Abstract
Fuzzy logic has rapidly become one of the most successful of today's technologies for developing sophisticated control systems. The reason for which is very simple. Fuzzy logic addresses such applications perfectly as it resembles human decision making with an ability to generate precise solutions from certain or approximate information. It fills an important gap in engineering design methods left vacant by purely mathematical approaches, and purely logic-based approaches in system design. While other approaches require accurate equations to model real-world behaviors, fuzzy design can accommodate the ambiguities of real-world human language and logic. It provides both an intuitive method for describing systems in human terms and automates the conversion of those system specifications into effective models.

What does it offer?
The first applications of fuzzy theory were primly industrial, such as process control for cement kilns. However, as the technology was further embraced, fuzzy logic was used in more useful applications. In 1987, the first fuzzy logic-controlled subway was opened in Sendai in northern Japan. Here, fuzzy-logic controllers make subway journeys more comfortable with smooth braking and acceleration. Best of all, all the driver has to do is push the start button! Fuzzy logic was also put to work in elevators to reduce waiting time. Since then, the applications of Fuzzy Logic technology have virtually exploded, affecting things we use everyday. Take for example, the fuzzy washing machine. A load of clothes in it and press start, and the machine begins to churn, automatically choosing the best cycle. The fuzzy microwave, Place chili, potatoes, or etc in a fuzzy microwave and push single button, and it cooks for the right time at the proper temperature. The fuzzy car, maneuvers itself by following simple verbal instructions from its driver. It can even stop itself when there is an obstacle immediately ahead using sensors. But, practically the most exciting thing about it, is the simplicity involved in operating it.

What do you mean by fuzzy??!!
Before illustrating the mechanisms which make fuzzy logic machines work, it is important to realize what fuzzy logic actually is. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth—truth values between "completely true" and "completely false". As its name suggests, it is the logic underlying modes of reasoning which are approximate rather than exact. The importance of fuzzy logic derives from the fact that most modes of human reasoning and especially common sense reasoning are approximate in nature. The essential characteristics of fuzzy logic as founded by Zadeh Lotfi are as follows.

- In fuzzy logic, exact reasoning is viewed as a limiting case of approximate reasoning.
• In fuzzy logic everything is a matter of degree.
• Any logical system can be fuzzified
• In fuzzy logic, knowledge is interpreted as a collection of elastic or, equivalently, fuzzy constraint on a collection of variables
• Inference is viewed as a process of propagation of elastic constraints.

The third statement hence, define Boolean logic as a subset of Fuzzy logic.

**Fuzzy Sets**

Fuzzy Set Theory was formalized by Professor Lofti Zadeh at the University of California in 1965. What Zadeh proposed is very much a paradigm shift that first gained acceptance in the Far East and its successful application has ensured its adoption around the world.

A paradigm is a set of rules and regulations which defines boundaries and tells us what to do to be successful in solving problems within these boundaries. For example the use of transistors instead of vacuum tubes is a paradigm shift - likewise the development of Fuzzy Set Theory from conventional bivalent set theory is a paradigm shift.

Bivalent Set Theory can be somewhat limiting if we wish to describe a 'humanistic' problem mathematically. For example, Fig 1 below illustrates bivalent sets to characterize the temperature of a room.

The most obvious limiting feature of bivalent sets that can be seen clearly from the diagram is that they are mutually exclusive - it is not possible to have membership of more than one set (opinion would widely vary as to whether 50 degrees Fahrenheit is 'cold' or 'cool' hence the expert knowledge we need to define our system is mathematically at odds with the humanistic world). Clearly, it is not accurate to define a transition from a quantity such as 'warm' to 'hot' by the application of one degree Fahrenheit of heat. In the real world a smooth (unnoticeable) drift from warm to hot would occur.

