INTRODUCTION TO FUZZY LOGIC

David I. Brubaker
The Huntington Group
Menlo Park, Calif.

David Brubaker is an independent consultant in the field of complex electronic systems, specializing in the design and development of fuzzy logic systems, embedded-processor systems, and systems with real-time constraints. Brubaker holds a BS, an MS, and a PhD in electrical engineering, all from Stanford University. He has 22 years of experience in the design and development of electronic systems, the past 16 as an independent consultant, and has been involved in fuzzy since 1988.
BACKGROUND

Traditional or bilevel (or Boolean or crisp) logic provides for variables with values that are either TRUE or FALSE. Every assertion made and every logic variable defined must be either TRUE or FALSE — no other value is allowed.

In 1964, Lotfi A. Zadeh, a professor of electrical engineering at the University of California, Berkeley (where he still remains today), became increasingly aware of the incongruent relationship between the black/white nature of traditional set theory (the basis of bilevel logic), and the many shades of gray inherent in the real-world. As a result and as part of his research involving the modeling of complex systems, Zadeh expanded the concept of a classically defined set to that of a fuzzy set. Elements no longer needed to be either entirely within or entirely outside a given set—partial membership of an element within a set is also allowed.

Fuzzy logic is based on fuzzy set theory and provides a rigorous framework for representing non-crisp situations. Consider a man who has lost 25% of his hair. The statement the man is bald is 25% true and 75% false. Fuzzy logic allows this degree of being true (and false), and also provides the rules and operations for manipulating fuzzy variables, in much the same way that Boolean logic is used to manipulate bilevel variables.

Fuzzy set theory is a generalization of classical set theory. Similarly, fuzzy logic is a generalization of Boolean logic, and therefore, a fuzzy logic based system can completely represent a crisp logic system. The converse is not true.

PHILOSOPHICAL UNDERPINNINGS

Traditional attempts to model or define complex systems have been in terms of precision. System models, from highest to lowest levels, are expressed precisely. Higher levels in organizational hierarchies exist to distribute complexity into more easily manageable and understandable components, while lower levels represent the detail contained within these components.

An alternate approach is the use of vagueness as a means of system representation. The motivation is two-fold. First, a significant number of systems do not lend themselves to precise, analytic modeling; they are simply too complex or their operating parameters are just too uncertain. Second, when a relatively accurate analytic model can be derived, the workings of the system may be masked, or even lost, in great degrees of over-quantification. By allowing fuzziness in a system model, by intentionally remaining vague, the complexity associated with a precise representation can be sidestepped, while still quite effectively achieving a workable system model.

Human language is wonderfully rich in vagueness, and yet is also a wonderfully rich communication medium. Human behavior is also rich and possibly even dominated by vagueness. Driving a car, a complex process, is handled by most people “by feel”, best

Brubaker
Introduction to Fuzzy Logic
described in vague terms of “speeding up”, “slowing down”, “braking lightly”, and so forth.

Combining the vagueness of natural language with our apparently excellent ability to handle vagueness exhibited by fuzzy logic suggests the approach most often taken to bring fuzzy systems into the real world. This is currently called a linguistic approach, and represents an analytical technique orthogonal to traditional quantitative analysis. Quoting Zadeh: Our main concern has centered on the development of a conceptual framework for what may be called a linguistic approach to the analysis of complex or ill-defined systems and decision processes.

Although a linguistic approach is at odds with the prevailing attitudes in scientific research, it is proving to be a step in the direction of lesser preoccupation with exact quantitative analysis and greater acceptance of the persuasiveness of imprecision in much of human thinking and perception. By accepting this reality, we are likely to make more real progress in the understanding of the behavior of humanistic systems than is possible within the confines of traditional methods.

**Figure 1** - Most current fuzzy systems are structured as a rule-base. Crisp inputs are measured and assigned fuzzy membership values as part of the fuzzification step, which are applied as conditions to the rules in the rule-base. The rules then specify what actions are to be taken, although in fuzzy terms. Fuzzy actions from typically two or more rules are combined and transformed back into executable system outputs.

