J. of Mult.-Valued Logic & Soft Computing., Vol. X, pp. 1–21
Reprints available directly from the publisher
Photocopying permitted by license only

Computation of Fuzzy Truth Values for The Liar and Related Self-Referential Systems

ATH. KEHAGIAS AND K. VEZERIDES

http://users.auth.gr/ ˜kehagiat
E-mail: kehagiat@auth.gr
E-mail: vezerid@ac.anatolia.edu.gr

Received: June 5, 2005. In Final Form: March 10, 2006.

We study the *Liar paradox* and related systems of self-referential sentences. Specifically, we consider the problem of assigning consistent fuzzy truth values to *systems of self-referential sentences*. We show that this problem can be reduced to the solution of a system of nonlinear equations and we prove that, under certain conditions, a solution (i.e. a consistent truth value assignment) always exists. Furthermore, we show that, for the min/max implementation of logical "and" / "or" and the standard negation, the "mid-point" solution is always consistent.

Keywords: Self-reference, liar paradox, fuzzy logic, nonlinear equations.

1 INTRODUCTION

In this paper we study the celebrated "Liar Sentence":

$$\text{“This sentence is false”,} \tag{1}$$

and some of its generalizations. We say that (1) is *self-referential* because it states something about itself. Self-reference becomes more obvious if we rewrite (1) as follows:

$$A = \text{“}A \text{ is false”.} \tag{2}$$

Here is another example (the "Inconsistent Dualist") of *two* sentences which talk about each other:

$$A_1 = \text{“}A_2 \text{ is true”}. \tag{3}$$

$$A_2 = \text{“}A_1 \text{ is false”}. \tag{4}$$

We will call systems such as (2) and (3)–(4) *self-referential systems* (of sentences). Further examples of self-referential systems will be presented in later sections.

It is well known that *some* self-referential systems generate logical *paradoxes*[1]. In this paper we generalize the analyses of [18] and [17] and show that a large family of self-referential systems admit consistent *fuzzy* truth value assignments.

More specifically, we define a language which can talk about the truth values of sentences; using this language we show to each self-referential system of sentences (which talk about each other) can be mapped a system of numerical equations. A solution of the numerical system corresponds to a consistent truth value assignment for the sentences of the self-referential system. Invoking Brouwer's fixed point theorem, we show that every numerical system of the above form (which satisfies some mild continuity assumptions) possesses at least one solution; it follows that every self-referential system possesses at least one consistent (fuzzy) truth value assignment. Hence, self-referential systems which are paradoxical in the Boolean context become non-paradoxical in the fuzzy context.

The following papers have directly influenced our work. The application of fuzzy logic to the Liar paradox goes back to a paper by Zadeh [18]; as already mentioned, we consider our work to be a generalization of Zadeh's approach. Another approach which is directly related to the current paper (and can be considered as complementary to [18]) appears in [17] where a procedure is presented to transform self-referential systems to systems of equations; however, this approach is confined to the Boolean context. An additional major influence to our work comes from a sequence of papers by Grim and his collaborators [3,9,10,12]. Grim et al. consider collections of self-referential sentences and models the *fuzzy reasoning process* as a *dynamical system*. A method is presented to map each self-referential

[1]For example, regarding (2) assume the sentence A to be true, then what it says must hold, i.e. the sentence must be false. But then its opposite must be true, i.e. it must be true that "This sentence is true". But then it is true that "This sentence is false". In short, our reasoning oscillates between two conclusions: first that the sentence is false, then that it is true. Notice that we would enter a similar oscillation if we started by assuming the sentence to be false; then we would conclude that the sentence is true, which would mean that the sentence is false etc. This is the famous *Liar Paradox*. Similarly, the pair (3)–(4) generates a paradox: if A_1 is true, then A_2 is also true; but then A_1 must be false and so A_2 must be false and so on ad infinitum.

collection to a *dynamical system* which represents the reasoning process. Each sentence of the self-referential collection has a *time-evolving* fuzzy truth value which corresponds to a *state variable* of the dynamical system. Grim et al. present several examples of self-referential collections and study the properties of the corresponding dynamical systems. One of the main points of [3] is that self-referential collections can generate *oscillating* or *chaotic* dynamical behavior. While Grim et al. concentrate on oscillatory/ chaotic behavior, in the current paper our main interest is in obtaining *stable,* consistent truth value assignments.

We will not attempt even a brief overview of the extensive philosophical literature on the Liar, *semantic paradoxes* and the related concept of *truth*. This literature has a much wider scope than the current paper (a large part of it concerns the possibility of defining a *global truth predicate*). Let us simply emphasize that we do **not** claim to have achieved a *complete* resolution of *all* the issues related to the Liar; we simply offer a fairly complete solution regarding consistent truth value assignments to a specific family of systems. However, we should mention Tarski's fundamental papers on the *concept of truth*, [13–15], where the idea of an *infinite* hierarchy of languages is introduced, and Kripke's seminal paper [8], which presents the idea of a fixed point (in [8] fixed points appear as the limit of iterative processes; actually his concept is somewhat different from our use of Brouwer's Fixed Point Theorem). The interested reader can find additional references in our technical report [16]. We also mention that the definition of a *global* ***fuzzy*** *truth predicate* (valid for the crisp Peano arithmetic) is addressed in [4, 5]. Finally, an interesting approach to the Liar paradox uses *multidimensional* logic [2].

2 THE LANGUAGE

In this section we will construct a language $\mathcal{L}$ which can talk about the fuzzy truth values of sentences. This language is very similar to the one presented in [10].

For motivation, consider the following restatement of the Liar sentence:

$$A = \text{“The truth value of } A \text{ is } 0\text{”}$$

or, more compactly,

$$A = \text{“Tr } (A) = 0\text{”}$$

where the expression “Tr $(A) = a$” is understood as shorthand for “The truth value of A is a”, A is assumed to belong to some collection of sentences and a belongs to the interval $[0, 1]$ (in the Boolean context, a actually belongs to $\{0, 1\}$: true sentences have truth value 1, false sentences have truth value 0; in the fuzzy context a can take any value in $[0, 1]$).

