FLC is developed to control a mathematical that simulates the autonomous rendezvous of two spacecraft, one actively performing the rendezvous (the chaser), the other passively orbiting (the target). The rendezvous problem has received attention in the literature¹⁴ and is a challenging control problem. The FLC is capable of performing the rendezvous of the vehicles as predicted by a mathematical model of the rendezvous system. Next, a GA learns more efficient fuzzy membership functions to be used with the FLC rules. Based on the results presented, GAs appear to be effective, versatile, and straightforward enough to learn high-performance membership functions in complex control problems.

THE PHYSICAL SYSTEM

Spacecraft rendezvous operations are of importance to many forthcoming space activities. Of increasing interest are those operations which are conducted autonomously, i.e., without a human pilot operating the control systems of the chaser spacecraft. This paper is concerned with the "terminal" phase of the rendezvous maneuver prior to docking.

Since terminal navigation is accomplished using sensors onboard the chase vehicle to measure relative range and closure rates, a relative coordinate system is a logical choice. When the equations of motion are expressed in a relative coordinate frame, the coordinates are given explicitly as functions of time. In this study, the relative coordinate system (Figure 1) is fixed to the target vehicle with the y-axis directed radially from the earth, the x-axis tangent to the orbit in the direction opposite the target's motion, and the z-axis normal to the plane of the orbit and forming the customary right hand system.

The complete differential equations of relative motion are easily derived⁹. When the distance between the spacecraft is small compared to the distance from the target to the center of the earth, the three equations of relative motion reduce to the Clohessy and Wiltshire (sometimes called Hill) equations⁵:

$$\begin{aligned}\ddot{x} - 2\omega\dot{y} &= F_x/m, \\ \ddot{y} + 2\omega\dot{x} - 3\omega^2y &= F_y/m, \\ \ddot{z} + \omega^2z &= F_z/m,\end{aligned}$$

where ω is the angular velocity of the relative coordinate system origin rotating about the earth; F_x , F_y , and F_z are thrust forces; and m is the mass of the chaser vehicle.

In the present study the equations of motion are numerically integrated using a predictor-corrector linear multistep method, specifically the explicit Adams-Bashforth 3-step method for the predictor and implicit Adams-Moulton 3-step method for the corrector².

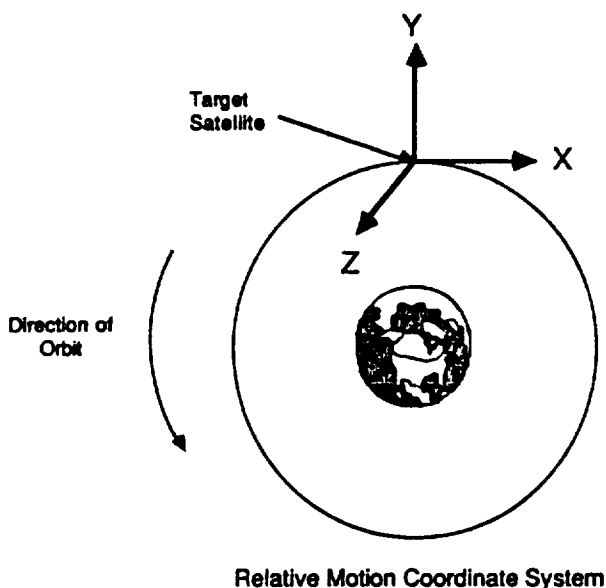


Figure 1.--The coordinate system is fixed to the target spacecraft.

A FUZZY PROCESS CONTROLLER

There are numerous approaches to developing FLCs. Generally, a compositional rule of inference¹²--a mathematical statement describing how the linguistic variables are to be manipulated--is employed to control the problem environment. In this paper, a hands-on, rational approach to the development of FLCs is taken. A step-by-step fuzzy procedure for controlling the rendezvous system is provided. This procedure is written in a generic form so that it may be easily adapted for the development of other FLCs.