This natural phenomenon can be described more accurately by Fuzzy Set Theory. Fig.2 below shows how fuzzy sets quantifying the same information can describe this natural drift.
The whole concept can be illustrated with this example. Let's talk about people and "youthness". In this case the set $S$ (the universe of discourse) is the set of people. A fuzzy subset $\text{YOUNG}$ is also defined, which answers the question "to what degree is person $x$ young?" To each person in the universe of discourse, we have to assign a degree of membership in the fuzzy subset $\text{YOUNG}$. The easiest way to do this is with a membership function based on the person's age.

$$\text{young}(x) = \begin{cases} 1, & \text{if age}(x) \leq 20, \\ (30 - \text{age}(x))/10, & \text{if } 20 < \text{age}(x) \leq 30, \\ 0, & \text{if age}(x) > 30 \end{cases}$$

A graph of this looks like:

Given this definition, here are some example values:

<table>
<thead>
<tr>
<th>Person</th>
<th>Age</th>
<th>degree of youth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Johan</td>
<td>10</td>
<td>1.00</td>
</tr>
<tr>
<td>Edwin</td>
<td>21</td>
<td>0.90</td>
</tr>
<tr>
<td>Parthiban</td>
<td>25</td>
<td>0.50</td>
</tr>
<tr>
<td>Arosha</td>
<td>26</td>
<td>0.40</td>
</tr>
<tr>
<td>Chin Wei</td>
<td>28</td>
<td>0.20</td>
</tr>
<tr>
<td>Rajkumar</td>
<td>83</td>
<td>0.00</td>
</tr>
</tbody>
</table>

So given this definition, we'd say that the degree of truth of the statement "Parthiban is $\text{YOUNG}$" is 0.50.

Note: Membership functions almost never have as simple a shape as age(x). They will at least tend to be triangles pointing up, and they can be much more complex than that. Furthermore, membership functions so far is discussed as if they always are based on a single criterion, but this isn't always the case, although it is the most common case. One could, for example, want to have the membership function for $\text{YOUNG}$ depend on both a person's age and their height (Arosha's short for his age). This is perfectly legitimate, and occasionally used in practice. It's referred to as a two-dimensional membership function. It's also possible to have even more criteria, or to have the membership function depend on elements from two completely different universes of discourse.

**Fuzzy Rules**

Human beings make decisions based on rules. Although, we may not be aware of it, all the decisions we make are all based on computer like if-then statements. If the weather is fine, then we may decide to go out. If the forecast says the weather will be bad today, but fine tomorrow, then we make a decision not to go today, and postpone it till tomorrow. Rules associate ideas and relate one event to another. Fuzzy machines, which always tend to mimic the behavior of man, work the same way. However, the decision and the means of choosing that decision are
replaced by fuzzy sets and the rules are replaced by fuzzy rules. Fuzzy rules also operate using a series of if-then statements. For instance, if X then A, if y then b, where A and B are all sets of X and Y. Fuzzy rules define fuzzy patches, which is the key idea in fuzzy logic. A machine is made smarter using a concept designed by Bart Kosko called the Fuzzy Approximation Theorem (FAT). The FAT theorem generally states a finite number of patches can cover a curve as seen in the figure below. If the patches are large, then the rules are sloppy. If the patches are small then the rules are fine.

**Fuzzy Patches**

In a fuzzy system this simply means that all our rules can be seen as patches and the input and output of the machine can be associated together using these patches. Graphically, if the rule patches shrink, our fuzzy subset triangles gets narrower. Simple enough? Yes, because even novices can build control systems that beat the best math models of control theory. Naturally, it is math-free system.

**Fuzzy Traffic Light Controller**

This part of the paper describes the design procedures of a real life application of fuzzy logic: A Smart Traffic Light Controller. The controller is suppose to change the cycle time depending upon the densities of cars behind green and red lights and the current cycle time.

**Background**

In a conventional traffic light controller, the lights change at constant cycle time, which is clearly not the optimal solution. It would be more feasible to pass more cars at the green interval if there are fewer cars waiting behind the red lights. Obviously, a mathematical model for this decision is enormously difficult to find. However, with fuzzy logic, it is relatively much easier.

**Design**

First, eight incremental sensors are put in specific positions as seen in the diagram below.

The first sensor behind each traffic light counts the number cars coming to the intersection and the second counts the cars passing the traffic lights. The amount of cars between the traffic lights is determined by the difference of the reading of the two sensors. For example, the number of cars behind traffic light North is s7-s8. The distance D, chosen to be 200ft., is used to determine the maximum density.
of cars allowed to wait in a very crowded situation. This is done by adding the number of cars between to paths and dividing it by the total distance. For instance, the number of cars between the East and West street is \((s_1-s_2)+(s_5-s_6)/400\).

