DESIGN AND IMPLEMENTATION OF A FUZZY LOGIC-BASED VOLTAGE CONTROLLER FOR VOLTAGE REGULATION OF A SYNCHRONOUS GENERATOR

Brock J. LaMeres
Montana State University
Bozeman, MT 59717
lameres@montana.edu

ABSTRACT

In this paper the design and implementation of a fuzzy logic-based controller is described for regulating the output voltage of a synchronous generator. An automated fuzzy logic-based control strategy is presented for controlling the armature voltage of a synchronous generator by varying the field current in real time. The fuzzy logic controller will take into account the voltage difference between the immediate output voltage and the rated voltage of the generator. It will also take into account the rate of change of voltage error. The controller will make an intelligent decision on the amount of field current that should be applied to the generator in order to keep the output voltage at its rated value. This control algorithm will be implemented on a Digital Signal Processor.

I. INTRODUCTION

Fuzzy logic controllers are rapidly becoming a viable alternative for classical controllers [2]. The reason for this is that a fuzzy controller can closely imitate human control processes. Fuzzy logic technology enables the use of engineering experience and experimental results in designing an embedded system. In many applications, this circumvents the use of rigorous mathematical modeling to derive a control solution.

Another advantage of using a fuzzy logic control strategy is that it allows a model-free estimation of the system. In other words, the designer does not need to state how the outputs depend mathematically upon the inputs. A fuzzy controller can be developed by encoding the structured knowledge of the system. This allows faster control algorithms to be developed in less time and at less cost. With the advance of microprocessors and Digital Signal Processor's (DSP's), fuzzy logic control devices are becoming a more attractive solution for real time control situations [1].

This paper describes the development of a fuzzy logic-based control strategy that is implemented in real time on a Digital Signal Processor. The theory of fuzzy logic, fuzzification, inference and the defuzzification method are discussed in the development of the controller. Also, the software used to run the DSP and the algorithms, which implement the controller, are presented. Finally, the experimental setup that is used to construct the closed loop control system and the experimental results are explained.

II. SYSTEM CONFIGURATION

Figure 1 shows the basic closed loop control system.

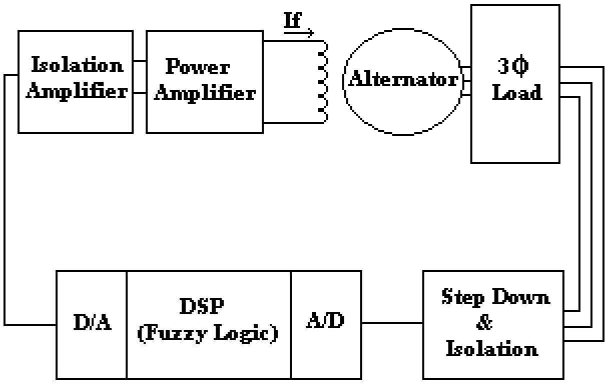
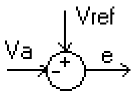


Figure 1. Closed Loop Control System

A three-phase, 5kVA laboratory-size alternator, running at rated speed and rated armature voltage experiences a varying 3-phase load. As the armature voltage fluctuates with the changing load, the excitation current into the field of the generator must be altered to bring the armature voltage back to its rated value. This is accomplished with an intelligent control strategy programmed into a *Texas Instruments TMS320C5x* DSP. The line to line voltage of the alternator will be fed into a 'step down and isolation' (20/1) stage which will produce a signal that is suitable for the DSP (i.e., 0-10 volts). This signal will first be converted from analog to digital using an onboard 12-bit signed A/D converter. The DSP will compare the immediate armature voltage (V_a) to its rated voltage (V_{ref}) to find the voltage error (e) that exists.



It will then find the rate of change of voltage by comparing the previous voltage value to the immediate armature voltage. These two inputs will be evaluated using a fuzzy logic-based control algorithm and an output signal will be produced. The digital output signal will then be fed into a 12 bit D/A converter giving an analog signal out ranging from 0 – 10 volts. This signal will be proportional to the field current that is to be applied to the field of the alternator (i.e., 5 volts => 1.5 Amps). This is then fed into an isolation amplifier. This amplifier is an opto-coupler that will provide additional isolation and safety. The signal is then fed to a power amplifier that will produce the necessary field current in order to regulate the armature voltage (i.e., 0 – 3 amps).