The first step in developing the rendezvous FLC is to determine which variables will be important in choosing an effective control action. Six decision variables are readily identified as being important in the rendezvous control system. First, the current position of the chaser spacecraft relative to the target vehicle (identified with three variables x , y , and z) is important because it is this position that the FLC must drive to a setpoint (a close rendezvous with the satellite). Second, the time rate of change of the position (identified with three variables \dot{x} , \dot{y} , and \dot{z}) is important because it describes the relative velocity of the chaser spacecraft and becomes pertinent to the decision as the setpoint is approached.

Once the decision variables have been chosen, the control variables must be identified. In the rendezvous problem there are only three parameters that can be adjusted to alter the position and velocity of the spacecraft: the specific thrust in the three respective directions, $T_x = F_x/m$, $T_y = F_y/m$, and $T_z = F_z/m$.

Once the important decision and control variables have been identified, the linguistic variables that will be used to describe these variables must be defined (fuzzy classes). For the terminal rendezvous system, six fuzzy classes are used to characterize each of the six decision variables:

NEGATIVE-BIG (NB), NEGATIVE-SMALL (NS), NEGATIVE-CLOSE (NC), POSITIVE-CLOSE (PC), POSITIVE-SMALL (PS), and POSITIVE-BIG (PB). Five fuzzy classes are used to characterize the specific thrusts, T_x , T_y , and T_z : NEGATIVE-BIG (NB), NEGATIVE-SMALL (NS), NO CHANGE (N_C), POSITIVE-SMALL (PS), and POSITIVE-BIG (PB). These fuzzy classes were chosen because they are similar to the descriptive terms a human operator might use when attempting to rendezvous the spacecraft.

The choice of fuzzy classes described above allows for the possibility of 36 different conditions that could exist in each of the x, y, and z coordinate direction when the rules are of the form (example for x direction):

$$\text{IF } [x \text{ is } A \text{ AND } \dot{x} \text{ is } B] \Rightarrow \text{THEN } [T_x \text{ is } C]$$

where A , B , and C are fuzzy classes characterizing the respective variables. The individual coordinate directions are considered separately in this rule which would apply when x, y, or z were large since a human operator would likely consider only the two decision variables in a single coordinate direction at a time. Additionally, rules are added to take into account the coupling of x and y, but only when the chaser is near the target. The 108 rules of the above form (36 rules for each of the 3 coordinate directions) are certainly not adequate to control the spacecraft since the equations of motion are coupled (the x and y equations), requiring the FLC to consider the x and y coordinate directions collectively. This coupling effect becomes particularly important when the chaser spacecraft is near the target. Therefore, a set of "coupled rules" is needed. These rules are of the form:

$$\text{IF } [x \text{ is } A \text{ AND } y \text{ is } B \text{ AND } \dot{x} \text{ is } C \text{ AND } \dot{y} \text{ is } D] \Rightarrow \text{THEN } [T_x \text{ is } E \text{ AND } T_y \text{ is } F]$$

where A , B , C , D , E , and F are fuzzy classes characterizing the respective variables. These rules are written for all of the possible combinations of the terms when the four relevant decision variables are in the NEGATIVE-CLOSE or POSITIVE-CLOSE classes. Thus, there are 16 coupled rules which when combined with the 108 original rules provide 124 total rules.

The driving force behind an FLC is the idea that some uncertainty exists in categorizing the values of the control variables; the linguistic variables mean different things to different people. As a result, there must exist some mechanism for interpreting the fuzzy classes. This mechanism is the fuzzy membership function. The fuzzy membership functions used in the rendezvous FLC to characterize relative positions are shown in Figure 2. Similar membership functions were used to characterize relative velocities and thrusts. Fuzzy membership functions allow the precise numeric values of the decision variables to be transformed into a fuzzy class and the fuzzy control actions of the production rules to be transformed into precise, discrete control actions. Actually, fuzzy membership functions are approximations to the confidence with which a precise numeric value is described by a fuzzy class, and fuzzy membership function values (μ) are numeric representations of these confidences. For example, an x of 1300 ft might be viewed as POSITIVE-BIG ($\mu_{PB}(1300) = 0.18$), POSITIVE-SMALL ($\mu_{PS}(1300) = 0.43$), POSITIVE-CLOSE ($\mu_{PC}(1300) = 0.0$), NEGATIVE-CLOSE ($\mu_{NC}(1300) = 0.0$), NEGATIVE-SMALL ($\mu_{NS}(1300) = 0.0$) or NEGATIVE-BIG ($\mu_{NB}(1300) = 0.0$). When a fuzzy membership function has a value of $\mu = 1$, the confidence level of the precise numeric value being accurately described by the fuzzy class is high. On the other extreme, when $\mu = 0$, the confidence level of the precise numeric value being accurately described by the fuzzy class is low. It is important to realize that for each precise decision value, each fuzzy class has a membership function value, i.e., $x = 1300$ ft is POSITIVE-SMALL with a certainty of 0.43, POSITIVE-BIG with a certainty of 0.18, and described by each of the other classes with a certainty of 0.0.

