Fuzzy logic essay

Therefore, fuzzy logic cannot be directly processed on computers but must be emulated by special code. The binary logic of modern computers often falls short when describing the vagueness of the real world. Fuzzy logic offers more graceful alternatives. Computers do not reason as brains do. Computers “ reason” when they manipulate precise facts that have been reduced to strings of zeros and ones and statements that are either true or false. The human brain can reason with vague assertions or claims that involve uncertainties or value judgments: The air is cool,” or “ That speed is fast” or “ She is young. Unlike computers, humans have common sense that enables them to reason in a world where things are only partially true. Fuzzy logic is a branch of machine intelligence that helps computers paint gray, commonsense pictures of an uncertain world. Logicians in the 1920s first broached its key concept: everything is a matter of degree. ? Fuzzy logic manipulates such vague concepts as “ warm” or “ still dirty” and so helps engineers to build air conditioners, washing machines and other devices that judge how fast they should operate or shift from one setting to another even when the criteria for making those changes are hard to define.

When mathematicians lack specific algorithms that dictate how a system should respond to inputs, fuzzy logic can control or describe the system by using “ commonsense” rules that refer to indefinite quantities. No known mathematical model can back up a truck-and-trailer rig from a parking lot to a loading dock when the vehicle starts from a random spot. Both humans and fuzzy systems can perform this nonlinear guidance task by using practical but imprecise rules such as “ If the trailer turns a little to the left, then turn it a little to the right. ” Fuzzy systems often glean their rules from experts.

When no expert gives the rules, adaptive fuzzy systems learn the rules by observing how people regulate real systems. Fuzzy logic is derived from fuzzy set theory dealing with reasoning that is approximate rather than precisely deduced from classical predicate logic. Fuzzy logic relies on a set of defined rules to be determined as being true or false. It relies on creating quantitative measurements of already pre-defined binary states. It can be thought of as the application side of fuzzy set theory dealing with well thought out real world expert values for a complex problem.

Fuzzy logic allows for set membership values to range (inclusively) between 0 and 1, and in its linguistic form, imprecise concepts like “ slightly”, “ quite” and “ very”. Specifically, it allows partial membership in a set. It is related to fuzzy sets and possibility theory. Basically, fuzzy logic allows a continuous range of truth values instead of just true and false. In fuzzy, set values strictly between 0 and 1 characterize the fuzzy members. Fuzzy logic is indeed a purely quantitative system and not the qualitative system many seem to assume is its key benefit.

Fuzzy logic is the same as “ imprecise logic”. Fuzzy logic is not any less precise than any other form of logic: it is an organized and mathematical method of handling inherently imprecise concepts. The concept of “ coldness” cannot be expressed in an equation, because although temperature is a quantity, “ coldness” is not. However, people have an idea of what “ cold” is, and agree that there is no sharp cutoff between “ cold” and “ not cold”, where something is “ cold” at N degrees but “ not cold” at N+1 degrees – a concept classical logic cannot easily handle due to the principle of bivalence.

The result has no set answer so it is believed to be a ‘ fuzzy’ answer. Fuzzy logic is a new way of expressing probability. Fuzzy logic and probability are different ways of expressing uncertainty. While both fuzzy logic and probability theory can be used to represent subjective belief, fuzzy set theory uses the concept of fuzzy set membership (i. e. how much a variable is in a set), probability theory uses the concept of subjective probability (i. e. how probable do I think that a variable is in a set).

While this distinction is mostly philosophical, the fuzzy-logic-derived possibility measure is inherently different from the probability measure; hence they are not directly equivalent. WHY USE FL? FL offers several unique features that make it a particularly good choice for many control problems. 1) It is inherently robust since it does not require precise, noise-free inputs and can be programmed to fail safely if a feedback sensor quits or is destroyed. The output control is a smooth control function despite a wide range of input variations.) Since the FL controller processes user-defined rules governing the target control system, it can be modified and tweaked easily to improve or drastically alter system performance. New sensors can easily be incorporated into the system simply by generating appropriate governing rules. 3) FL is not limited to a few feedback inputs and one or two control outputs, nor is it necessary to measure or compute rate-of-change parameters in order for it to be implemented. Any sensor data that provides some indication of a system’s actions and reactions is sufficient.