III. THE FUZZY LOGIC CONTROLLER

Fuzzy logic control is a non-mathematical decision algorithm that is based on an operator's experience. This type of control strategy is suited well for non-linear systems such as the synchronous generator, which exhibits non-linearity between the field current in and the armature voltage out [3]. Figure 2 shows the no-load saturation curve of the 5kVA, laboratory size synchronous generator to be controlled. As shown, the curve behaves in a non-linear fashion as it enters its saturation region at $If = 3.3$ amps. The fuzzy logic controller can easily be programmed to handle this region.

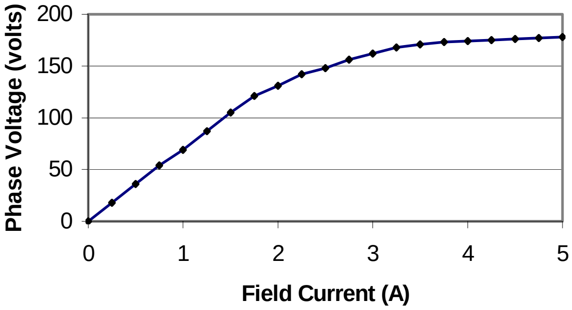


Figure 2. No Load Saturation Curve of the Synchronous Generator

The first input to the fuzzy logic controller is the immediate armature *voltage.* This voltage, when compared to the rated voltage of the generator, will tell the controller if the voltage needs to be raised/lowered or if the machine is operating at its rated voltage. The input voltage is the phase voltage of the alternator stepped down with a ratio of 20 to 1. This 'step down' stage will produce a signal that is suitable for the DSP (< 10 V_p). For the generator in this experiment, the rated voltage of 120 V_{RMS} is stepped down to a voltage of +/- 8.5 V_p.

The second input to the controller is the *rate of change of voltage*. This input describes how fast the output voltage is changing. This is an important factor in a real time control strategy for increasing the time response of the system. For example, the *voltage* may be 2 volts lower than its rated voltage. In one case the output voltage is static and a slight increase in field current will bring the generator back to its rated voltage. In another case, the output voltage may still be 2 volts lower but decreasing rapidly. For this situation a larger field current must be applied (at least temporarily) in order to bring the generator back into control.

The output of the fuzzy logic controller is the *field current*. A proportional voltage from the DSP will be output into an isolation amplifier and then to a power amplifier that will provide 0-3 amps (I_f = 1.5 A_{RATED}) to the field of the alternator. This signal will increase or decrease the current to the field of the synchronous generator in order to regulate the output voltage. This can be seen if figure 1. Figure 3 shows the variables of the fuzzy logic controller.

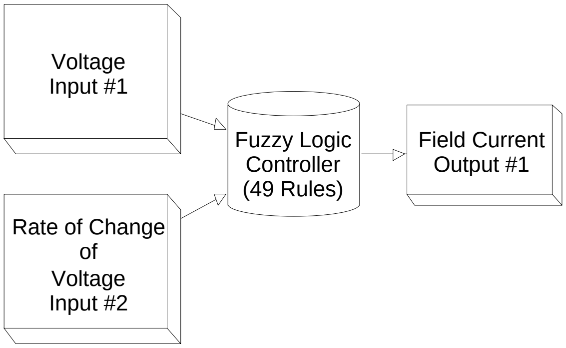


Figure 3. Variables of the Fuzzy Logic Controller

Figure 4 shows the structure of the fuzzy logic control algorithm.

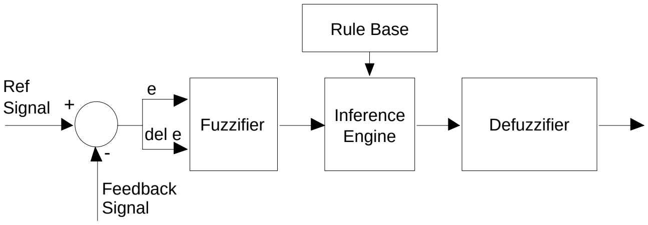


Figure 4. Structure of Fuzzy Logic Controller

FUZZIFIER

Fuzzy logic uses linguistic variables instead of numerical variables. In a closed loop control system, the *error (e)* between the reference voltage and the output voltage and the *rate of change of error (del e)* can be labeled as zero (ZE), positive small (PS), negative small (NS), etc. In the real world, measured quantities are real numbers (crisp). The process of converting a numerical variable (real number) into a linguistic label (fuzzy number) is called fuzzification. Figure 5 shows the *membership functions* that are used to fuzzify the inputs.

