Computing under Vagueness

Apostolos Syropoulos

Greek Molecular Computing Group, Xanthi, Greece
asyropoulos@yahoo.com

1 What is Vagueness?

Nowadays, the terms *fuzziness* and *vagueness* are used interchangeably, nevertheless, the term fuzziness is closely associated to fuzzy set theory and its accompanying logic. Since the use of the term vagueness precedes the use of the term fuzziness, I have opted to use this term in the title of this paper.

Vagueness is a very important notion. Unfortunately, most scientific theories, including computability theory, are "ostensibly expressed in terms of objects never encountered in experience" as Max Black [2] has pointed out. For example, there are no spheres in nature, but there are objects that are spheres to a certain degree. Bertrand Russell [8] has defined vagueness as follows:

Definition 1. *Per contra*, a representation is *vague* when the relation of the representing system to the represented system is not one-one, but one-many.

For instance, Russel suggests that a photograph which is so smudged that it might equally represent Brown or Jones or Robinson is vague. In addition, Russell and Black agree that vagueness should not be confused with *generality*, as the former applies in cases where there is lack of specification of boundaries. In addition, vagueness should not be confused with lack-of-information.

Black was the first who tried to formalize vagueness, but he did not managed to propose a full-fledged mathematical theory. This was achieved to a certain degree (!) by Zadeh who introduced fuzzy set theory.

2 Fuzzy Set Theory in a Nutshell

Fuzzy set theory was proposed by Lotfi Askar Zadeh [19] as an extension of ordinary set theory. Zadeh defined fuzzy sets by generalizing the memebrship relationship. In particular, given a universe X, he defined a fuzzy subset of X to characterized by a function $A : X \to \mathrm{I}$, where I is the unit interval. The value $A(x)$ specifies the degree to which some element x belongs to A. Despite its superficial similartiy to probability theory, fuzzy set theory is a different theory. Zadeh [18] has argued that the theories are different facets of vagueness. However, Bart Kosko [4] and other researchers, including this author, have argued that fuzzy set theory is more fundamental than probability theory.

Assume that $A, B : X \to \mathrm{I}$ are two fuzzy subsets of X. Then, $(A \cup B)(x) = \max\{A(x), B(X)\}$ and $(A \cap B)(x) = \min\{A(x), B(X)\}$. Also, if B is the complement of the fuzzy subset A, then $B(x) = 1 - A(x)$. A main deficiency of the theory is that Zadeh *fuzzified* the membership relationship, but he did not fuzzyfied the equality relationship.

In the years following the publication of Zadeh's paper, various researchers proposed and defined various fuzzy structures (e.g., fuzzy algebraic structures, fuzzy topologies, etc.). In particular, the concept of fuzzy languages was introduced by E.T. Lee and Zadeh [5]:

Definition 2. A fuzzy language λ over an alphabet S (i.e., an ordinary set of symbols) is a fuzzy subset of S^*.

If $s \in S^*$, then $\lambda(s)$ is the grade of membership that s is a member of the language.

3 Fuzzy Turing Machines

As expected, Zadeh [20] was the first researcher who mentioned fuzzy Turing machines and fuzzy algorithms or programs. According to Zadeh, a program is fuzzy if it contains fuzzy commands, that is, commands like the following one:

Make y approximately equal to 10, if x is approximately equal to 5.

Zadeh hinted about the way fuzzy programs can be executed, but it was Shi-Kuo Chang [3], Kokichi Tanaka and Masaharu Mizumoto [16], and Eugene S. Santos [10] who made precise the notion of fuzzy programs and their execution. Santos [9] was the first researcher who had given a formal definition of a fuzzy Turing machine:

Definition 3. A fuzzy Turing machine is a septuple $(S, Q, q_i, q_f, \delta, W, \delta_W)$ where:

1. S represents a finite non-empty set of input symbols,

2. Q denotes a finite non-empty set of states such that $S \cap Q = \emptyset$,

3. $q_i, q_f \in Q$ are the symbols designating the initial and final state, respectively,

4. $\delta \subset (Q \times S) \times (Q \times (S \times \{-1, 0, 1\}))$ is the next-move relation,

5. W is the semiring $(W, \wedge, \vee)$,

6. $\delta_W : (Q \times S) \times (Q \times (S \times \{-1, 0, 1\})) \rightarrow W$ is a W-function that assigns a degree of certainty to each machine transition.

Modern versions of this machine use t-norms and t-conorms instead of semirings. In particular, Jiří Wiedermann [17] has defined such a machine and proved that fuzzy languages accepted by these machines with a computable t-norm correspond exactly to the union $\Sigma_1^0 \cup \Pi_1^0$ of recursively enumerable languages and their complements. However, Benjamín Callejas Bedregal and Santiago Figueira [1] have shown that this very important result is not true in general. These researchers have partially shown also that there are no universal fuzzy Turing machines. Also, Yongming Li [6] has shown the nonexistence of an unrestricted universal fuzzy Turing machine.

