CHAPTER 6

OPTIMIZATION OF FUZZY LOGIC CONTROLLER PARAMETER FOR ATTITUDE CONTROL AND COUPLED CONTROL

6.1 INTRODUCTION

Fuzzy expert systems are gaining acceptance in both the process control and the computer modeling communities. However, fuzzy systems often provide only approximate solutions because they employ inexact reasoning. This has been well explained in the previous chapter in which a fuzzy controller is designed for the satellite attitude controller. As explained in the conclusion part of the previous chapter, the response of the fuzzy controller can be improved by properly optimizing the parameters of the fuzzy controller. Genetic algorithms can be used to produce very precise fuzzy systems (Karr 1996). While designing and developing a spacecraft control system, the controller must at all time respect constraints like fuel consumption and maneuvering time. Fuzzy systems have two parameters, which can be optimized: a rule database and the fuzzy sets. Two approaches can be taken to obtain an optimum solution: either using the heuristic method, in which the control engineer obtains the best system to satisfy the criteria, or by finding an analytical solution to the problem. On most occasions the design of optimum system requires the knowledge of an expert operator. The optimization of the fuzzy controller parameters is one of the most widely investigated subjects in the research into fuzzy expert systems. Many recent
publications have demonstrated the possibility of optimizing those parameters automatically (analytically) by means of Genetic Algorithms (GAs) (Ray-Guang and Chung 1996), reinforcement learning, and Artificial Neural Networks (ANN). Genetic algorithms were introduced 30 years ago, but only recently they have been recognized as a promising technique to optimize these types of functions.

Recently numerous papers and applications combining fuzzy concepts and GAs have appeared and there is increasing concern about the integration of the two topics. In particular, a great number of publications explore the use of GAs for designing fuzzy systems. These approaches receive the general name of Genetic Fuzzy Systems (GFSs) (Francisco et al 2001). This is applied in numerous spacecraft applications. For example since spacecraft attitude control is approached as a nonlinear control problem, the robustness quality of the Genetic Algorithm (GA)-optimized Fuzzy Logic (FL) control is demonstrated using a nonlinear model of the Space Station Freedom (Satyadas and Krishna Kumar 1994).

In the real world, the process of natural selection controls organism evolution. Organisms most suited for their environment tend to live long enough to reproduce new organisms. The genetic algorithm emulates the evolutionary process in an artificial world, by populating pseudo-organisms and giving those organisms a goal to achieve. A genetic algorithm emulates the behaviors and characteristics of the successful organisms and discovers solution to a given problem (Goldberg 1989). In an artificial world, a pseudo-organism represents a group of parameters to be optimized in a specific problem.

The fundamental underlying mechanism consists of three operations

- Evaluation of individual fitness
• Formation of gene pool (intermediate population) through selection mechanism and
• Recombination through crossover and mutation.

In genetic algorithm, each organism is generally referred to as an individual and represented by a binary string (i.e. a string consisting of a ‘0’ and ‘1’). Each individual represents a potential solution to the problem in hand.

The genetic algorithm is a model of machine learning, which derives its behavior from a metaphor of the process of evolution in nature. This is done by the creation of a population of individuals represented by chromosomes. In essence a set of character strings that are analogous to the base-4 chromosomes present in DNA. The individuals in the population then go through a process of evaluation. Evolution (in nature or any where else) is not a purposive or direct process. Indeed, the process of nature seems to boil down to different individuals competing for resources in the environment. Some are better than others. Those that are better are more likely to survive and propagate their genetic material.

At the molecular level a pair of chromosomes burns up into one another, exchange chunks of generic information and drift apart. This is the recombination operation, which GAs generally refers to as crossover because of the way that generic material crosses over from one chromosome to another. The crossover operation happens in an environment where the selection of who gets to mate is a function of the fitness of the individual, i.e. how good the individual is at competing in its environment.

Some genetic algorithm uses a simple function of the fitness measure to select individuals (probabilistically) to undergo genetic operations such as crossover. This is fitness-proportionate selection. Other
implementations use a model in which certain randomly selected individuals in a subgroup compete and the fittest is selected. This is called tournament selection. The two processes that contribute most to evolution are crossover and fitness based selection/reproduction.

Mutation also plays a role in this process, although the importance of its role continues to be a matter of debate (some refer to it as a background operator, while others view it as playing the dominant role in the evolutionary process).

Genetic algorithms are used for a number of different application areas. In practice, this genetic model of computation can be implemented by having arrays of bits or characters to represent the chromosomes. Simple bit manipulation operations allow the implementation of crossover, mutation and other operations.

Although a substantial amount of research is performed on variable length strings and other structures, the majority of the work with genetic algorithm is focused on fixed length character strings.