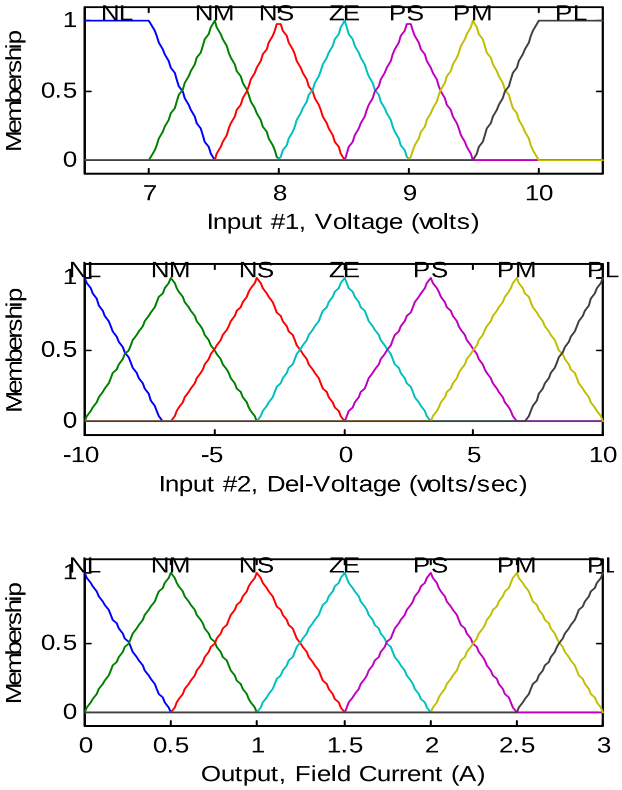


Figure 5. Membership Functions of
The Fuzzy Logic Controller

The inputs are mapped into these membership functions and a degree of membership is found for how much the input belongs to that particular linguistic label. The membership can take on a value from zero to unity for each of the linguistic labels. The waveforms are evenly distributed about the range of operation of the variables. For each of the input and output variables, the following seven linguistic labels are assigned to the membership functions:

NL = Negative Large
NM = Negative Medium
NS = Negative Small
ZE = Zero
PS = Positive Small
PM = Positive Medium
PL = Positive Large

Once the membership is found for each of the linguistic labels, an intelligent decision can be made unto what the output should be. This decision process is called *inference*.

INFERENCE

In conventional controllers, there are *control laws*, which are combinations of numerical values that govern the reaction of the controller. In fuzzy logic control, the equivalent term is *rules*. Rules are linguistic in nature and allow the operator to develop a control decision in a more familiar human environment [4]. A typical rule can be written as follows:

If the "voltage" is negative large (NL), AND the "rate of change of voltage error" is negative large (NL), then the "field current" is positive large (PL).

In this design, a *minimum correlation inference* technique was used. This means that the logic operation of AND will return the minimum of all inputs. For the linguistic rule stated earlier, the output, *field current - PL*, would receive a membership that was equal to the minimum of the two inputs, *voltage – NL* and *rate of change of voltage – NL*. For example:

Membership (V – NL) = .8
Membership (del V – NL) = .2

(.8) AND (.2) = .2

Membership (F – PL) = .2

The rules of a fuzzy logic controller give the controller its *intelligence*, assuming the rules are developed by a person who has a experience with the system to be controlled. A programmer with more experience with the system will create a better controller.

In the case of the fuzzy logic synchronous generator controller, the desired effect is to keep the output voltage of the generator at its rated voltage under varying loads. From this desired goal, rules are made for every combination of *voltage* and *rate of change of voltage* on what the *field current* should be in order to stabilize the generator. It is convenient when dealing with a large number of combinations of inputs, to put the rules in the form of a rule table. Figure 6 shows the rule table for controlling the synchronous generator output voltage where *Del Volt* refers to the rate of change of output voltage.

