7-2013

How to Explain (and Overcome) 2% Barrier in Teaching Computer Science: Fuzzy Ideas Can Help

Olga Kosheleva
The University of Texas at El Paso, olgak@utep.edu

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Comments:

Technical Report: UTEP-CS-13-21a

How to Explain (and Overcome) 2% Barrier in Teaching Computer Science: Towards New Applications of Fuzzy Ideas

Olga Kosheleva[1] and Vladik Kreinovich[2]
[1]Department of Teacher Education
[2]Department of Computer Science
University of Texas at El Paso
500 W. University
El Paso, TX 79968, USA
olgak@utep.edu. vladik@utep.edu

Abstract

Computer science educators observed that in the present way of teaching computing, only 2% of students can easily handle computational concepts – and, as a result, only 2% of the students specialize in computer science. With the increasing role of computers in the modern world, and the increasing need for computer-related jobs, this 2% barrier creates a shortage of computer scientists. We notice that the current way of teaching computer science is based on easiness of using two-valued logic, on easiness of dividing all situations, with respect to each property, into three classes: yes, no, and unknown. The fact that the number of people for whom such a division is natural is approximately 2% provides a natural explanation of the 2% barrier – and a natural idea of how to overcome this barrier: to tailor our teaching to students for whom division into more than three classes is much more natural.

This means, in particular, emphasizing fuzzy logic, in which for each property, we divide the objects into several classes corresponding to different degrees with which the given property is satisfied. We also analyze which are the best ways to implement the corresponding fuzzy ideas.

1 2% Barrier in Teaching Computer Science: Formulation of the Problem

2% barrier: a brief description. Computer science educators have observed that only about two out of every 100 students enrolling in introductory programming courses really resonate with the subject and seem to be natural-born computer scientists; see, e.g., [4, 6, 7, 8]. This observation is in good

accordance with the fact that in many universities, computer science students form about 2% of the total number of students, both on the undergraduate and on the graduate level.

Donald E. Knuth, one of the world leading computer scientists and computer science educators, uses this know fact to conclude that "roughly 2% of all people 'think algorithmically', in the sense that they can rapidly reason about algorithmic processes" [6, 7, 8].

Why the 2% barrier is a problem. As the world becomes more and more computerized, the society needs more and more computer scientists – or at least people who can think algorithmically. At present, there is a shortage of computer scientists – and in spite of the numerous efforts by computer science programs, the number of computer science students increases too slowly to cover this shortage.

What we do in this paper. How can we overcome this 2% barrier? A natural way to solve a problem is to analyze it. For the 2% problem, this means that we need to first understand the reasons behind the 2% barrier. In this paper, we provide a possible explanation for this barrier. We then use this explanation to describe a possible path to solving this problem, and discuss ways to follow this path.

2 Why 2% Barrier? A Possible Explanation

What computational thinking means now? In most computer science education programs, computational thinking includes the ability to take imprecise problems and reformulate them in precise terms, in terms which make it easier to explain these problems in a language that a computer can understand. A large part of this thinking is related to the traditional crisp 2-valued logic, whether it is the actual logic with "and", "or", and "not" used in programming languages, whether it is the fact that in the computer, everything has to be represented as a sequence of 0s and 1s.

What crisp computational thinking means in psychological terms. For people to use this logic, they need to be able, with respect to each property, to easily classify objects into three categories:

- objects which satisfy this property,
- objects which do not satisfy this property, and
- objects about which we do not know whether they satisfy the given property or not.

What is known about people's ability to easily classify into classes. Classification of objects into classes is a process well-studied by psychologists. A well-known "7 plus minus 2 law" states that people are most comfortable with classifying into 7 ± 2 classes; see, e.g., [9, 10]. This general psychological law has also been confirmed in our specific area of formalizing expert knowledge: namely, in [1, 3], it was shown that this law explains why in intelligent control, experts normally use ≤ 9 different degrees (such as "small", "medium", etc.) to describe the value of each characteristic (see also [14]).