Similarly, the Inconsistent Dualist can be written as follows

$$A_1 = \text{“Tr } (A_2) = 1\text{”}.$$
$$A_2 = \text{“Tr } (A_1) = 0\text{”}.$$

We construct the language $\mathcal{L}$ in several steps. The first step is well known for both Boolean and fuzzy logic. We denote the finite set of *1st-level elementary sentences* (also called *variables*) by $\mathbf{V}_1$:

$$\mathbf{V}_1 = \{A_1, A_2, \ldots, A_M\}.$$

From the 1st level variables *recursively* build the set of *1st level sentences* denoted by $\mathbf{S}_1$:

$$\text{If} \quad A_m \in \mathbf{V}_1 \text{ then } A_m \in \mathbf{S}_1 \, (m = 1, 2, \ldots, M)$$
$$\text{If} \quad B_1, B_2 \in \mathbf{S}_1 \quad \text{then } B_1 \vee B_2, B_1 \wedge B_2, B_1' \in \mathbf{S}_1$$

(note that $\mathbf{V}_1 \subseteq \mathbf{S}_1$). The set of *2nd level elementary sentences* is denoted by $\mathbf{V}_2$ and defined by

$$\mathbf{V}_2 = \{\text{“Tr}(B) = b\text{”} : B \in \mathbf{S}_1, b \in [0, 1]\}.$$

The set of *2nd level sentences* is denoted by $\mathbf{S}_2$ and defined recursively as follows:

$$\text{If} \quad C \in \mathbf{V}_2 \text{ then } C \in \mathbf{S}_2$$
$$\text{If} \quad D_1, D_2 \in \mathbf{S}_2 \text{ then } D_1 \vee D_2, D_1 \wedge D_2, D_1' \in \mathbf{S}_2$$

(again we have $\mathbf{V}_2 \subseteq \mathbf{S}_2$). Finally, the language $\mathcal{L}$ is the set of *all* sentences previously defined: $\mathcal{L} = \mathbf{V}_1 \cup \mathbf{S}_1 \cup \mathbf{V}_2 \cup \mathbf{S}_2$.

The above definitions specify the *syntactic form* of the sentences in which we are interested. The *meaning* we attach to these sentences is understood as follows.

1 The symbols “$\vee$”, “$\wedge$”, “ $'$ ” denote the usual logical operators *or, and, negation*. (Note that we treat *parentheses* in an informal manner and we assume that precedence of operators, grouping of terms etc. are well understood from the context).
2 The expression “Tr $(B) = b$” means “The truth value of B is b”. Note that at this point we have not provided a mechanism for evaluating the truth value of B (where $B \in \mathbf{S}_1$). *In other words, Tr* $(\cdot)$ *is not* **(yet)** *a function*. Neither have we provided a mechanism for evaluating the truth value of “Tr $(B) = b$” (where $B \in \mathbf{S}_1$ and “Tr $(B) = b$” $\in \mathbf{S}_2$). This will be done in Section 3.

Let us summarize. We have defined a family of 1st level logical sentences which combine a finite number of *primitive* (i.e. undefined) variables. We have also defined a family of 2nd level logical sentences which talk about the truth values of the 1st level sentences. Here are some examples of sentences from $\mathbf{V}_1, \mathbf{S}_1, \mathbf{V}_2, \mathbf{S}_2$.

$\mathbf{V}_1 : A_1, A_2, \dots, A_M.$

$\mathbf{S}_1 : A_1 \vee A_3, A_2', (A_2 \wedge A_4) \vee A_5' \;\dots$ etc.

$\mathbf{V}_2$: "$\mathrm{Tr}(A_1) = 0$", "$\mathrm{Tr}\left(A_7'\right) = 0$", "$\mathrm{Tr}((A_2 \vee A_5) \wedge A_1) = 0.3$" ... etc.

$\mathbf{S}_2$: ["$\mathrm{Tr}\left(A_1'\right) = 0$" $\wedge$ "$\mathrm{Tr}((A_1 \vee A_4) \wedge A_2) = 0.3$"] $\vee$ "$\mathrm{Tr}(A_3) = 0.8$" ... etc.

Obviously, we could keep building up the hierarchy of sentences, defining $\mathbf{V}_n$ in terms of $\mathbf{S}_{n-1}$, and $\mathbf{S}_n$ in terms of $\mathbf{V}_n$; but, for the purposes of this paper, going up to $\mathbf{V}_2, \mathbf{S}_2$ will be sufficient. However, it will be useful to define a special subset of $\mathbf{V}_2$ and the corresponding subset of $\mathbf{S}_2$ as follows. The *2nd level **elementary** Boolean sentences* are denoted by $\widetilde{\mathbf{V}}_2$ and defined by

$$\widetilde{\mathbf{V}}_2 = \{\text{"Tr}(B) = b\text{"} : B \in \mathbf{S}_1, b \in \{0, 1\}\}.$$

The *2nd level Boolean sentences* are denoted by $\widetilde{\mathbf{S}}_2$ and defined recursively as follows

$$\text{If} \quad C \in \widetilde{\mathbf{V}}_2 \text{ then } C \in \widetilde{\mathbf{S}}_2$$

$$\text{If} \quad D_1, D_2 \in \widetilde{\mathbf{S}}_2 \text{ then } D_1 \vee D_2, D_1 \wedge D_2, D_1' \in \widetilde{\mathbf{S}}_2.$$

Hence $\widetilde{\mathbf{V}}_2$ contains the sentences which claim *crisp* truth values for 1st level sentences, and $\widetilde{\mathbf{S}}_2$ contains the sentences which are formed from combinations of $\widetilde{\mathbf{V}}_2$ sentences. Again we have $\widetilde{\mathbf{V}}_2 \subseteq \widetilde{\mathbf{S}}_2$.

3 TRUTH VALUE ASSIGNMENT

We want to assign truth values to self-referential systems, i.e. to sentences which talk about each other. However, let us first recall (in Section 3.1) a classical method of truth value assignment which works for the *non*-self-referential case; then (in Section 3.2) we will adapt this method for the self-referential case.

3.1 "Explicit" Truth Value Assignment

The "explicit" assignment of truth values to elements of $\mathbf{S}_1$ is classical and works for both Boolean and fuzzy logics. We start by assigning an arbitrary truth value to every element of $\mathbf{V}_1$ (1st level variable). This is equivalent to selecting a mapping $x : \mathbf{V}_1 \to [0, 1]$, i.e. $\forall A_m \in \mathbf{V}_1$ we have

Family	$x \wedge y$	$x \vee y$	x'
Standard	$\min(x, y)$	$\max(x, y)$	$1 - x$
Algebraic	xy	$x + y - xy$	$1 - x$
Bounded	$\max(0, x + y - 1)$	$\min(1, x + y)$	$1 - x$
Drastic	$\begin{pmatrix} x \text{ when } y = 1 \\ y \text{ when } x = 1 \\ 0 \text{ else} \end{pmatrix}$	$\begin{pmatrix} x \text{ when } y = 0 \\ y \text{ when } x = 0 \\ 1 \text{ else} \end{pmatrix}$	$1 - x$

$\mathrm{Tr}\,(A_m) = x\,(A_m)$; for the sake of brevity we will henceforth use the simpler notation

$$\forall A_m \in \mathbf{V}_1 : \mathrm{Tr}\,(A_m) = x_m.$$

Next, take any $B \in \mathbf{S}_1$; it is a logical formula with variables $A_1, \dots, A_M$. If we replace every occurrence of A_m with x_m then we obtain a formula containing the variables $x_1, \dots, x_M$ and the operators "$\vee$", "$\wedge$", " $'$ ", which are now understood as numerical operators; in fuzzy logic (which subsumes Boolean logic as a special case) "$\vee$" denotes a *t-conorm*, "$\wedge$" denotes a *t-norm*, and " $'$ " denotes a negation. These operators have been studied extensively by fuzzy logicians [6]. Several typical implementations of t-norms, t-conorms and negations[2] are presented in Table 1.

