Darrin Ginder

CSc-180
Spring 2018
Project 1 - EXPERT SYSTEMS / FUZZY LOGIC

# Introduction

This report details the nuances of an expert system that, based off of user inputs, helps to determine what could be wrong with the user's computer and possibly fix it. It does so by a combination of yes and no questions and fuzzy logic to determine the actions the user should take and what the problem might be. This expert system was created for those with an intermediate level of computer efficiency that are able to do things such as determine if some of the hardware is faulty, run a diagnostic, find CPU usage and hard disk usage, and the ability to delete programs from the computer.

# Expert System

My expert is Daerik Vaughan is a 22-year-old tech specialist that was also interviewed by Milton Chiem. He has an entire lifetime of experience when it come to the inner hardware of computers. Officially though he has had 5 years of professional experience and because of that has the technical experience equivalent to a CompTIA A+ and a CompTIA Network+ certified persons. He currently though works at the Last Stop Computer Repair tech support shop and is attending college in order to obtain a mechanical engineering degree.

The main way I went about getting the information question for this expert system was mainly simple interview questions. The first questions I asked were pertaining to what problems Daerik generally encounters within his workplace. This got me a total of 9 different categories he generally gets. These categories include: nothing is wrong, a power issue, to turn off start up programs, the computer has a virus, the hardware has failed, there is a software failure, combination of startup and viruses, a combination of hardware and viruses, and that the date doesn't make sense.

When these were determined is was a simple step by step process of if this succeeds or fails what do you do. This allowed me to build the trees corresponding to each problem. Next I asked how he determines which problem it is and learned that with 8 of them he may bases it off of the boot tome, the CPU use while idle, and the hard disk use while idle. This immediately made me think that fuzzy logic could be applied here with a few simplifications. I began split the 3 inputs based off of his opinion into different categories such as fine, not fine, taken, etc. and where these categories cross over. Using these categories I began building a 3D FAM to get what conditions. Using this information, I was able to create a fuzzy system for this expert system.

Unfortunately, to make the fuzzy system work I had to make a few simplifications. I had to combine hardware failure and software failure into 1 to make the system output nicely. Luckily the expert was able to help me work around this and combine the trees into one. I also had to contain the application a bit because the expert began to go into lingo and techniques that you must be in the field for years to understand.

# Expert System Design

## Description

The way I implemented the knowledge of my expert was in a three-step process. The first step is that system begins with a ruleset that creates a tree detailing if the issue is a power issue or not. If the user enters that the issue is not with the power, then the system proceeds to the second step which is the fuzzy logic. This fuzzy logic requires the asks for three numbers: the boot time, CPU usage while idle, and hard disk use while idle. Based off of those numbers the system will generate a number. This number is then used to determine the problem that is most likely affecting the pc. The system will then fire one or more facts by itself pertaining to the Startup Programs tree, the Virus tree, or a Hardware Software tree. The system will then transverse the trees in a manner similar to the first part while giving precedence to the startup and hardware tree so that they are always done before the virus tree.

## Overview

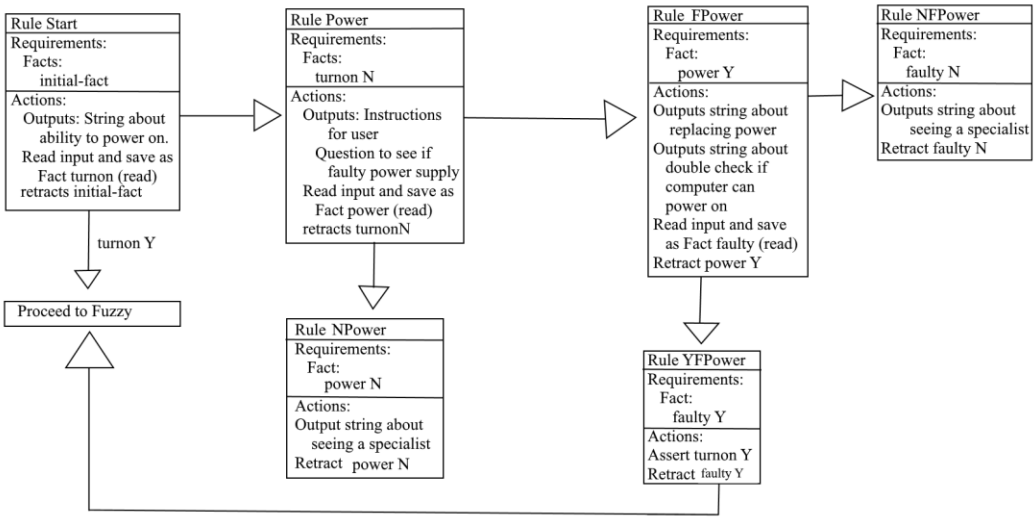
My rules can be split into 9 sections. I will rules, facts created and retracted, and the outputs of each section.

