

Jerry M. Mendel¹, *University of Southern California, USA*
 Lotfi A. Zadeh, *University of California at Berkeley, USA*
 Enric Trillas, *European Centre for Soft Computing, SPAIN*
 Ronald Yager, *Iona College, USA*
 Jonathan Lawry, *University of Bristol, UK*
 Hani Hagra, *University of Essex, UK*, and
 Sergio Guadarrama, *European Centre for Soft-Computing, SPAIN*

What Computing with Words Means to Me

Abstract: Computing with words (CWW) means different things to different people. This article is the start of a position paper, written by some of the members of the CIS Fuzzy Systems Technical Committee Task Force on CWW, that answers the question “What does CWW mean to me?”

I. Introduction

In September, 2008, Jerry Mendel sent the following e-mail to the members of the Task Force on Computing With Words^{2,3} (CWW): “CWW means different things to different people, so I was thinking that if we could each provide our own short interpretation of what this means we could put together a magazine article (for *CI Magazine*) about this. To that end, could you each send me a 100–200 word discussion on ‘What CWW means to you?’”

Replies were obtained from Lotfi A. Zadeh (who is the originator of the phrase CWW), Enric Trillas, Ronald Yager, Jonathan Lawry, Hani Hagra, Sergio Guadarrama and Jerry Mendel. Then,

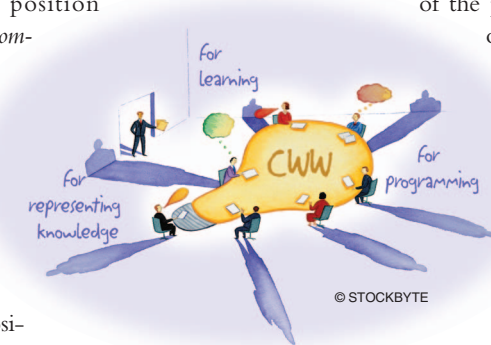
Mendel requested that members of the CWW Task Force comment on each of the replies if they chose to. Comments were obtained from Zadeh, Hagra and Guadarrama. Finally, the comments were replied to only by the authors to which they applied. By the time this process ended, many of the discussions far exceeded the 100–200 word suggestion. Regardless, the results of all of this are quite interesting and rather diverse. They are collected in this article and represent the start of a position paper on what *computing with words* means. Note that a *position paper*⁴ presents an arguable opinion about an issue.

The goal of a position paper is to convince the audience that the opinion is valid and worth listening to, without the need to present completed research work and/or validated results. Hopefully, each person’s argument is supported with evidence to ensure the validity of its claims.

and make rational decisions in an environment of imprecision, uncertainty, incompleteness of information and partiality of truth. And second, the capability to perform a wide variety of physical and mental tasks without any measurements and any computations. In large measure, Computing with Words is inspired by these remarkable capabilities.

Achievement of human level machine intelligence has long been one of the principal objectives of AI. Progress toward achievement of this objective has been and continues to be slow. Anybody who had the experience of struggling with a dumb automated customer service system will readily agree.

In a paper entitled “A New Direction in AI—Toward a Computational Theory of Perceptions,” [*AI Magazine*, Spring 2001, pp. 73–84] I argued that one of the principal reasons for slowness of progress has been AI’s failure to address the problem of reasoning and decision-making with perception-based information. A key idea that was suggested in my paper was that of dealing not with perceptions per se but with their descriptions in a natural language. This idea is a natural derivative of the fact that a natural language is basically a system for



© STOCKBYTE

¹ Chairman of the CWW task force. All of the other authors are members of this task force.

² This is a task force of the Fuzzy Systems Technical Committee of the IEEE Computational Intelligence Society.

³ Different acronyms have been used for Computing With Words—CW, CWW and CWP. The latter is short for Computing With Perceptions. Since the phrase Computing With Words has three words in it, I will use CWW in this article.

II. Lotfi A. Zadeh on CWW

Humans have many remarkable capabilities. Among them there are two that stand out in importance. First, the capability to converse, communicate, reason

⁴ Taken from e-mail “CFP-Position Paper Submission for ICAART 2009;” (International Conference on Agents and AD), V. Rosario, ICAART Secretariat.

description of perceptions. It is at this point that Computing with Words enters the picture. In essence, Computing with Words is a methodology for reasoning, computing and decision-making with information described in natural language. It is this methodology, described in my paper, “From Computing with Numbers to Computing with Words—From Manipulation of Measurements to Manipulation of Perceptions,” [*IEEE Trans. on Circuits and Systems*, vol. 45, no. 1, pp. 105–119, 1999] that is needed for computation with natural language descriptions of perceptions.