This chapter deals with the implementation of genetic fuzzy systems for satellite attitude control through the optimization of fuzzy logic membership function width for the various attitude control modes explained in the previous chapter. The selection of the GA parameters and problem representation has a crucial effect on the GA performance, optimization speed and reliability. In literature, these have been intensively studied both theoretically (Alander 1992, Alander 2002, Gao 2003, Rees and Koehler 1999a) and empirically (Alander 1992, Alander 2002, Rees and Koehler 1999a). Finally the chapter ends with the coupling of attitude controller and the orbit controller for the distance keeping. The inference of the coupling has also been discussed.
6.2 ADVANTAGES OF GA

Genetic algorithm is different from normal search methods encountered in engineering optimization in the following ways:

- GAs work with a coding of the parameter set and not the parameter themselves.
- GAs searches from a population of points and not a single point.
- GAs use probabilistic transition rules and not deterministic transition rules.

6.3 GENERAL PROCEDURE OF GA OPTIMIZED FUZZY CONTROLLER

The procedure is in steps as explained below:

i) The fitness function design is very important to evaluate each individual in one generation. For these problems, minimization of settling time is taken as the performance criteria and tuning is done. So, \( E = \) Desired output - Actual output is taken as error, \( E \). Therefore, a smaller \( E \) represents a higher fitness (GA maximizes performance). The \( E \) is converted to a fitness value of a GA by using, \( \text{Fitness} = 1/|E| \) (Jinwoo et al 1994).

ii) In genetic operations, the entire individuals are expressed as binary strings, not the parameters themselves. In the coding method, the scaling factors generated randomly, are first coded into binary strings. While calculating the fitness values for each individual, these binary strings are converted into corresponding values in the parameter space by using a
decoding procedure. For tuning the width of the membership function random binary initialization technique is used and a mapping equation given in equation (3.72) is used to map the binary string to membership function width.

3. Once the fitness values of all individuals in the population are evaluated, the fittest individuals are selected for survival and reproduction. The selection process is based on proportional selection method, i.e. an individual with a high fitness value has a high probability of being selected, as given by equations (3.73) and (3.74). The selected individuals are randomly mated to perform genetic operations.

4. Two mating parents exchange information through simple crossover and are replaced with the new individuals. For a simple crossover, the cut off position is randomly determined.

5. After completing the simple crossover, mutation operation is performed.

During the process of iteration, the genetic algorithm maintains a constant population of individuals. Each individual will undergo evaluation and selection. The surviving individuals will perform crossover and mutation operations. The iteration process is repeated until the termination conditions are satisfied.

6.4 IMPLEMENTATION

Based on the procedure explained in the section 6.3, the fuzzy controller parameter namely the width of the membership function is
optimized by GA. The implementation procedure is given in the form of flowchart as in Figure 6.1.

**Figure 6.1** GA based parameter optimization

The optimization of the fuzzy logic controller by genetic algorithm is depicted in the form of a block diagram in Figure 6.2.
6.5 SIMULATION AND RESULTS

6.5.1 GA Specifications

The specifications adopted for optimization using GA are given below:

Initial population : Random binary string

No. of Bits : 4 bits for each parameter

Crossover : Simple crossover

Mutation : Shift mutation

Performance Index : Settling time
The population size of the generation is taken to be 10 and also the optimization is tried for a population size of 20 for the attitude controller modes like detumbling with initial spin up, spin rate controller (spin up and spin down) and spin axis orientation controller. The results are analyzed and compared with the other control techniques.

6.5.2 Satellite Specifications

Satellite Configuration is already given in the previous chapter and the same is followed in the optimization, which is again repeated below.

\[
\begin{align*}
I_x &= 1.255427 \text{ kg-m}^2 \\
I_y &= 1.338694 \text{ kg-m}^2 \\
I_z &= 1.442010 \text{ kg-m}^2
\end{align*}
\]

6.5.3 Initial Rates

Since the beginning of each simulation, an initial angular rate of 6 degree per second for each axis has been assumed and these rates are decided according to the specification given by the launch vehicle team.

\[
\begin{align*}
\omega_x &= 6 \text{ deg/sec} \\
\omega_y &= 6 \text{ deg/sec} \\
\omega_z &= 6 \text{ deg/sec}
\end{align*}
\]

6.5.4 Detumbling with Initial Spin Up Response

The response of the fuzzy controller with its width before and after optimization by genetic algorithm for population size 10 and 20 are given in Figures 5.7, 6.3 and 6.4, for the detumbling with initial spin up mode.
Figure 6.3  Detumbling with initial spin up response after optimization for FLC with population size of 10

Figure 6.4  Detumbling with initial spin up response after optimization for FLC with population size of 20
6.5.5 Spin Rate Control

The responses of the fuzzy controller with its width before and after optimization by genetic algorithm for the spin rate controller for both spin up and spin down are given in Figures 6.5 through 6.8. The optimization is done for population size 10 and 20.

![Figure 6.5 Spin rate control up response after optimization for FLC with population size of 10]
Figure 6.6  Spin rate control up response after optimization for FLC with population size of 20

Figure 6.7  Spin rate control up response after optimization for FLC with population size of 10
Figure 6.8  Spin rate control up response after optimization for FLC with population size of 20

6.5.6  Spin Axis Orientation Control

The response of the fuzzy controller with its width before and after optimization by genetic algorithm for the spin axis orientation controller for orbit normal are given in Figures 6.9 and 6.10. The simulation is done for population size 10 and 20.
Figure 6.9  Spin axis orientation controller response after optimization for FLC – orbit normal with population size 10

Figure 6.10  Spin axis orientation controller response after optimization for FLC – orbit normal with population size 20
6.6 COUPLED CONTROLLER

The coupled controller is a coupled attitude and orbit controller developed using the GA optimized controllers developed for attitude and orbit control discussed in previous chapters. The coupled system controls the orbit and hence maintains the formation and the attitude controller controls the attitude of the individual satellite. This part describes the overall coupled system for a spacecraft using magnetic torquer for attitude control and maintains the distance between the satellites by absolute station keeping.