### Special Note

All inputs for expert system to work must be either a Y or N or a decimal between the ranges. Nothing else will work.

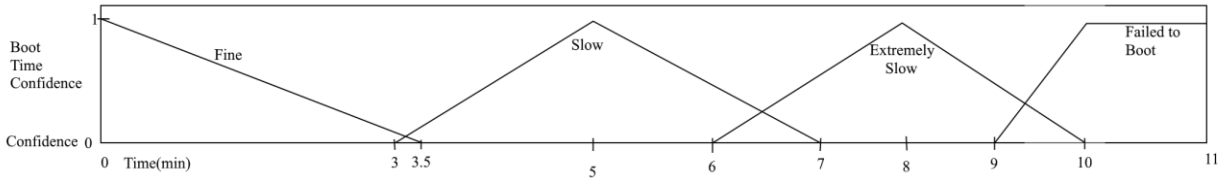
Section 1: Power Issue

The power issue of the system is really simple. The flow of its is as follows:

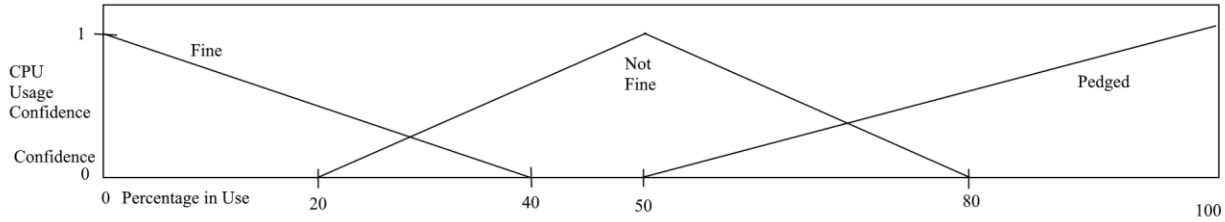



Section 2: Fuzzy System

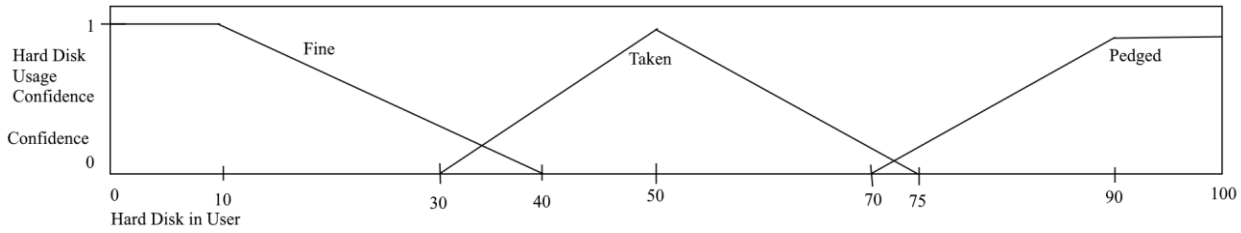
With the fuzzy system the first thing that it has the user do is enter in 3 values. The first value entered is Boot Time to OS as a number between 0-10. 0 represents around 0 minutes to boot while ten represents ten or more minutes or the facts that it doesn't boot to OS. Depending on the value it can fall into one of the following categories.




The Second number entered is the CPU usage while the computer is idle. The User must enter in a number ranging from 0-100. Each number represents the percent of CPU used while 100 is also used if the computer failed to boot to OS.