Hence, every sentence of $\mathbf{S}_1$

$$B = F_B\,(A_1, \cdots, A_m)$$

is translated to a numerical formula

$$\mathrm{Tr}\,(B) = f_B\,(\mathrm{Tr}\,(A_1)\,, \dots, \mathrm{Tr}\,(A_M))$$

or, more concisely,

$$\mathrm{Tr}\,(B) = f_B\,(x_1, \dots, x_M)\,.$$

In this manner, the *truth function* originally defined on $\mathbf{V}_1$ (i.e. $\mathrm{Tr}\,(A_m) = x_m$, $m = 1, 2, \dots, M$) has been extended to $\mathbf{S}_1$. More precisely, the truth value of any sentence B is a function $f_B\,(x_1, \dots, x_M)$ of the truth values $x_1, \dots, x_M$ of the 1st level elementary sentences $A_1, \dots, A_M$.

Let us now extend the truth function so that it is defined everywhere on $\mathbf{S}_2$. We will do this in two moves. First, take any 2nd level elementary sentence $C \in \mathbf{V}_2$. This has the form

$$C = \text{“}\mathrm{Tr}\,(B) = b\text{”}$$

[2]In particular, note that we only mention one implementation of negation, namely $x' = 1 - x$; this the *standard* negation, by far the most popular in the literature. In the rest of the paper we will only work with the standard negation.

with $B \in \mathbf{S}_1$, $b \in [0, 1]$. Let us *define* the *truth-membership function* of sentence C (for every $B \in \mathbf{S}_1$ and $b \in [0, 1]$) to be

$$\mathrm{Tr}\,(C) = 1 - |\mathrm{Tr}\,(B) - b|\,. \tag{5}$$

Tr (B) in (5) has already been defined and so we can compute the truth value of C (for every $C \in \mathbf{V}_2$). Finally, we can extend truth values from $\mathbf{V}_2$ to $\mathbf{S}_2$ in exactly the same manner as we extended truth values from $\mathbf{V}_1$ to $\mathbf{S}_1$. This completes the definition of the truth function Tr: $\mathcal{L} \to [0, 1]$.

While the first part of the construction is classical, the second part is not as well known (but it *has* been used in the past, see for example [3, 10]). The key step is the definition of the truth-membership function in (5). This has been used in [11] and [3, 10]. Note that, according to (5), the maximum truth value of C is 1 and it is achieved when Tr $(B) = b$; the latter is exactly what C says. More generally, the truth value of C is a decreasing function of the absolute difference between Tr (B) and b. This certainly appears reasonable[3] .

In conclusion, explicit truth value assignment consists in choosing arbitrary truth values $x_1, \ldots, x_M$ for the elementary sentences $A_1, \ldots, A_M$ and then expressing the truth value of every sentence $D \in \mathcal{L}$ as a function $\mathrm{Tr}\,(D) = f_D\,(x_1, \ldots, x_M)$, where $f_D : [0, 1]^M \to [0, 1]$. Explicit truth value assignment does not involve any self-reference or circularity.

3.2 "Implicit" Truth Value Assignment

In the previous section we discussed the assignment of truth values to sentences which do not refer to each other. We will now present an approach for the assignment of truth values to self-referential sentences. The crux of our approach is a procedure which maps every system of self referential sentences to a system of numerical equations; then truth value assignment consists in solving the numerical system.

3.2.1 Two Simple Examples

Example A: The Liar. Consider the sentence

$$D = \text{“}\mathrm{Tr}\,(A) = 0\text{”}.$$

where $A \in \mathbf{S}_2$ and hence $D \in \mathbf{V}_2$. If we knew Tr (A), we would be able to compute

$$Tr\,(D) = 1 - |\mathrm{Tr}\,(A)|\,. \tag{6}$$

Suppose however that Tr (A) is unknown, but some other information is available, namely that D and A are the same sentence. Taking $D = A$

[3]For further justification of (5) see [3]. Note however, that a number of other functions could be used; as a simple example consider $\mathrm{Tr}(C) = 1 - (\mathrm{Tr}\,(B) - b)^2$.

in (6), we recover the Liar Sentence. It is also rather clear that $D = A$ implies Tr $(D) =$ Tr (A). Hence, setting Tr$(D) =$ Tr $(A) = x$ (where x is *unknown*), replacing in (6) and keeping in mind that we are looking for $x \in [0, 1]$, we obtain the numerical equation

$$x = 1 - x$$

which has the unique solution $x = 1/2$, i.e. the Liar Sentence is *half-true*. This is the approach used by Zadeh in [18]. Hence the paradox is removed[4].

Example B: The Inconsistent Dualist. Now consider the pair of sentences

$$D_1 = \text{“Tr}\,(A_2) = 1\text{”}$$
$$D_2 = \text{“Tr}\,(A_1) = 0\text{”}$$

where $A_1, A_2 \in \mathbf{S}_2$ and hence $D_1, D_2 \in \mathbf{V}_2$. To recover the Inconsistent Dualist, we make the correspondences $D_1 = A_1$, $D_2 = A_2$, we set Tr $(D_1) =$ Tr $(A_1) = x_1$, Tr $(D_2) =$ Tr $(A_2) = x_2$ and hence we obtain the system

$$x_1 = 1 - |x_2 - 1| \tag{7}$$
$$x_2 = 1 - |x_1| \tag{8}$$

which in $[0, 1] \times [0, 1]$ is equivalent to

$$x_1 = x_2$$
$$x_2 = 1 - x_1.$$

(7)–(8) has the unique solution $x_1 = x_2 = 1/2$, i.e. the Inconsistent Dualist is *half-true*.

3.2.2 Description of the Procedure

Generalizing the approach of the previous two examples, we obtain a procedure which maps every system of self-referential sentences to a system of numerical equations.

The basic idea is the following. Suppose that we have a collection of M sentences, which talk about the truth values of each other. Consider

---

[4]Before proceeding any further, let us note that the notation $A =$“Tr $(A) = a$” is not entirely rigorous. If taken literally, it would mean that a sentence can be written in two different ways, first using a single symbol A, and then using the string “Tr $(A) = a$”. But sentences have been defined to be *unique* strings of symbols; furthermore if $A =$“Tr $(A) = a$” is taken as a definition of A, then it is circular.