4 Fuzzy P Systems

P systems [7] is a model of computation inspired by the way living cells function. Basically, a P system is structure that consists of nested, porous membranes that contain indistinguishable copies of objects. Attached to each compartment is a set of rewrite rules, that is, equations that roughly specify how the contents of a compartment should be modified. In particular, such rules may specify that copies of certain objects should be deleted or moved to another compartment or that copies of objects should be introduced from outside or be created out of thin air. Rules are applied in parallel in such a way that only optimal output is generated. When there is no more activity, the result of the computation is equal to the number of (copies of the) objects found in a designated compartment—the output compartment. P systems operate in a massively parallel way while they can interact with their environment.

Fuzzy P systems has been introduced by this author [12]. Typically, a P system is modelled by multisets (see [11] for an overview of the theory of multisets) and multiset rewrite rules that operate on these sets. If we consider that the multisets are actually fuzzy multisets, then we get a version of fuzzy multisets. Since the cardinality of fuzzy multisets is a real number, one can compute real numbers with fuzzy P systems. It is possible to generalize fuzzy P systems by replacing fuzzy multisets with L-fuzzy multisets or even by L-fuzzy hybrid sets, but it is not clear whether this will increase the computational power of the resulting system. However, it seems that these go beyond the Church-Turing barrier [13, 15].

The fuzzy chemical abstract machine [14] is model of computation that is similar to fuzzy P systems. However, the study of this model has just started!

References

[1] Benjamín Callejas Bedregal and Santiago Figueira. On the computing power of fuzzy turing machines. *Fuzzy Sets Systems*, 159:1072–1083, 2008.

[2] Max Black. Vagueness. An Exercise in Logical Analysis. *Philosophy of Science*, 4(4):427–455, 1937.

[3] Shi-Kuo Chang. On the Execution of Fuzzy Programs Using Finite-State Machines. *IEEE Transactions on Computers*, C-21(3):241–253, 1972.

[4] Bart Kosko. Fuzziness vs. Probability. *International Journal of General Systems*, 17(2):211–240, 1990.

[5] E.T. Lee and Lotfi Askar Zadeh. Note on Fuzzy Languages. *Information Sciences*, 1:421–434, 1969.

[6] Yongming Li. Fuzzy turing machines: Variants and universality. *IEEE Transactions on Fuzzy Systems*, 16:1491–1502, 2008.

[7] Gheorghe Păun. *Membrane Computing: An Introduction*. Springer-Verlag, Berlin, 2002.

[8] Bertrand Russell. Vagueness. *Australasian Journal of Philosophy*, 1(2):84–92, 1923.

[9] Eugene S. Santos. Fuzzy Algorithms. *Information and Control*, 17:326–339, 1970.

[10] Eugene S. Santos. Fuzzy and Probablistic Programs. *Information Sciences*, 10:331–345, 1976.

[11] Apostolos Syropoulos. Mathematics of Multisets. In C.S. Calude, Gh. Păun, Gr. Rozenberg, and A. Salomaa, editors, *Multiset Processing*, number 2235 in Lecture Notes in Computer Science, pages 347–358. Springer-Verlag, Berlin, 2001.

[12] Apostolos Syropoulos. Fuzzifying P Systems. *The Computer Journal*, 49(5):619–628, 2006.

[13] Apostolos Syropoulos. *Hypercomputation: Computing Beyond the Church-Turing Barrier*. Springer New York, Inc., Secaucus, NJ, USA, 2008.

[14] Apostolos Syropoulos. Fuzzy chemical abstract machines. *CoRR*, abs/0903.3513, 2009.

[15] Apostolos Syropoulos. On Generalized Fuzzy Multisets and their Use in Computation. To appear in the "Iranian Journal of Fuzzy Systems", 2011.

[16] Kokichi Tanaka and Masaharu Mizumoto. Fuzzy programs and their executions. In Lotfi A. Zadeh, King-Sun Fu, Kokichi Tanaka, and Masamichi Shimura, editors, *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*, pages 41–76. Academic Press, New York, 1975.
[17] Jiří Wiedermann. Characterizing the super-Turing computing power and efficiency of classical fuzzy Turing machines. *Theoretical Computer Science*, 317:61–69, 2004.
[18] Lotfi A. Zadeh. Discussion: Probability Theory and Fuzzy Logic Are Complementary Rather Than Competitive. *Technometrics*, 37(3):271–276, 1995.
[19] Lotfi Askar Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.
[20] Lotfi Askar Zadeh. Fuzzy Algorithms. *Information and Control*, 12:94–102, 1968.