Computer Science 773

Robotics and Real-time Control

FUZZY CONTROL

The idea of linearising a non-linear system around some convenient operating point is well established, and an effective strategy for developing a controller in cases where it makes sense to restrict operations to such a single point, and where it can be guaranteed that deviations from the region will not occur. If these conditions are not satisfied, linearisation is less successful, and other strategies must be found.

GAIN SCHEDULING.

An obvious development is to repeat the linearisation calculations for several different operating points, producing different control functions for different regions of operation. The control system can then select the appropriate function by inspecting the system parameters to determine the current operating conditions. This is the basis of the control method given the non-obvious name *gain scheduling*.

The name makes a lot more sense if you approach it as an engineer building an electronic analogue controller for the plant. For a linear system (recall the linearisation), such a controller is built of adders and integrators (and occasionally differentiators) to do the calculations and potentiometers to set the coefficients. For the sort of system we are discussing, a different controller is required, in principle, for each operating region, but there is a simplifying factor. As all the individual controllers are intended to control the same physical plant, it is very unlikely that the control functions will be drastically different. For each region, the true non-linear function is approximated by a linear equation over the range of interest, and the only difference as the region changes is in the constants (for the one-dimensional case, slope and intercept or equivalents) which select the required linear approximation. These constants will work through to the analogue controllers as potentiometer settings which define the gains of the circuits. It is therefore possible to use a single hardware controller by adjusting the potentiometers, or gains, according to the region.

Gain scheduling works in practice, though to achieve very precise control over a wide range of operating conditions it might be necessary to use a very large number of linearisation regions. In theory, it is less attractive, because there is always some sort of discontinuity where the regions meet, and control theory does not handle discontinuities well.

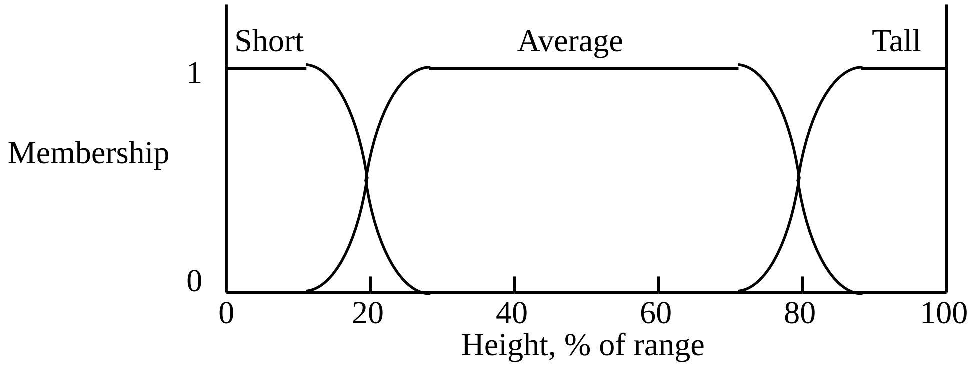
FUZZY LOGIC.

(Skip this if you know about fuzzy logic and fuzzy sets.)

Fuzzy logic was developed by Lotfi Zadeh as a way of dealing with the way we reason about approximate categories. A word such as "tall" does not have a simple meaning, and it's hard to make it precise : a tall building is much taller than a tall person, and even restricting the range to people we find that in practice a tall woman would not necessarily become a tall man by a sex-change operation.

Perhaps, then, the meaning of "tall" is something like "with a height within the upper 20% of the usual range of heights of the group concerned". But does that mean that someone just 19% from the top is unambiguously tall, while someone 21% from the top is clearly of average height ? No, it doesn't. We just don't measure people as accurately as that, and if forced to make a decision there are extensive ranges of height (or whatever) where we hesitate.

This is not a matter of probability. It isn't really sensible to say that someone on the 20% boundary is tall with a probability of 50%. You can say that the probability of someone describing the height as "tall" is 50%, but that isn't the same thing. Zadeh found a way of describing the situation by allowing entities to be part-members of sets : in his fuzzy set terminology, the 20% person might have a membership of 0.5 in the set of tall people and 0.5 in the set of average people. One can represent the situation by drawing a diagram something like this :



The values "Short", "Average", "Tall" which take the place of the arithmetic variable percentage in the fuzzy description are called *linguistic variables*.