**STRUCTURE**

Most fuzzy logic systems use a rule-base as their central structure (Figure 1). Rules, typically cast in an if ... then ... syntax, represent system operation, mapping inputs to outputs. Measured crisp input values are “fuzzified”, using membership functions, into fuzzy truth values (or degrees of membership). These are then applied as conditions to the
rules contained in the rule-base, with triggered rules specifying necessary actions, again as fuzzy truth values. These actions are combined and “defuzzified” into crisp, executable system outputs. Where inputs and outputs are continuous (as in control applications), this fuzzify-infer-defuzzify process is performed on an ongoing basis, at regular sampling intervals.

Conceptually this process is similar to the use of a Fast Fourier Transform and its inverse to transform time domain signals into the frequency domain, to process the resulting frequencies, and then to transform the results back into the time domain. The added expense of transforming between time and frequency domains is justified because the system model is easier to understand and to manipulate in terms of frequencies.

Similarly, a fuzzy system “transforms” signals from the “crisp domain” to the “fuzzy domain”, makes decisions based on these fuzzy values and a knowledge of desired system operation cast in fuzzy terms (rules), and then transforms the results back into the crisp domain for execution. The justification is, as with frequency domain processing, that the system model is easier to understand and manipulate in the fuzzy domain than in the crisp domain.

This basic fuzzy rule-based structure can be used in many different types of applications, including control, process control, decision making, scheduling, prediction, and estimation. By allowing for flexibility in the definition of fuzzy logic operators, and especially in how action combination/defuzzification is performed, the breadth of application is even further increased.

FUZZIFICATION

Consider a system input for rotational velocity, assigned the label $\omega$, and for which we expect an operational range $0 \leq \omega \leq 2000$ rpm. Assume that $\omega$ is a measured input value to be used in controlling the drive to a motor.

In a traditional system, the drive control algorithm would in some way include $\omega$ as a variable, with one or more outputs calculated as functions of the measured value of $\omega$, a crisp value.

If the system model is represented in fuzzy terms, we are more interested in the fuzzy or linguistic values of the fuzzy input Rotational_Velocity (Figure 2). These are assigned by the system designer, and are given labels such as Near_Zero, Very_Low, Low, Medium, High, and Very_High. Each of these labels represents a fuzzy set positioned in the operational domain of possible crisp values.

Each set can be considered a fuzzy value the fuzzy variable Rotational_Velocity can take. Each set is defined as a function, called a membership function, with domain over the possible crisp values, and range from 0 to 1.

Brubaker
Introduction to Fuzzy Logic
Actual measured values are crisp (e.g., 1025 rpm). Rules are expressed using fuzzy terms (e.g., “if Rotational_Velocity is High then ...”). Transformation from crisp to fuzzy values is performed by identifying to what degree the crisp value is a member of each of the fuzzy sets (Figure 3). This calculated value is called the degree of membership or truth value, is typically labeled \( \mu \), and as stated previously, ranges in value over \( 0 \leq \mu \leq 1 \). The degrees of membership associated with a given crisp input value are calculated directly from the (membership) functions used to represent the fuzzy sets.

![Figure 2](image.png)

**Figure 2** - When a variable is treated as being fuzzy, its values are fuzzy as well. So although the measured RPM may be a crisp 1025, the system is more interested in whether and to what degree it is High or Very_High. Other fuzzy values are variable Rotational_Velocity may take are Near-Zero, Very_Low, Low, and Medium..

For example, \( \omega = 1025 \text{ rpm} \) results in a degree of membership in the set \( \text{High} \) of 0.69. This is also expressed as

\[
\mu_{\text{High}}(\omega = 1025) = 0.69
\]

The same value also is contained in the fuzzy set \( \text{Very_High} \), with

\[
\mu_{\text{Very_High}}(\omega = 1025) = 0.31
\]

and has zero membership in all other sets. These truth values are now applied as conditions to the rules of the rule-base in what is called the inference step (Figure 3).