		Voltage						
		NL	**NM**	**NS**	**ZE**	**PS**	**PM**	**PL**
Del Volt	**NL**	PL	PL	PL	PL	PM	PS	ZE
	NM	PL	PL	PM	PM	PS	ZE	NS
	NS	PL	PM	PS	PS	NS	NM	NL
	ZE	PL	PM	PS	ZE	NS	NM	NL
	PS	PL	PM	PS	NS	NS	NM	NL
	PM	PM	ZE	NS	NM	NM	NL	NL
	PL	ZE	NS	NM	NL	NL	NL	NL

Figure 6. Fuzzy Logic Rule Table

After the rules are evaluated, each output membership function will contain a corresponding membership. From these memberships, a numerical (crisp) value must be produced. This process is called *defuzzification*.

DEFUZZIFICATION

Defuzzification plays a great role in a fuzzy logic-based control system. It is the process in which the fuzzy quantities defined over the output membership functions are mapped into a non-fuzzy (crisp) number. It is impossible to convert a fuzzy set into a numeric value without loosing some information. Many different methods exist to accomplish defuzzification. Naturally there are trade-offs to each method.

The method that was chosen for implementation in this project was the *Weighted Average* method. This method is defined as the sum of the products of each membership function's center and height, divided by the sum of all the membership functions' heights [4].

$$Z_O = \frac{Z_1 \cdot h_1 + Z_2 \cdot h_2 + \cdots + Z_n \cdot h_n}{h_1 + h_2 + \cdots + h_n}$$

Figure 7 & 8 shows the graphical representation of this method.

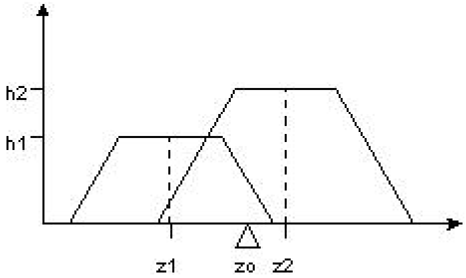


Figure 7. Weighted Average Defuzzification Method (1)

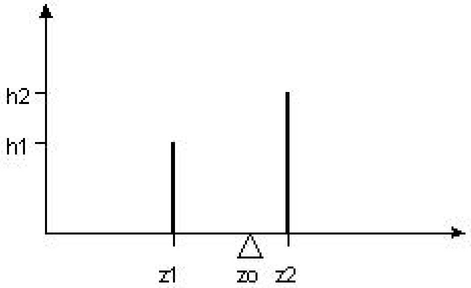


Figure 8. Weighted Average Defuzzification Method (2)

This method was chosen primarily for its speed. The crisp value that it produces is fairly easy to calculate computationally. The output also reflects the membership of each contributing membership function. Also the value depends on the shape of the waveform. The output will not move as smoothly from value to value as it would have if a more complicated defuzzification method was chosen, but the speed advantages outweigh the small gain in accuracy in a real time control situation where time response is a concern.

IV. IMPLEMENTATION

The implementation of this closed loop control strategy has many parts. The ultimate goal is to increase/decrease the field current of a synchronous generator to raise/lower the output voltage to its rated value. The output voltage of the generator must be stepped down to a level that is suitable for the DSP, i.e. 0 – 10 volts. The generator is assumed to be at constant 60 Hz with a balanced three-phase load. With this assumption, it is necessary to monitor only one phase of the output voltage. This signal is converted into a digital signal by going through an Analog to Digital (A/D) converter and fed to the DSP where the fuzzy logic decision will be made to determine the amount of field current needed. This digital value will then be converted to an analog signal (D/A). The analog signal will be fed to the isolation and power amplifiers that will in turn change the value of the field current to maintain a constant output voltage at the synchronous generator terminals. The experimental setup of this real time procedure is given below.

EXPERIMENTAL SETUP

The main piece of hardware that was used in this design was the DSP. The components that were used on it were the 12.5 μs/12-bit A/D converter, the 12-bit D/A converter, the timer, and the 40 MHz, 32-bit processor.

The synchronous generator that was controlled in this design was a *General Electric*, 5kVA, three phase, 1800 rpm, 120 V_{rms} machine. The field current was rated at 1.5 Amps.