This allows the sensors to be inexpensive and imprecise thus keeping the overall system cost and complexity low. 4) Because of the rule-based operation, any reasonable number of inputs can be processed (1-8 or more) and numerous outputs (1-4 or more) generated, although defining the rulebase quickly becomes complex if too many inputs and outputs are chosen for a single implementation since rules defining their interrelations must also be defined. It would be better to break the control system into smaller chunks and use several smaller FL controllers distributed on the system, each with more limited responsibilities.) FL can control nonlinear systems that would be difficult or impossible to model mathematically. This opens doors for control systems that would normally be deemed unfeasible for automation. HOW IS FL USED? 1) Define the control objectives and criteria: What am I trying to control? What do I have to do to control the system? What kind of response do I need? What are the possible (probable) system failure modes? 2) Determine the input and output relationships and choose a minimum number of variables for input to the FL engine (typically error and rate-of-change-of-error).) Using the rule-based structure of FL, break the control problem down into a series of IF X AND Y THEN Z rules that define the desired system output response for given system input conditions. The number and complexity of rules depends on the number of input parameters that are to be processed and the number fuzzy variables associated with each parameter. If possible, use at least one variable and its time derivative. Although it is possible to use a single, instantaneous error parameter without knowing its rate of change, this cripples the system's ability to minimize overshoot for a step inputs.) Create FL membership functions that define the meaning (values) of Input/Output terms used in the rules. 5) Create the necessary pre- and post-processing FL routines if implementing in S/W, otherwise program the rules into the FL H/W engine. 6) Test the system, evaluate the results, tune the rules and membership functions, and retest until satisfactory results are obtained. FL does not require precise inputs, is inherently robust, and can process any reasonable number of inputs but system complexity increases rapidly with more inputs and outputs.

Distributed processors would probably be easier to implement. Simple, plain-language IF X AND Y THEN Z rules are used to describe the desired system response in terms of linguistic variables rather than mathematical formulas. The number of these is dependent on the number of inputs, outputs, and the designer’s control response goals. An example of fuzzy reasoning Fuzzy Set Theory defines Fuzzy Operators on Fuzzy Sets. The problem in applying this is that the appropriate Fuzzy Operator may not be known.

For this reason, Fuzzy logic usually uses IF/THEN rules, or constructs that are equivalent, such as fuzzy associative matrices. Rules are usually expressed in the form: IF variable IS set THEN action For example, an extremely simple temperature regulator that uses a fan might look like this: IF temperature IS very cold THEN stop fan IF temperature IS cold THEN turn down fan IF temperature IS normal THEN maintain level IF temperature IS hot THEN speed up fan Notice there is no “ ELSE”. All of the rules are evaluated, because the temperature might be “ cold” and “ normal” at the same time to differing degrees.

The AND, OR, and NOT operators of boolean logic exist in fuzzy logic, usually defined as the minimum, maximum, and complement; when they are defined this way, they are called the Zadeh operators, because they were first defined as such in Zadeh's original papers. So for the fuzzy variables x and y: NOT x = (1 – truth(x)) x AND y = minimum(truth(x), truth(y)) x OR y = maximum(truth(x), truth(y)) There are also other operators, more linguistic in nature, called hedges that can be applied. These are generally adverbs such as " very", or " somewhat", which modify the meaning of a set using a mathematical formula.

In application, the programming language Prolog is well geared to implementing fuzzy logic with its facilities to set up a database of " rules" which are queried to deduct logic. This sort of programming is known as logic programming. Once fuzzy relations are defined, it is possible to develop fuzzy relational databases. The first fuzzy relational database, FRDB, appeared in Maria Zemankova's dissertation. After, some other models arose like the Buckles-Petry model, the Prade-Testemale Model, the Umano-Fukami model or the GEFRED model by J. M. Medina, M. A. Vila et al.

In the context of fuzzy databases, some fuzzy querying languages have been defined, highlighting the SQLf by P. Bosc et al. and the FSQL by J. Galindo et al. These languages define some structures in order to include fuzzy aspects in the SQL statements, like fuzzy conditions, fuzzy comparators, fuzzy constants, fuzzy constraints, fuzzy thresholds, linguistic labels and so on. Examples where fuzzy logic is used It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information.

