Parabola Volume 27, Issue 2 (1991)

FUZZY LOGIC

James Franklin*

It all started (as we keep saying) with the Greeks. In this case with a certain Eubulides, philosopher-about-town in the Athens of the 4th century BC. Eubulides' specialty was the invention of paradoxes. His winning entry in this department was 'If someone says, "Am I now lying", does he speak truly or falsely?' But it is another idea that leads to the topic of fuzzy logic, his 'Sorites' or 'paradox of the heap'.

The paradox is as follows: how many grains of wheat are necessary to make a 'heap'? Well, if I have a heap, it is still a heap if I add or take away just one grain – it seems ridiculus to say, for example, that 1347 grains is a heap but 1346 grains is not. But then, if I start with a heap, and keep taking away one grain at a time, I eventually get to one grain, which is not a heap. Yet there seems no point at which the pile of grains suddenly changed from being a heap to not being a heap.

As paradoxes go, this one, it must be admitted, is not too impressive. Nevertheless, it needs some care to explain precisely what is wrong with it. The natural answer seems to be this: the concept 'heap' is not an all-or-nothing one. While some things are definitely heaps and some things are definitely not heaps, there are borderline cases, where we would be unwilling to say definitely. Among these borderline cases, some are more nearly definite heaps than others. That is things can be more or less a heap. We say that 'being a heap' is a 'fuzzy' or 'vague' concept. The answer to the paradox of Eubulides is, then, that taking away grains one by one from a heap makes it turn into a non-heap gradually.

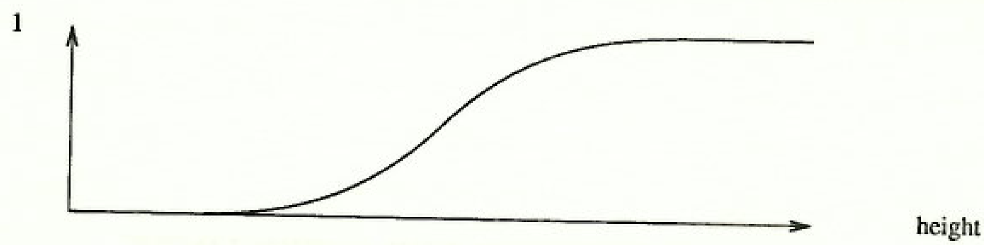
Obviously, a great many of the concepts in natural language are fuzzy in this sense: consider 'red', 'tall', 'intelligent', 'cost-effective' 'justifiable'. Mathematics, and science generally, have mostly avoided using these concepts, and sought to use completely precise language instead. Thus 'prime' is not a fuzzy concept (it is 'crisp', in the jargon). A whole number is either prime or not prime, and there are no borderline cases. Likewise science replaces 'tall' with '177.3cm high': exact measurements are not fuzzy.

---

* Jim is a pure mathematician/philosopher/computer scientist at the University of NSW

Precision has worked very well for science. Nevertheless, there are certain areas where it becomes necessary to treat fuzzy concepts on their own terms. The most important is in the design of user-friendly interfaces for computer software. Ideally, a company should have all the information about itself stored in a management information system, and a manager should be able to type in (or even speak in) a request like "Who have we got that has a low salary and lives fairly close to Sydney?" The system should interpret sensibly "low" and "fairly close", search the information, and print out (or voice synthesise) an intelligible report in natural English.

Present technology is far from being able to do this. The first problem is to decide how to represent fuzzy concepts in a computer. There is an obvious answer: give a number between 0 and 1 to the degree to which borderline case fits the concept. Thus someone definitely not tall would be "tall to degree 0", professional basketballers would be "tall to degree 1", while most of us would be "tall to degree 0.3" or 0.7 or some other degree in between 0 and 1. "Tall" would have a "membership function" something like:



Given such a function, a computer can respond to a question "Give me all the tall people on file" with "The tall ones are $X\&Y$; the ones tall to degree at least 0.9 are $Z, W, \ldots$; do you want ones shorter than this?" Simple. But there are subtleties. For a start, "tall" is context-dependent, so that, for example, the membership functions for men and women are a little different. If the computer is to deal with many kinds of objects, it needs to be told to do something different with "tall tree", and something different again with "tall story". This is the kind of thing that needs a good deal of interdisciplinary research, with cooperation between mathematicians and computer people on the one side and psychologists and linguistics experts on the other. Unfortunately, these people rarely talk to one another.

Where fuzzy logic has had success, and made some people a lot of money, is in the smooth control of Japanese computerised trains and domestic appliances like vacuum cleaners and washing machines. A train, can only be controlled in one way: by accelerating it or decelerating it, by an amount under the driver's, or computer's, control. On the other hand, there are many things to be considered in deciding what degree of acceleration would be safe. They include safety, punctuality, stopping accurately and smoothly at the next station, fuel efficiency, and smoothness of ride (that is, changing the acceleration as little as possible). The idea that the Japanese engineers used was to decide as a fuzzy membership function for each of these objectives, then to evaluate each of these functions for each possible setting of acceleration. The acceleration chosen would then be the one with the **highest** performance on the objective for which it performed **worst** – thus, the chosen acceleration would, so to speak, attend to the objective that needs the most urgent action.

The mathematics is simple. It is powerful. It works.

A HAIRY PROBLEM FOR FUZZY LOGIC

In a town the following facts are true:

i) no two inhabitants have exactly the same number of hairs;

ii) no inhabitant has exactly 215,698 hairs;

iii) there are more inhabitants than there are hairs on the head of any one inhabitant.

What is the larger possible number of inhabitants of this town?

(from the **New Zealand Mathematics Magazine** August 1990)