What the 7 ± 2 law tells us about the number of people who easily classify into 3 classes. What is the precise meaning of the 7 ± 2 law? This law does not mean that for *all* people, the number of classes into which they naturally classify is always between $7-2=5$ and $7+2=9$: there are people for whom the natural number of classes is smaller than 5, and there are people for whom the natural number of classes is larger than 9. A natural way to interpret this law is to treat it the same way as $\pm$ notations in science and engineering are usually interpreted (see, e.g., [12]): namely, to understand this law as saying that the among the human population, the mean value of the natural number of classes is 7 and the standard deviation is 2.

As in many other real-life sitiuations, the number of classes is affected by many different factors; in such situations, it is reasonable to apply the Central Limit Theorem, according to which the joint effect of many independent effects leads (under some reasonable conditions) to a normal distribution; see, e.g., [13]. Thus, it is reasonable to conclude that the natural number of classes is a normally distributed random variable with mean $\mu = 7$ and standard deviation $\sigma = 2$.

Based on these ideas, what is the proportion of people for whom computational thinking is natural? As we have mentioned, computational thinking means that it is natural for a person to divide everything into $X = 3$ classes – or even sometimes into 2 classes ("yes" and "no"). In other words, computational thinking means that the natural number of classes does not exceed 3: $X \leq 3$.

Now that we know the probability distribution for the natural number of classes X, we can estimate the probability $P(X \leq 3)$. A usual way of computing such probabilities for a normal distribution with given mean μ and standard deviation μ is to reduce it to the *standard* normal distribution $Z \stackrel{\text{def}}{=} \dfrac{X-\mu}{\sigma}$, with 0 mean and standard deviation 1. For each real number x, the inequality $X \leq x$ is equivalent to $Z \leq z \stackrel{\text{def}}{=} \dfrac{x-\mu}{\sigma}$. Thus, the desired probability $P(X \leq x)$ is equal to $\Psi(z)$, where $\Psi(z) \stackrel{\text{def}}{=} P(Z \leq z)$ is a (well-tabulated) cumulative distribution function for the standard normal distribution.

In our case, we have $x = 3$, $\mu = 7$, and $\sigma = 2$, so $z = \dfrac{x-\mu}{\sigma} = \dfrac{3-7}{2} = -2$ and thus, $P(X \leq 3) = \Phi(-2) \approx 2.3\%$.

These computations explains the 2% barrier. The above computations show that approximately 2% of people have the ability to easily classify all the objects into three classes (yes, no, and unknown), an ability which is crucial to computational thinking as it is taught now.

3 How Can We Overcome the 2% Barrier? Fuzzy Ideas Can Help

Our explanation of the 2% barrier: reminder. Our explanation seems to indicate the origin of the 2% barrier to increasing the number of computer scientists:

- a traditional way to study computer science is based on emphasizing two-valued logic, while
- for 98% of the people, this logic is *not* very natural.

How to overcome this barrier? From this viewpoint, a reasonable way to overcome the 2% barrier is to do more to tailor computer science education to folks for whom division into more than 3 classes is much more natural. In particular, instead of focusing on the two-valued logic, such tailored approaches should emphasize multiple-valued logic.

Fuzzy ideas can help. In principle, from the mathematical viewpoint, there are many different multiple-valued logics. Which one should we choose?

A good criterion for selection is taking into account that applications are always a good stimulus for learning. It is reasonable to select a multiple-valued logic which has the largest number of practical applications – and this is undoubtedly *fuzzy logic*; see, e.g., [5, 11, 15].

How to overcome the 2% barrier: resulting suggestion. As a result of the above discussion, we arrive at the following conclusion: to overcome the 2% barrier, to satisfy the society's need for computer scientists, to increase the number of computer scientists, we need to introduce fuzzy logic as early as possible into the computer science education.

Discussion. Of course, simply introducing fuzzy logic is not a panacea, we need to go further and reemphasize all the usual binary concepts of computer science in terms of scales consisting of 7 ± 2 elements. Some of this reformulation has already been done in fuzzy research: there are notions of (and results about) fuzzy automata, fuzzy graphs, even fuzzy algorithms. However, all this is just the beginning, the main work is still ahead.