We have used the “=” symbol because it is more suggestive of self-reference. A more rigorous notation would be to write $A \leftrightarrow$“Tr $(A) = a$” where $\leftrightarrow$ is understood as a *biconditional*: it means that the sentence A is true iff the sentence “Tr $(A) = a$” is true or, more generally, that A and “Tr $(A) = a$” always have exactly the same truth value. This latter property (identity of truth value) is the one that we really needed to move from the self-referential sentence to the numerical equation and (we believe) it captures the crucial self-reference of the Liar (and similar systems of sentences, as will be seen in what follows).

the set $\mathbf{S}_2$ which is generated from M elementary sentences. We can pick sentences $D_1, \ldots, D_M \in \mathbf{S}_2$ which have the same structure as the original self-referential sentences. The only difference is that the M self-referential sentences talk about each other, while $D_1, \ldots, D_M$ talk about some elementary, unspecified sentences $A_1, \ldots, A_M$. However, since $A_1, \ldots, A_M$ are unspecified, we can identify A_m with D_m (for $m \in \{1, \ldots, M\}$). Intuitively, this means that D_m says something about the truth values of $A_1, \ldots, A_M$, i.e. about the truth values of $D_1, \ldots, D_M$. This is exactly the situation which we were trying to model in the first place.

Here are the details. As mentioned at the end of Section 3.1, the truth value of every 2nd level sentence $D \in \mathbf{S}_2$ (for fixed M and a specific choice of t-norm, t-conorm and negation) is a numerical function $f_D(x_1, \ldots, x_M)$, the independent variables $x_1, \ldots, x_M$ being the truth values of $A_1, \ldots, A_M$. To obtain specific truth values by the procedure of Section 3.1, it is necessary to specify $x_1, \ldots, x_M$. To this end, choose a function $\Phi : \{1, 2, \ldots, M\} \to \mathbf{S}_2$, where $\Phi(m)$ is defined (for $m \in \{1, \ldots, M\}$) so that it says about the (unspecified) $A_1, A_2, \ldots, A_M$ the same things that the m-th self-referential sentence says about the self-referential collection of sentences (examples of the procedure appear in Section 3.2.3). $\Phi(m)$ is a 2nd level sentence which can also be denoted as D_m. Now D_m is a logical formula $F_m(A_1, \ldots, A_M)$. In other words, we have (for $m = 1, 2, \ldots, M$):

$$\begin{aligned} D_1 &= F_1(A_1, \ldots, A_M) \\ D_2 &= F_2(A_1, \ldots, A_M) \\ &\ldots \\ D_M &= F_M(A_1, \ldots, A_M). \end{aligned}$$

Identifying A_m with D_m we can form the system of logical equations[5]

$$\begin{aligned} A_1 = D_1 &= F_1(A_1, \ldots, A_M) \\ A_2 = D_2 &= F_2(A_1, \ldots, A_M) \\ &\ldots \\ A_M = D_M &= F_M(A_1, \ldots, A_M). \end{aligned} \tag{9}$$

The "natural" interpretation of (9) is that A_m *says* (or is, or means) D_m.

[5]Keeping again in mind that $A_m \leftrightarrow F_m(A_1, \ldots, A_M)$ could be used in place of $A_m = F_m(A_1, \ldots, A_M)$.

(9) implies that:

$$\begin{aligned} \mathrm{Tr}(A_1) &= \mathrm{Tr}(D_1) = f_1(\mathrm{Tr}(A_1), \ldots, \mathrm{Tr}(A_M)) \\ \mathrm{Tr}(A_2) &= \mathrm{Tr}(D_2) = f_2(\mathrm{Tr}(A_1), \ldots, \mathrm{Tr}(A_M)) \\ &\ldots \\ \mathrm{Tr}(A_M) &= \mathrm{Tr}(D_M) = f_M(\mathrm{Tr}(A_1), \ldots, \mathrm{Tr}(A_M)) \end{aligned} \tag{10}$$

where $f_m : [0,1]^M \to [0,1]$ is the numerical formula obtained (by the procedure of Section 3.1) from F_m. A simpler way to write (10) is

$$\begin{aligned} x_1 &= f_1(x_1, \ldots, x_M) \\ x_2 &= f_2(x_1, \ldots, x_M) \\ &\ldots \\ x_M &= f_M(x_1, \ldots, x_M). \end{aligned} \tag{11}$$

(11) is a system of M numerical equations in M unknowns; we will refer to it as the system of *truth value equations*.

Depending on the particular Φ used, (11) may have none, one or more than one solutions in $[0,1]^M$. Hence, by specifying a particular Φ, we obtain a *set* of possible *consistent* truth value assignments for the 1st level elementary sentences. In other words, every solution of (11) is a consistent truth value assignment. At this point we must consider the possibility that the set of solutions is empty, i.e. that there is no consistent truth value assignment. However, as we will see in Section 4, under mild conditions there always exists at least one consistent assignment. Assuming that (11) has one or more solutions, we can choose one of these to assign truth values to the 1st level elementary sentences; next, using exactly the same construction as in Section 3.1, we can assign truth values to 1st level sentences, then to 2nd level elementary sentences and finally to 2nd level sentences. In particular, it is easy to check that at the end of the procedure the 2nd level sentences $D_1, \ldots, D_M$ will receive the truth values originally specified by the solution of (11) – hence the truth value assignment is, indeed, consistent.

3.2.3 More Examples

Example C: The Consistent Dualist. Now consider the pair of sentences: $A_1 =$"A_2 is true", $A_2 =$"A_1 is true". This can be written in $\mathcal{L}$ as

$$\begin{aligned} A_1 &= \text{“}\mathrm{Tr}(A_2) = 1\text{”} \\ A_2 &= \text{“}\mathrm{Tr}(A_1) = 1\text{”}, \end{aligned}$$

which in turn gives

$$\left\{\begin{array}{l}\mathrm{Tr}\,(A_1) = 1 - |\mathrm{Tr}\,(A_2) - 1| \\ \mathrm{Tr}\,(A_2) = 1 - |\mathrm{Tr}\,(A_1) - 1|\end{array}\right\} \text{ or } \left\{\begin{array}{l}x_1 = 1 - |x_2 - 1| \\ x_2 = 1 - |x_1 - 1|\end{array}\right\}.$$

Since we want solutions $x_1 \in [0, 1]$, $x_2 \in [0, 1]$, we finally get

$$x_1 = x_2$$

$$x_2 = x_1.$$

Any vector of the form $\overline{x} = (\beta, \beta)$ ($\beta \in [0, 1]$) is a solution; i.e. there is an infinite number of consistent truth value assignments including complete truth (Tr (A_1) =Tr $(A_2) = 1$) and complete falsity (Tr $(A_1) =$ Tr $(A_2) = 0$); in accordance to Proposition 4, (1/2, 1/2) is also a solution.
Example D. Now consider

$$A_1 = \text{“}A_2 \text{ is true and } A_3 \text{ is false”} \tag{12}$$

$$A_2 = \text{“}A_1 \text{ is true and } A_3 \text{ is false”} \tag{13}$$

$$A_3 = \text{“}A_1 \text{ is false”}. \tag{14}$$

which translates to

$$A_1 = \text{“Tr}\,(A_2) = 1\text{”} \wedge \text{“Tr}\,(A_3) = 0\text{”}$$

$$A_2 = \text{“Tr}\,(A_1) = 1\text{”} \wedge \text{“Tr}\,(A_3) = 0\text{”}$$

$$A_3 = \text{“Tr}\,(A_1) = 0\text{”}.$$

We will consider two different implemementations of $\wedge$.