Underlying Computing with Words are three principal rationales. First, much of human knowledge is described in natural language. Second, words are less precise than numbers—we use words when we do not know the numbers. In this perspective, Computing with Words may be viewed as a powerful formalism for dealing with imprecise information. And third, precision carries a cost. If there is a tolerance for imprecision, it can be exploited through the use of words in place of numbers. This is the key idea that underlies the machinery of linguistic variables and fuzzy if-then rules—a machinery which is employed in almost all applications of fuzzy logic, especially in the realms of consumer products and industrial systems.

By adding to the armamentarium of AI an effective tool for reasoning and computing with information described in natural language, Computing with Words opens the door to a wide-ranging enlargement of the role of natural languages in science and engineering.

Hani Hagra: I agree with the above definition and comments, however as mentioned above words are imprecise and uncertain and this is why as mentioned above “we use words when we do not know the numbers”. Hence, can we say that when employing fuzzy sets, it will be better to employ higher order fuzzy sets like type-2 fuzzy sets rather than type-1 fuzzy sets where the higher order fuzzy sets will be able to better model the uncertainty rather than type-1 sets which use precise and crisp membership function. For example, for

Computing with Words is a methodology for reasoning, computing and decision-making with information described in natural language.

the word of “warm” temperature, it seems that using a type-1 set will constrain the uncertainty associated with this word across multiple users and even for the same user his concept of warm will change according to the location, season, location, etc. However using type-2 fuzzy sets will be able to better model the faced uncertainties associated with this word through the Footprint of Uncertainty and the third dimension of the type-2 fuzzy set.

Sergio Guadarrama: You have said “a natural language is basically a system for description of perceptions.” It is usually said that a natural language is basically a system for communicating information. But we can communicate descriptions of perceptions, descriptions of actions, descriptions of intentions, descriptions of feelings⁵. [Can you clarify this?]

Lotfi Zadeh: There are many misconceptions about what Computing with Words (CW or CWW) is and what it has to offer. A common misconception is that CW and natural language processing are closely related. In reality, the opposite is the case. More importantly, what is widely unrecognized at this juncture is that moving from computation with numbers to computation with words has the potential for evolving into a basic paradigm shift—a paradigm shift which would open the door to a wide-ranging enlargement of the role of natural languages in scientific theories. The constructive comments of Sergio Guadarrama, Hani Hagra, Jonathan Lawry, Jerry M. Mendel, Enric Trillas and Ronald Yager serve to clarify what CW is and suggest new avenues for exploration of its applications.

In the Appendix, I add to my earlier comment a brief updated exposition of

the basic concepts and ideas that underlie CW. The updated version is a significant improvement over earlier versions.

III. Enric Trillas on CWW

CWW is not yet a well established field of research, but is one in course of development whose name cannot be still the subject of a clear-cut definition. It represents a good opportunity for extending fuzzy logic to copy with some complex problems related with linguistic, contextual, and purpose-oriented meaning.

In my view, Zadeh’s seminal ideas on CWW could lead to results with a big impact in science, technology, and society.

In addition, CWW means a “challenge” for people working in both the theoretic and the technology sides of fuzzy logic. A challenge that eventually if successful, even could help to pose CWW itself as a new experimental science, managing the concepts and tools of computer science, and actually interacting with other disciplines.

I think that a good mastering on current fuzzy logic is essential to work in CWW.

On the theoretic side what right now seems extremely interesting is the design and realization of experiments within natural language related with the meaning’s representation of larger and more complex linguistic expressions than those currently considered in the applications of fuzzy logic. Experiments should pursue two goals: (i) Finding the limits of the current theoretic armamentarium of fuzzy logic, and (ii) Finding and developing new mathematical and algorithmic models, that could allow us to represent what cannot be done by means of, for instance, the standard algebras of fuzzy sets. Such experiments could also help to advance the comprehension of the uncertainty coming from imprecision.

⁵ Bracketed comment is by Mendel, made in order to expedite a reply.

CWW means a “challenge” for people working in both the theoretic and the technology sides of fuzzy logic.