The step down and isolation stage of the closed loop control system was implemented using a *Probe Master Inc.*, 20/1 voltage attenuater. This stepped down the rated voltage of the generator with a phase voltage of 120 Vrms (170 V_P), to 8.5 V_P. The membership functions of the controller were centered on this voltage.

Since the A/D converter could do a full conversion in 12.5 μs, it was more cost feasible to handle the problem of trying to read the peak value of the sinusoidal voltage entering, with a software solution. For every cycle to be measured, a sample number of A/D readings (100) were taken and the highest value was selected as the immediate peak voltage.

The *rate of change of voltage* was found by storing the previous voltage value until the next value was calculated.

$$\frac{dV}{dt} = \frac{V_{current} - V_{previous}}{\Delta t}$$

Since the time through each repetition of the program remained constant (Δt = *constant*), it was not necessary to divide by the change in time. The difference between $V_{current}$ and $V_{previous}$ gave the necessary information on the rate of change of the *voltage*. The membership functions for *rate of change of voltage* were scaled to accommodate this alteration. The rate of change that was implemented was:

$$\frac{dV}{dt} \equiv V_{current} - V_{previous}$$

Once the fuzzy logic decision has been made according to the two inputs, the D/A converter would turn the output into an analog signal that could be fed to the power amplifier. The power amplifier would step up the output signal generator field current accordingly.

The isolation amplifier used was a *Burr-Brown* opto-coupling device that provided additional isolation and safety.

The power amplifier portion of the system was implemented with a *Dayton, Motor Control Power Amplifier*. This device was driven by a 0-10 V_{DC} signal and would produce a 0 – 3 Amp signal that was fed to the field of the synchronous generator

Figure 9 shows the laboratory setup for the implementation of the fuzzy logic controller.

Figure 9. Laboratory Setup.

SOFTWARE

The software in this design refers to the fuzzy logic control algorithm that was programmed into the DSP that makes the *intelligent* decision of how much field current is to be applied. Also, there is software that was developed to run the hardware of the DSP and its peripheral board. This software handles the A/D and D/A converters and the timers also.

All the code for the DSP was written in the TI *C5x*'s assembly language. The assembly language was then examined and optimized for faster time performance.

The fist step in this program was to initialize the processor, program memory, A/D, D/A, and timers. The main program begins by reading from the A/D 100 times per 60 Hz cycle. These values are evaluated and the largest value is selected as the peak voltage. This peak voltage is the first input to the controller (*voltage*). The next stage of the program was to find the rate of change of voltage. Each time through the program, the current voltage is compared against the previous voltage. This difference is the *rate of change of voltage* and is the second input to the controller. After the two inputs are found, the main fuzzy logic decision algorithm is executed to find the appropriate output value. Once this occurs, the value is output through the D/A converter and held until the next output value is ready. This process is repeated indefinitely. Figure 9 shows the flow chart for the fuzzy logic-based controller implemented on the DSP.

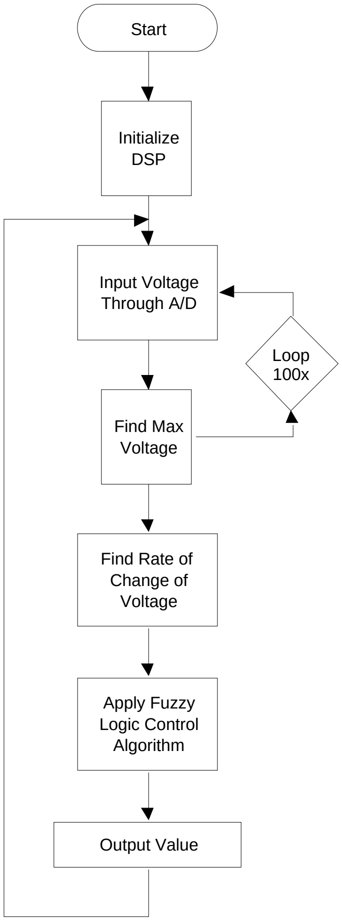


Figure 10. Flow Chart for the Fuzzy Logic-Based Synchronous Generator Controller.

