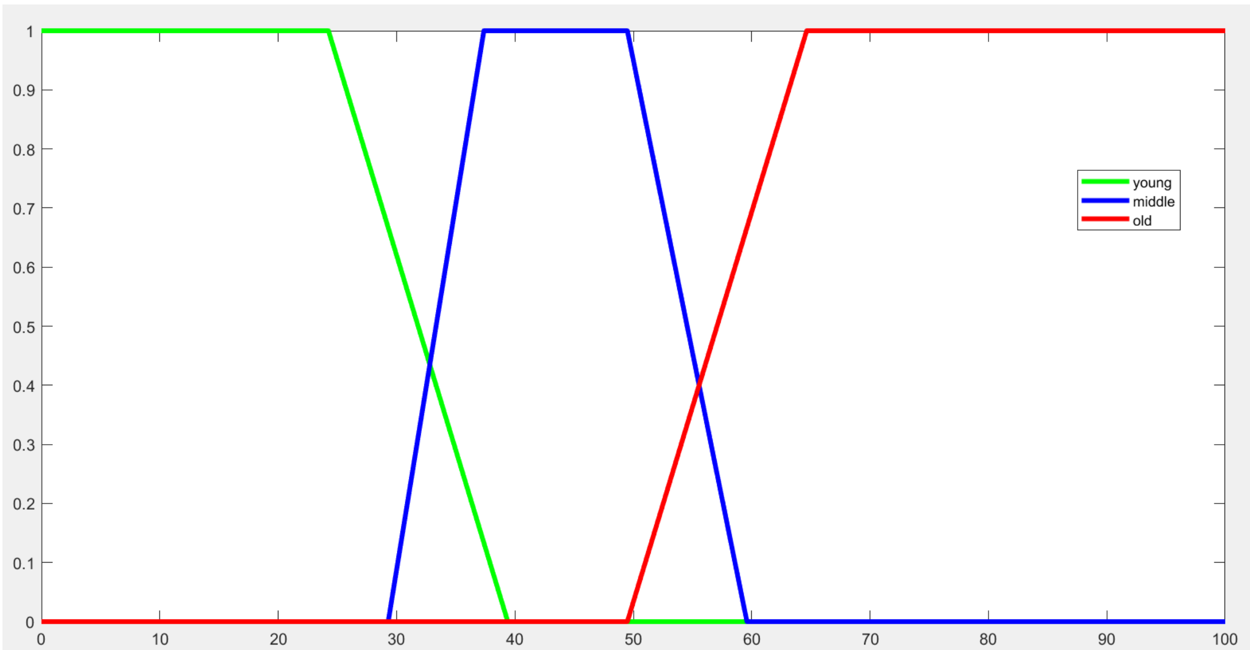
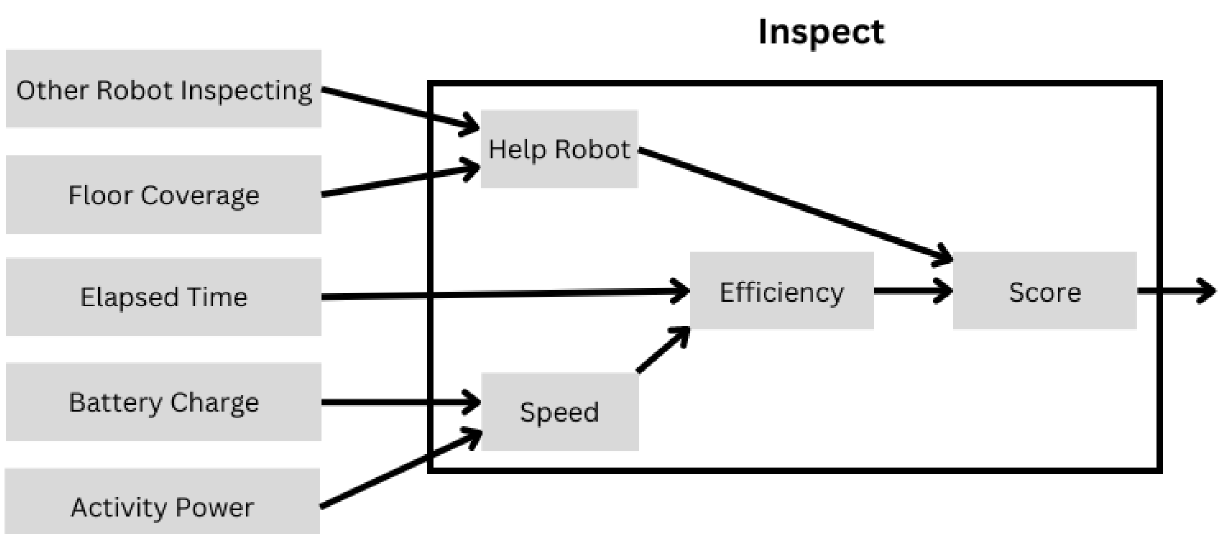
Problem 1

The thought process behind the middle-aged values is that most people roughly consider mid-thirties to mid-fifties as middle-aged. This set up has 33 – 55 being considered middle-aged, with 29 – 33 young but somewhat middle-aged, and 56-60 old but somewhat middle-aged.