The third number entered is similar to the CPU usage while the computer is idle in that it is hard disk used while in idle. The User must enter in a number ranging from 0-100. Each number represents the percent of hard disk used while 100 is also used if the computer failed to boot to OS.



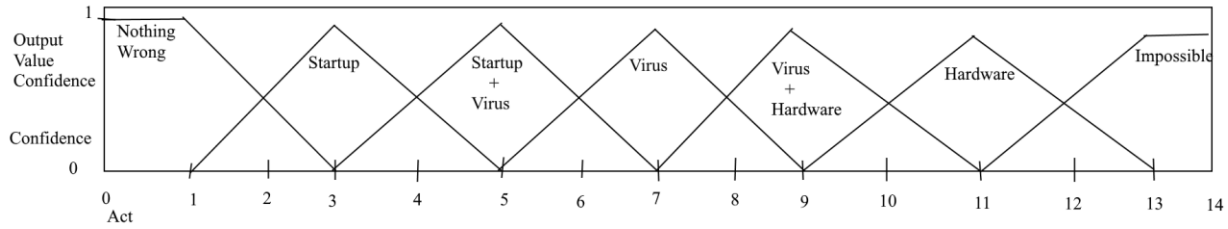

The fuzzy system then and these three values together and outputs a value based on the FAM charts as follows

| Hard Disk Fine / CPU / Boot Time | Fine | Not Fine | Pedged |
|---|---|---|---|
| Fine | Nothin Wrong | Virus/Spyware | Virus/Spyware |
| Slow | Turn Off Startup Items | Turn Off Startup Items + Virus/Spyware | Software/Hardware |
| Extremely Slow | Turn Off Startup Items + Virus/Spyware | Software/Hardware | Software/Hardware |
| Failed to Boot | Software/Hardware | Software/Hardware | Software/Hardware |

| Hard Disk Taken / CPU / Boot Time | Fine | Not Fine | Pedged |
|---|---|---|---|
| Fine | Virus/Spyware | Virus/Spyware | Virus/Spyware |
| Slow | Turn Off Startup Items + Virus/Spyware | Turn Off Startup Items + Virus/Spyware | Turn Off Startup Items + Virus/Spyware |
| Extremely Slow | Virus/Spyware + Software/Hardware | Virus/Spyware + Software/Hardware | Software/Hardware |
| Failed to Boot | Software/Hardware | Software/Hardware | Software/Hardware |

| Hard Disk Pedged / CPU / Boot Time | Fine | Not Fine | Pedged |
|---|---|---|---|
| Fine | Impossible | Impossible | Impossible |
| Slow | Virus/Spyware + Software/Hardware | Virus/Spyware + Software/Hardware | Virus/Spyware + Software/Hardware |
| Extremely Slow | Software/Hardware | Software/Hardware | Software/Hardware |
| Failed to Boot | Software/Hardware | Software/Hardware | Software/Hardware |

Each of these FAMs possible combinations was made a rule that requires the corresponding inputs categories and asserts the proper output category as a fact called Act as shown in this graph.




When all of the appropriate Rules have fired the fuzzy System will then take the fuzzy Act fact and defuzzify it and assert the result as a fact called Action.

