



Computing with Words — A Framework for Human-Computer Interaction

Dan Tamir¹(✉), Shai Neumann², Naphtali Rische³, Abe Kandel⁴,
and Lotfi Zadeh⁵

¹ Texas State University, San Marcos, TX 78666, USA
dan.tamir@txstate.edu

² Eastern Florida State College, Cocoa, FL 32922, USA

³ Florida International University, Miami, FL 33199, USA

⁴ University of South Florida, Tampa, FL 33620, USA

⁵ University of California Berkeley (Deceased), Berkeley, CA 94720, USA

Abstract. In this paper we explore the possibility of using computation with words (CWW) systems and CWW-based human-computer interface (HCI) and interaction to enable efficient computation and HCI. The application selected to demonstrate the problems and potential solutions is in the context of autonomous driving. The specific problem addressed is of a machine instructed by human word commands to execute the task of parking two manned or unmanned cars in a two-car garage using CWW. We divide the interaction process into two steps: (1) feasibility verification and (2) execution. In order to fulfill the task, we begin with verifications of feasibility in terms of assessing whether the garage is unoccupied, checking general ballpark dimensions, inspecting irregular shapes, and classifying the cars that need to be parked, in terms of size, types of vehicles, ranges of acceptable tolerances needed if the cars are manned or not, and means of collision avoidance. The execution of the autonomous driving part is directed by sensory non-numeric fuzzy information that indicates distances from walls or obstacles. The execution algorithm uses a sequence of driving instructions aimed at using the available space in a simple and efficient way without resorting to elaborate numerical calculations, such as making sure that the car is within 2 inches of the wall. The system and its usability are qualitatively analyzed. The analysis shows that the approach has a potential for reducing computational complexity and improving system usability.

Keywords: Fuzzy logic · Computation with words · Autonomous vehicle · Human computer interface · Usability · Learnability · Operability · Understandability

1 Introduction

Humans have a remarkable capability to reason, compute, and make rational decisions in environmental imprecision, uncertainty, and incompleteness of information. In so doing, humans employ modes of reasoning which are approximate rather than exact.

For example, consider the task of driving a car in a highway and maintaining a “safe distance¹” from other cars. To this end, a computer based system might deploy numerous high resolution sensors for accurately measuring location, speed, velocity, distance from other cars, and routing information as well as general information (e.g., the meaning of speed governing road signs). The sensor data and information are generally represented via numerical data. The machine processes the data; potentially solving differential equations, evaluating the values of mathematical functions and deriving state and change of state conclusions in real time. Clearly, even this task, of maintaining “safe distance” from traffic, which is relatively simple for most humans, requires the autonomous system to obtain large amounts of data, maintain a relatively large database/knowledge-base of general knowledge, and perform complex real time inference. This approach of employing Computation with Numbers (CWN) is very promising and finds numerous applications. Furthermore, the increase in computation capacity and ability to acquire and process enormous amounts of data and information enables advances in Artificial Intelligence (AI), whereby many systems that are currently operated by humans can be replaced by CWN systems. Nevertheless, the efficiency and effectiveness of human beings provides numerous examples where the human approach, which involves computation using natural language reasoning and “calculations,” that is, Computation with Words (CWW), seems to be more simple, efficient, and precise. This, in-turn, might raise an intriguing research question concerning the computational and human computer interaction benefits of using CWW for AI-related tasks.

The goal of our present research is to explore the utility of CWW as a tool for devising systems for AI, automatic control and robotics. There are numerous cases where the CWN system is the most cost effective approach to solving a specific problem of this type. Nevertheless, there are three main reasons (rationales) for preferring CWW systems over CWN systems for specific problems. The rationales are governed by the basic premise that words are less precise than numbers and precision carries a cost. The three rationales are:

Rationale A: Numbers are not known or are too costly to obtain. In this case, the use of words is a necessity.

Rationale B: Words are good enough. Assumed that the numbers are known and/or obtainable, there may be a tolerance for imprecision, which can be expressed by words in place of numbers, aiming at a reduction in cost and achieving simplicity. Use of words is advantageous.

Rationale C: Using CWW might simplify the human-computer interaction (HCI) of interactive components of the AI system.

Consequently, the CWW endeavor is at the forefront of advancing society by expanding the capabilities of computing machines. Its roots can be traced to the work by Babbage and Ada, Turing, Von Neumann, Zadeh, Minsky, Chomsky, and literally thousands of other researchers and research efforts in AI, automatic control, and robotics.

¹ The terms “safe distance” and “too far” have fuzzy connotation. Nevertheless, the CWN engineer might attempt to provide a crisp and accurate definition for this term.

Although the concept of CWW, originated by the late professor L. Zadeh, is not new, the research and development literature on this topic is quite scarce. One of the authors of the present paper has had the privilege and honor of working with Professor Zadeh on a National Science Foundation (NSF) proposal concerning CWW. Several portions of this paper's text were written by Prof. Zadeh in our 2016 NSF proposal, before he passed away. Therefore, the late Professor Zadeh is a co-author of the present paper.

This paper explores the possibility of using CWW-based interface to enable efficient computation and human machine interaction. The application selected to demonstrate the problems and potential solutions is in the context of autonomous driving. The specific problem addressed is of a CWW-based machine instructed by human word commands to execute the task of parking two manned or unmanned cars in a two-car garage, a typical scenario associated with single family homes in America.