4 Using Fuzzy Techniques in Computer Science Education: A Challenge

At first glance, the situation is straightforward: let us use membership functions. The main idea behind the usual approaches to fuzzy logic is that we describe our degree of confidence in different statements by numbers from the interval $[0, 1]$:

- degree of confidence 1 means that we are absolutely sure that the given statement is true;
- degree of confidence 0 means that we are absolutely sure that the given statement is false;
- intermediate degrees of confidence mean that we have some reasons to believe in the given statement, but we may also have some reasons to believe that the given statement is false.

In this approach, an expert information about a quantity – e.g., that the value of this quantity is approximately 1 – can be described by assigning, to each possible value x of this quantity, a degree $\mu(x)$ to which we are confident that this value is consistent with the given expert information. A function that maps a real number x to the corresponding degree $\mu(x)$ is known as a *membership function*.

Problem: a description by generic membership functions is too complicated. Membership functions are a known successful tool in many applications of fuzzy techniques. Sometimes, the resulting mathematics is somewhat complicated, but no one complains as long as the resulting control works well, the resulting image processing technique leads to a good quality image, etc.

Our objective, however, is to use fuzzy techniques and membership functions in introductory computer science courses. One of the main basic concepts of computer science is the concept of a number. If we want to introduce fuzzy logic into computer science education, we need, in effect, to replace usual (crisp) numbers with fuzzy numbers, i.e., with objects described by membership functions.

The problem is that even operations with simple numbers may sound too complicated to students who are just taking introductory computer science courses. If we use generic functions instead, the situation will become even more complicated – so, instead of helping student learn, we will be leading them to failure.

5 An Empirical Fact that Helps to Apply Fuzzy Techniques: A 4-Parametric Family of Membership Functions Is Usually Sufficient

Do we really need generic membership functions? An empirical observation. The above-mentioned complexity arises if we need to use all kinds of membership functions. But do we really need all kinds of membership functions?

When applications of fuzzy logic started, researchers and practitioners tried their best to come with membership functions which best reflect expert knowledge. As a result, first textbooks containing applications of fuzzy techniques listed sophisticated methods for eliciting membership functions and shows the pictures of the resulting complex functions.

As practitioners gained more experience in applications, they start noticing that after we get a crude picture of the membership function, further elicitation does not lead to a better quality control or a better quality image; see, e.g., [5, 11]. Moreover, it was observed that in most cases, it is sufficient to use *trapezoid* membership functions. Each such function is determined by four parameters $a < b < c < d$:

- it is equal to 0 for all $x \le a$;
- it linearly increases from 0 to 1 on the interval $[a, b]$;
- it is equal to 1 for all values x from the interval $[b, c]$;
- it linearly decreases from 1 to 0 on the interval $[c, d]$; and
- it is again equal to 0 for all $x \ge d$.

In precise terms:

- $\mu(x) = 0$ for $x \le a$;
- $\mu(x) = \dfrac{x - a}{b - a}$ for $a \le x \le b$;
- $\mu(x) = 1$ for $b \le x \le c$;
- $\mu(x) = 1 - \dfrac{x - c}{d - c}$ for $c \le x \le d$; and
- $\mu(x) = 0$ for $x \ge d$.

Each of these functions needs only four parameters to describe, and the corresponding formulas are rather easy. Hence, the resulting description is much less complicated – and thus much more suitable for our purpose – than for generic membership functions.

Comment. Further simplification is not always possible. For example, in some cases, an even simpler 3-parametric class of *triangular* membership functions is sufficient – but not always. In many practical situations, trapezoid functions result in better quality of applications.

Why is the 4-parametric family of membership functions sufficient? The limitation to a 4-parametric family of membership functions is motivated by an empirical success in applications such as control and image processing. The fact that this family is sufficient for many different applications provides a good reason to believe that this same family will turn out to be sufficient useful in other applications as well.

However, to be more confident in its sufficiency in new applications, it would be nice to have some a theoretical explanation for this empirical sufficiency – an explanation that does not depend on a specific application and would therefore be equally applicable to new applications as well. In other words, it would be nice to have a theoretical explanation of why a 4-parametric family of membership functions is often sufficient to describe human opinion?