If we implement $\wedge$ by the min t-norm, the truth value equations become

$$x_1 = \min\,[x_2, (1 - x_3)] \tag{15}$$

$$x_2 = \min\,[x_1, (1 - x_3)] \tag{16}$$

$$x_3 = 1 - x_1 \tag{17}$$

and they can be solved analytically. From (17) we obtain

$$x_1 = 1 - x_3$$

and then (15) – (16) become

$$x_1 = \min\,[x_2, x_1]\,, \qquad x_2 = \min\,[x_1, x_1]$$

from which follows that

$$x_1 = x_2, \qquad x_3 = 1 - x_1.$$

In other words, the general solution of (15) – (17) is

$$x = (\beta, \beta, 1 - \beta)$$

with $\beta \in [0, 1]$. Note that this includes the extremal solutions $(1, 1, 0)$ and $(0, 0, 1)$ as well as the mid-point solution $(1/2, 1/2, 1/2)$.

If we implement $\wedge$ by the product t-norm we obtain the truth value equations

$$x_1 = x_2 \cdot (1 - x_3)$$
$$x_2 = x_1 \cdot (1 - x_3)$$
$$x_3 = 1 - x_1$$

We can still use $x_1 = 1 - x_3$ to simplify the truth value equations to

$$x_1 = x_2 \cdot x_1, \qquad x_2 = x_1^2, \qquad x_3 = 1 - x_1$$

from which we obtain

$$x_1 = x_1^3, \qquad x_2 = x_1^2, \qquad x_3 = 1 - x_1$$

and finally

$$x_1 \cdot \left(1 - x_1^2\right) = 0, \qquad x_2 = x_1^2, \qquad x_3 = 1 - x_1.$$

This has the solutions

$$(0, 0, 1), \qquad (1, 1, 0), \qquad (-1, 1, 2);$$

the last solution, however, is inadmissible as a truth value assignment. Hence we see that for the same self-referential collection, the product implementation of $\wedge$ yields a subset of the solutions obtained through the min implementation.

Example E. Now consider

$$A_1 = \text{“Tr}(A_2) = 0.90\text{”} \wedge \text{“Tr}(A_3) = 0.20\text{”}$$
$$A_2 = \text{“Tr}(A_1) = 0.80\text{”} \wedge \text{“Tr}(A_3) = 0.30\text{”}$$
$$A_3 = \text{“Tr}(A_1) = 0.10\text{”}.$$

If we implement $\wedge$ with the min operator the truth value equations become

$$x_1 = \min\left[1 - |x_2 - 0.90|, 1 - |x_3 - 0.20|\right]$$
$$x_2 = \min\left[1 - |x_2 - 0.80|, 1 - |x_3 - 0.30|\right]$$
$$x_3 = 1 - |x_1 - 0.10|.$$

These equations cannot be further reduced and while in principle they can be solved analytically by distinguishing cases, the amount of work required is excessive. However, the equations can be solved numerically (using the Newton-Raphson or some other root finding algorithm – for details see [16]). One solution is $\overline{x} = (0.95, 0.85, 0.15)$. Repeated runs of the numerical algorithm (with different initial conditions) always give the same solution, so it is possible that this is the *unique* consistent truth value assignment for this problem.

The situation is similar when we implement $\wedge$ by product.. The truth value equations become

$$x_1 = (1 - |x_2 - 0.90|) \cdot (1 - |x_3 - 0.20|)$$
$$x_2 = (1 - |x_2 - 0.80|) \cdot (1 - |x_3 - 0.30|)$$
$$x_3 = 1 - |x_1 - 0.10|$$

and Newton-Raphson yields at least *two* solutions:

$$\overline{x} = (0.6784\ldots, 0.7715\ldots, 0.4216\ldots)$$
$$\overline{x} = (0.0473\ldots, 0.0872\ldots, 0.9473\ldots).$$

Each of these is a consistent truth value assignment

Example F. Our approach can also handle the case where the self-referential system contains more sentences than self-rerefences. Consider the following system

$$A_1 = \text{“}A_2 \text{ is true and } A_3 \text{ is false.”} \tag{18}$$
$$A_2 = \text{“}A_1 \text{ is true and } A_3 \text{ is false.”} \tag{19}$$

Using the min t-norm and the standard negation, the above translate to

$$x_1 = \min(x_2, 1 - x_3)$$
$$x_2 = \min(x_1, 1 - x_3)$$

and it can be checked that this system of *two* equations in *three* unknowns is solved by any triple from the set $\{(\alpha, \alpha, \beta) : \alpha \leq 1 - \beta\}$; the intersection of this set with $[0, 1]^3$ yields the acceptable truth values for the original self referential system. We see that this system is less specified (contains fewer self-references) than the one of Example D and hence admits more solutions. Of course, we could introduce additional, even *non*-self-referential constraints; for example we could add

$$A_3 = \text{“Snow is white”} \tag{20}$$

which would specify that $x_3 = 1$ and hence the expanded system would only admit the (reasonable) truth value assignment (0, 0, 1). If on the other hand, we used instead

$$A_3 = \text{“Snow is black”} \tag{21}$$

then the system would admit any truth value assignment $(\alpha, \alpha, 0)$ with $\alpha \leq 1$. The reader may want to speculate on the intuitive justification of these truth value assignments. Note that neither (20) nor (21) are self-referential; but both [(18),(19),(20)] and [(18),(19),(21)] are self-referential systems.
Example G: The Strengthened Liar. The final example is the so-called "Strengthened Liar", which involves the following sentence

$$A = \text{“}A \text{ is not true”}. \tag{22}$$

The "Strengthened Liar" has been often used as a test of proposed solutions to the Liar paradox. To treat this and similar sentences in the fuzzy context we must translate it in terms of a membership function for the property of being not true. To this end, consider the sentence

$$C = \text{“The truth value of } A \text{ is not } a\text{”}. \tag{23}$$

A possible truth value assignment for (23) is

$$\mathrm{Tr}\,(C) = \begin{cases} 1 \text{ when } & \mathrm{Tr}\,(A) \neq a \\ 0 & \text{else} \end{cases}.$$