Section 3: Nothing Wrong

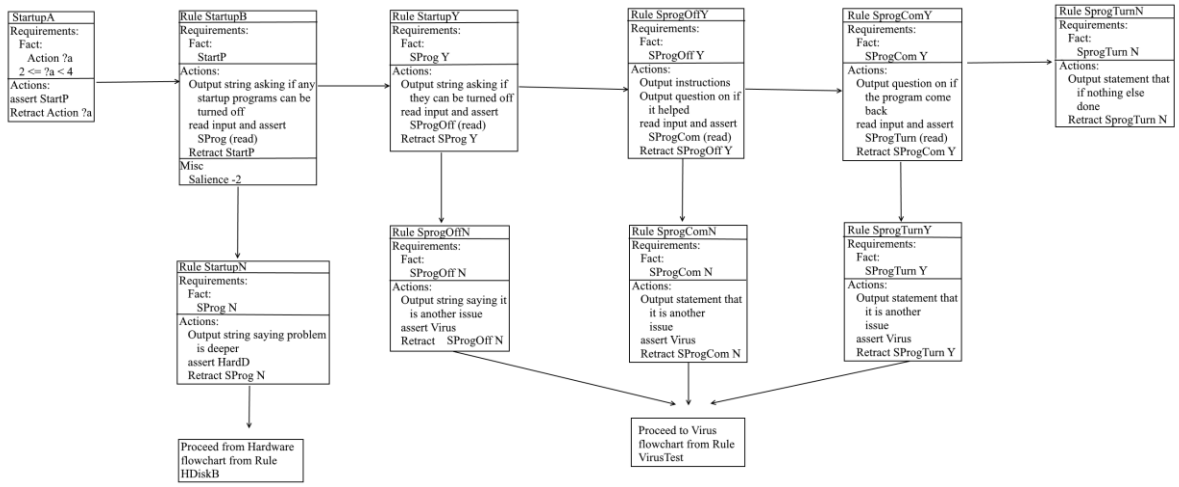
This Section of the Cody is quiet easy. It has one rule called NothingWrong that requires the Output to be defuzzified and asserted as the fact Action ?a. The rule then tests if the value is less than 2 and if it is then it will print a message and retract the Action fact.

Section 4: Impossible

This section of the code works exactly like the NothingWrong rule except that it is called Imposs and checks if the Action fact is greater or equal to 12.

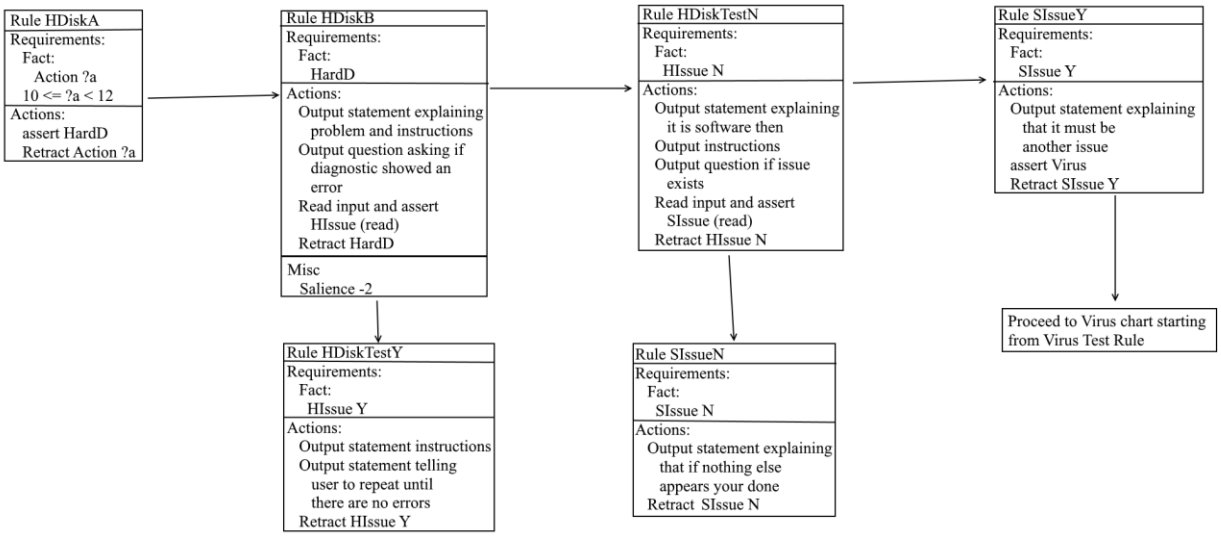
Section 5: Startup Programs Only

This section of code follows the following flowchart and only activates when the defuzzification has happened.