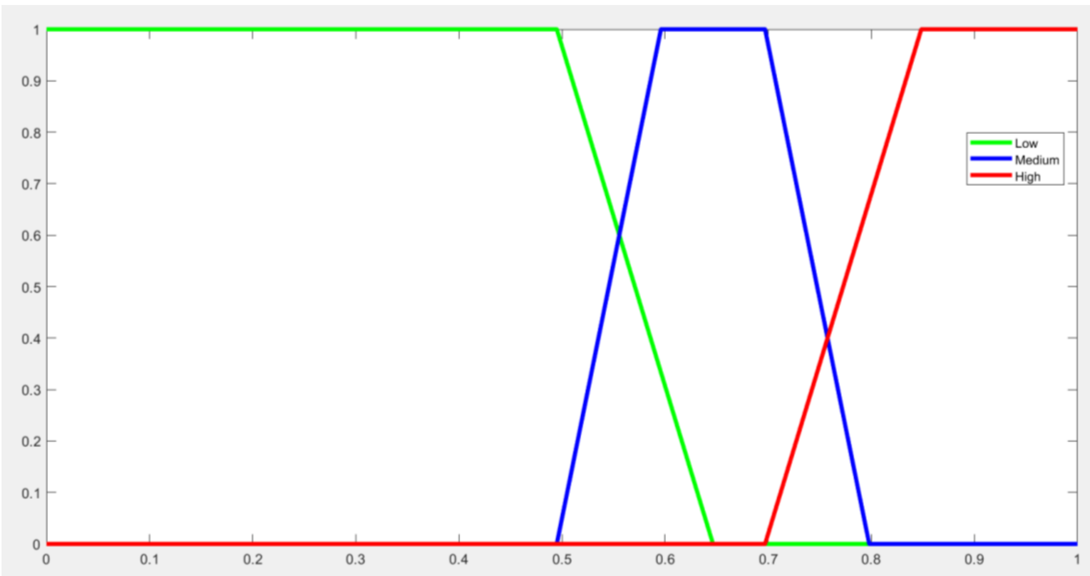


Problem 2



Help Robot

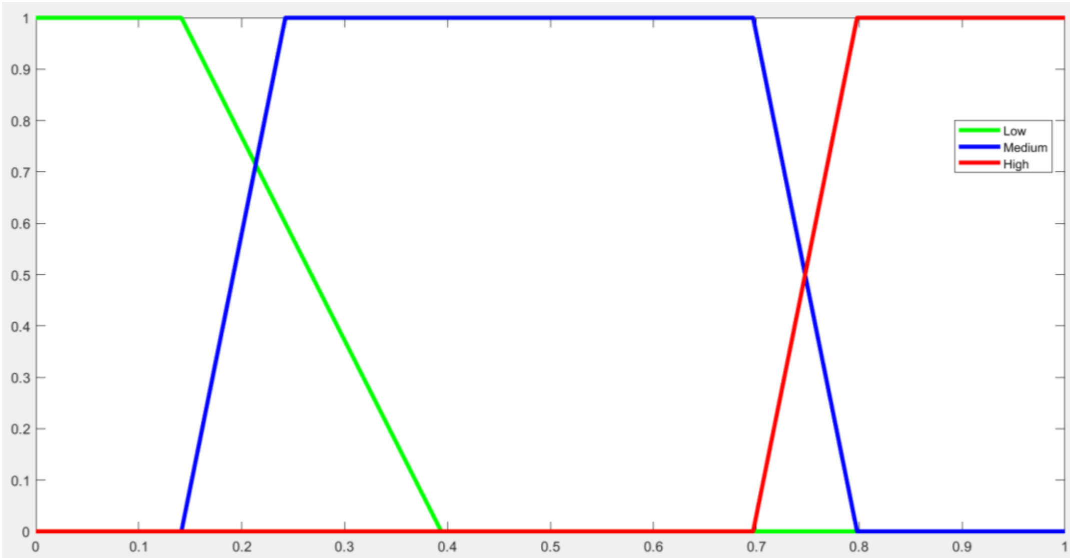
Help robot is a subsystem designed to consider if another robot is covering an area, and if so, is the inspect area so big that it would be more efficient to split it between both robots. Ideally robots should focus on separate tasks, so the fuzzy graph has a very large Low region with lesser Medium and High regions. If “Other Robot Inspecting” is Low, no other robot is inspecting this area and therefore the “help robot” output will be High. If “Other Robot Inspecting” is High, that means the other robot is very motivated for this task even if it’s a large region.