**INFERENCE**

In rules, both conditions (antecedents) and actions (consequences) are expressed as fuzzy terms in the form “if (conditions) then (actions)”. An example is:
if Rotational Velocity is High AND Flow Rate Error is Near_Zero then Accelerator Delta is Near_Zero, Brake is Off;

Figure 3 - A measured value of 1025 rpm is found to belong to both High and Very High sets. The degree to which it belongs to each set is determined by solving the respective membership function. Thus, for rpm = 1025, Rotational Velocity is said to have value Very-High with degree of membership 0.31.

The degree to which the conditions are true were calculated as part of the fuzzification step and are applied as inputs to the rule-base. If a rule contains several logically linked conditions, the respective degree of membership values are combined using fuzzy logic operators, for example AND, OR, and NOT; these are conceptually the same as bilevel logic operators, but are defined differently.

Once input degrees of membership have been applied, any rule that has a non-zero value for its condition is said to have been triggered, or to have fired. In a correctly designed fuzzy system, more than one rule will typically fire at a time. The strength of the firing of each individual rule is the degree to which its conditions are true. This same strength is then applied to the actions indicated by the specific rule. Thus if a rule fires weakly, the specified actions will only weakly impact the subsequent system output, while if the rule fires strongly, the system output will be greatly affected.

COMBINATION/DEFUZZIFICATION

Each rule that fires at each system iteration specifies its own action. Each action is fuzzy, represented at this stage by a degree of membership in an output fuzzy set. For each system output two steps remain: a combination of all fuzzy actions into a single fuzzy action, and a transforming the single resulting fuzzy action into a crisp, executable system output.

Several different combination/defuzzification techniques are popular. The one most often
used is called the “center of mass” or “centroid” technique (Figure 4).

![Diagram of membership functions and defuzzification process]

**Figure 4** - If two triggered rules specify output actions of Medium ($\mu=0.35$) and Large ($\mu=0.75$), the centroid method of defuzzification lops the tops of the respective membership functions at the specified $\mu$'s, then finds the center of mass of the resulting areas. This is the same as if the indicated regions were treated as two metal plates, glued together, and the balancepoint located. In this example, the defuzzification output, that which will be executed, is 62%, and is indicated by the heavier, vertical arrow.

Each rule action specifies an output fuzzy value (for example, “Accelerator_Delta is Near_Zero”, or “Investment_Risk is Minimal”) and a strength (the indicated truth value). In the centroid method, the membership function representing each fuzzy value is topped at the value of the truth value $\mu$. Centroids of the resulting areas are calculated and their center of mass is then determined. The single centroid of all areas defines the crisp, executable output. This process is most easily visualized by considering the individual areas to be rigid sheets that are glued together. The center of mass of the resulting collection of sheets identifies the crisp output value to be executed.

**APPLICATIONS**

The basic rule-based structure can be used in many different applications. This section shall discuss three: control, decision making, and prediction modeling.

**Control**

In control, the fuzzy component is placed in a feedback path around the system to be controlled. The operation of such a controller is as follows (Figure 5).

- Controlled system outputs are sensed as crisp inputs to the fuzzy controller, and their (crisp) values are translated into degrees of membership in the (fuzzy) values, represented by input membership functions.
- These fuzzy values and calculated degrees of membership are used as conditions to
the fuzzy rules of the controller. The rules describe the desired system response.

- Those rules with conditions that are at least partially true (that is, are not absolutely false) are said to have “fired”. Each fired rule submits an action, stated as a fuzzy value with an associated degree of membership to be included in the executed output. The fuzzy value is represented as an output membership function.

- All active output (fuzzy) values associated with each single controller output, together with their degrees of membership, are combined and translated back into a single crisp value, to be executed as the fuzzy controller output.