Next comes the fuzzy decision process which uses the three step mentioned above (fuzzyification, rule evaluation and defuzzification).

**Step 1**

As before, firstly the inputs and outputs of the design has to be determined. Assuming red light is shown to both North and South streets and distance \(D\) is constant, the inputs of the model consist of:

1) Cycle Time
2) Cars behind red light
3) Cars behind green light

The cars behind the light is the maximum number of cars in the two directions. The corresponding output parameter is the probability of change of the current cycle time. Once this is done, the input and output parameters are divided into overlapping member functions, each function corresponding to different levels. For inputs one and two the levels and their corresponding ranges are zero \((0,1)\), low \((0,7)\), medium \((4,11)\), high \((7,18)\), and chaos \((14,20)\). For input 3, the levels are very short \((0,14)\), short \((0,34)\), medium \((14,60)\), long \((33,88)\), very long \((65,100)\), limit \((85,100)\). The levels of output are no \((0)\), probably no \((0.25)\), maybe \((0.5)\), probably yes \((0.75)\), and yes \((1.0)\). Note: For the output, one value (singleton position) is associated to each level instead of a range of values. The corresponding graphs for each of these membership function is drawn in the similar way above.

**Step 2**

The rules, as before are formulated using a series of if-then statements, combined with AND/OR operators. Ex: if cycle time is medium AND Cars Behind Red is low AND Cars Behind Green is medium, then change is Probably Not. With three inputs, each having 5, 5, and 6 membership functions, there are a combination of 150 rules. However using the minimum or maximum criterion some rules are combined to a total of 86.

**Step 3**

This process, also mentioned above converts the fuzzy set output to real crisp value. The method used for this system is center of gravity:

\[
\text{Crisp Output} = \frac{\text{Sum(Membership Degree*Singleton Position)}}{\text{(Membership degree)}}
\]

For example, if the output membership degree, after rule evaluation are:

<table>
<thead>
<tr>
<th>Change Probability</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>0</td>
</tr>
<tr>
<td>Probably Yes</td>
<td>0.6</td>
</tr>
<tr>
<td>Maybe</td>
<td>0.9</td>
</tr>
<tr>
<td>Probably No</td>
<td>0.3</td>
</tr>
<tr>
<td>No</td>
<td>0.1</td>
</tr>
</tbody>
</table>

then the crisp value will be: Crisp Output =

\[0.1 \times 0 + 0.3 \times 0.25 + 0.9 \times 0.5 + 0.6 \times 0.75 + 0 \times 1.00] / 0.1 + 0.3 + 0.9 + 0.6 + 0 = 0.51

**Is Fuzzy Controller better?**

**Testing of the controller**

The fuzzy controller has been tested under seven different kinds of traffic conditions from very heavy traffic to very lean traffic. 35 random chosen car densities were grouped according to different periods of the day representing those traffic conditions.

**Performance evaluation**

The performance of the controller was
compared with that of a conventional criteria used for comparison were number of cars allowed to pass at one time and average waiting time. A performance index which maximizes the traffic flow and reduces the average waiting time was developed. A means of calculating the average waiting time was also developed, however, a detailed calculation of this evaluation is beyond the scope of this article. All three traffic controller types were compared and can be summarized with the following graph of performance index in all seven traffic categories.

Conclusion
The fuzzy controller passed through 31% more cars, with an average waiting time shorter by 5% than the theoretical minimum of the conventional controller. The performance also measure 72% higher. This was expected. However, in comparison with a human expert the fuzzy controller passed through 14% more cars with 14% shorter waiting time and 36% higher performance index. In conclusion, as Man gets hungry in finding new ways of improving our way of life, new, smarter machines must be created. Fuzzy logic provides a simple and efficient way to meet these demands and the future of it is limitless.

References


6. [http://www.csie.ntu.edu.tw/_cjlin/libsom](http://www.csie.ntu.edu.tw/_cjlin/libsom)