Help Robot				
		Other Robot Inspecting		
		Low	Medium	High
Floor Coverage	Low	High	Low	Low
	Medium	High	Low	Low
	High	High	High	Medium

Speed

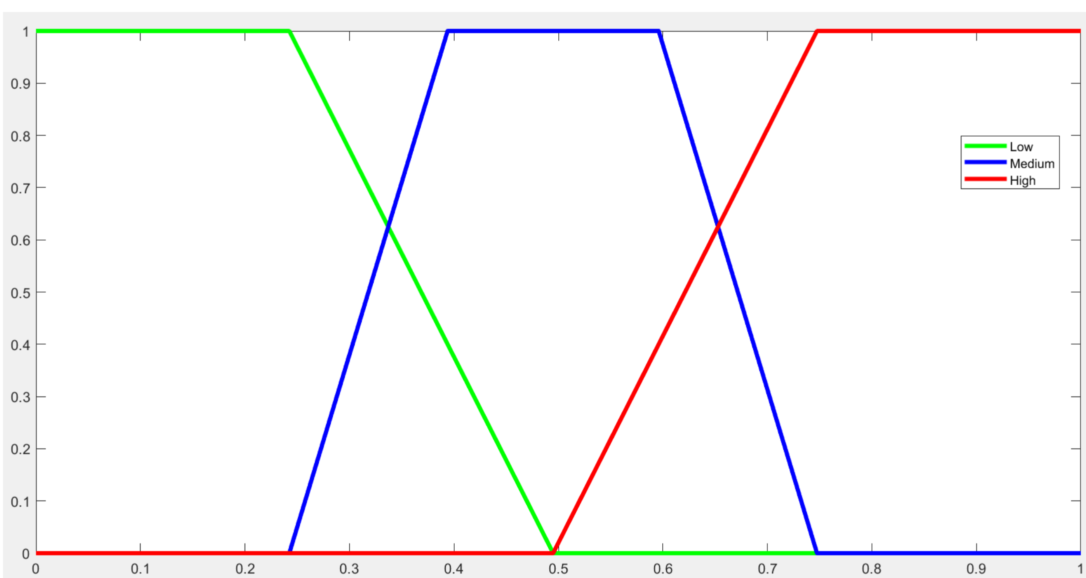
Speed is the robot’s movement speed. The fuzzy graph is set up with a large Medium region because most actuators and vehicles are most energy efficient around a medium speed. Additionally, moving to slowly may be ineffective while inspecting larger regions, as areas it has already inspected may have new activity happening. The robot wants to maintain this efficient medium speed, but only will do so if the activity power is not consuming too much power already.



Speed				
		Activity Power		
		Low	Medium	High
Battery Charge	Low	Medium	Low	Low
	Medium	Medium	Medium	Low
	High	High	Medium	Medium

<u>Efficiency</u>

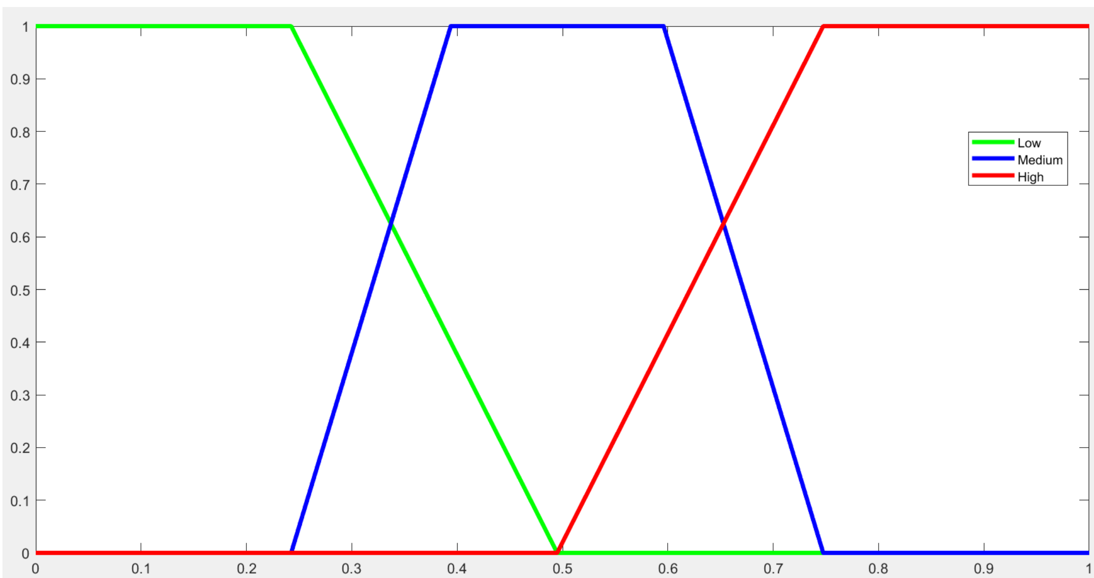
Efficiency of the inspection is a function of speed and time spent on the task. If a robot spends large amounts of time on a task because of a slow speed (from low battery or large activity power requirements), it's more efficient to focus on other, lower energy tasks. A different robot with more battery can inspect instead of this robot. This is a balanced fuzzy graph that doesn't have a preference on an output (like medium for speed).