However, this falls outside the framework of the the previous sections (in the language $\mathcal{L}$ we have not defined a predicate of the form "the truth value of A is different from a") and also it is too strict. Consider the case when $a = 1$ and $\mathrm{Tr}(A) = 0.99$. Do we really want to assign $\mathrm{Tr}(C) = 1$? How about the case $\mathrm{Tr}(A) = 0.99999$? A more reasonable truth value assignment can be obtained within the language as follows:

$$(\mathrm{Tr}(A) \neq a) = (\mathrm{Tr}(A) = a)'$$

and so

$$\begin{aligned} \mathrm{Tr}\,(\mathrm{Tr}(A) \neq a) &= 1 - \mathrm{Tr}\,(\mathrm{Tr}(A) = a) = 1 - (1 - |\mathrm{Tr}(A) - a|) \\ &= |\mathrm{Tr}\,(A) - a| \end{aligned} \tag{24}$$

which takes the maximum value of 1 when $|\mathrm{Tr}\,(A) - a| = 1$, i.e. in the cases: $\mathrm{Tr}(A) = 1$ and $a = 0$; $\mathrm{Tr}(A) = 0$ and $a = 1$.

Let us accept (24) and set $A = C$, i.e.

$$A = \text{“}\mathrm{Tr}\,(A) \neq a\text{”}. \tag{25}$$

(25) is more general than (22); to obtain (22) we set $a = 1$:

$$A = \text{“}\mathrm{Tr}\,(A) \neq 1\text{”}.$$

Hence, setting $x = \mathrm{Tr}(A)$, we must solve the truth value equation

$$x = |x - 1| = 1 - x$$

which has the unique solution $x = 1/2$.

3.2.4 Discussion

Consider the Liar sentence. In the classical (Boolean) context this sentence has been considered paradoxical because it cannot be consistently considered either true or false. Now consider its numerical version: $x = 1 - x$; what was previously considered a paradox, reduces to a simple algebraic fact: the equation $x = 1 - x$ has no solution in the set $\{0, 1\}$. If we enlarge the solution space to be the interval $[0, 1]$, then the equation *has* a solution ($x = 1/2$) and, in the fuzzy context, the sentence becomes non-paradoxical.

In Section 3.2.2 we have shown that, using the language $\mathcal{L}$, we can write a large number of self-referential systems and reduce each of these to a system of numerical equations. Several examples of self-referential systems have been given in Section 3.2.3. For each such system of sentences, we have found one (or more) solution to the corresponding system of equations, i.e. a consistent truth value assignment for the original system of sentences. In other words, self-referential systems which are paradoxical in the Boolean context, are non-paradoxical in the fuzzy context.

It is natural at this point to investigate the existence and uniqueness of solutions for the *general* self-referential system. We will show in Section 4 that *every* self-referential system of the form (11) possesses *at least* one solution, i.e. a consistent truth value assignment[6]. On the other hand, it is clear from the examples that uniqueness *cannot* be guaranteed: a self-referential system may possess more than one consistent truth value assignment. We will further discuss this point in Section 5.

4 EXISTENCE OF CONSISTENT TRUTH VALUE ASSIGNMENTS

We now turn to our main concern: we will investigate the consistency of implicit truth value assignment. More specifically, we will show that (in the fuzzy context) implicit truth value assignment (under very mild conditions on the numerical implementation of fuzzy connectives) always results in *at least* one consistent truth value assignment. In other words, we will show that every Φ function specifies *at least* one consistent truth value assignment. This is the subject of Proposition 3. However, we first need two auxiliary propositions.

Proposition 1 *If the implementations of* $\vee, \wedge, '$ *are, respectively, a continuous t-conorm, a continuous t-norm and the standard negation, then* $f_1, f_2, \ldots, f_M$ *in (11) are continuous functions of* $(x_1, x_2, \ldots, x_M)$.

Proof: We give a sketch of the proof (we omit the complete proof for the sake of brevity; the basic idea is clear). Suppose that $\vee, \wedge, '$ are a

[6]More precisely, this statement holds for every self-referential system *which can be expressed using the language* $\mathcal{L}$. Systems of more general sentences will be discussed in Section 5.

continuous t-conorm, t-norm and negation. Take some $m \in \{1, 2, \dots, M\}$. Recall that $f_m(x_1, x_2, \dots, x_M) = \mathrm{Tr}(D_m)$, where $D_m \in \mathbf{S}_2$. Now, take any $B \in \mathbf{S}_1$; then $\mathrm{Tr}(B) = y(x_1, x_2, \dots, x_M)$ and y is a finite combination of $\vee, \wedge, '$ and $x_1, \dots, x_M$, which is clearly a continuous function of the vector $(x_1, x_2, \dots, x_M)$. Furthermore, let $C =$ "$\mathrm{Tr}(B) = b$"; then

$$\mathrm{Tr}(C) = 1 - |\mathrm{Tr}(B) - b| = 1 - |y(x_1, x_2, \dots, x_M) - b|$$

which is also a continuous function of $(x_1, x_2, \dots, x_M)$. Since this is true for every $B \in \mathbf{S}_1$ and every $b \in [0, 1]$, we conclude that $\mathrm{Tr}(C)$ is a continuous function of $x = (x_1, x_2, \dots, x_M)$ for *every* $C \in \mathbf{V}_2$. Finally, since $D_m \in \mathbf{S}_2$, and $\mathrm{Tr}(D_m)$ is a finite combination of $\vee, \wedge, '$ and a finite number of terms $\mathrm{Tr}(C_1)$, $\mathrm{Tr}(C_2)$, ..., $\mathrm{Tr}(C_L)$ (where $C_1, C_2, \dots, C_L \in \mathbf{V}_2$) it follows that $\mathrm{Tr}(D_m)$ is a continuous function of $(x_1, x_2, \dots, x_M)$. □

Proposition 2 *Suppose that X is a nonempty, compact, convex set in R^M. If the function $f : X \to X$ is continuous, then there exists at least one fixed point $\overline{x} \in X$ satisfying*

$$\overline{x} = f(\overline{x}).$$

Proof: This is the well-known *Brouwer's Fixed Point Theorem*. Its proof can be found in a number of standard texts, for instance in [1, pp.323–329]. □

Now we can easily prove the existence of consistent truth value assignments.