Now that the precise numeric conditions existing in the rendezvous system at any given time can be categorized in a fuzzy class with some certainty, a process for determining a precise action to take on the rendezvous system must be developed. This process involves rules which must provide a fuzzy action for any condition that could possibly exist in the problem environment. Therefore, a human expert provides a fuzzy action for each condition possible in the environment (there are 124 rules in the rendezvous FLC). The formation of the rule set is comparable to the process that must exist

in the development of any expert system, except the rules incorporate linguistic variables that human operators are generally comfortable using. An example of a fuzzy rule used in the rendezvous system follows:

IF [y is NEGATIVE-BIG AND y is NEGATIVE-BIG] => THEN [T_y is POSITIVE-BIG].

This sample rule simply says that if the chaser spacecraft is well below the target vehicle and moving rapidly toward the earth, the thrust in the y direction should be increased as much as possible.

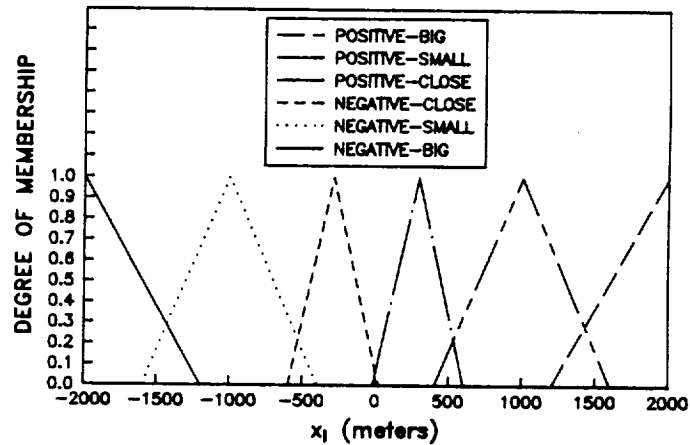


Figure 2.--The fuzzy membership functions shown provide a mechanism for converting a numeric value of position ($x_i = x, y, z$) into a fuzzy class.

At this point a means for converting a precise set of conditions existing in the rendezvous system to a set of fuzzy conditions, and a set of fuzzy rules prescribing a fuzzy action associated with a particular set of fuzzy conditions have been developed. There still remains the task of converting the 124 fuzzy actions provided by the fuzzy rules into a single, precise action to be taken on the chaser spacecraft system. Larkin¹¹ found that a procedure known as the center of area (COA) scheme is an efficient method for determining this precise action. In the COA method, a single action is defined by weighting the 124 individual actions described by the rules in proportion to the confidence that exists in each rule. The rules with high degrees of membership play proportionately higher roles in determining the action to be taken on the system.