Efficiency				
		Elapsed Time		
		Low	Medium	High
Speed	Low	Low	Low	Low
	Medium	High	Medium	Low
	High	High	High	Medium

<u>Score</u>

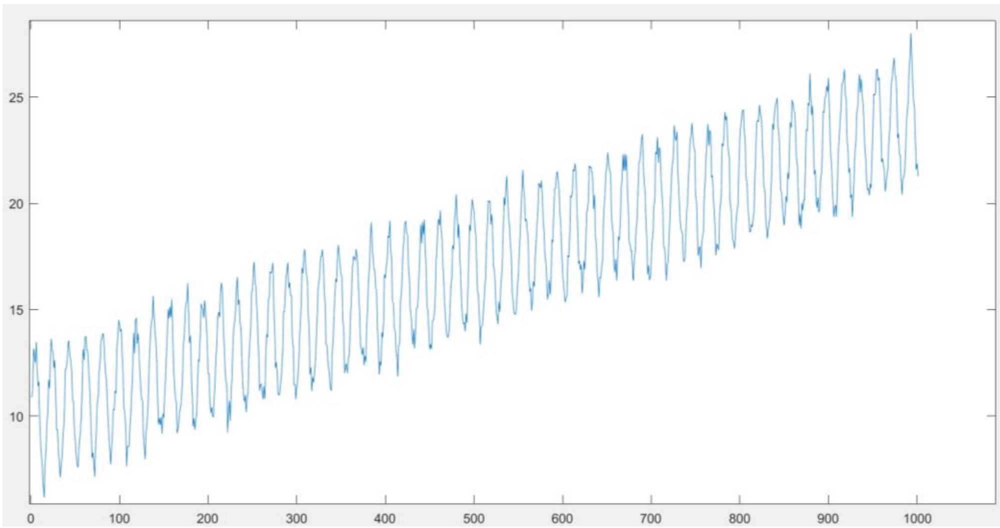
The score indicates if the desire of the robot to perform inspection. If other robots are not inspecting or need help, this robot needs to inspect the area. However, if this task is very inefficient for the robot due to power constraints, it will not inspect and perform other tasks instead.



Score				
		Efficiency		
		Low	Medium	High
Help Robot	Low	Low	Low	Medium
	Medium	Low	Medium	Medium
	High	Medium	High	High

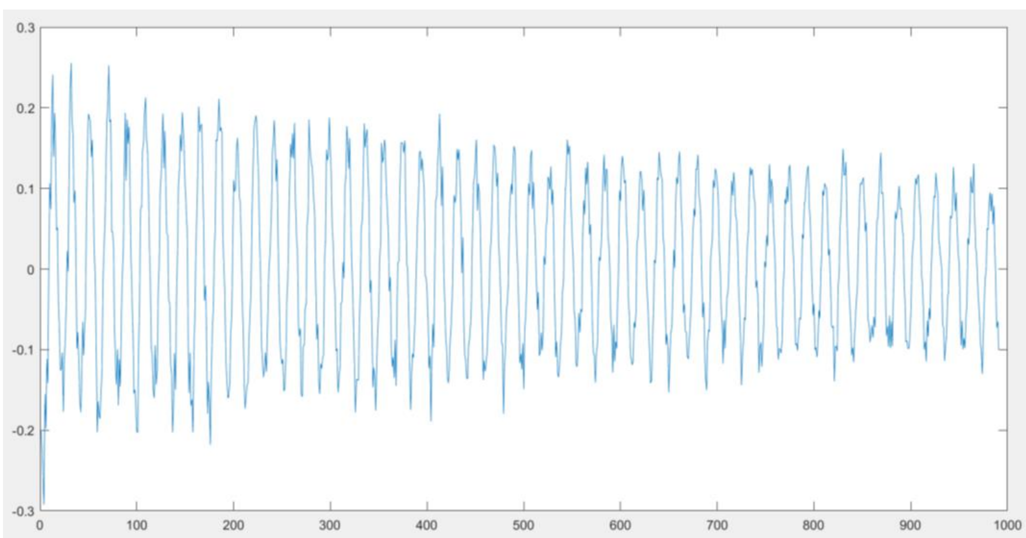
Programming Problem

My approach was to use 2 subsystems, the first with inputs of XYZ and TMA to create an Equivalent price variable. MAD and Equivalent Price are then input into the second fuzzy subsystem, which outputs Action. I wanted to normalize the inputs for the fuzzy system but noticed that there was a gradual upward trend to the stock price (XYZ) and TMA values.