6 Why Is a 4-Parametric Family of Membership Functions Often Sufficient to Describe Human Perceptions: Towards an Explanation

Why fuzzy techniques: a brief reminder. Why is a 4-parametric family of membership functions often sufficient to describe human perceptions? To answer this question, let us go back from mathematical terms back to the original problem. In fuzzy logic, membership functions are used to describe words from natural language, words that we humans use to describe our knowledge.

In our everyday life, words from natural language is what we routinely use all the time. Whether we want to convey a cooking recipe, describe a stranger's appearance, express our opinion of a faculty candidate, or even describe walking instructions, we normally use words from a natural language. Even if we use numbers in these descriptions, it is usually in the "fuzzy" context: cook for about half an hour, my friend is almost two meters tall, etc. These words form our natural language, and our ability to use them and process them is behind most of our actions and decisions.

Our perceptions are often not very accurate. Our perceptions are often not very accurate. For example, without a measuring instrument, we do not easily distinguish between 1.99 m and 2.00 m, between 29 and 30 minutes, etc. In other words, these perceptions do not carry too many details about the corresponding objects.

This inaccuracy makes sense: we use these terms to make decisions, sometimes decisions that need to be made urgently, and if we carried too many unnecessary details, we would not be able to process all these details fast – after

all, in comparison to computer cells, neural cells in our brains are known to be very slow. In urgent situations, e.g, to avoid a falling tree, it is much better to jump away very as soon as possible than to spend more time computing the optimal trajectory of the jump.

How many details do we need to adequately describe perceptions? How many details are needed? We live in a 3-dimensional world. When we look around, we can characterize all the surrounding objects in the 3-D space by appropriate words, whether these words describe color of a leaf, height of a person, etc. If we use a family with p parameters $v_1, \ldots, v_p$ to describe each word, this means that at each of the 3-D points $x = (x_1, x_2, x_3)$, we store the values $v_1(x), \ldots, v_p(x)$ of the corresponding p parameters v_i.

Towards a precise formulation of the problem. To properly react to a sudden emergency, we need to make sure that an unusual object is immediately detected based on these appropriate observations. How can we describe this in precise terms?

At each moment, our perceived world consists of several continuous zones: for example, when we walk in a countryside, we see a road, a field to the left, a field to the right, a house, one or two clouds, the sky, the Sun. Within each zone, the qualities continuously change with x and thus, the corresponding parameter values $v_i(x)$ also continually change with x.

To make sure that a new object, with values $v'_1, \ldots, v'_p$, can be detected against this background, we must make sure that it is perceptually different from all the points from the zone, i.e., that, in general, there is no x for which $v_1(x) = v'_1$, $v'_2(x) = v'_2$, …, and $v_p(x) = v'_p$.

These equalities form p equations with 3 unknowns x_1, x_2, and x_3. We want to make sure that, in general, this system of equations does not have a solution.

Towards solving the resulting mathematical problem. In general:

- if we have as many equations as unknowns, this system has a unique solution (or at least finitely many solutions);
- if we have fewer equations than unknowns, then, in general, we have infinitely many solutions; and
- if we have more equations than solutions, this means that, in general, we have no solutions.

We want to make sure that the system of p equations with 3 unknowns has, in general, no solutions. Thus, we need to make sure that $p > 3$.

Final explanation. We have concluded that the number of parameters p needed to adequately describe human perceptions should be larger than 3: $p > 3$.

On the other hand, as we have mentioned, the smaller the number of parameters, the better – because we will able to process data faster. Of all the integers p which are larger than three, the smallest is $p = 4$.

We thus have a desired theoretical explanation of why in many cases, our intuitive perceptions are best described by 4-parametric families. The existence of this explanation makes us more confident that the corresponding 4-parametric family of membership functions is sufficient not only in control and image processing, but also in desired educational applications.

7 Discussion

Our conclusion that intuitive perceptions are best described by 4-parametric families is not limited to fuzzy techniques. Let us give two more examples.