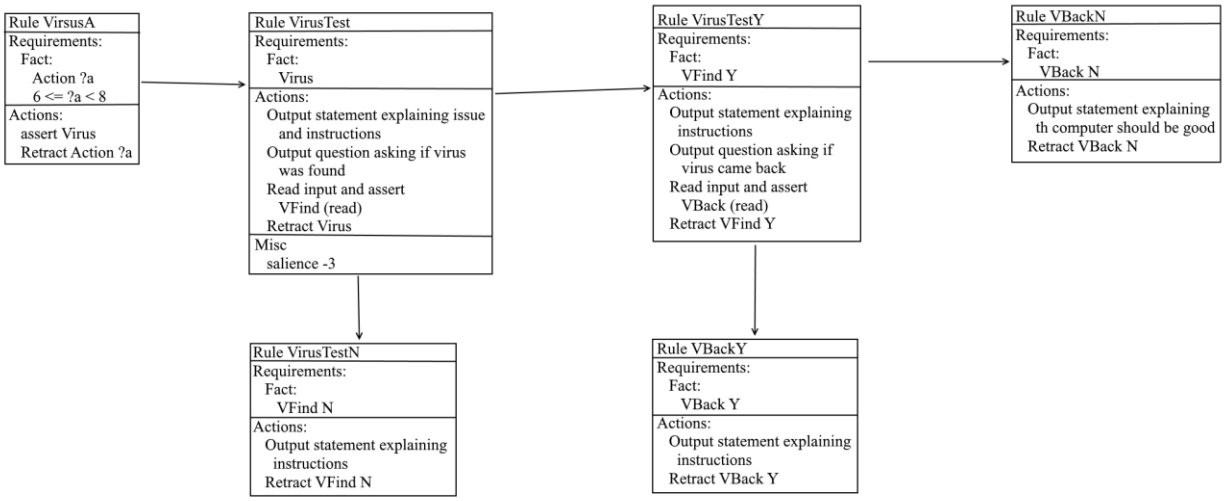

Section 6: Hardware/Software Issue

This sections begins at rule HardDiskA if coming from the fuzzy system or it can start from HardDiskB if coming from somewhere else. It follows this flowchart.



Section 7: Virus/Spyware

This section is very similar to Section 6 in that if it is called by the fuzzy Facts then it will start at VirusA. If it is called elsewhere it will start at VirusTest. This Section also has the lowest salience so if it is called it will always go last.