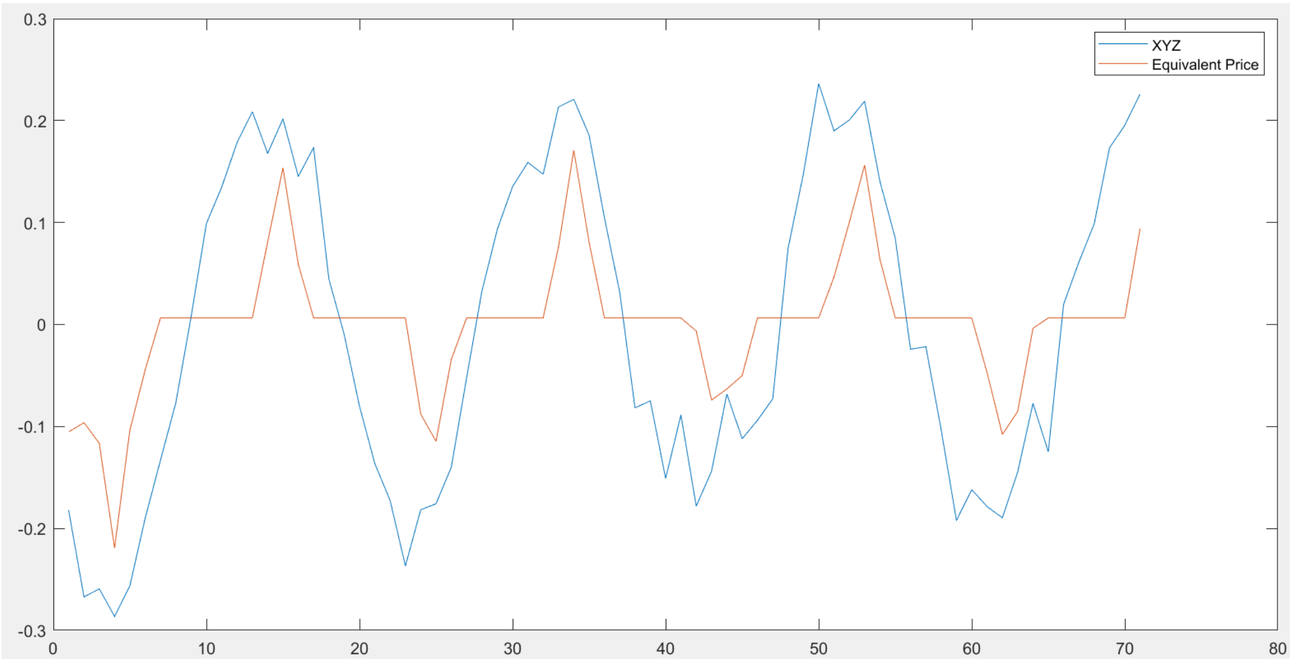
Graph 1: Stock Price (XYZ)

I normalized XYZ with respect to the last 20 days, then shift that down to 0 by subtracting the mean of the last 20 days. This gave it the following pattern, which is more consistent and easier to apply a reliable rule base to:



Graph 2: Normalized Stock Price (XYZ)

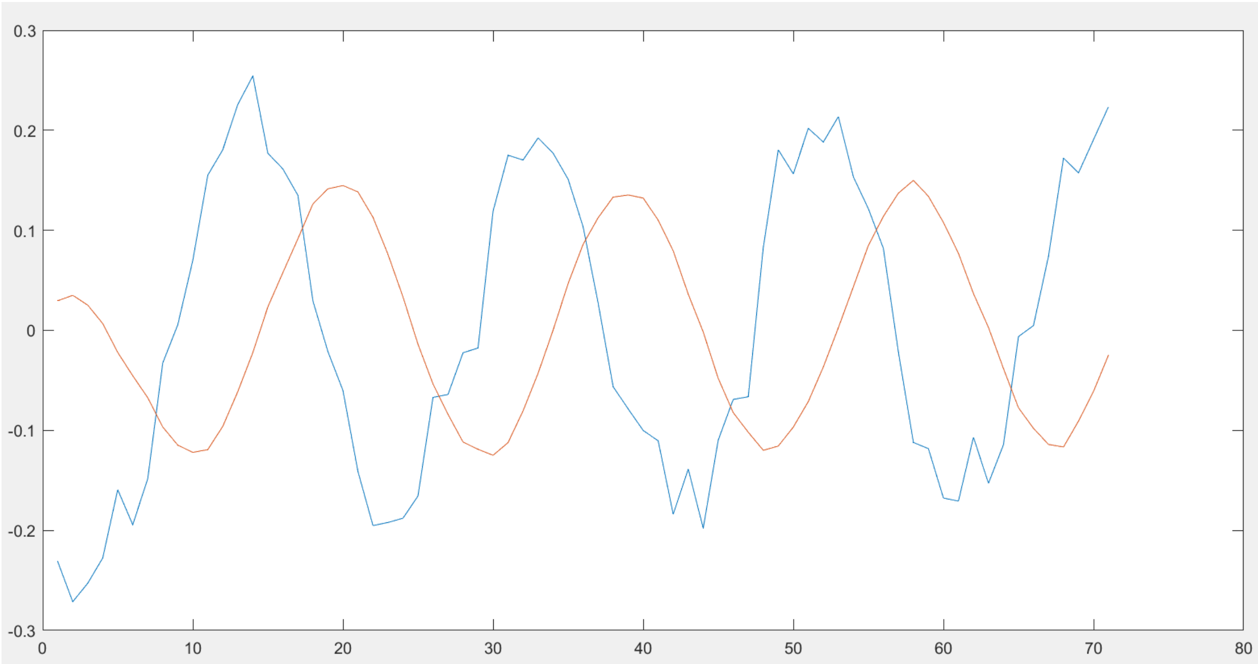
The output of the first subsystem, Equivalent Price, is shown below in Graph 3. The purpose of it is to convert the XYZ into a form with thin peaks that have a slight delay with maximums and minimums of the XYZ peaks, which is where I want to buy and sell. If the XYZ is between these peaks, then I want to wait for action. This makes it easier for the final subsystem to choose an action at optimal points.