On the technology side I can foresee two overlapping lines corresponding to hardware and software to build up respectively, electronic devices endowed with sensors and soft computing based systems directed to perform actions for social, medical, economical, technological, etc. purposes. All that is to be accomplished with an extensive use of spoken or written natural language and for recognizing, understanding, answering, and, in the end, acting in or interacting with machines, animals, persons, and organizations of several types.

Hani Hagras: Again, I agree with the above comments, and especially the comment “Experiments [should be] made with the goal of finding the limits of the current theoretic armamentarium of fuzzy logic, and new mathematical and algorithmic models [should be developed] that could allow us to represent what cannot be done by means of, for example, the standard algebras of fuzzy sets.” I think also besides exploring the limits of type-1 fuzzy sets, the limits of the higher order fuzzy sets and especially general type-2 fuzzy sets need to be explored as in my opinion they appear very promising to represent words in the CWW paradigm.

Enric Trillas: I took for granted, of course, that type-2 fuzzy sets are in “the current theoretic armamentarium” of fuzzy logic. Anyway, I would like to make an additional and short comment at the respect. A good reason argued in pro of type-2 fuzzy sets is that of reaching some control on the uncertainty of the membership degrees. But, since in my view its objective control deserves again more scientific study, I also wrote “Such experiments could also help to advance the comprehension of the uncertainty coming from imprecision”.

IV. Ronald Yager on CWW

Human beings most naturally express and understand various types of information using language. One objective of Computing with Words (CWW) is to enable the inclusion of human sourced information in the formal computer based decision-making models that are becoming more and more pervasive. Central to CWW is a translation process. This process involves taking linguistically expressed information and translating into a machine manipulative format. The types of information that have to be translated are not restricted to the linguistic values of variables but must also include linguistically expressed information for processing information. The use of OWA provides an example of this translation of processing rules.

Another objective of CWW is to help in the human understanding of the results of information acquisition and information processing. This involves techniques of linguistic summarization and retranslation. Retranslation involves taking the results of the manipulation of formal objects and converting them into linguistic terms understandable to the human. Here we are going in the opposite way of the previous objective. With linguistic summarization we are trying to summarize large sets of data, with the aid of words, in a way that a human can get a global understanding of the content of the data.

Sergio Guadarrama: It is not clear how OWA is an example of translation. [Can you clarify this?]

Ronald Yager: The OWA operator allows one to translate and implement aggregation rules expressed using linguistic quantifiers. For example, in the context of multi-criteria decision-making, consider the linguistically stated requirement “*most criteria should be satisfied.*” Here “most” is an example of a lin-

guistic quantifier. We can represent “most” as a fuzzy subset M on the unit interval where $M(p)$ indicates the degree to which the proportion p satisfies the concept “most”. In my paper “Quantifier guided aggregation using OWA operators” [*Int'l. J. of Intelligent Systems*, vol. 11, pp. 49–73, 1996] I show how this fuzzy set can be used to obtain the weights of an OWA operator that can be used to provide an aggregation function that combines the satisfactions of the individual criteria to determine how well they have met the stated requirement.

V. Mendel on CWW

I begin with Zadeh’s seminal statement that Computing With Words (CWW) is *a methodology in which the objects of computation are words and propositions drawn from a natural language. [It is] inspired by the remarkable human capability to perform a wide variety of physical and mental tasks without any measurements and any computations.* This statement is very general and means that “CWW” is a broad overarching methodology, which makes it very rich because it is open to interpretations and different instantiations.

As an engineer, I begin by asking the question “How can I *solve* a specific (class of) problem(s) using CWW? If I am able to do this, then CWW will have been interpreted/defined (for me) for this class of problems. In this question, the word “solve” is very important because to me it contains two important steps—*formulate* [a solution] and *implement* [that solution]. Unless both steps can be accomplished, then CWW will remain a fantasy for the solution of the specific class of problems.

Although the *formulation* of a CWW solution to a specific class of problems may appear to be quite generic, in that it may be applicable to a broad spectrum of problems, and not just the one that I am presently interested in, it is the *implementation* of the formulation that distinguishes one kind of CWW instantiation from another for different classes of problems.

Let me be specific. I am interested in CWW for assisting people in making subjective judgments (this is a class

of problems). I call my CWW *formulation a Perceptual Computer (Per-C)*. The Per-C has three components: (1) Encoder that maps words into fuzzy set models; (2) CWW Engine⁶ that maps its input fuzzy sets into its output fuzzy sets; and, (3) Decoder that maps the CWW Engine's output fuzzy sets into a linguistic recommendation. This formulation is quite generic, and, as I mentioned above, may be applicable to other classes of problems.