FL’s approach to control problems mimics how a person would make decisions, only much faster. Fuzzy technology is also resurfacing in information technology, where it provides decision support and expert systems with powerful reasoning capabilities bound by a minimum of rules. It also proved to be an excellent tool in building memory caches, and hard disk controllers, as well as compression algorithms for speech and video. Also, telecom applications such as echo cancellation, network routing, and speech recognition benefit from fuzzy logic.

Automobile and other vehicle subsystems, such as automatic transmissions, ABS and cruise control (e. g. Tokyo monorail) Air conditioners The Massive engine used in the Lord of the Rings films, which helped show huge scale armies create random, yet orderly movements Cameras Digital image processing, such as edge detection Rice cookers Dishwashers Elevators Washing machines and other home appliances Video game artificial intelligence Language filters on message boards and chat rooms for filtering out offensive text Pattern recognition in Remote Sensing

Fuzzy logic has also been incorporated into some microcontrollers and microprocessors A more sophisticated practical example is the use of fuzzy logic in high-performance error correction to improve information reception over a limited-bandwidth communication link affected by data-corrupting noise using turbo codes. The front-end of a decoder produces a likelihood measure for the value intended by the sender (0 or 1) for each bit in the data stream. The likelihood measures might use a scale of 256 values between extremes of “ certainly 0” and “ certainly 1” FL was conceived as a better method for sorting and handling data but has proven to e a excellent choice for many control system applications since it mimics human control logic. It can be built into anything from small, hand-held products to large computerized process control systems. It uses an imprecise but very descriptive language to deal with input data more like a human operator. It is very robust and forgiving of operator and data input and often works when first implemented with little or no tuning. Problems with Fuzzy logic This criticism is mainly because there are problems with conditional possibility, the fuzzy set theory equivalent of conditional probability. This makes it difficult to perform inference.

However there have not been many studies comparing fuzzy-based systems with probabilistic ones. It’s just a collection of simple interpolations supported by informal verbal rationalizations that attempt to achieve effective control without having to work with differential equations and “ really understand” the system Implementation may not be as straightforward as binary logic is. It is difficult to estimate membership function. There are many ways of interpreting fuzzy rules, combining the outputs of several fuzzy rules and de-fuzzifying them. Also, defining functions using this new “ fuzzy” logic paradigm may be tricky.

Note that there is nothing fuzzy about the way fuzzy logic defines logic functions. The only fuzzy thing about it is its use of linguistic terms which are inherently ambiguous, or as we’d like to call them fuzzy. In fact, fuzzy logic fills ambiguous qualifiers, such as tall, hot, etc. , with crisp mathematical models for machine processing The chief arguments against FL in my view are: a) Exaggerated claims are made for it. The claim that it is a generalization of set theory is simply false, as membership functions are functions, and functions are defined in terms of sets. Thus FL is built on set theory, and is so not a generalization of it.) FL is used for both deterministic purposes and decision-making under uncertainty. For deterministic purposes it does not offer much of an advantage over simple percentages. For decision-making under uncertainty it should give the same answers as decision theory or there should a good reason why not. It does not give the same answers as decision theory. The reason is that the solutions it provides are, in decision theory terms ‘ inadmissible’ (i. e. non-optimal). FL is simply a ‘ quick and dirty’ ad hoc technique. There is a place for ‘ quick and dirty’ techniques in engineering, as long as one knows that that is what one is using.

However, I suspect that many people using FL think they are using a rigorous technique. c) Conventional Popperian philosophy of science lays emphasis on statements which empirically falsifiable. The FL set membership functions are not empirically falsifiable, whereas probability statements (even Bayesian subjective probabilities) are capable of refutation with probability 1 – epsilon, for any positive epsilon To add my own opinion, I don’t think the implementation of fuzzy logic really accomplishes anything that can’t be done with other math.

I think where fuzzy logic wins is the way you look at a problem. Sometimes a more linguistic approach is more appropriate, and in my opinion, proper fuzzy logic (as opposed to the way it is often applied) is all about being able to phrase a problem and its solution in linguistic terms. Then the solution becomes obvious, and should be easy to implement.