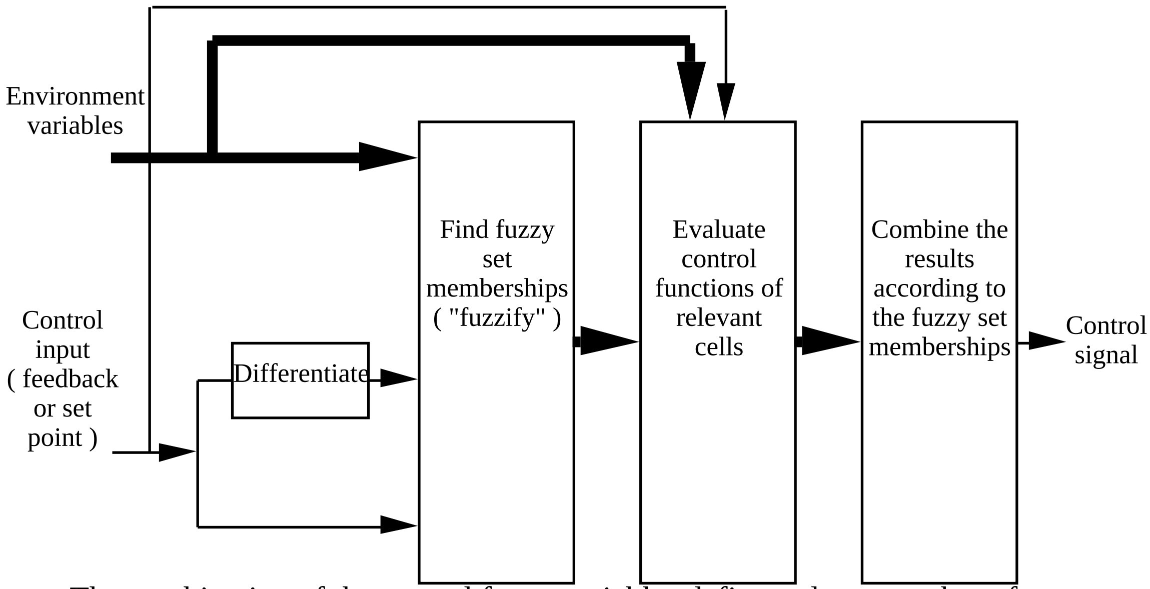
Ordinary predicate logic is strongly tied to the ordinary idea of sets and set membership : "All cows eat grass" can alternatively be stated "for all X, if X is a member of the set of cows, then X eats grass". By taking a little more trouble, the idea of fuzzy sets can be developed into fuzzy logic. With this, one can make statements about tall people, and have a much better chance of applying such statements to specific situations and getting useful results. Alternatively, one can combine statements about tall, average, and short people (that's one statement for each case) into a general statement about people, and handle it systematically. This can get complicated if several statements incorporating fuzzy variables interact, but it still works in principle. For example, consider a statement about { tall | average | short } people banging their heads on { high | medium | low } doorways; there are altogether nine component statements, one for each possible interaction, but one commonly finds simplifications. In this example, it might be reasonable to state that tall people always bang their heads on low doorways, while short people never bang their heads on high doorways.

FUZZY CONTROL.

That is more or less how the ideas of fuzziness are used in control systems. The linguistic variables correspond, more or less, to different ranges of linearisation, but these are now taken to be overlapping according to some appropriate distribution.

The diagram below shows what happens. First, the incoming variables are converted into fuzzy equivalents. This is called *fuzzifying* or *fuzzification* (don't blame me for the vocabulary, which is fairly ridiculous). The incoming information depends on the nature of the feedback. The set point and any relevant environment variables are always necessary; for a closed-loop system, the feedback from the plant is also required. Both the measured variables and their rates of change are often included if the feedback signal is used; recall that the rate of change was useful in the example from the *CONTROL THEORY PRINCIPLES* sheet. Each variable is converted into a number of set memberships, strictly one for each of the linguistic variables which describe its range.

The ranges are usually chosen so that at most two of these memberships are non-zero, but there's no deep reason why that should be so except that it makes less work.



The combination of the several fuzzy variables defines a large number of separate cells, each corresponding to one combination of linguistic variables – recall the people and the doorways. An appropriate control function must be provided for each of these which could possibly be used in practice; following the precedent of the gain scheduling discussion, it might well be sufficient to store a set of constants which adapts a standard function to each case. The control functions are evaluated in every cell for which all the memberships are non-zero. These evaluations will probably use the real values of the variables rather than their fuzzy equivalents.

The results of the evaluation are then combined together using some function which takes into account the memberships of the cells. This is generally some sort of weighted mean. This process is called *defuzzifying* or *defuzzification*. The result is the computed control signal.

It works. The result, given care with the defuzzifying, is a smooth function, though not necessarily easily analysable. Perhaps because of the smoothing, fuzzy control seems to work with fewer linguistic variables than the number of linear ranges which would be needed to achieve comparable performance by a traditional gain scheduling method covering the same ranges of variables.

Fuzzy control can be used for open-loop and closed-loop systems; it's just another control function.

Alan Creak,
March, 1997.