In order to *implement* this three-stage formulation, I: (1) Establish a vocabulary of words that is application dependent; (2) Collect data from a group of subjects about all of the words in the vocabulary; (3) Map that data into a fuzzy set model for each word [the results of Steps (1)–(3) lead to the *Codebook* for the application, and Steps (2) and (3) implement the Encoder]; (4) Establish which CWW Engine will be used for the specific application of the Per-C (e.g., IF-THEN rules, Linguistic Weighted Average, linguistic summarization, etc.); (5) Implement the specific CWW Engine, i.e. develop (or, if it is already available, use) the mathematics for its input–output relationships; (6) Map the fuzzy set outputs of the CWW Engine into a recommendation (e.g., words, a mixture of words and numerical data that explain the words, linguistic rankings, linguistic classes) [the results of this step implement the Decoder, and make very heavy use of similarity, ranking or classification].

For this class of problems, there are, in my opinion, some guidelines for when its solution should be branded “CWW.” Without such guidelines “CWW” will just be a re-labeling of what we are already doing. The following *proposed guidelines* are in the form of three tests all of which I propose must be passed or else the work should not be called CWW.

- 1) *A word must lead to a membership function rather than a membership function leading to a word.*
- 2) *Numbers alone may not activate the CWW engine.*

⁶ Examples of CWW Engines are Linguistic Weighted Averages and IF-THEN Rules.

What has been largely missing from the CWW literature is the connection between data and fuzzy set model. This connection should be made at the start of works about CWW because those works need to incorporate the uncertainties about words.

- 3) *The output from CWW must be at least a word and not just a number.*

A fourth test is optional but is strongly suggested. It is “optional” so as not to exclude much research on CWW that uses type-1 fuzzy sets, even though I strongly believe that this test should also be a requirement for CWW.

- 4) *Because words mean different things to different people, they should be modeled using at least interval type-2 fuzzy sets.*

These tests are easy to apply to any paper that uses CWW in its title or claims to be about CWW. If tests 1–3 are passed, then using the phrase CWW is okay; otherwise, it is not. Failing any one or all of these tests does not diminish the work's importance; it serves to distinguish the work from works about CWW. In this way, a clear distinction will appear between FL as in CWW and FL as not in CWW. Confusion will be reduced and this will benefit the entire FL field.

Hani Hagra: I think the tests mentioned above are a good way forward to formalize the CWW paradigm.

Sergio Guadarrama: Although words could mean different things to different people, I don't know why it is required for them to be modeled by type-2 fuzzy sets.

Jerry Mendel: I believe that *words mean different things to different people*. In my papers “Fuzzy Sets for Words: a New Beginning,” [*Proc. IEEE FUZZ Conf.*, St. Louis, MO, May 26–28, 2003, pp. 37–42] and “Computing with words: Zadeh, Turing, Popper and Occam,” [*IEEE Computational Intelligence Magazine*, vol. 2, pp. 10–17, Nov. 2007], I apply Karl Popper's *Falsificationism* to show that it is scientifically incorrect to model a word using a type-1 fuzzy set, and that an interval type-2 fuzzy set is a scientifically

correct first-order uncertainty model for a word.

I also believe that fuzzy set models for words must be derived from data that are collected from a group of subjects. Such data will collectively contain each subject's (intra-) uncertainty about a word as well as the group's (inter-) uncertainty about the word (*words mean different things to different people*). The intra- and inter-uncertainties about a word can be modeled by an interval type-2 fuzzy set but cannot be modeled by a type-1 fuzzy set.

What has been largely missing from the CWW literature is the connection between data and fuzzy set model. I believe that this connection should be made at the start of works about CWW because those works need to incorporate the uncertainties about words. Since a type-1 fuzzy set is a special case of a type-2 fuzzy set, it seems to me that it is more appropriate to develop new ideas about CWW in the setting of type-2 fuzzy sets than in the setting of type-1 fuzzy sets, since the former can reduce to the latter, but the latter can not be generalized to the former.

VI. Lawry on CWW

Here is my definition partly borrowed from the cover material for my recent book:

- *very short definition:* computing with words = The incorporation of vague linguistic concepts into intelligent computer systems.
- *Slightly longer definition:* Vagueness is central to the flexibility and robustness of natural language descriptions. Vague concepts are robust to the imprecision of our perceptions, while still allowing us to convey useful, and sometimes vital, information.