We divide the interaction process into two steps: (1) feasibility verification and (2) execution. In order to fulfill the task, we begin with verifications of feasibility in terms of assessing whether the garage is unoccupied, checking general ballpark dimensions, inspecting irregular shapes, and classifying the types of cars that need to be parked, in terms of size (sizes might range from subcompact to full size), types of vehicles (e.g., sedans, SUV, and pickup trucks), ranges of acceptable tolerances needed if the cars are manned or not, and means of collision avoidance.

A simple example of computation in this case is that the result of the expression *subcompact* + *subcompact* should yield $subcompact + subcompact \leq 2$. That is, "adding" the two cars, results in a value of 2 or less, where 2 represents the typical minimum requirements to be classified as a two-car garage. On the other hand, "adding" two full size cars may lead to $fullsize + fullsize \geq 2$ which reflects the possibility of an infeasible case in some two-car garage scenarios but feasible in other situations. A priori, and fuzzy measures in this case may mean a "small-2", a "medium-2" or a "big 2" car garage.

The execution of the autonomous driving part is directed by sensory non-numeric information that indicates distances from walls and obstacles. For example, the sensor information might indicate: being safely away, being close, being very close, and being dangerously close to an obstacle. The execution algorithm uses a sequence of driving instructions aimed at using available space in a simple and efficient way without resorting to elaborate numerical calculations, such as making sure that the car is within 2 inches of the wall. This applies more easily when the situation has symmetries (e.g., two identical cars in a perfectly symmetric garage).

If the vehicles being parked are unmanned, then a simple inverse process can take the cars out of the two-car garage back to points of origin with an additional safety requirement of checking for new obstacles or occupied spaces.

The hypothesis of the research presented in this paper is that, due to the rationalized, above several computational HCI tasks can be addressed more efficiently via CWW than via CWN. This hypothesis is qualitatively addressed using the autonomous parking example.

This paper provides an introduction to CWW. Additionally, the problem of parking stated above is described in detail and analyzed. The main contribution of this paper is that it should serve as an initiator of theoretical and applied research in the field of CWW and its applications in areas such as autonomous vehicles. To the best of our knowledge, there is no published research concentrating on the disruptive concepts of using CWW for efficient computation and HCI in AI/RI applications.

The rest of the paper is organized as follows. Background concepts are defined in Sect. 2. Section 3 includes literature review. Section 4 introduces the CWW-based Autonomous Parking System (CWWAPS). Section 5 provides a qualitative analysis of the CWWAPS effectiveness and efficiency, as well as its usability. Finally, Sect. 6 draws conclusions and proposes further research.

2 Background

In this section we provide background concerning the fuzzy logic based concept of CWW [1–11], System Control Using CWW [11–28], and the CWW Autonomous Parking System Background. The CWW concepts are based on a previously submitted NSF proposal [29].

2.1 Computation with Words

Restrictions

Definitions: One of the basic operations in human cognition is that of assignment of a value, v , to a focal variable X through an assignment statement. The assignment statement is generalized if a collection of values, rather than a singleton, is assigned to X . The collection is referred to as a Generalized Assignment value (GA-value) of X . A **restriction** on X written as $R(X)$ is a statement which places a limitation on the values which X can take. A restriction is crisp if its borders are sharply defined. A restriction is fuzzy if its borders are not sharp.

A **proposition**, p , is a description of a restriction. The **meaning** of p is the restriction which p describes. A **canonical form** of a restriction is an expression of the form $R(X) : X \text{ is_} r R$. In this expression, X is a restricted variable (focal variable), R is a restricting relation (focal relation), and r is an index variable which defines the way in which R restricts X . If X is n -ary variable, then R is n -ary relation. Frequently, the term *restriction* is applied to R , rather than to X . Hence, disambiguation, which is context-dependent, is required. The meaning of a restriction is assigned to its canonical form. Equivalently, the meaning of a restriction is the triple (X, R, r) . Typically, both X and R are described in a natural language. A restriction is referred to as a *precisiated restriction* if X and R are described mathematically.

Example: Assume that $X = \text{Vera's age} = \text{Age}(\text{Vera})$. The statement “Vera is middle-aged” places a limitation on the values which X can assume. Hence, “Vera is middle-aged” is a restriction on X . Another example, Robert is “tall” is a restriction on the variable $\text{Height}(\text{Robert})$. As yet another example, $X = (\text{length of time it takes to drive from Berkeley to the San Francisco Airport (SFO)})$. $R(X) = (\text{usually it takes about$

90 min to drive from Berkeley to SFO). Finally, consider $X = (\text{birthday of Robert})$. The statement “Robert is 41 years old” made on August 22, 2016 limits the values of X to the interval [1974.08.22 to 1975.08.22].

Combination of Restrictions

Restrictions can be combined in various ways. The principal modes of combination are the following:

Conjunction: $X \text{ is } A \wedge X \text{ is } B \rightarrow (X, Y) \text{ is } A \cap B$

Disjunction: $X \text{ is } A \vee X \text{ is } B \rightarrow (X, Y) \text{ is } A \cup B$

Implication: if $X \text{ is } A \text{ then } Y \text{ is } B \rightarrow (X, Y) \subseteq A \times B$ ($A \times B$ is the Cartesian product of A and B).

There are close connections between restrictions, information, and propositions. A restriction on X is information about X . Consequently, it can be asserted that: **Restriction = Information.** A proposition p is a carrier of information. Equivalently, a proposition p is a description of a restriction and, consequently, the meaning of the proposition is the restriction which p describes on the focal variable, which is implicit in the proposition. Hence, **Proposition = Restriction.**

Example: The proposition “Vera is middle-aged” is a restriction on the variable $\text{Age}(\text{Vera})$. The proposition “most Swedes are tall” is a restriction on the variable $X = (\text{fraction of tall Swedes among Swedes})$. In the proposition “Robert is much taller than most of his friends”, the focal variable is the fraction of Robert’s friends in relation to whom Robert is much taller. In this case, the restricting relation is “most”.