EXPERIMENTAL RESULTS

The controller was tested under four different types of loads, a light resistive load (20%), a heavy resistive load (85%), a capacitive, and an inductive load. In all cases the fuzzy logic controller was able to bring the output voltage of the generator back to its rated voltage. Figures 10 shows the armature voltage of the generator as a sudden resistive load is applied. Figure 11 shows the voltage that the controller applies to the field in order to bring the generator back to its rated voltage.

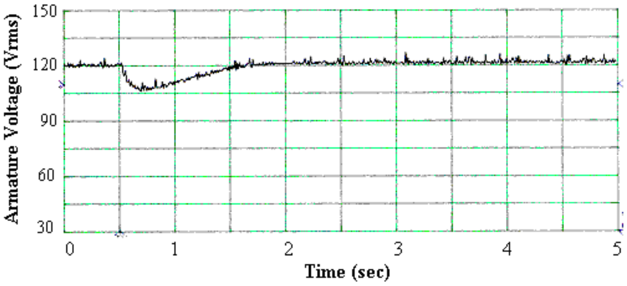


Figure 11. Armature Voltage Under a Light Resistive Load (20%).

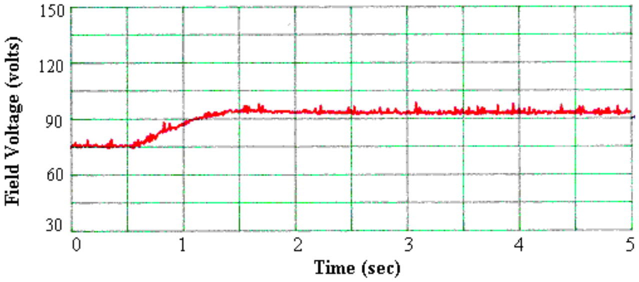


Figure 12. Field Voltage Applied Under a Light Resistive Load (20%).

Figures 13-18 show the armature and field voltages of a heavy resistive load (85%), a capacitive, and an inductive load.

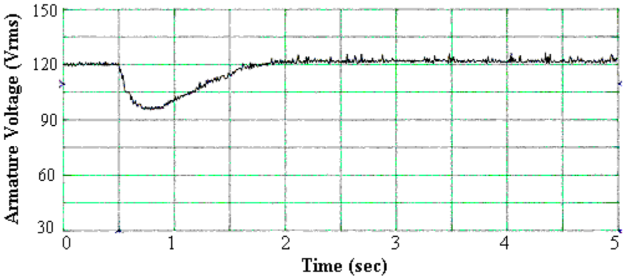


Figure 13. Armature Voltage Under a Heavy Resistive Load (85%).

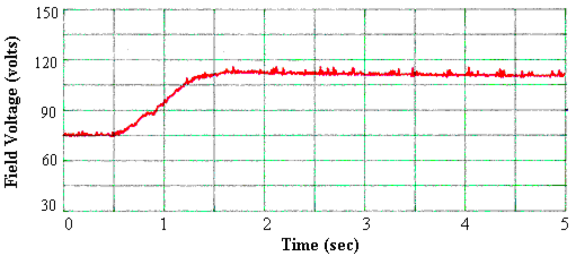


Figure 14. Field Voltage Applied under a Heavy Resistive Load (85%).

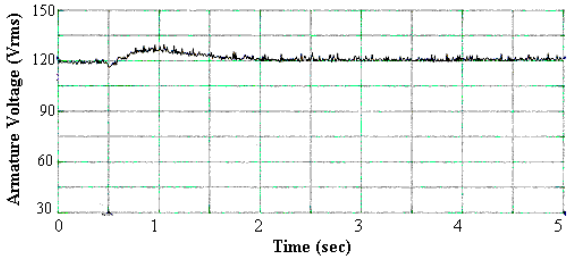


Figure 15. Armature Voltage Under a Capacitive Load.

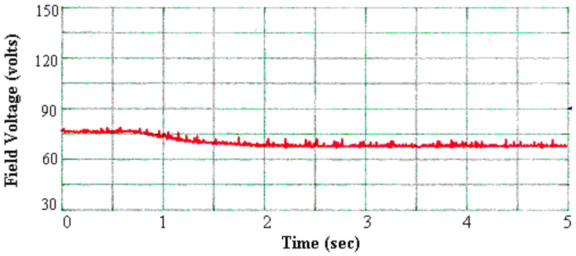


Figure 16. Field Voltage Applied under a Capacitive Load.