A critical point for the CWW paradigms is to develop reasoning mechanisms that are able to map inputs words, perceptions and propositions to words, decisions, etc.

Computing with words is therefore motivated by the desire to incorporate this robustness and flexibility into intelligent computer systems. Such a goal, however, requires a formal model of vague concepts that will allow us to quantify and manipulate the uncertainty resulting from their use as a means of passing information between autonomous agents.

Lotfi Zadeh: A word about vagueness. Lawry uses the term “vague” in the same sense as fuzzy. Actually, vague and fuzzy have different meanings. Fuzzy relates to un-sharpness of class boundaries, while vagueness relates to insufficient specificity. As an illustration, “I’ll be back in a few minutes,” is fuzzy but not vague. While “I’ll be back sometime,” is both fuzzy and vague. Similarly, “Meet me at the lobby at about 6pm,” is fuzzy but not vague. Usually, what is vague is fuzzy but not vice-versa.

Jonathan Lawry: For me vagueness refers to uncertainty concerning the conventions for the use of linguistic terms across a population of communicating agents. From this perspective the term “few minutes” is vague since for a given number of minutes x there is uncertainty as to whether the assertion “ x is a few minutes” can be appropriately used (based on linguistic convention). I do agree that there is a distinction between vagueness and imprecision though. So, for example, given the assertion “the exam was between 20 and 40 minutes long” we have imprecision and not vagueness, i.e., there is no concept level uncertainty but rather the information provided is imprecise in that it does not tell us the exact length of the exam.

Hani Hagra: Would vagueness be associated with words and what they mean to different people or would it be

associated with how humans can compute with words and move from inputs words to output actions? Hence, would the framework of type-2 fuzzy systems be more appropriate to handling the uncertainty in CWWs?

Jonathan Lawry: Second order vagueness may be important, however in my view the central issue is that uncertainty concerning the meaning of linguistic terms arises from the distributed way that meaning is communicated through communications between agents across a population.

Sergio Guadarrama: Although vagueness is an important feature of words it is not the only one.

Jonathan Lawry: I agree but I think that the fundamental aim of computing with words is to enable representation and reasoning with granular models, where information granules are represented by linguistic concepts. For this an effective representation of concept vagueness is of central importance since it allows for the development of robust and transparent granular models.

VII. Hagra on CWW

Computing with words relates to developing intelligent systems that are able to receive as input words, perceptions, and propositions drawn from the natural language and can then produce a decision or output based on these words. Computing with words involve different needed components which are:

- Developing the mechanisms that can handle the uncertainties existing with natural language.
- Dealing with problems associated with change of the meaning of words according to context, region and culture.
- Developing reasoning mechanisms that can deal with words, perceptions, and propositions and result in words

that address a problem in a similar way a human would address it.

Computing with words becomes a necessary tool when the available information is perception-based or not precise enough to use numbers, as is the case of most real world applications involving humans. Computing with words adds to conventional modes of computing the capability to compute with interpreted words and propositions drawn from a natural language.

Sergio Guadarrama: How would a human do the third bulleted item above?

Hani Hagra: What is meant in the third bulleted point is to employ social and psychological sciences in order to gain more understanding of how humans can reason with words, perceptions and propositions and try to come up with algorithms that will mimic these reasoning mechanisms. A critical point for the CWW paradigms is to develop reasoning mechanisms that are able to map inputs words, perceptions and propositions to words, decisions, etc.

VIII. Guadarrama on CWW

CWW tries to extend fuzzy logic to understand, describe and design more complex, interactive and flexible computational systems by means of linguistic expressions. An extension that, if successful, could represent an increase in the abstractions used, the variety of the models used and in the kind of aspects captured.

There are three main aspects to be considered when trying to develop CWW:

- 1) *CWW for representing knowledge.* In the same way that over many years Artificial Intelligence before and more recently Soft Computing, especially Fuzzy Logic, have pursued knowledge representation, CWW now aims to provide more flexible mathematical and computational models that can capture and represent different kinds of knowledge that is expressed by humans who use linguistic expressions. In this case, models have to be developed that capture both imprecision and uncertainty.

2) *CWW for learning*. In this case the goal is to learn the meaning of linguistic expressions through interaction. Models have to be developed that can capture the meaning of linguistic expressions given by users in specific contexts, and with specific purposes.