Z-numbers and Z-restrictions

In many real-world settings, values of variables are not known with precision and certainty. To deal with such variables, CWW offers two important concepts: The concept of a Z -number and the related concept of a Z -restriction. For clarity, we reintroduce the concept of a Z -number.

Definition: A Z -number is an ordered pair (A, B) in which A is an estimate of the value of X and B is an estimate of the goodness/correctness of A as an estimate of X .

Hence, if X is a variable whose value is not known with precision and certainty, then the value of X can be represented as a Z -number.

If the Probability Distribution function (PDF) of X is known to be $p(u)$ then B may be expressed as $B = \int P(u)\mu_A du$, where μ_A is the membership function of A . If the PDF of X is not known then an estimation of B becomes a matter of perception employing a similarity-based definition of probability and any available information about X [6]. There are several special cases of Z -numbers depending on B . Z is a u -number if $B = \text{usually}$. Z is an h -number if $B = \text{high}$, with high being a fuzzy number which is close to 0.9. Z is a g -number (a good number) if $B = \text{good}$, with good being a fuzzy number which is close to 0.7. Z is an m -number if $B = \text{moderate}$, with moderate being a fuzzy number which is close to 0.5. The default value of B is good . More precisely, if X is a variable whose PDF, P , is known, then B can be equated to the probability that X is A . Concretely,

$B = Prob(X \text{ is } A) = \int_u \mu_A(u) \times P_x(u)du$. Or, more simply, as $B = \mu_A \cdot P_x$, where ‘ \cdot ’ is the inner product operation.

Typically, both A and B are linguistic fuzzy numbers in the sense that they are fuzzy numbers which are described in natural language.

Examples:

$X =$ projected deficit.	$X =$ price of oil in 2020
$A =$ <i>about</i> \$500,000.	$A =$ (<i>about</i> \$60/barrel)
$B =$ <i>good</i> .	$B =$ high
$Z =$ (<i>about</i> \$500,000, <i>good</i>).	$Z =$ (<i>about</i> \$60, <i>high</i>).
$X =$ length of time of driving from Berkeley to SFO.	$X =$ price of oil in 2025
$A =$ <i>about</i> 90 minutes.	$A =$ (<i>about</i> \$50/barrel)
$B =$ <i>usually</i>	$B =$ <i>moderate</i>
$Z =$ (<i>about</i> 90 minutes, <i>usually</i>)	$Z =$ (<i>about</i> \$60, <i>moderate</i>).

Z-numbers can be combined via arithmetical operations, in particular via addition. Operations on Z-numbers are described in detail in [2, 6]. In the following, the focus is on addition.

Let $Z (Z = (A, B))$ be the sum of $Z_1 = (A_1, B_1)$ and $Z_2 = (A_2, B_2)$, where Z_1 and Z_2 are Z-numbers,

By definition, $A = A_1 + B_1$. Let p_1 and p_2 be the PDF of X_1 and X_2 respectively. If the probability distributions of $X_1 + X_1, p_{12}$, is a convolution of p_1 and p_2 , then :

$$p_{12} = \int_u p_1(u) \times p_2(v - u)du \tag{1}$$

Or more briefly,

$$p_{12} = p_1 \cdot p_2 \text{ (inner product due to the fact that both are real).}$$

Hence,

$$p_{12} = \int_u p_{12} \times \mu_a(v - u)du \tag{2}$$

Thus, B can be expressed as $= p_{12} \cdot \mu_A$.

Similar formulas can be derived for multiplication and division [2]. An important special case is one in which $B_1 = B_2 = B$. In this case, an approximate expression of the sum is $Z = (A_1 + B_1, B)$. This equality is approximate rather than exact.

In the proposed research area, Z-numbers and their applications will be an important object of consideration. An important application example is one in which the issue is assessment of the global impact of a major economic event, such as an increase in the minimum wage, increase in interest rate, increase in oil price, etc. In such problems the global impact is represented as a fuzzy number which is the sum of local impacts, Z_1, \dots, Z_n . $Z = Z_1, \dots Z_n \cdot Z_n$.

As an illustration, suppose that the issue of interest is the impact on the profit of McDonald Corporation by a to 10% increase in minimum wage. Assume that the interest focuses on the value of X which is the sum of Z_1, \dots, Z_n , where Z_i is the profit of the McDonald restaurants in region, e.g. Oakland. Z_i is supplied by the manager of region R_i . Thus, the problem is of estimating a large number of Z -numbers.

A special case of addition is one where all the Z -numbers have the same confidence, B . In this case, an approximate formula for addition is: $Z_1 + Z_2 = (A_1 + A_2, B)$.

The formulation of approximate rules for addition, subtraction, multiplication and division of Z -numbers should be considered in the proposed research area. Approximate validity of such rules should be assessed through the use of computer simulations.

2.2 System Control Using CWW

Figure 1 presents a schematic diagram of a CWW-based process. One of the most commonly used methods for implementing the inference is via a fuzzy rule based system [12–18].

In [4], Zadeh points to the relations between the Fuzzy rule-based solutions of the control problems, such as the pendulum stabilization task (see below), and CWW. He shows a fuzzification methodology that reflects human perception that can be described in natural language and provides a set of perception-based linguistic model of inference rules.

