On the Computer Certification of Fuzzy Numbers

Adam Grabowski
Institute of Informatics, University of Białystok
Akademicka 2, 15-267 Białystok, Poland
Email: adam@math.uwb.edu.pl

***Abstract*—The formalization of fuzzy sets in terms of corresponding membership functions is already available in machine-verified mathematical knowledge base. We show how it can be extended to provide the development of fuzzy numbers fully benefitting from the existing framework. The flexibility which is offered by automated proof-assistants allowed us to overcome some initial difficulties. Although fuzziness stems from the same background as rough set theory, i.e. incomplete or imprecise information, both formal approaches are substantially different.**

I. Introduction

DURING the past decades, mathematics would evolve from the pen-and-paper model in the direction of use of computers. As fuzzy set theory proposed by Zadeh offered new mathematical insight for the real data in the world of uncertain or incomplete information, dealing mainly with those contained in digital archives, it is not surprising that similar methods will be used in order to obtain the properties of objects within the theory itself. The original approach to fuzzy numbers met some criticism and various ways of improvement were offered as yet. But usually computers serve as an assistant offering calculations – why not to benefit from their more artificial intelligence strength? We try to address some issues concerned with the digitization of this specific fragment of fuzzy set theory, representing a path to fuzzy numbers, so it can be considered as a case study in a knowledge management, being a work on fuzzy sets in the same time.

The paper is organized as follows. The next section is devoted specifically to the situation in the area of computer-checked formalization of mathematics and contains a brief primer to formal fuzzy sets; in the third we gave an example of the proof to show how it looks like; fourth is devoted to the connection of our work with classical and the rough set theory. The other two sections explain specific issues we met during our work while the final brings some concluding remarks and the plans for future work.

II. A Formal Primer of Fuzzy Sets

We were surprised that within the rough set theory the notion of a rough set is not formally chosen as unique. On the one hand, it is a class of abstraction with respect to the rough equality, on the other – the pair of approximation operators. As both theories have much in common, we expected the same from fuzzy sets. But – the membership function itself can be just treated as a fuzzy set. Obviously, there is something unclear with the domain vs. support of a function (as what we call 'fuzzy sets' in fact is a fuzzy subset), but it is not that dangerous. As the author developed the formalization of rough sets, he could make the decision of how much of the existing apparatus should be used also in this case. Eventually all relational structure framework [3] was dropped as completely useless here. We could take the Cartesian product of the original set and the corresponding function, but it is enough to deal only with the latter one.

"Computer certification" is a relatively new term describing the process of the formalization via rewriting the text in a specific manner, usually in a rigorous language. Now this idea, although rather old (taking Peano, Whitehead and Russell as protagonists), gradually obtains a new life. As the tools evolved, the new paradigm was established: computers can potentially serve as a kind of oracle to check if the text is really correct. And then, the formalization is not *l'art pour l'art,* but it extends perspectives of knowledge reusing. The problem with computer-driven formalization is that it draws the attention of researchers somewhere at the intersection of mathematics and computer science, and if the complexity of the tools will be too high, only software engineers will be attracted and all the usefulness for an ordinary mathematician will be lost. But here, at this border, where there are the origins of MKM – Mathematical Knowledge Management, the place of fuzzy sets can be also. To give more or less formal definition, according to Wiedijk [9], *the formalization* can be seen presently as "the translation into a formal (i.e. rigorous) language so computers check this for correctness."

In this era of digital information anyone is free to choose his own way; to quote V. Voevodsky, Fields Medal winner's words: "Eventually I became convinced that the most interesting and important directions in current mathematics are the ones related to the transition into a new era which will be characterized by the widespread use of automated tools for proof construction and verification". If we take into account famous Four Colour Theorem, automated tools can really enable making some significant part of proofs, so hard to discuss with this opinion.

Among many available systems which serve as a proof-assistant we have chosen Mizar. The Mizar system [4] consists of three parts – the formal language, the software, and the database. The latter, called Mizar Mathematical Library (MML for short) established in 1989 is considered one of the largest repositories of computer checked mathematical knowledge. The basic item in the MML is called a Mizar article. It reflects roughly a structure of an ordinary paper, being considered at two main layers – the declarative one, where definitions and

978-1-4673-4471-5/$25.00

theorems are stated and the other one – proofs. Naturally, although the latter is the larger, the earlier needs some additional care.

