5-2001

Overview of Fuzzy Logic

Mohamed Trabia
University of Nevada, Las Vegas, mbt@me.unlv.edu

Follow this and additional works at: https://digitalscholarship.unlv.edu/me_presentations
Part of the Acoustics, Dynamics, and Controls Commons, Controls and Control Theory Commons, and the Control Theory Commons

Repository Citation
Available at: https://digitalscholarship.unlv.edu/me_presentations/1

This Presentation is brought to you for free and open access by the Mechanical Engineering at Digital Scholarship@UNLV. It has been accepted for inclusion in Mechanical Engineering Faculty Presentations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
Overview of Fuzzy Logic

Mohamed B. Trabia, Professor
Department of Mechanical Engineering
University of Nevada, Las Vegas
Las Vegas, NV 89154-4027
USA
• Dr. Lotfi Zadeh, a professor of mathematics from U.C. Berkeley, proposed the fuzzy theory 1965/1967.

• Dr. Zadeh originally started his research within traditional control theory.
• He was unsatisfied with the failure of the traditional control theory to explain many phenomena such as, why a person can control a complex system that he/she cannot describe mathematically (Driving car is a good example).
**Thesaurus:**
- *Fuzzy*: uncertain / unclear / vague
- *Logic*: reason

**Definition:**
- Fuzzy logic is a mean to transform linguistic experience into mathematical information.

- Fuzziness may be related to possibility as opposed to probability.
<table>
<thead>
<tr>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author</strong></td>
</tr>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td><strong>Publ info</strong></td>
</tr>
<tr>
<td><strong>Author</strong></td>
</tr>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td><strong>Publ info</strong></td>
</tr>
<tr>
<td><strong>Author</strong></td>
</tr>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td><strong>Publ info</strong></td>
</tr>
<tr>
<td><strong>Author</strong></td>
</tr>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td><strong>Publ info</strong></td>
</tr>
<tr>
<td><strong>Edition</strong></td>
</tr>
</tbody>
</table>
Example: Height

<table>
<thead>
<tr>
<th>Height (cm)</th>
<th>Truthfulness</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>short (sure)</td>
</tr>
<tr>
<td>150</td>
<td>short (sure)</td>
</tr>
<tr>
<td>165</td>
<td>short (?) medium (sure) tall (?)</td>
</tr>
<tr>
<td>170</td>
<td>short (no way) medium (sure) tall (?)</td>
</tr>
<tr>
<td>175</td>
<td>short (no way) medium (?) tall (?)</td>
</tr>
<tr>
<td>180</td>
<td>short (no way) medium (no way) tall (sure)</td>
</tr>
</tbody>
</table>

..
Fuzzy Sets (Membership Functions):

- Membership function (fuzzy set) relates the degree of truthfulness $\mu$ (between 0,1) for a linguistic term.

$\mu=0$  No truth / Absolutely false / No possibility
$\mu=1$  Full confidence / Absolutely correct / Sure thing

- The same value of a variable can be represented using more than one membership function (check 165 cm).
- Membership sets can be of any shape (trapezoid, triangle, gaussian, etc….)

![Diagram showing height vs. degree of truthfulness with membership functions for Short, Medium, and Tall categories. ]
Fuzzy Logic Controller

Fuzzy logic can be used to control systems that do not have well-defined models. Fuzzy logic controller is a feedback controller that looks like:

![Diagram of Fuzzy Logic Controller](image)
Fuzzifier

- The fuzzifier uses the fuzzy membership sets to define the truthfulness of a variable as shown before.
Fuzzy Inference Rules

- These rules determine the actions of the controller. Each rule is in the form of:

\[
IF \text{ \{variable #1\} is \text{fuzzy term}\} \\
and \text{\{variable #2\} is \text{fuzzy term}\}} \\
and \text{\{variable #3\} is \text{fuzzy term}\}} \\
and \ldots \\
\ldots \\
\ldots \\
\ldots \\
\ldots \\
\ldots \\
THEN \\
\text{controller input #j} \text{ is \text{fuzzy term}}
\]
Defuzzification:
- Defuzzification starts by assigning a truth value for the control output as follows,

\[ \mu(\text{fuzzy term of control input } #j) = \min(\mu(\text{fuzzy term of sensor output } #1), \mu(\text{fuzzy term of sensor output } #2), \ldots, \mu(\text{fuzzy term of sensor output } #n)) \]
At this stage, we know two things:
1. What fuzzy term(s) control input \( j \) belongs to.
2. What is the truthfulness of each of these terms.

However, the controller input has to be a crisp (definitive) number.

Different processes of defuzzifications can achieve this goal.

We will present here the moment of area method:

\[
x = \frac{\sum_{i=1}^{n} A_i \ast r c g_i}{\sum_{i=1}^{n} A_i}
\]

![Diagram showing fuzzy rule outcomes and controller output.]
Example

Consider the control of a vehicle,

Model:

\[ F(\phi) = m\ddot{x} \]
\[ \dot{x} \geq 0 \]

Objectives of the controller:

- Reach a target point at \( x_d \) meters away from the starting point
- Velocity is equal to zero at the end of the motion
Variables:
Sensor Outputs:
- Distance_to_target, \( d=(x_d-x) \)
- Velocity, \( v=\dot{x} \)

Controller input
- Accelerator/Brake_angle, \( \phi \)
<table>
<thead>
<tr>
<th>Variable</th>
<th>Fuzzy Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d)</td>
<td>close</td>
</tr>
<tr>
<td></td>
<td>medium</td>
</tr>
<tr>
<td></td>
<td>far</td>
</tr>
<tr>
<td>(v)</td>
<td>zero</td>
</tr>
<tr>
<td></td>
<td>slow</td>
</tr>
<tr>
<td></td>
<td>medium</td>
</tr>
<tr>
<td>(\phi)</td>
<td>Negative Medium</td>
</tr>
<tr>
<td></td>
<td>Negative Small</td>
</tr>
<tr>
<td></td>
<td>Zero</td>
</tr>
<tr>
<td></td>
<td>Positive Small</td>
</tr>
<tr>
<td></td>
<td>Positive Medium</td>
</tr>
</tbody>
</table>
Fuzzy membership sets

![Fuzzy membership sets graph](image-url)
Fuzzy rules:

\[
IF \{<\text{Distance\_to\_target}\text{ is }<\text{close}\}\} \text{and} \{<\text{Velocity}\text{ and }<\text{medium}\}\} \text{ THEN } <\text{Accelerator/Brake\_angle}\text{ is }<\text{Negative Medium}\>
\]

All these rules can be combined in one Table as follows,

<table>
<thead>
<tr>
<th>Fuzzy Rules for the Autonomous Vehicle Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance → Velocity ↓</td>
</tr>
<tr>
<td>Z</td>
</tr>
<tr>
<td>S</td>
</tr>
<tr>
<td>M</td>
</tr>
</tbody>
</table>

Note:

C = close  B = big  S = small
M = medium  Z = zero
P = positive  N = negative
A 3D surface graph is shown with the axes labeled as follows:

- X axis: \( d \)
- Y axis: \( v \)
- Z axis: \( \phi \)

The grid settings are:

- X grids: 15
- Y grids: 15

The interface includes options for input and output:

- X (input): \( d \)
- Y (input): \( v \)
- Z (output): \( \phi \)

There are buttons for "Evaluate," "Help," and "Close."
Results