Stabilization and Related Applications

Following the observation that humans perform numerous complicated control tasks, including tasks that relate to the control of complicated nonlinear systems, without resorting to the use of complex, complicated and computationally intensive numerical methodologies, Zadeh has introduced a different approach to the problem of system control. Namely, fuzzy logic-based and CWW-based approaches [1–5]. In this context, Yamakawa introduced a fuzzy rule based system for the challenging problem of pendulum stabilization [15].

The inverted pendulum stabilization is an instance of a set of fundamental problems in control theory and robotics. The system can be described by a set of nonlinear equations [15] and it often serves as a benchmark for solutions for the underlying set of problems [16–19]. Several approaches involving numerical computation have been applied to this problem [19, 30]. These methods fall under the CWN paradigm and in general they require a relatively complicated and expensive control and computation systems employing computer control algorithms concerned with extensive data acquisition and processing of numerical solutions to the equations [19].

One approach that can be used to address this problem falls under the paradigm of CWW [1–5]. Indeed, in [5] Zadeh shows the relations between Yamakawa and CWW.

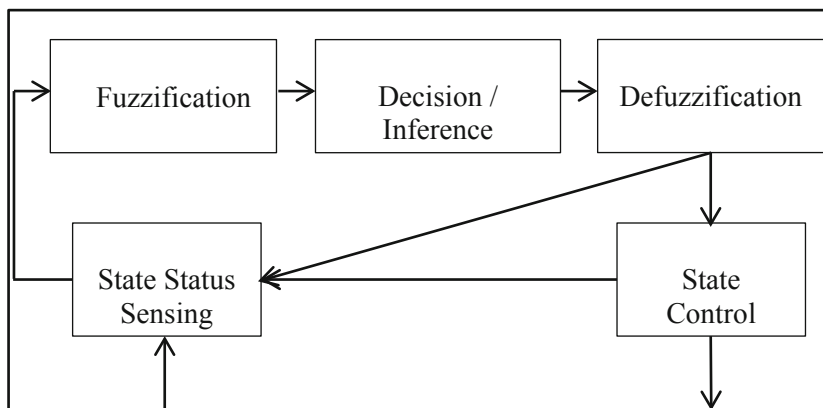


Fig. 1. A Generalized Structure of a Fuzzy Logic Control System.

The process described in Fig. 1 includes a closed loop of (1) sensing external information, (2) fuzzifying the information, (3) using fuzzy logic-based decision and inference system, (4) defuzzifying of the decisions and inferences, and (5) controlling the state according to the inference.

2.3 CWW Autonomous Parking System Background

We assume that the drivers of two cars leave their vehicles one behind the other, both facing the center of a two-car garage, about 30 to 50 feet from the garage door. Identical CWW Autonomous Parking Systems (CWWAPS) are installed in both cars and the systems are synchronized “over the air” so that each car “knows” about the other. Additionally, we assume that the specifications of both vehicles in terms of dimensions, such as length, width, height, ground clearance and potentially other features, are available to the synchronized systems of the cars. The initial conditions place the vehicles fairly close to the targeted parking locations for both cars. Hence, we expect the maneuvers required to be of small to moderate in magnitude. That is, the steering wheel corrections, braking, and velocities, are not of the “race track” type. An exception to such expectation is safety related, when the system is confronted with a “surprise” such as a running toddler, a pet, falling objects, and other discontinuities that call for immediate pause/response in the process.

We assume that the car is equipped with numerous high resolution sensors, [31] and the sensory data coming from those sources is fed into fuzzy sensors that accumulate relevant high resolution data, fuzzify it, and transmit relevant fuzzy sensor input to the control unit. We further assume that the system can dynamically “map” the garage and its vicinity, providing identification and distances from various points of the vehicle to locations of interest. Such input is numerical but the decision making and maneuvering of vehicles are based on CWW. One form of mapping can be obtained via the production of point cloud from radar, Lidar, or camera systems and conversion of the point cloud into a 3D computer graphics mesh.

Fuzzy decisions are made for steering (S), corrective distance decisions (CDD), corrective obstacle decisions (COD), and velocity (V). These CWW decisions rely on fuzzy sensor inputs, vehicle fuzzy measurements of its own states, and fuzzy desired results. The following is a list of fuzzy sensors that are expected to be available for the CWWAPS and their outputs (output is denoted in parenthesis):

- (1) The Fuzzy Heading sensor (H) provides a fuzzy score of the current heading relative to the location of interest on the “map.” The sensor output is one of the following:
 - a. Straight to the selected location (0)
 - b. A bit left of the selected location (-1)
 - c. Left of the selected location (-2)
 - d. Too much left of the selected location (-3)
 - e. A bit right of the selected location (1)
 - f. Right of the selected location (2)
 - g. Too much right of the selected location (3)
- (2) The Fuzzy Distance Sensor #1 (DS1) provides a fuzzy score of current distance relative to a point of interest on the map. The sensor output is one of the following:
 - a. The distance to the selected location is as desired (0)
 - b. The distance to the selected location is a bit too small (-1)
 - c. The distance to the selected location is way too small (-2)
 - d. The distance to the selected location is a bit too large (1)
 - e. The distance to the selected location is way too large (2)
- (3) The Fuzzy Distance Sensor #2 (DS2) provides a fuzzy score for severity of danger associated with vehicle distance to obstacles relevant to projected path. (Higher magnitudes of scores reflect greater danger). The sensor output is one of the following:
 - a. No obstacles along the current/projected path to point of interest or its vicinity (0)
 - b. One or more stationary obstacles far to the left of the current/projected path to the selected location (-1)
 - c. One or more stationary obstacles to the left of the current/projected path to the selected location (-2)
 - d. One or more stationary obstacles a bit to the left of the current/projected path to the selected location (-3)
 - e. Pop-up close moving object on the left side (-4)
 - f. One or more stationary obstacles far to the right of the current/projected path to the selected location (1)
 - g. One or more stationary obstacles to the right of the current/projected path to the selected location (2)
 - h. One or more stationary obstacles a bit to the right of the current/projected path to the selected location (3)
 - i. Pop-up close moving object on the right (4)
 - j. Pop-up close moving object in front of the current/projected path to selected location (∞)