As lattice theory and functional analysis are the most developed disciplines within the MML, further codification of fuzzy numbers, including their lattice-theoretic flavour, looks very promising. As a by-product, apart of readability of the Mizar language, also presentation of the source which is accessible by ordinary mathematicians and pure HTML form with clickable links to notions and theorems are available after the acceptance of the development into the library. As far as we know, this is the first attempt to formalize fuzzy sets in such extent using any popular computerized proof assistant.

Recall that a fuzzy set A over a universe X is a set defined as

$$A = \{(x, \mu_A) : x \in X\},$$

where $\mu_A \in [0, 1]$ is membership degree of x in A. Because the notions in the MML make a natural hierarchy (as the base set theory is Tarski-Grothendieck, which is close to ordinary Zermelo-Fraenkel axiomatics, accepted by most mathematicians): functions $\rightarrow$ relations = subsets of Cartesian product $\rightarrow$ sets, so it is a relation. Zadeh's approach assumes furthermore that μ_A is a function, extending a characteristic function χ_A. So, for arbitrary point x of the set A, the pair (x, μ_A) can be replaced just by the value of the membership function $\mu_A(x)$, which is in fact, formally speaking, the pair under consideration. Then all operations can be viewed as operations on functions, which appeared to be pretty natural in the set-theoretic background taken in the MML as the base. All basic formalized definitions and theorems can be tracked under the address http://mizar.org.

```
definition let C be non empty set;
  mode Membership_Func of C is
    [.0,1.]-valued Function of C,REAL;
end;
```

The aforementioned definition introduces membership function just as a function from given non-empty set into a subset of the set of all real numbers, and the values belong to the unit interval. Of course, `Membership_Func` is not uniquely determined for C – the keyword `mode` starts the shorthand for a type in Mizar, that is, in fact C variable can be read from the corresponding function rather than vice versa.

```
definition let C be non empty set;
  mode FuzzySet of C is Membership_Func of C;
end;
```

We collected translations of selected formalized notions in Table I. As we can read from this table, there are standard operations of fuzzy sets available, usually taken componentwise (note that `F.x` stands for the value of the function F on an argument x). Note that the Mizar repository extensively uses a difference between functions and partial functions; (`Function of X, Y` and `PartFunc of X, Y` in Mizar formalism); because in case of partial functions only the inclusion of the domain in the set X is required, hence the earlier type expands to the latter automatically.

TABLE I
FORMALIZED NOTIONS AND THEIR FORMAL TRANSLATIONS

The notion	Formal counterpart
the membership function	`Membership_Func of C`
fuzzy set	`FuzzySet of C`
$\chi_A(x)$	`chi(A,X).x`
α-set	`alpha-set C`
supp C	`support C`
$F \cap G$	`min (F,G)`
$F \cup G$	`max (F,G)`
cF	`1_minus F`

III. An Illustrative Example of the Proof

In this section we focus on the example of formalized theorem about level sets. Before we start, we explain some plain ASCII symbols which will be used as all Mizar articles have the limitation of using only this narrow set of codes (but automated translation enables to use ordinary mathematical notations, usually based on LaTeX). `>=` stands for $\geq$, `c=` is set-theoretic inclusion, `"` means counterimage or the converse of the relation (including function); `in` stands for $\in$. Of course, `[.a,b.]` is a interval of real numbers a and b. `dom` denotes a domain of a function [4]. It appeared to be pretty feasible, because we could formalize natural properties in a rather compact way, as shown below [2]:

```
definition let C be non empty set;
           let F be FuzzySet of C;
           let a be Real;
  func a-cut F -> Subset of C equals
    { x where x is Element of C : F.x >= a };
end;
```