Graph 3: Normalized XYZ compared with Equivalent Price output

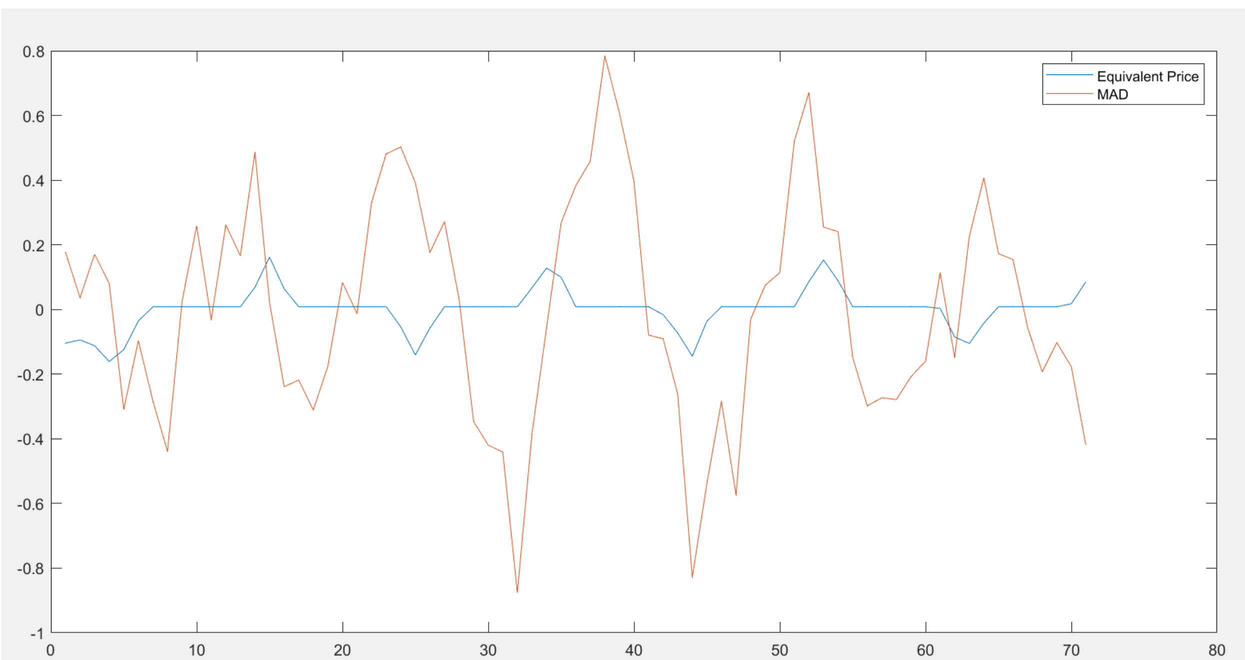
Equivalent Price						
		TMA				
		VL	L	M	H	VH
Stock Price	VL	M	M	VL	M	M
	L	M	M	L	M	M
	M	M	M	M	M	M
	H	M	M	H	M	M
	VH	M	M	VH	M	M

When creating rules for this subsystem, I saw a trend with XYZ and TMA. Graph 4 shows that XYZ maximums and minimums only occur when TMA is medium. The rule chart was designed on the idea that when TMA is not medium, I want to wait. If the TMA is medium, a peak in XYZ is occurring and I want to act upon it. If XYZ is high and TMA medium, then Equivalent Price is high. If XYZ is low and TMA is medium, then Equivalent Price is low.