The vehicle control unit is capable of providing the car with the following instructions:

- (1) Steering instructions (S):
 - a. Stay on current heading, no steering correction needed (0)
 - b. Turn a bit to the left (-1)
 - c. Turn left (-2)
 - d. Turn sharply to the left (any CWW calculated value of -3 or smaller is treated as -3)
 - e. Turn a bit to the right (1)
 - f. Turn right (2)
 - g. Turn sharply to the right (any CWW calculated value of 3 or higher is treated as 3)
- (2) Corrective distance decisions, CDD, as one of the following:
 - a. Stop, the distance is as desired (0)
 - b. Move forward to decrease the distance at velocity level 1 [slow] (-1)
 - c. Move forward to decrease distance at velocity level 2 [fast] (any CWW calculated value of -2 or lower is treated as -2) (-2)
 - d. Move backward to increase the distance at velocity level 1 [slow] (1)
 - e. Move backward to increase the distance at velocity level 2 [fast] (any CWW calculated value of 2 or higher is treated as 2) (2)
- (3) Corrective obstacle decision, COD, as one of the following:
 - a. Continue on the current/projected path (0)
 - b. Start preparing an alternative path slightly to the left of the current path if conditions worsen (-1)
 - c. Slow down a bit and be ready to change to an alternative path slightly to the left of the current path (-2)
 - d. Slow down a lot and be ready to change to an alternative path slightly to the left of the current path (-3)
 - e. Emergency stop, evaluate safety level of alternative paths to the left of the current path (-4)
 - f. Emergency stop, evaluate potential move backward and evaluate safety level of an alternative path ($-\infty$)
 - g. Start preparing an alternative path slightly to the right of the current path if conditions worsen (1)
 - h. Slow down a bit and be ready to change to an alternative path slightly to the right of the current path (2)
 - i. Slow down a lot and be ready to change to an alternative path slightly to the right of the current path (3)
 - j. Emergency stop, evaluate safety level of alternative path to the right of current one (4)
- 4) Velocity (V) fuzzy levels are set at two levels forward [level 1 = 1-3 mph (slow), level 2 = 4-6 mph (fast)] and two backward [level 1 = 1-2 mph (slow), level 2 = 3-4 mph (fast)]

Note that the output of fuzzy sensors is used as input to the CWWAPS and, without potential confusion, the same naming (e.g., H for heading) is used as output to describe sensor output and as input to describe a CWWAPS input).

3 Literature Review

Numerous publications address the topic of CWW from the theoretical point of view and/or analyze relevant applications [1–6, 15–19]. Nevertheless, a thorough search for literature that is using the approach presented in this paper did not yield any relevant publications.

Several papers, simultaneously address similar control problems (e.g., “garaging robots”) and CWW [32–35]. Yet, they do not elaborate on the way that the CWW is implemented. Other papers refer to CWW but use different approaches, e.g., genetic fuzzy based control for accomplishing tasks such as garaging [36, 37].

Mitrovic et al. introduce the concept of fictitious fuzzy magnets for robot navigation, obstacle avoidance and garaging [33]. Their system displays some similarities to our proposed system. However, they do not use a CWW-based system.

Numerous papers address the topic of HCI ([38–41]) in CWW-based system and conclude that the affinity between the way that the CWW-based system operates and natural language-oriented HCI significantly improves these systems’ usability [42–45]. These observations are in line with our expectations. Nevertheless, we could not identify papers that specifically address the HCI of CWW-based applications such as the AI applications of interest in this research (e.g., control of autonomous vehicles and robots [42–46]).

4 CWWAPS Procedure

We begin the CWWAPS procedure with the simple case of tasking the system to autonomously park two identical subcompact cars in a symmetric, vacant, open, free of obstacles, two-car garage. Under initial conditions as described in the background section, the CWWAPS system does not face size-related feasibility or assignment issues. Furthermore, it is assumed that the two cars fit into the garage. Hence, the task is reduced to a sequence of maneuvers that safely and smoothly fulfills this assignment. Note that the task of parking a single car in a single car garage is a sub-task of the current task. This is due to the fact that after parking the first car the problem reduces to parking a single car.

Due to the symmetry, the selected side of the garage for either car does not matter and the system chooses a side. To move the first car to the selected side of the garage (say the left side), the system makes a sequence of simultaneous decisions with respect to S, CDD, COD, and V. The output of Fuzzy Sensor H is used as CWWAPS input, where the input is relative to the first location of interest, the corner of the front side of the garage opening of the selected side for vehicle #1. Based on the current heading input, H, and on the desired heading, D, the system provides steering instructions, S,

such as: Stay on current heading, no steering correction needed (0), Turn a bit left (-1), Turn a bit right (1) etc.