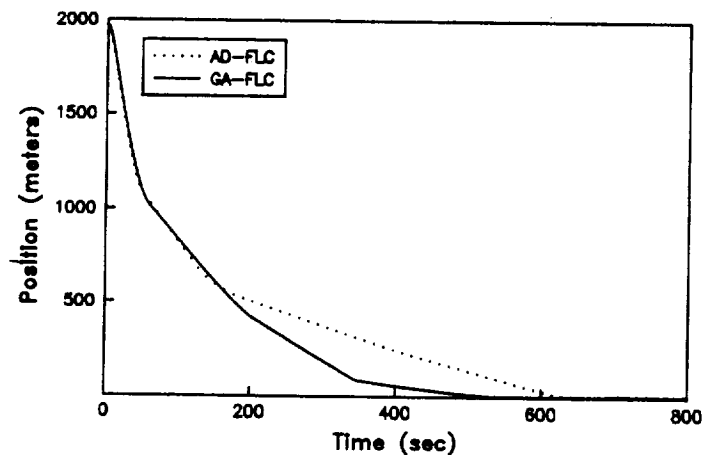


Figure 3.--The author-developed (AD) FLC is able to successfully rendezvous the spacecraft in approximately 300 seconds.

As a demonstration of the effectiveness of this fuzzy approach to control, consider a computer program that implements an FLC for manipulating a mathematical model of the rendezvous system. Figure 3 shows the position of the chaser relative to the target as a function of time for one particular initial condition for the problem environment. The FLC uses only the set of 124 fuzzy rules to govern its selection of actions. The FLC is able to rendezvous the spacecraft in approximately 600 seconds. In the remainder of this paper, a technique for improving the selection of membership functions (a GA) is introduced and applied to the rendezvous FLC.

THE MECHANICS OF A SIMPLE GENETIC ALGORITHM

GAs are powerful search algorithms based on the mechanics of natural genetics. They ensure the proliferation of quality solutions while investigating new solutions via a systematic information exchange that utilizes probabilistic decisions. It is this combination which allows GAs to exploit historical information to locate new points in the search space with expected improved performance.

GAs are unlike many conventional search algorithms in the following ways:

- 1) GAs consider many points in the search space simultaneously, not just 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 differences establish inviting characteristics for a search technique. Namely, these differences preclude the requirement of derivative information and continuity of the search space. For this reason, GAs avoid convergence to local optima.

A simple GA that has given good results in a variety of engineering problems is composed of three operators: (1) reproduction, (2) crossover; and (3) mutation. These operators are implemented by performing the basic tasks of copying strings, exchanging portions of strings, and generating random numbers; tasks that are easily performed on a computer. Before looking at the operators, consider the overall processing of a GA during a single generation. The GA begins by randomly generating a population of N bit strings each of length ℓ . Each string represents one possible combination of the parameter set; one possible solution to the problem (in this case, one particular set of fuzzy membership functions). Although the coding of parameter sets to bit strings may at first seem to be a problem, many imaginative codings exist for representing large parameter sets. Each of the N strings is decoded so that the character strings yield the actual parameters. The parameters are sent to a mathematical model of the rendezvous process, evaluated with some objective function (i.e., told how good an FLC the parameters produce), and assigned a fitness value which is simply a measure of relative worth (a reward based on the quality of the solution). This fitness is then used when employing the three operators that produce a new population of strings (a new generation). Hopefully, this new generation will contain more efficient membership functions. The new strings are again decoded, evaluated, and transformed using the basic operators. The process continues until convergence is achieved or a suitable solution is found.

Reproduction is simply a process by which strings with large fitness values, good solutions to the problem at hand, receive correspondingly large numbers of copies in the new population. In this study use is made of tournament selection. In tournament selection, pairs of strings compete with each other on a head-to-head basis for the right to be reproduced in the next generation. The participants in these competitions are selected in a probabilistic fashion based on the relative fitnesses of the strings. Once the strings are reproduced, or copied for possible use in the next generation, they are placed in a mating pool where they await the action of the other two operators.