The simple overall block schematic for a coupled controller is given in Figure 6.1. The figure shows how the station keeping and attitude control are done. After the launch of the satellite, the satellite attains the required orbit. The attitude of the satellite is maintained through detumbling and then by spinning the satellite to the required speed. Later the orientation of the spin axis is also done. The orbit of the satellite is checked for deviation in distance. At the end of the day or if the deviation in semi-major axis exceeds 4m, the orbital controller controls and helps in maintaining the orbit. The attitude of the satellite is checked for deviation. The attitude controller then corrects any change in the attitude of the satellite. If there is no change then system proceeds with the orbit estimation.

The coupled system is implemented using the orbit estimation, orbit control, attitude estimation and control methods discussed in previous chapters. The Gaussian planetary equation is used for finding the change in the orbit of the satellite due to perturbation and GA based orbit controller is used. The attitude of the satellite is found by the magnetic field prevailing at the orbit. The magnetic field changes, as there is deviation in the orbital elements due to perturbation in atmospheric drag.
The attitude is based on the magnetic field existing in the orbit at the location of the satellite and since the deviation in the orbit is very less and is corrected every day at frequent intervals of time there is no significant deviation in the attitude of the satellite. Hence it is evident that, the orbital deviation does not affect the attitude of the satellite. This is due to the appropriate distance keeping method followed.

6.6.1 Decentralized Control

A spacecraft formation is a distributed system. A distributed system is a large system consisting of multiple smaller subsystems. The attitude control systems of the individual spacecraft act as the local control agents. The control decisions of the local control agents must be coordinated to ensure the stability and convergence of the global system. Coordinated controllers are generally categorized into centralized and decentralized types. The distinction is based on where the control decisions are made. Centralized control is a type of coordinated control where a single control agent, called the global control agent, determines the control actions for the distributed
system. Figure 6.12 shows a block diagram of a distributed system using centralized control.

![Diagram](image)

**Figure 6.12 Centralized control**

The global control agent commands are represented by the unidirectional arrows directed toward the local control agents. In decentralized control, control decisions are relegated to the local control agents. The local control agents use local observations and any information communicated by the other control agents to determine control actions. A block diagram of a distributed system using decentralized control is presented in Figure 6.13.
The bi-directional arrows represent the two-way communication of information between the local control agents. The two primary benefits of decentralized control over centralized control are fault-tolerance and simpler control laws. Failure of a single local control agent in a decentralized controlled system does not lead to the destabilization of the entire system. The failure is confined to the region of the failed local control agent resulting in a graceful degradation of system performance. Decentralized control results in relatively simple control laws, because the design of the global controller can be decomposed into smaller control agents. The local control agents are designed so that they perform their local control tasks, and coordinate with one another to control the global system. The coordination is implemented by means of communication between the local control agents. Centralized controllers require greater information and information processing than what is required by the local control agents of the equivalent decentralized
controller. The primary drawback of decentralized controllers is that they are difficult to analyze analytically.

The decentralized control is utilized in the spacecraft attitude control, whereby the attitude of one spacecraft is dependent on the other. For simulation 2 satellites were considered and the spin-axis orientation of one satellite was changed with respect to the other. The requirement for change was communicated by satellite 1 to satellite 2. The satellite 2 changed its orientation depending upon the requirement given by satellite 1 and the change has also been communicated, which forms a two-way communication.

6.7 CONCLUSION

The optimization of fuzzy logic parameters by genetic algorithm as applied to the attitude controller for the micro-satellite is explained in this chapter. The settling time taken for detumbling with initial spin-up for population size of 10 and 20 are 24,802 s and 18,752 s respectively. For SAOC the settling time for population size of 10 and 20 are 17,390 s and 14,475 s respectively.

In the present work the design of GFS was implemented using Matlab and the responses were plotted for the various modes. The optimization was done by varying the population sizes and the results shows that as the population size is increased the results provided are better which means that optimization is improved as the population size is increased. The results prove that the selection of the GA parameters and problem representation has a crucial effect on the GA performance. It also has impact on optimization speed and reliability. The time taken to attain the attitudes are less than the conventional B-Dot control for all attitude control modes when the population size is increased from 10 to 20. The results prove the effect of GA parameter variation on optimization.
The chapter has also finally integrated the orbital control method hence the distance keeping given in chapter 3 and the GA optimized fuzzy attitude controller and gives a summarized picture of the thesis. From the simulation it has been inferred that for a satellite using magnetic based attitude control, if the orbit controller and hence the distance keeping is properly taken care then there is no change in the attitude of the satellite due to orbital deviation. Decentralized control of satellites in formation is also explained along with its advantages.