Probability theory. In probability theory, one of the known ways to describe a probability distribution with probability density $\rho(x)$ is to list its moments $M_k = \int x^k \cdot \rho(x)\,dx$ of arbitrary orders $k = 1, 2, \ldots$

- Theoretically, we may need moments of 5-th, 6-th, etc. orders.
- In practice, however, the overwhelming majority of applications use only the first four moments; see, e.g., [13].

The above argument explains why the first four moments M_1, M_2, M_3, and M_4 are sufficient – and why fewer moments are not always sufficient.

Physics. In physics, when we have a non-linear dependence, we often expand it in Taylor series

$$f(x) = a + b \cdot (x - x_0) + c \cdot (x - x_0)^2 + d \cdot (x - x_0)^3 + \ldots,$$

where

$$a = f(x_0), b = f'(x_0), c = \frac{1}{2!} \cdot f''(x_0), d = \frac{1}{3!} \cdot f'''(x_0), \ldots$$

and keep only the first few terms in this expansion. In many cases, a 2-parametric linear approximation $f(x) \approx a + b \cdot (x - x_0)$ is good enough. In other cases, we need a 3-parametric quadratic approximation

$$f(x) \approx a + b \cdot (x - x_0) + c \cdot (x - x_0)^2$$

or even a 4-parametric cubic approximation

$$f(x) \approx a + b \cdot (x - x_0) + c \cdot (x - x_0)^2 + d \cdot (x - x_0)^3.$$

- Theoretically, we can envision situations in which higher order terms are also needed.
- However, in the overwhelming majority of applications, it is sufficient to use a cubic approximation – corresponding to a 4-parametric family; see, e.g., [2].

Again, the above argument explains why this is the case.

Acknowledgment

This work was supported in part by the National Science Foundation grants HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and DUE-0926721, by Grants 1 T36 GM078000-01 and 1R43TR000173-01 from the National Institutes of Health, and by a grant on F-transforms from the Office of Naval Research.

References

[1] P. Bonissone and K. Decker, "Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-off Precision and Complexity", In: L. N. Kanal and J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence*, North Holland, Amsterdam, 1986, pp. 217–247.

[2] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.

[3] L. Godo, R. Lopez de Mantaras, C. Sierra, and A. Verdaguer, "MILORD: The Architecture and management of Linguistically expressed Uncertainty", *International Journal of Intelligent Systems*, 1989, Vol. 4, pp. 471–501.

[4] F. Gruenberger, "The role of education in preparing effective computing personnel," in: F. Gruenberger (ed.), *Effective vs. Efficient Computing*, Prentice-Hall, Englewood Cliffs, New Jersey, 1973, pp. 112–120.

[5] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.

[6] D. E. Knuth, "Algorithms in modern mathematics and computer science", *Proceedings of the International Symposium on Algorithms in Modern mathematics and Computer Science, Khiva, Uzbekistan, September 1979*, Springer Lecture Notes in Computer Scinece, 1981, Vol. 122, pp. 82–99.

[7] D. E. Knuth, "Algorithmic thinking and mathematical thinking", *American Mathematical Monthly*, 1985, Vol. 92, pp. 170–181.

[8] D. E. Knuth, *Selected Papers on Computer Science*, Cambridge University Press, 1996.

[9] G. A. Miller, "The magical number seven plus or minus two: some limits on our capacity for processing information", *Psychological Review*, 1956, Vol. 63, pp. 81–97.

[10] P. M. Milner, *Physiological psychology,* Holt, NY, 1970.

[11] H. T. Nguyen and E. A. Walker, *First Course In Fuzzy Logic*, CRC Press, Boca Raton, Florida, 2006.

[12] S. Rabinovich, *Measurement Errors and Uncertainties: Theory and Practice*, American Institute of Physics, New York, 2005.

[13] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC Press, Boca Raton, Florida, 2011.

[14] R. Trejo, V. Kreinovich, I. R. Goodman, J. Martinez, and R. Gonzalez, "A Realistic (Non-Associative) Logic And a Possible Explanations of 7 ± 2 Law", *International Journal of Approximate Reasoning*, 2002, Vol. 29, pp. 235–266.

[15] L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.