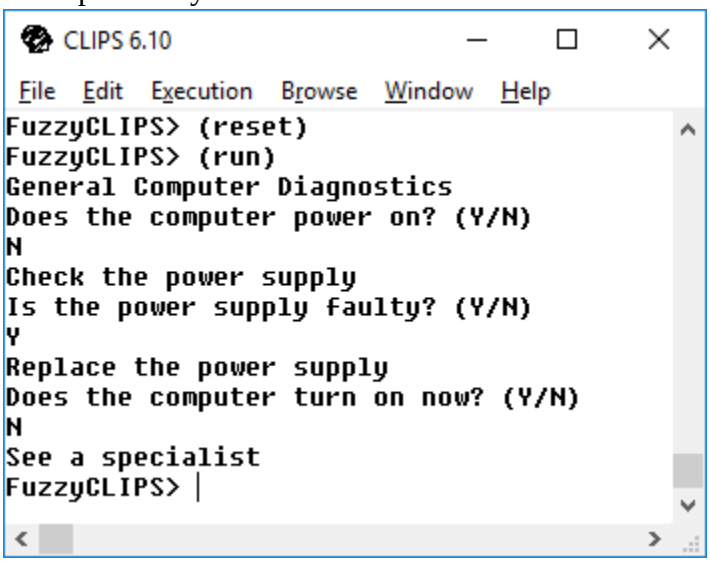

Section 8: Software + Virus

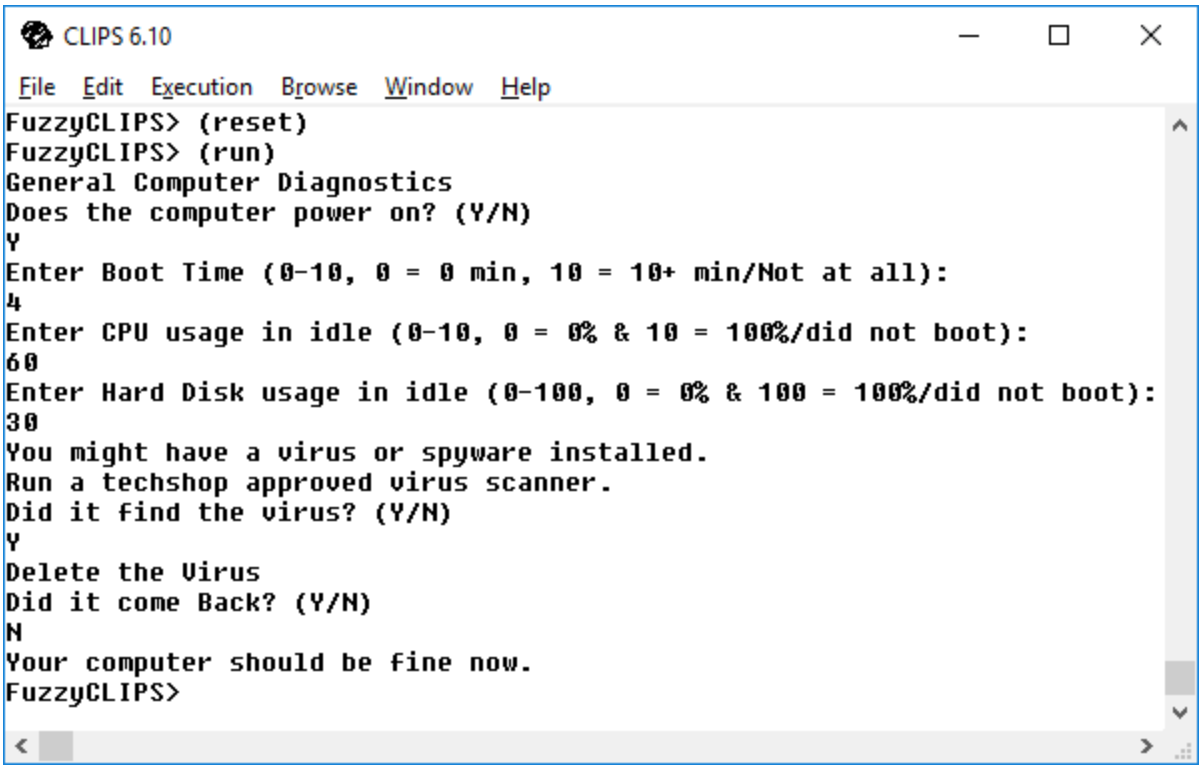
This Section is easy. There is only one rule call HDiskAndVirus that check if the defuzzified output is between 8 and 10. If it is it will assert both StartP and Virus.

Section 9: Hardware/Software + Virus

This Section is also easy. There is only one rule call VirsusAndStart that check if the defuzzified output is between 4 and 6. If it is it will assert both HardD and Virus.

Examples of System






# Conclusion

Overall I would like to think that this program does its job decently. It has no known bugs within it and the only problem is that it I had to cut corner to get it to function perfectly and for a more average person. I could not go into full detail on some things to keeps its purpose the same and I had to simplify and combine a few things in order to get rid of any errors.

# Appendix A

## Installation Instructions

In order to run this file you first need to download FUZZYCLIPS from a site such as http://athena.ecs.csus.edu/~gordonvs/180/180homework.html. When you have extracted all of the downloaded files open the CLIPSwin application and either click Filr then Load Constructs or hit Ctrl + L. Navigate to the ComputerTest.CLP file and open it. When that is done type (reset) and hit enter and then (run) and then enter. The program should be loaded now.

# Appendix B

## User Guide

After the user has installed and loaded this program the user only need to follow the on screen instructions. The program will tell the user the general idea of what to do. How the user goes about this is up to them. Examples of variable actions include which hardware diagnostics to run, or how the search out corrupt programs. When a yes or no question is asked the user only needs to type Y to say yes to the questions or N to say no. Finally, when the program asks for a number the user must enter either a decimal or a whole number within the range given. Anything else but these specified inputs will break the program.

# References

## People

Vaughan, Daerik. 15 Feb. 2018. Same Interviewee as Milton Chiem

## Websites

"Wine and Food Pairing Chart." Wine Folly, 30 Jan. 2017, winefolly.com/tutorial/basic-wine-and-food-pairing-chart/.