The systematic information exchange utilizing probabilistic decisions is implemented by the second operator, crossover. Crossover provides a mechanism for strings to mix and match their desirable qualities through a random process. After reproduction, simple crossover proceeds in three steps. First, two newly reproduced strings are selected from the mating pool produced by

reproduction. Second, a position along the two strings is selected uniformly at random. This is illustrated below where two binary coded strings A and B of length six are shown aligned for crossover:

$$\begin{array}{r} \overline{\hspace{1.5cm}} \\ A = 110 \quad | \quad 101 \\ B = 001 \quad | \quad 011 \end{array}$$

Notice how crossing site 3 has been selected in this particular example through random choice, although any of the other four positions were just as likely to have been selected. The third step is to exchange all characters following the crossing site. The two new strings following this crossing are shown below as A' and B':

$$\begin{array}{r} A' = 110011 \\ B' = 001101 \end{array}$$

String A' is made up of the first part of string A and the tail of string B. Likewise, string B' is made up of the first part of string B and the tail of string A. Although crossover has a random element, it should not be thought of as a random walk through the search space. When combined with reproduction, it is an effective means of exchanging information and combining portions of high quality solutions.

Reproduction and crossover give GAs the majority of their search power. The third operator, mutation, enhances the ability of the GA to find near optimal solutions. Mutation is the occasional alteration of a value at a particular string position. It is an insurance policy against the permanent loss of any simple bit. A generation may be created that is void of a particular character at a given string position. For example, a generation may exist that does not have a one in the third string position when, due to the chosen coding, a one at the third position may be critical to obtaining a quality solution. Under these conditions, neither reproduction nor crossover will ever produce a one in this third position in subsequent generations. Mutation, however, causes a zero in the third position to be changed to a one occasionally. Thus, the critical piece of information can be reinstated into the population. Although mutation can serve a vital role in a GA, it should be noted that it occurs with a small probability (on the order of one mutation per thousand string positions), and is secondary to reproduction and crossover.

This has been a brief overview of a simple three-operator GA. For a more complete description of GAs including discussions of coding schemes, high-order operators, and fitness assignment, reference should be made to Goldberg⁶.

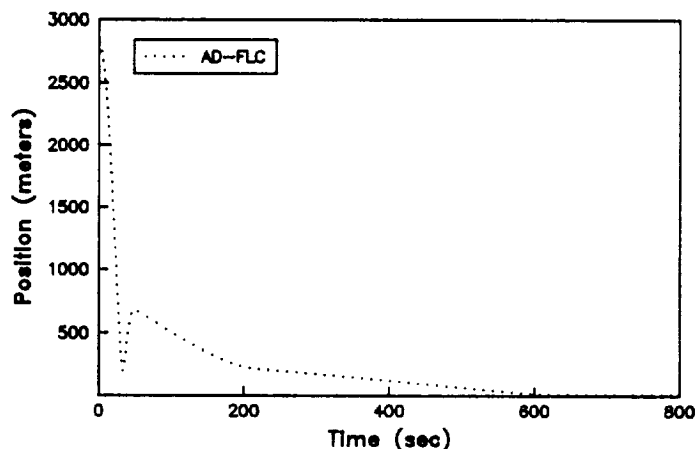
GENETIC ALGORITHM LEARNS FUZZY MEMBERSHIP FUNCTIONS

In this section, a three-operator GA learns membership functions that produce a rendezvous FLC that is more efficient than the author-developed FLC. The GA is essentially assigned the task of learning the proper definition of the linguistic variables as required for optimal performance when used with the given rule set.

An objective function was developed that tended to force the GA to locate membership functions that drove the relative position of the spacecraft to zero and held it there. The objective function tracked the vehicles for 8000 seconds. For a given set of membership functions, an error (where error is the sum of the absolute values of the relative distances between the chaser and the target taken over time) was calculated for a particular initial condition with the intent of using a GA to minimize this error term. So that a general purpose set of membership functions would be developed, four different initial conditions were considered in the evaluation of each bit string. The four initial conditions were chosen to ensure that the FLC could rendezvous the spacecraft as effectively when the chaser was approaching the target from a higher orbit as it could when it was approaching from a lower orbit.

The GA learned membership functions that provided for better control than those defined by the authors in the previous section. Figure 4 compares the GA-FLC to the author-developed FLC for one of the four initial conditions considered. In each initial condition case, the GA-FLC completed the spacecraft rendezvous more favorably than the author-developed FLC.