- All controller outputs are applied as inputs to the controlled system, thereby controlling the feedback loop.

\[\text{Figure 5 - A fuzzy, rule based component becomes a feedback controller when controlled system outputs are applied as controller inputs, and controller outputs are returned to the controlled system as inputs. Rather than having to mathematically model both controlled and controller systems, the rules of the fuzzy controller "linguistically" describe the desired operations.}\]

**Decision making**

In a decision system, a fuzzy rule-based structure is used to calculate relative strengths of desirability between various alternatives. A fuzzy approach is best when making decisions under uncertainty, that is, when available information is incomplete or inaccurate.

As a simple, single rule example, consider a scheduling decision of when to allocate resources to a project start (Figure 6). The goal and three constraints are:

- **(Goal)** To start as soon as possible.
- **(Constraint 1)** The company president has announced to the board of directors a start date “early next week”; it is important to at least be close.
- **(Constraint 2)** The project team leader has just been rehired after being away from the company for three years. There is an anticipated start-up delay for her to come up to speed.
- **(Constraint 3)** A small team is currently involved in preliminary work. It would be cost effective to not start the larger team until the preliminary results are available, which will occur gradually over the next three weeks.
The "rule" is represented as an assertion:

Start_Date is Goal AND Constraint_1 AND Constraint_2 AND Constraint_3;

**Constraint 1:**
- **Goal:** President's promise
  - "As soon as possible"

**Constraint 2:**
- Key person
  - "up to speed"

**Constraint 3:**
- Preliminary work complete

**Figure 6** - The decision system's output results from a fuzzy AND of goal and constraints. Goal and constraints are represented as fuzzy membership functions, as is the decision. The optimum decision (the one that best satisfies goal and constraints) is at the maximum value of the decision, or Tuesday of the second week.

Fuzzy values, represented by membership functions of time, for the goal and constraints, are defined. A "consensus" is taken to be the fuzzy AND of these four functions, which results in an "optimum" start date.

**Prediction modeling**

In a prediction system, fuzzy logic can be used to accumulate current state information and assign strengths to possible future states.

An example is of an actual production planning model, designed to investigate the relationship between fluctuating sales and profits over twelve week segments (Figure 7). The model is structured as ten internal modules that communicate to each other by passing data. The operation of each module is defined by its rules, stated in fuzzy terms and employing
linguistic values defined as membership functions. An example rule is:
if Price is Reasonable AND Product_Quality is Good AND Service_Level is High
then Customer_Demand_Policy is Planned_Sales;
State variables from any of the modules are available for observation as system outputs. In
addition, as time passes and actual data becomes available, it is used to refine subsequent
predictions. Finally, the model structure allows "rationale" queries of what reasoning went
into a given predicted system state.

Figure 7 - A production planning model. Data, as system inputs or as generated by the
system components, flows between the modules. Operation of the modules is defined by
expert rules stated in fuzzy terms.
Conclusion

When does it make sense to use fuzzy logic in place of classical, bilevel logic? Fuzzy logic has been applied to many different and diverse applications, including categorization of weather patterns and of sea gull behaviors, control of cement kilns, passenger trains, and elevators, scheduling of subway trains and service technicians, and as a prediction mechanism in risk management. Empirically, five general categories have emerged within which a fuzzy logic based system is beneficial, and often even necessary:

1) Complex systems, where an adequate system model is difficult or impossible to define.
2) Human expert controlled systems.
3) Systems with moderately to very complex continuous (or semi-continuous) inputs and outputs, for example PID based control systems.
4) Systems with human observations as control rules and/or inputs.
5) Systems where vagueness is common, for example in economic systems, natural sciences, and behavioral sciences.

As a mathematical foundation, a generalization on the age-old but quite limiting concept of absolute truth, fuzzy logic can be successfully applied across a broad range of disciplines, and has the potential of having as significant an impact on the types of systems developed over the next twenty years as the use of the microprocessor has had over the past twenty.