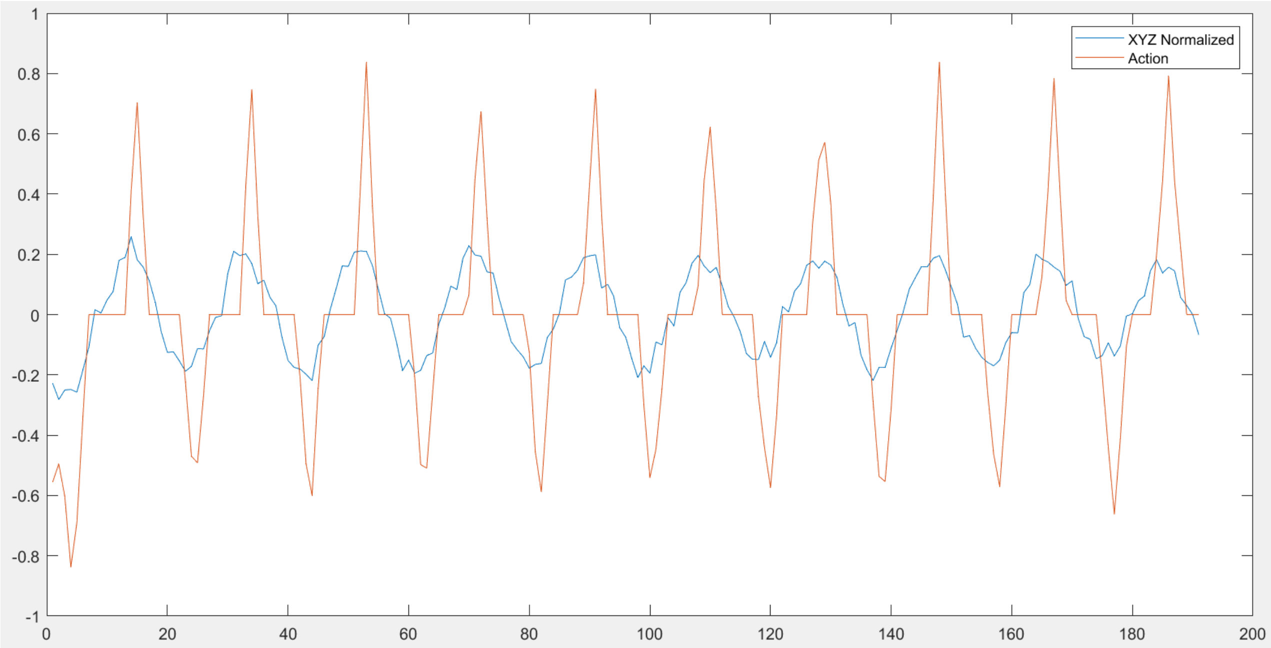
Graph 4: Stock Price (XYZ) compared with TMA (10 day)

The second fuzzy system relies mostly on the Equivalent Price variable. As seen in Graph 5, there isn't a distinct pattern relating XYZ trend and MAD (even though it is stated in the assignment there is); this made it difficult to design a rule structure around MAD. However, there is a clear distinction between the Equivalent Price and XYZ trends. Therefore, it made much more sense to rely on Equivalent Price.