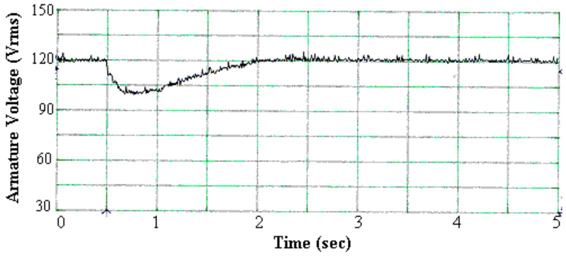


Figure 17. Armature Voltage Under an Inductive Load.

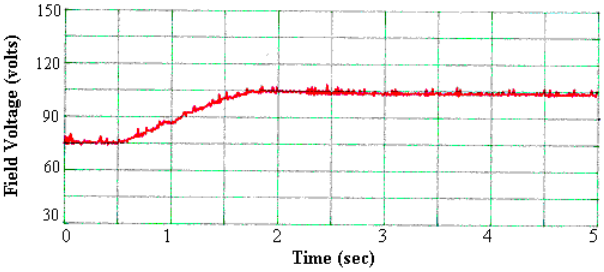


Figure 18. Field Voltage Applied under an Inductive Load.

Figures 19 and 20 show the armature and field voltages of the generator during the start-up routine of the controller.

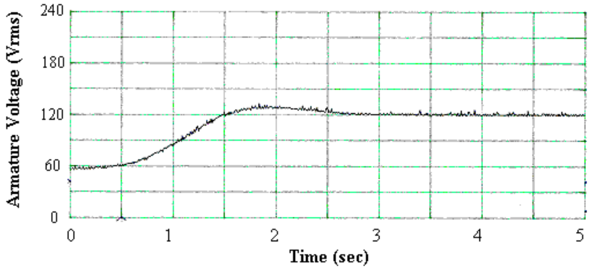


Figure 19. Armature Voltage of the Generator During Start-Up.

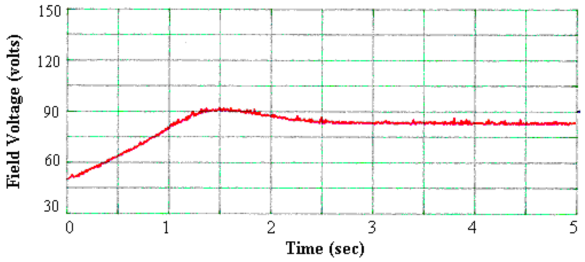


Figure 20. Field Voltage Applied to the Generator During Start-Up.

V. CONCLUSION

In this paper the design of a fuzzy logic controller was discussed. The controller was to regulate the output voltage of a synchronous generator by varying the current that was applied to the field of the generator in real time. This was accomplished by a closed loop control system. This system stepped down the output of one phase of the generator to a suitable microprocessor level, converted it to a digital signal, and performed a fuzzy logic control algorithm on the input and the rate of change of input. After the output signal was calculated, it was fed to a D/A converter to produce an analog signal. This signal was then fed to the isolation and power amplifiers. The isolation amplifier provided additional isolation and safety. The power amplifier provided the necessary current to the field of the generator in order to keep the generator's output voltage at its rated value.

The relevant results are presented and discussed. Despite the highly non-linear nature of the system, the transient and steady state performance with the fuzzy controller are seen to be quite satisfactory.

REFERENCES

1) Hasan, Abul R., Martis, Thomas S., Sadrul, A.H.M, "Design and Implementation of a Fuzzy Controller Based Automatic Voltage Regulator for a Synchronous Generator", *IEEE Transactions on Electrical Machinery*, Vol.9, No.6, Sep. 1994

2) M. Jamshidi, "Fuzzy Logic and Control, Software and Hardware Applications", University of New Mexico, PTR Prentice-Hall, Inc. Ch 1-4, 1993.

3) Theodore Wildi, "Electrical Machines, Drives, and Power Systems" Prentice Hall, Columbus, OH, Ch 6, 1997

4) D. Panda, V. Ramanarayanan, "Fuzzy Logic Based Control of Switched Reluctance Motor", International conference on Signal Processing Applications and Technology (ICSPAT), Toronto, Feb., 1998.