Proposition 3 *If the implementations of $\vee, \wedge, '$ are, respectively, a continuous t-conorm, a continuous t-norm and the standard negation, then (11) has at least one solution $\overline{x} = (\overline{x}_1, \overline{x}_2, \dots, \overline{x}_M) \in [0, 1]^M$.*

Proof: We define the vector function $f(x_1, x_2, \dots, x_M)$ as follows:

$$f(x_1, x_2, \dots, x_M) = \begin{pmatrix} f_1(x_1, x_2, \dots, x_M), f_2(x_1, x_2, \dots, x_M), \dots, \\ f_M(x_1, x_2, \dots, x_M) \end{pmatrix}$$

where (for $m \in \{1, 2, \dots, M\}$) $f_m(x_1, x_2, \dots, x_M)$ is the function appearing in (11). Since $f_m(x_1, x_2, \dots, x_M)$ computes a truth value, we have $f_m : [0, 1]^M \to [0, 1]$ and hence $f : [0, 1]^M \to [0, 1]^M$. Furthermore, by Proposition 1 each f_m is a continuous function and so f is also a continuous vector function. Now we can apply Proposition 2 with $X = [0, 1]^M$. □

When we use Boolean truth value assessments, we can prove an additional result about consistent truth value assignments.

Proposition 4 *Suppose that in (9) $D_1, D_2, \dots, D_M \in \widetilde{\mathbf{S}}_2$ and the implementations of $\vee, \wedge, '$ are, respectively, max, min and the standard negation. Then (11) admits the solution* $(1/2, 1/2, \dots, 1/2)$.

Proof: Take any $m \in \{1, 2, \ldots, M\}$; then $F_m(A_1, A_2, \ldots, A_M)$ is a combination (through $\vee, \wedge, '$) of a finite number of elements $C_1, C_2, \ldots, C_L \in \widetilde{\mathbf{V}}_2$. Take any C_l (with $l \in \{1, 2, \ldots, L\}$); it has the form

$$C_l = \text{“Tr}(B_l) = b_l\text{”}$$

where $B_l \in \mathbf{S}_2$ and $b_l \in \{0, 1\}$. The corresponding numerical term will have the form

$$\mathrm{Tr}(C_l) = 1 - |\mathrm{Tr}(B_l) - b_l|\,.$$

or

$$z_l = 1 - |y_l - b_l| \tag{26}$$

where $y_l = \mathrm{Tr}(B_l)$ and $z_l = \mathrm{Tr}(C_l)$. Now, y_l will be a finite combination of $x_1, \ldots, x_M$ through max, min and negation operators, hence when $x_1 = x_2 = \ldots = x_M = 1/2$ we also get $y_l = 1/2$. Then, for $b_l \in \{0, 1\}$ we also get from (26) that $z_l = 1/2$.

Hence every term appearing in $f_m(1/2, 1/2, \ldots, 1/2)$ (the numerical version of $F_m(A_1, A_2, \ldots, A_M)$) will be equal to 1/2. Since these terms will be combined with max, min and negation operators it follows that $f_m(1/2, 1/2, \ldots, 1/2) = 1/2$ and this satisfies the m-th truth value equation:

$$x_m = \frac{1}{2} = f_m(1/2, 1/2, \ldots, 1/2)\,. \tag{27}$$

Since (27) holds for all $m \in \{1, 2, .., M\}$, it follows that (11) admits the solution $(1/2, 1/2, \ldots, 1/2)$. □

5 DISCUSSION

Our goal in this paper has been the investigation of the class of self-referential systems which can be resolved (i.e. can receive a consistent truth value assignment) within the context of fuzzy logic[7]. In this direction, we have shown that *every* self-referential system which is expressed using the language $\mathcal{L}$ admits a consistent *fuzzy* truth value assignment and hence is non-paradoxical in the fuzzy context. Note that $\mathcal{L}$ does not denote a single language; depending on the implementation of the logical connectives and the truth-membership function $\mathrm{Tr}(\cdot)$ we obtain a different language; hence our result holds with even greater generality. Indeed, further generalizations of our result are possible. Consider the following cases.

[7]We repeat that we do *not* concern ourselves with the more general philosophical problems associated withe Liar; in particular we do not make any claims regarding the definition of a global truth predicate.

1 It is possible to include in $\mathcal{L}$ the set $\mathbf{S}_3$ (of sentences which talk about the truth values of $\mathbf{S}_2$ sentences), $\mathbf{S}_4, \ldots$ and so on to any *finite* number of levels; and also by combining sentences from several levels of the hierarchy (for example, a sentence such as "Tr($A_4 \wedge A_2$) = 0.3"$\vee A_3$).
2 $\mathcal{L}$ can be extended to include sentences which talk about truth values without using the "=" relationship (for instance, see Example F, the Strengthened Liar; also consider the system "Tr(A_1) $\neq 0.3$"$\wedge$"Tr(A_2) < 0.8").
3 $\mathcal{L}$ can also be extended to iclude linguistic hedges (such as "very", "more or less" etc.); Proposition 3 still holds, provided the hedges are implemented by *continuous* numerical functions (e.g. second powers, square roots etc.).
4 Finally, one can conceive of even more general self-referential claims about sentences, e.g. sentences which talk about properties of the solutions of the truth value equations. For example consider the self-referential system: A_1= "Tr(A_2) = 0.3 ", A_2= "The system A_1, A_2 has no consistent truth value assignment".

Our results can be extended to handle many instances of cases 1–3 above; case 4 is more open-ended and hence may contain more problematic, i.e. unsolvable, systems. At any rate, we find it entirely possible that one can construct a self-referential system which corresponds to an unsolvable system of numerical equations; such a system would be paradoxical even in the context of fuzzy logic (the results of [4] point in this direction).

On the other hand, we believe that even in such cases the paradox would "merely" consist in the fact that a certain system of equations has no solution. In other words, we believe Liar-like "paradoxes" do not entail a paradox, but a false assumption, namely that a certain system of equations has solutions within a certain set[8]; the paradox can be removed by enlarging the set of admissible solutions.

It is perhaps worthwhile to elaborate the issue somewhat. The "explicit" method to assign truth values to a collection of sentences is to give truth values to the elementary variables / sentences and then evaluate all constituent sentences. This method works always, for both Boolean and fuzzy logic. In our opinion many of the "classical" self-reference paradoxes originate in the following manner: we are given a collection of sentences and the additional information does not consist in the truth values of some of them but in the interrelationship between the sentences. It turns out that in the Boolean context this information may be *too* restrictive, resulting in an unsolvable numerical system; while in the fuzzy context the information may be *not sufficiently* restrictive, resulting in a numerical system with more than one solution.

[8]This is exactly the way in which [17] treats Liar-like paradoxes.

An alternative point of view regarding existence and uniqueness of truth value assignments is the following: explicit truth value assignment is a form of selecting *axioms* for a particular systems, namely postulating that certain (elementary) sentences are true (have truth value 1) or false (have truth value 0; in this case the axioms can be understood as the negations of the elementary sentences); from this point of view one also has the choice of postulating "half true axioms" in the fuzzy context. At any rate, if explicit assignment of truth values is a method of specifying axioms, an alternative method is to specify (postulate, axiomatize) *relations* between sentences; these (self-referential) relations will implicitly determine the truth values of the elementary sentences (but maybe not uniquely, resulting in several equally possible axiomatizations).