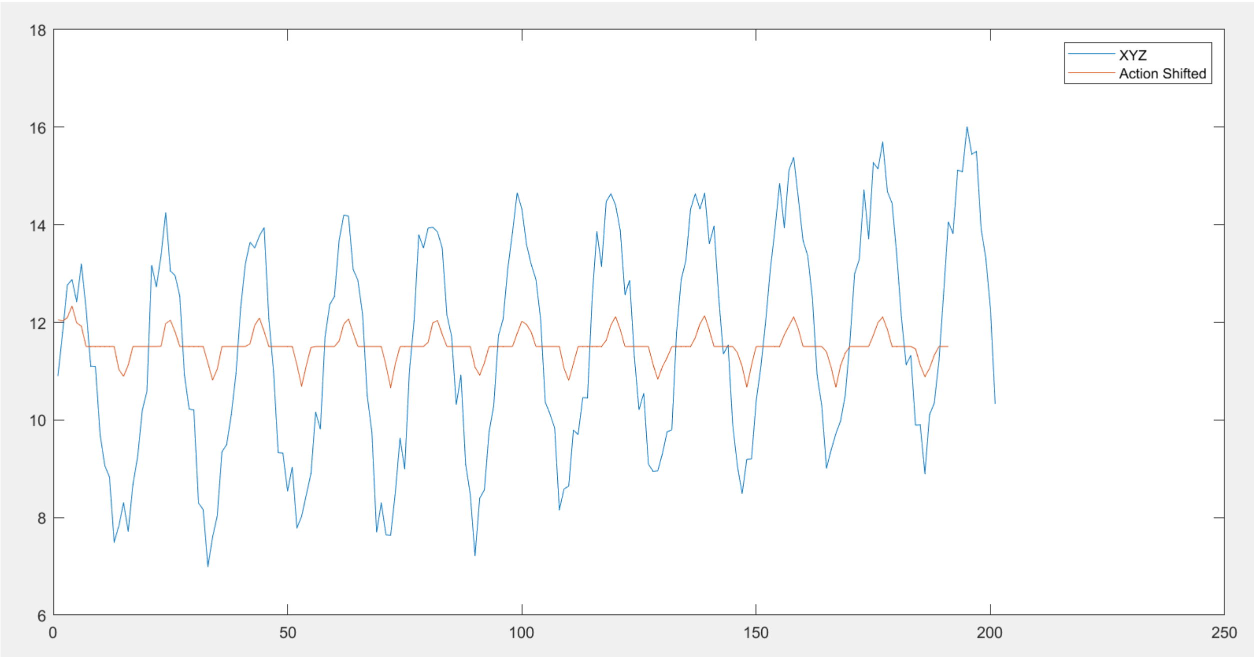


Graph 5: Equivalent Price compared with MAD

When tweaking this fuzzy subsystem, I used another graph comparison and altered the system's parameters so that days with low XYZ peaks overlapped with the values for "buy" action output, and days with high XYZ peaks overlapped with values for "sell" action output. See the graphs below:



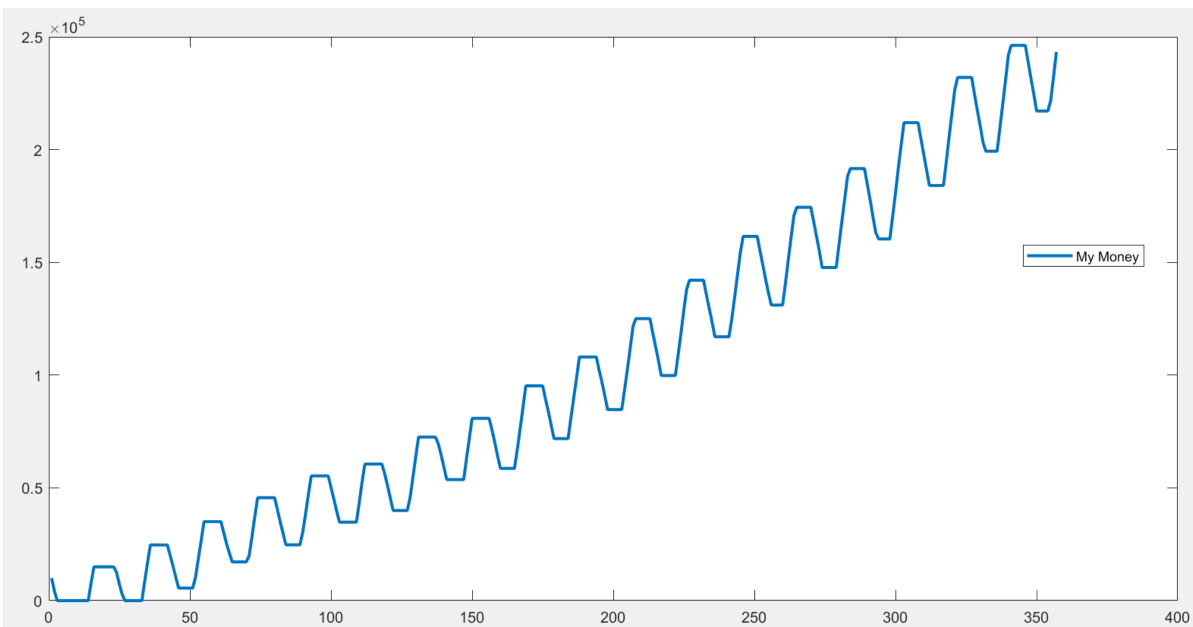
Graph 6: Normalized XYZ compared with Action



Graph 7: XYZ compared with Action (shifted for easier visualization)

Action				
		MAD		
		N	Z	P
Equivalent Price	VL	VL	VL	VL
	L	L	L	L
	M	M	M	M
	H	H	H	H
	VH	VH	VH	VH

Once the final fuzzy system was designed, I ran the simulation with actions occurring. I multiplied the -1 to 1 Action output by a factor to make it proportional to the number of stocks I wanted to trade. After trying several different proportionalities, I found that maxing out the stock size to 600 for most transactions had the best result. In the graph shown below, I made $ 233,300 and held 988 stocks after 365 days.



Graph 8: Simulation results by day

Discussion

Overall, this approach seems to work very well. If there is a consistent pattern with multiple variables, it's fast to develop a rule system that can be used to use decide an action. It's very useful to have an action output that is proportional to number of stocks to buy/sell instead of simply a binary buy or sell action. MAD turned out to be useless as there was no consistent pattern to create a rule system around.