3) *CWW for programming*. One of the initial goals of “high level” programming languages was to leverage the abstraction of the language used to program a computer in order to get closer to human language. In the same way, CWW tries to get closer to those languages by admitting more kinds of linguistic descriptions, e.g., imprecise as well as precise descriptions.

These three aspects are complementary rather than competitive, and should be preferably pursued simultaneously, since they are intertwined and the development in one aspect could help in the development of the others.

Appendix (Zadeh)

CW is rooted in my paper “Outline of a new approach to the analysis of complex systems and decision processes,” [*IEEE Trans. on Systems, Man and Cybernetics*, vol. 3, pp. 28–44, 1973] in which the concepts of linguistic variable and fuzzy if-then rule were introduced; and in my paper “Outline of a computational approach to meaning and knowledge representation based on the concept of a generalized assignment statement,” [*Proc. of Int’l. Seminar on Artificial Intelligence and Man-Machine Systems*, M. Thoma and A. Wyner (eds.), pp. 198–211, Heidelberg: Springer-Verlag, 1986] which introduced the concept of a generalized constraint.

In essence, CW is a system of computation which adds to traditional systems of computation two important capabilities: (a) the capability to precisiate the meaning of words and propositions drawn from natural language; and (b) the capability to reason and compute with precisiated words and propositions.

Conventional systems of computation do not have the capability to deal with linguistic valuations.

As a system of computation, a CW-based model, or simply CW-model, has three principal components, as shown in Fig. 1, and explained next.

- 1) A question, Q , of the form: What is the value of a variable, Y ?
- 2) An information set, $I = (p_1, \dots, p_n)$, where p_i ($i = 1, \dots, n$) are propositions which individually or collectively are carriers of information about the value of Y , i.e., are question-relevant. One or more of the p_i may be drawn from world knowledge. A proposition, p_i , plays the role of an assignment statement which assigns a value, v_i , to a variable, X_i , in p_i . Equivalently, p_i may be viewed as an answer to the question: What is the value of X_i ? X_i and v_i may be explicit or implicit. Basically, an assignment statement constrains the values X_i is allowed to take. To place this in evidence, X_i and v_i are referred to as the constrained variable and the constraining relation, respectively. More concretely, what this implies is that the meaning of a proposition, p , may be represented as a generalized constraint, $X \text{ is } R$, in which X is the constrained variable, R is the constraining relation and r defines the modality of the constraint, i.e., the way in which R constrains X . For example, in the proposition “Robert is tall,” the

constrained variable is Height (Robert), the constraining relation is “tall,” and the modality of the constraint is possibilistic, implying that “tall” is the fuzzy set of possible values of Height (Robert). When v_i is a word or a combination of words, X_i is referred to as a linguistic variable, with v_i being its linguistic value. When it is helpful to stress that p_i assigns a value to a variable, p_i is referred to as a valuation. Correspondently, the information set, I , is referred to as a valuation system, V .

- 3) The third component is an aggregation function, f , which relates Y to the X_i , i.e. $Y = f(X_1, \dots, X_n)$.

Example. Consider a CW-model in which

P_1 : Usually it takes Robert about an hour to get home from work

P_2 : Robert left work at about 5pm

Q: What is the probability that Robert is home at about 6:15pm?

The implicit constrained variable in P_1 is “Travel time,” and the constraining relation is “usually (about 1hour).” The constrained variable in P_2 is “Time of departure,” and the constraining relation is “about 5pm.” The aggregation function is

$$\text{Time of arrival} = \text{Time of departure} + \text{Travel time.}$$

What is important to observe is that, in this and similar examples, the methodology of CW is required to compute the answer. Conventional systems of computation do not have the capability to deal with linguistic valuations.

The principal difference between CW and conventional systems of computation is that CW allows inclusion in the information set, I , of propositions expressed in a natural language,

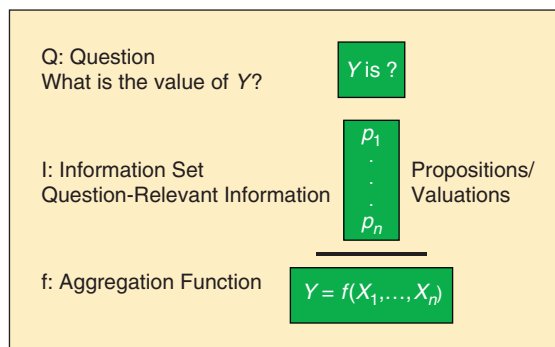


FIGURE 1 Basic structure of CW-based model of computation.