Steering instruction (S) are provided based on the CWW equation:

$$S = D - H \quad (3)$$

Note that in a simple case described in this scenario, $D = 0$, but there may be situations with a desired small bias. For example, when the “map” suggests an easily identified object close to where we want to guide the vehicle, D may be 1 or -1 depending on the side of that object relative to actual intended location. This may cause CWW calculated values to be less than -3 or greater than 3, which would lead to steering instructions in the correct direction with the highest fuzzy magnitude resulting in a favorable outcome for that particular step in the process.

Thus, if the system decides to park vehicle #1 on the left side of the garage and the vehicle starting position is as described earlier, facing the center of the garage, then for a sufficiently distant starting point, the heading sensor is likely to state a moderate reading of $H = 1$ or $H = 2$ (A bit to the right or left of the desired location, which is the left corner) and the initial steering is $S = 0 - 1 = -1$, ‘Turn a bit left,’ or $S = 0 - 2 = -2$, ‘Turn left.’ Now, suppose that steering left is too sharp or lasts longer than needed, then H may switch to -1, and the steering switches to $S = 0 - (-1) = 1$, ‘Turn a bit right.’ This is done with decreasing velocities as car gets closer (see below) $H = D$, which is equivalent to: $S = 0$ can be maintained as long as needed.

At the same time that steering instructions are calculated to keep the vehicle heading to the selected location, the fuzzy sensor DS1 provides input regarding the current distance to the first location of interest, i.e., the same corner of the front side of the garage. Based on the current distance DS1 input, and the desired distance (DD), the system makes corrective distance decisions, CDD , such as: Stop, the distance is as desired (0), Move forward to decrease the distance at velocity level 1 [slow] (-1), etc.

Corrective distance decisions (CDD) are made based on the CWW:

$$CDD = DD - DS1 \quad (4)$$

Note, that in a simple case $DD = 0$, but there may be situations where the appropriate value is 1 or -1 with similar consequences as in the case of CWW for S .

Assuming that the process starts as described earlier with vehicle #1 at sufficiently distant starting point, DS1 is likely to state $DS1 = 2$, then the initial CDD is -2, and the vehicle starts moving forward fast in combination with concurrent steering instruction. Obviously, as the vehicle moves according to correct steering, its distance to the location of interest is decreasing and DS1 is changes to $DS1 = 1$, leading to $CDD = -1$ and the vehicle decreases velocity to level 1, the distance keeps decreasing until DS1 inputs the value 0, $CDD = 0$. Consequently, and promptly, the vehicle stops at the end of this step.

In order to address aspects of safety concerns, the fuzzy sensor DS2 input allows the system to make safety level evaluations, avoiding causing harm to humans and animals and preventing damage to the vehicles being parked and to objects outside the vehicles.

Based on desired obstacle level, DO (in this case $DO = 0$), corrective obstacle decisions are made based on the CWW:

$$COD = DO - DS2 \quad (5)$$

As noted earlier, in this case higher magnitude readings of sensor $DS2$ reflect greater danger, calling for more deliberate decisions. This time, however, the most deliberate safety decision means ‘stop moving immediately.’ This is different than the case of $DS1$ where higher magnitude readings leads to decisions to move faster. Additionally, note the potential of multiple obstacles from different sides. This might require taking into account all of the constraints in terms of actual implementation of multiple calculated COD values. Obviously, evolving safety related circumstances may change the feasibility of a solution to the problem.

After vehicle #1 has reached its selected location, facing the selected corner of the garage door opening at an appropriate distance, it is ready to iterate the process with a new selected location, this time the back wall of the garage, with appropriate distances from obstacles, to the left, right, and the front sides. Hence, if vehicle #1 ends the first iteration facing the left corner of the garage, when new selected location becomes effective, the heading is to the left of the new selected location, $H < 0$, and the steering decision turns vehicle #1 to the right, $S > 0$. All other decision components in each iteration function in the same way. To complete the task, vehicle #2 repeats the same iterations with initial selected location being the complement of the choice made initially for vehicle #1.

Maintaining initial conditions of the garage and tasking the system to park two identical vehicles of increasing size require no change to the process. It only requires closer scrutiny of the feasibility conditions. As long as the garage is deemed to be large enough to accommodate the two vehicles, the iterations are followed the same way as described above, with parameter values adjusted for the types of vehicles.

If the conditions are asymmetric in terms of vehicles and the garage, say a subcompact and a full-size SUV and/or the garage contains obstacles (for example, the garage contains items such as equipment or appliances), then the system has to evaluate the adequacy of available space. In this case, it is possible that the parking task is performed with CWW, while the feasibility test requires CWN. Once the feasibility is assured, the only adjustment required to the process as described above, is that the choice of side assignment to vehicles does matter now. Now, the system can follow the simple principle that each vehicle must be assigned to the side that matches the side determined as a part of the feasible solution found earlier. In principle, of course, the smaller vehicle is assigned the smaller space, the larger to the larger space (this is a CWW version that may be used in simpler cases).

5 Analysis

5.1 Additional Scenarios

The CWWAPS illustrated above, can be expanded to address additional scenarios. For example, the initial conditions regarding the way that the drivers leave their cars in relation to one another and in relation to the front side of the garage, have been set in order to simplify the formulation of a feasible path fulfilling the task. It is obvious, that all the maneuvers described above can be reprogrammed for initial conditions where the drivers leave their vehicles with their rear sides facing the garage. This solely requires reorientation of fuzzy sensors, reorientation of steering instructions, and interchanging forward and backward motions.