Yet another point of view regarding the solutions of the truth value equations (11) (and the existence and uniqueness of their solutions) is to compare them to the dynamical systems used by Grim et al. [3, 9, 10, 12]. These evolve in discrete time, according to equations of the form

$$x_m(t+1) = f_m(x_1(t), \dots, x_M(t)), \quad \text{for } m = 1, 2, \dots, M \text{ and } t = 0, 1, 2, \dots \tag{28}$$

where $f_m(\cdot)$ are exactly the functions we have used in (11). Clearly, every solution of (11) (every consistent truth value assignment) is an equilibrium point of (28). Because, if for some $\overline{x} = (\overline{x}_1, \dots, \overline{x}_M)$ we have

$$\overline{x}_m = f_m(\overline{x}_1, \dots, \overline{x}_M) \qquad \text{for } m = 1, 2, \dots, M$$

and we take for some t that

$$x(t) = (\overline{x}_1, \dots, \overline{x}_M)$$

then clearly we will have

$$x(t+n) = f(\overline{x}_1, \dots, \overline{x}_M) = (\overline{x}_1, \dots, \overline{x}_M) \qquad \text{for } n = 1, 2, \dots$$

However, some of the equilibria $\overline{x}$ may be *unreachable* from certain initial conditions $(x_1(0), \dots, x_M(0))$ and / or *unstable*, i.e. a small perturbation send the system (28) away from $\overline{x}$. Hence either unreachable or unstable truth value assignments are not obtained with Grim's approach. This is a consequence of the fact that Grim uses the specific time update of (28), perhaps because it is analogous to human reasoning. But many other dynamical update schemes could be used; we have used Newton-Raphson (and also steepest descent, see [16]) which give access to truth value assignments not accessible by (28). Of course, Grim et al. are *not* interested in solving (11) but in discovering oscillatory and chaotic behavior; we, on the contrary, are interested in solving the truth value assignment, hence we want to *suppress* such behavior. It is also interesting to compare the appearance of instability, oscillation and chaos in such "dynamical reasoning systems" with the efforts expended for establishing the *stability* of more conventional dynamic fuzzy rule bases (for instance consider [7] and the references therein).

We have already mentioned in Section 3.2.3 that in certain cases the resolution of a self-referential system requires the use of numerical methods. Let us discuss two additional situations where numerical methods may prove useful.

First, consider a self-referential system where multiple truth value assignments are possible. Are some assignments "better" than other? For instance, solutions on the vertices of the hypercube $[0, 1]^M$ are "crisper" than solutions on the faces, which in turn are crisper than solutions in the interior. In certain circumstances one might be interested in obtaining the crispest solution to a self-referential system. The crispness can be described by an entropy function, so one might want to select the truth value assignment $(x_1, \ldots, x_M)$ which minimizes the function

$$J = \sum_{m=1}^{M} x_m \cdot \log(x_m)$$

subject to

$$x_m \in [0, 1], \quad x_m = f(x_1, \ldots, x_M), \quad m = 1, 2, \ldots, M.$$

This problem might be attacked by methods of constrained optimization.

Second, consider a logic with a truth value set $\{0, 1/K, 2/K, \ldots, 1\}$ (it has $K + 1$ truth values). In this context a self-referential system may fail to have a consistent truth value assignment, i.e. the truth value equations (11) may have no solution in the set $\{0, 1/K, 2/K, \ldots, 1\}$. In this case an *approximate solution* of the corresponding paradox can still be obtained by minimizing the *inconsistency* of the system; where inconsistency could be defined as

$$J = \sum_{m=1}^{M} (x_m - f_m(x_1, \ldots, x_M))^2$$

(or some similar function) and the minimization of J must be subject to

$$x_m \in \{0, 1/K, 2/K, ..., 1\}, \quad m = 1, 2, \ldots, M.$$

This is a combinatorial optimization problem.

Finally, a goal for further research is to establish an analog of Proposition 3 for *lattice-valued logics*; in this case a *lattice fix-point theorem* might be used in place of Proposition 2.

REFERENCES

[1] Binmore, K., (1992). *Fun and Games,* Heath and Company.

[2] Gersheson, C. "Multidimensional logic: A model for paraconsistent logic". Web preprint, http://homepages.vub.ac.be/˜cgershen/jlagunez/mdl/mdl.html.

[3] Grim, P., (1993). "Self-reference and chaos in fuzzy logic'". *IEEE Trans. on FS, 1*, 237–253.

[4] Hajek, P., Paris, J. and Shepherdson, J., (2000). "The liar paradox and fuzzy logic". *The Journal of Symbolic Logic, 65,* 339–346.

[5] Hajek, P. and Shepherdson, J. "A note on the notion of truth in fuzzy logic". *Annals of Pure and Applied Logic, 109,* 65–69.

[6] Klir, G.I. and Yuan, B., (1995). *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall.

[7] Kosko, B., (1988). "Global stability of generalized additive fuzzy systems". *IEEE Trans. on SMC – Part C, 28,* 441–452.

[8] Kripke, S. (1975). "Outline of a theory of truth". *J. of Philosophy, 72,* 690–716.

[9] Mar, G. and Grim, P., (1991). "Pattern and chaos: new images in the semantics of paradox". *Nous, 25,* 659–693.

[10] Mar, G. and St. Denis, P., (1999). "What the liar taught Achilles". *J. of Phil. Logic, 28*, 29–46.

[11] Rescher, N., (1969). *Multi-valued Logic*. McGraw-Hill.

[12] St. Denis, P. and Grim, P., (1997). "Fractal images of formal systems". *J. of Phil. Logic, 26,* 181–222.

[13] Tarski, A., (1935). "Der Wahrheitsbegriff in den formalisierten Sprachen". *Studien Philosophica, 1,* 261–405.

[14] Tarski, A., (1939). "On undecidable statements in enlarged systems of logic and the concept of truth". *The Journal of Symbolic Logic, 4,* 105–112.

[15] Tarski, A., (1944). "The semantic conception of truth". *Philosophy and Phenomenlological Research, 4,* 13–47.

[16] Vezerides, K. and Kehagias, Ath., (2003). "The Liar and Related Paradoxes: Fuzzy Truth Value Assignment for Collections of Self-Referential Sentences", CS Arxiv TR cs.LO/0309046.

[17] Wen, Lan., (2001). "Semantic paradoxes as equations". *The Mathematical Intelligencer, 23,* 43–48.

[18] Zadeh, L.A., (1979). "Liar's paradox and truth-qualification principle". Elect. Radio Lab Memo M79/34, Univ. of California, Berkeley.