CWW is a broad overarching methodology, which makes it very rich because it is open to interpretations and different instantiations.

that is, linguistic valuations. Legalization of linguistic valuations has three important implications.

- 1) It greatly enhances the capability of computational methodologies to deal with imperfect information, i.e., information that in one or more respects is imprecise, uncertain, incomplete, unreliable, vague or partially true. In realistic settings, such information is the norm rather than exception.
- 2) In cases in which there is a tolerance for imprecision, linguistic valuations serve to exploit the tolerance for imprecision through the use of words in place of numbers.
- 3) Linguistic valuations are close to human reasoning and thus facilitate the design of systems that have a high level of machine intelligence, that is, high level of MIQ (machine IQ).

Words are less precise than numbers, implying that linguistic valuations are less precise than numerical valuations. In large measure, science is driven by a quest for precision. What this seems to suggest is that moving from computation with numbers to computation with words is a step in a wrong direction. If this is the case, what useful purpose is served by employing linguistic valuations? To answer this question it is necessary to take a fresh look at the very basic concept—the concept of precision.

To begin with, what is needed is a differentiation between two meanings of precision—precision in value (ν -precision) and precision in meaning (m -precision). Furthermore, what is needed is a differentiation between human-oriented precisiation, e.g. definition of a word in a dictionary, and machine-oriented precisiation that is mathematically well

defined. These modes of precisiation are referred to as mh -precisiation and mm -precisiation, respectively. Two simple examples are: $X = 5$ is ν -precise and m -precise; and X is between 2 and 8 is ν -imprecise and m -precise. Replacement of $X = 5$ with “ X lies between 2 and 8” is an instance of ν -imprecisiation. In CW, the object of precisiation and the result of precisiation are referred to as the precisend and precisand, respectively. Precisiation is cointensive if the precisand is a close fit to the precisend in some specified sense of “close fit.” More specifically, if precisiation is viewed as modelization, and $M(S)$ is a model of a system S , then $M(S)$ is cointensive if it is a close fit to S .

We are now in a position to address the question: *What does Computing with Words have to offer?* The answer rests on two important tools that are provided by the machinery of fuzzy logic.

- The first tool is a formalism for mm -precisiation of propositions expressed in a natural language through representation of the meaning of a proposition as a generalized constraint of the form $X \text{ is } r$, where as noted earlier X is the constrained variable, R is the constraining relation and r is the modality of the constraint.
- The second tool is a formalism for computing with mm -precisiated propositions through propagation and counter-propagation of generalized constraints. The principal rule governing constraint propagation is the Extension Principle, introduced in my paper “Fuzzy sets,” *Information and Control* [vol. 8, pp. 338–353, 1965], and generalized in my paper “The concept of a linguistic variable and its application to approximate reasoning, Parts I, II and III” [*Inf.*

Sci., vol. 8, pp. 199–249 and pp. 301–357, 1975; *Inf. Sci.*, vol. 9, pp. 43–80, 1975]. In combination, these two tools provide an effective formalism for computation with information described in a natural language. And it is these tools that serve as a basis for legalization of linguistic valuations.

What useful purpose is served by linguistic valuations? Let’s consider two cases.

- Case 1: We either do not have sufficient reliable information to justify assignment of numerical values to variables, or can obtain such information but at an excessively high cost. In effect, ν -imprecision is forced. In this case, CW provides an effective alternative—assignment of linguistic values to variables and computing with mm -precisiated propositions. Using this alternative, we can construct computable models that have a higher degree of cointension than models employing numerical variables. In plain language, what this implies is that realistic imprecise models may be preferable to unrealistic precise models.
- Case 2: We do have precise information; however, precision carries a cost. If there is a tolerance for imprecision, it may be exploited through deliberate ν -imprecisiation, reducing Case 2 to Case 1. What is involved then, is a deliberate ν -imprecisiation followed by mm -precisiation of linguistic values. This artifice is referred to as the *Fuzzy Logic Gambit*, since a deliberate sacrifice is involved. What is gained by the sacrifice is reduction in cost and complexity. This is the principal reason why the Fuzzy Logic Gambit is widely used in the design of consumer products.

These are persuasive rationales for the use of Computing with Words. In coming years, Computing with Words is likely to grow in acceptance, visibility and importance. 