Furthermore, the initial conditions may be simpler or more complicated than the presented scenario, requiring fewer steps or a greater number of steps in the process. For instance, the drivers may leave the two cars side by side facing available and feasible parking spaces inside the garage. Under this scenario, the problem is significantly simplified and only the second step is required. Moreover, the CWWAPS may be useful in providing a parking solution that does not include the requirement that the drivers are getting in and out of the parked vehicles inside the garage. This scenario allows smaller tolerances between the doors of the parked vehicles. Alternatively, in case that the two drivers leave their vehicles parked side by side outside the garage in a way that they face available but infeasible spaces, the CWWAPS can provide a solution by verifying feasibility after commuting the two vehicles and then performing the second step of our procedure.

5.2 Usability

Tamir et al. have developed an effort-based theory and practice of measuring usability [38–41]. Under this theory, learnability, operability, and understandability are assumed to be inversely proportional to the effort required from the user in accomplishing an interactive task. A simple and useful measure of effort can be the time on the task. Said theory can be used to determine usability requirements, evaluate the usability of systems (including comparative evaluation of “system A vs. System B”), verify their compliance with usability requirements and standards, pinpoint usability issues, and improve usability of system versions. It is quite obvious, and supported by research literature, that an interface that uses a natural language or a formal language that are “close” to natural languages reduces the operator effort and can improve system usability [42–44].

A natural language interface can accompany a CWN-based as well as a CWW-based system. A CWW-based system, however provides the advantage that the system itself operates and is being controlled in a way that is closer to human reasoning, decision making, and operation. This increases the coherence between the system and its user interface and, thus, it simplifies the system design and human-controlled operation of the system.

6 Conclusion and Further Research

In this paper we have demonstrated the possibility of using CWW systems and CWW-based human computer interface (HCI) and interaction to enable efficient computation and HCI. The application selected to demonstrate the problems and potential solutions is in the context of autonomous driving. The specific problem addressed is of a machine instructed by human word commands to execute the task of parking two manned or unmanned cars in a two-car garage using CWW. The system and its usability have been qualitatively analyzed. The analysis shows that the approach has a potential for reducing computational complexity and improving system usability.

In the future we plan to perform simulations, potentially via augmented and virtual reality environments, of the current system and systems addressing other autonomous vehicle tasks. Further, we plan to implement some of these systems. The simulations and implementations will be used to evaluate the feasibility and utility of applying CWW in these AI applications as well assessing the usability of this approach.

References

1. Zadeh, L.A.: Toward a restriction-centered theory of truth and meaning (RCT). In: Tamir, D. E., Rische, N.D., Kandel, A. (eds.) *Fifty Years of Fuzzy Logic and its Applications*. SFSC, vol. 326, pp. 1–24. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19683-1_1
2. Zadeh, L.A.: *Computing With Words: Principal Concepts And Ideas*. Springer, Heidelberg (2012). <https://doi.org/10.1007/978-3-642-27473-2>
3. Zadeh, L.A.: *Computing with Words—Principal Concepts and Ideas Updated*. Computer Science Division, Department of EECS, UC Berkeley. <https://people.eecs.berkeley.edu/~zadeh/presentations%202010/CW-Principal%20Concepts%20and%20Ideasupdated%20Jan%2021%202011.pdf>. Accessed 29 Jan 2019
4. Zadeh, L.A.: Fuzzy logic = computing with words. *IEEE Transact. Fuzzy Syst.* **4**(2), 103–111 (1996)
5. Zadeh, L.A.: From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions. *Int. J. Appl. Math. Comput. Sci.* **12**(3), 307–324 (2002)
6. Aliev, R.A., Alizadeh, R.V., Huseynov, R.A.: The arithmetic of discrete Z-numbers. *Inf. Sci.* **290**(C), 134–155 (2014)
7. Aliev, R.A.: *Fundamentals of The Fuzzy Logic-Based Generalized Theory of Decisions*. Springer, Heidelberg (2013). <https://doi.org/10.1007/978-3-642-34895-2>
8. Zadeh, L.A.: Fuzzy Sets. *Inf. Control* **8**, 338–353 (1965)
9. Klir, G.J.: Fuzzy arithmetic with requisite constraints. *Fuzzy Sets Syst.* **91**(2), 165–175 (1997)
10. Zadeh, L.A.: A note on similarity-based definitions of possibility and probability. *Inf. Sci.* **267**, 334–336 (2014)
11. Tamir, D.E., Rische, N.D., Kandel, A.: Complex fuzzy sets and complex fuzzy logic an overview of theory and applications. In: Tamir, D.E., Rische, N.D., Kandel, A. (eds.) *Fifty Years of Fuzzy Logic and its Applications*. SFSC, vol. 326, pp. 661–681. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19683-1_31
12. Passino, K.M., Yurkovich, S.: *Fuzzy control*. Addison-Wesley, Boston (1998)