The desirable characteristics of the rendezvous must be conveyed to the GA through the objective functions. In this study, the objective of the GA was simply to reduce the total distance between the two vehicles summed over a period of time. The GA accomplished this goal. The FLC can be forced to exhibit other properties by altering the objective function. For example, if it is important to prevent the spacecraft from overshooting the target in the x-coordinate, the GA can be dissuaded from locating such solutions by incorporating a penalty into the objective function (see Goldberg⁶ for information on penalty methods in objective function formulation).



SUMMARY

In this paper, a GA was used to improve the performance of an FLC. Initially, an approach to FLC development was outlined. The approach was developed to be straightforward and intentionally avoided an abundance of fuzzy mathematics. Next, this approach was used to develop a fuzzy system for controlling the rendezvous of a spacecraft. The FLC was able to maintain control over the spacecraft (as simulated by a computer) by relying exclusively on fuzzy rules to determine its next action.

Altering the membership functions used in an FLC affects the performance of the controller, and the selection of appropriate membership functions can be cast in the light of a search problem. A simple three-operator GA was used to learn high-performance membership functions for the rendezvous FLC. The GA-FLC outperformed the author-developed FLC on four specific initial conditions chosen to represent a cross-section of potential initial conditions.

REFERENCES

1. Assilian, S., 1974, "Artificial Intelligence in the Control of Real Dynamic Systems," Ph.D. Thesis, Queen Mary College, London, 215 pp.
2. Burden, R. L., Faires, J. D., and Reynolds, A. C., 1978, Numerical Analysis, Prindle, Weber, & Schmidt, Boston, MA, 579 pp.

3. Clohessy, W. H., and Wiltshire, R. S., 1960, "Terminal Guidance System for Satellite Rendezvous," Journal of the Aerospace Sciences, Sept., pp. 653-674.
4. De Jong, K. A., 1975, "Analysis of the Behavior of a Class of Genetic Adaptive Systems," Ph. D. Thesis, University of Michigan, Ann Arbor, 256 pp.
5. Goldberg, D. E., 1983, "Computer-Aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning," Dissertation Abstracts International, Vol. 44, No. 10, p. 3174B.
6. Goldberg, D. E., 1989, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 412 pp.
7. Holland, J. H., 1975, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, 183 pp.
8. Hollstien, R. B., 1971, "Artificial Genetic Adaptation in Computer Control Systems," Ph. D. Thesis, University of Michigan, Ann Arbor, 297 pp.
9. Kaplan, M. H., 1976, Modern Spacecraft Dynamics & Control, John Wiley & Sons, New York, 415 pp.
10. Karr, C. L., Meredith, D. L., and Stanley, D. A., 1990, "Fuzzy Process Control with a Genetic Algorithm", Proceedings, Control '90 Mineral/Metallurgical Processing-Second International Symposium, Salt Lake City, UT, February.
11. Larkin, L. I., 1985, "A Fuzzy Logic Controller for Aircraft Flight Control," Industrial Applications of Fuzzy Control, Sugeno, M., Ed., North-Holland, Amsterdam, pp. 87-104.
12. Mamdani, E. H., 1974, "Application of Fuzzy Algorithms for Control of Simple Dynamic Plant," Proceedings, IEEE, Vol. 121, No. 12, pp. 1585-1588.
13. Mamdani, E. H., and Assilian, S., 1975, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller," International Journal of Man-Machine Studies, Vol. 7, pp. 1-13.
14. Natenbruk, P., and Ragnitt, D., 1983, "Control Aspects as Elaborated in Space Rendezvous Simulations," Conference Proceedings on Guidance and Control Techniques for Advanced Space Vehicles, Florence, Italy.
15. Procyk, T. J., and Mamdani, E. H., 1978, "A Linguistic Self-Organising Process Controller," Automatica, Vol. 15, pp. 15-30.
16. Sugeno, M., Ed., 1985, Industrial Applications of Fuzzy Control, North-Holland, Amsterdam, 269 pp.
17. Waterman, D. A., 1970, A Guide to Expert Systems, Addison-Wesley, Reading, MA, 398 pp.
18. Zadeh, L. A., 1973, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-3, No. 1, pp. 28-44.