13. Lee, C.C.: Fuzzy logic in control systems: fuzzy logic controller I. *IEEE Transact. Syst. Man Cybern.* **20**(2), 404–418 (1990)
14. Lee, C.C.: Fuzzy logic in control systems: fuzzy logic controller II. *IEEE Transact. Syst. Man Cybern.* **20**(2), 419–435 (1990)
15. Yamakawa, T.: A fuzzy inference engine in nonlinear analog mode and its application to a fuzzy logic control. *IEEE Transact. Neural Netw.* **4**(3), 496–522 (1993)
16. Dorf, R.C., Bishop, R.H.: *Modern Control Systems*, 12th edn. Prentice-Hall, Upper Saddle River (2010)
17. Bugeja, M.: Non-linear swing-up and stabilizing control of an inverted pendulum system. *EUROCON Comput. Tool* **2**, 437–441 (2003)
18. Yoshida, K.: Swing-up control of an inverted pendulum by energy-based methods. *Proc. Am. Control Conf.* **6**, 4045–4047 (1999)
19. Ross, M., Gong, Q., Fahroo, F., Kang, W.: Practical stabilization through real-time optimal control. In: *American Control Conference*, pp. 1–6 (2006)
20. Lee, C.C., Berenji, H.R.: An intelligent controller based on approximate reasoning and reinforcement learning. In: *IEEE International Symposium on Intelligent Control*, pp. 200–205 (1989)
21. Berenji, H.R.: A reinforcement learning-based architecture for fuzzy logic control. *Int. J. Approx. Reason.* **6**(2), 267–292 (1992)
22. Sutton, R.S.: *Temporal credit assignment in reinforcement learning*. Ph.D. thesis, Univ. Massachusetts, (1984)
23. Barto, G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transact. Syst. Man Cybern.* **13**(5), 834–846 (1983)
24. Sutton, R.S.: Learning to predict by the methods of temporal differences. *Mach. Learn.* **3**, 9–44 (1988)
25. Barto, G., Sutton R.S., Watkins, C.H.: Sequential decision problems and neural networks. In: Touretzky, A.B. (ed.) *Advances in Neural Information Processing Systems*. Morgan Kaufmann, pp. 686–693 (1990)
26. Michie, D., Chambers, R.A.: Boxes: an experiment in adaptive control. In: Tou, J.T., Wilcox, H. (eds.) *Machine Intelligence*, Edinburgh, vol. 2, pp. 137–152 (1968)
27. Barto, G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transact. Syst. Man Cybern.* **13**, 834–846 (1983)
28. Anderson, A.W.: *Learning and problem solving with multilayer connectionist systems*. Ph. D. thesis, University of Massachusetts (1986)
29. Zadeh, L.A., Tamir D.E.: Proposal Number: 1704840, NSF Program: NSF: CISE: RI: Medium: Collaborative Research: Computing with Words, not funded (2016)
30. Widrow, B., Stearns, S.D.: *Adaptive Signal Processing*. Prentice-Hall, Upper Saddle River (1985)
31. Tesla, Inc.: Tesla Autopilot. <https://www.tesla.com/autopilotlast>. Accessed 29 Jan 2019
32. Milanés, M., Onieva, E., Rastelli, J.P., Godoy, J., Villagra, J.: An approach to driverless vehicles in highways. In: *14th International IEEE Conference on Intelligent Transportation Systems*, pp. 205–210 (2011)
33. Mitrović, S.T., Đurović, Z.M.: Fictitious fuzzy-magnet concept in solving mobile-robot target navigation, obstacle avoidance and garaging problems. In: Topalov, A.V. (ed.) *Recent Advances in Mobile Robotics*, pp. 235–260 (2011)
34. Borrero, G.H., Becker, M., Archila, J.F., Bonito, R.: Fuzzy control strategy for the adjustment of the front steering angle of a 4WSD agricultural mobile robot. In: *7th Colombian Computing Congress*, pp. 1–6 (2012)

35. Mitrović, S.T., Đurović, Z.M.: Fuzzy logic controller for bidirectional garaging of a differential drive mobile robot. *Adv. Robot.* **24**(8–9), 1291–1311 (2010)
36. Vicente, E.O., De Pérez, M.J., Pedro, R.T.: Genetic fuzzy-based steering wheel controller using a mass-produced car. *Int. J. Innov. Comput. Inf. Control ICIC Int.* **8**(5B), 3477–3494 (2009)
37. Nedjah, N., Renato, S.S., Mourelle, S.D.: Customizable hardware design of fuzzy controllers applied to autonomous car driving. *Expert Syst. Appl.* **41**(16), 7046–7060 (2014)
38. Tamir, D.E., Komogortsev, O.J., Mueller, C., Venkata, D.V., LaKowski, G., Jamnagarwala, A.: Detection of software usability deficiencies. In: *Proceedings of HCII 2011* (2011)
39. Komogortsev, O.J., Tamir, D.E., Mueller, C., Camou, J., Holland, C.: Automated eye movement-driven approach for identification of usability issues. In: *Proceedings of HCII 2011* (2011)
40. Tamir, D.E., Komogortsev, O.J., Mueller, C., Venkata, D.V., LaKowski, G.: Detecting software usability deficiencies through pinpoint analysis. *Int. J. Adv. Softw. Syst.* **7**(1,2), 220–229 (2014)
41. Tamir, D.E., Komogortsev, O.J., Mueller, C.: A learning-based framework for evaluating software usability. *ARPN J. Syst. Softw.*, 45–59 (2013)
42. Godfrey, L.B., Gashler, M.S.: A parameterized activation function for learning fuzzy logic operations in deep neural networks. In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (2017)
43. Kouatli, I.: Computing with words: Are we ready to talk to our cars? In: *International Conference on Connected Vehicles and Expo (ICCVE)*, pp. 212–213 (2015)
44. Rubin, S.H., Lee, G.: Human-machine learning for intelligent aircraft systems. In: *International Conference on Autonomous and Intelligent Systems: Autonomous and Intelligent Systems*, pp. 331–342 (2011)
45. Pandey, K.K.: *Design and Analysis of Intelligent Navigational Controller for Mobile Robot*. Master's thesis, Department of Mechanical Engineering, National Institute of Technology, Rourkela, India (2014)
46. Mahapatra, S.: *Analysis and Control of Mobile Robots in Various Environmental Conditions*. Master's thesis, Department of Industrial Design National Institute of Technology, Rourkela, India (2012)