One can easily notice near one-to-one correspondence with the well-known definition of α-cuts (or level sets):

$$A_\alpha = \{x \in X : \mu_A(x) \geq \alpha\}$$

As it is the counterimage of the interval $[\alpha, 1]$, we can prove the following theorem:

```
theorem AlphaCut1:
  for F being FuzzySet of C,
      alpha being Real holds
    alpha-cut F = F " ([. alpha, 1 .])
  proof
    let F be FuzzySet of C,
        alpha be Real;
    thus alpha-cut F c= F " ([. alpha, 1 .])
    proof
      let x be element;
      assume x in alpha-cut F; then
      consider y being Element of C such that
A1:   x = y & F.y >= alpha;
      x in C by A1; then
A2:   x in dom F by FUNCT_2:def 1; then
      F.y in [. 0, 1 .] by A1,PARTFUN1:4; then
      0 <= F.y & F.y <= 1 by XXREAL_1:1; then
      F.y in [. alpha, 1 .] by XXREAL_1:1,A1;
      hence thesis by A1,FUNCT_1:def 7,A2;
    end;
    :: the other inclusion omitted
  end;
```

IV. Rough and Fuzzy Formal Approaches

In the usual informal mathematical jargon it is easy to say that e.g., two objects are identical up to the isomorphism, formal language has to deal somehow with it. In the fuzzy set theory this can be noticed at the very beginning – some people treat fuzzy sets as the pair of the set and corresponding membership function. Fuzzy sets are subsets of ordinary sets; as we can take membership function just as χ of ordinary sets, it clearly shows the feasibility of this approach. Of course, it is impossible then, at least without any additional preparing work, to find the common bottom ground for ordinary sets and fuzzy sets; however all sets can be made fuzzy in view of the simple lemma cited below:

```
theorem
  for C being non empty set holds
    chi(C,C) is FuzzySet of C;
```

Although two widely-known views (rough and fuzzy approaches) for incomplete or imprecise information have much in common in principle, there are essential differences between both of them [10]. In fuzzy sets, every element has its own membership measure. In rough approach, the degree of membership is rather calculated from the set as a whole, so it is pretty close to Bayes' probability theory, as we quote this below.

```
definition let A be finite Tolerance_Space;
           let X be Subset of A;
  func MemberFunc (X, A) ->
    Function of the carrier of A, REAL means
  for x being Element of A holds  it.x =
  card (X /\ Class (the InternalRel of A,x)) /
    (card Class (the InternalRel of A,x));
end;
```

Paradoxically, even the notion of the rough set was defined in two ways, as pairs of the lower and the upper approximations of a set (in the sense of Iwiński), and as classes of abstraction with respect to given reflexive, symmetric, and transitive binary relation (original Pawlak's approach). MML reflects both approaches, concentrating on the properties of approximation operators and various types of binary relations generalizing equivalence relations.

V. Some "Easy" Problems

Virtually any mathematician uses a formal language; as engineers have also a higher math course in his curriculum, it shouldn't be a big problem to validate facts formally. But if we claim the proof is correct, some natural questions arise: the correctness with respect to which assumptions? What about foundations? If it comes to machine, what are properties of the checker? The original de Bruijn's dream was to have a small checker with the transparent kernel. Most of contemporary proof assistants are rather far from this requirement. Although a theorem can look easy, formal mathematics can bring some unpredictable problems; it is enough to mention e.g. Kepler's conjecture about the densest sphere packing (a part of Hilbert's 18th problem) or Jordan curve theorem that any simple closed curve cuts the plane into two disjoint areas. Intuitively, they are nearly trivial and understandable virtually for any human being. However, even at the very foundational level of used logic (constructive proofs do not claim the law of excluded middle) we can find some unexpected difficulties. Especially important example in our fuzzy context is the so-called glueing lemma – the proof of a simple fact about pasting some continuous functions together to make e.g. triangular (or trapezoidal) fuzzy set, intuitively trivial, draws some surprising dependencies.

It is rather hard to approximate the real complexity of a proof; one of the most popular measures is the de Bruijn factor, i.e. the ratio between the formal translation of the mathematical paper and the original (usually after packing the source and corresponding LaTeX file). Although it is claimed to be about 4 in the case of the Mizar library, in our case is about six (i.e. formal proofs are six times longer than their informal counterparts). Such relatively high number is caused by technical calculations in the process of glueing continuous functions.

VI. Towards Fuzzy Numbers

As all Mizar types should have non-empty denotation, it would force us to define both triangular and trapezoidal fuzzy sets. The natural definition is usually written as conditional definition of parts of the function. We used intervals `[.a,b.]` and `AffineMaps` to save some work (e.g., affine maps are proven to be continuous, one-to-one, and monotone real maps under underlying assumptions). The operator `+*` glues two functions if their domains are disjoint; if not, then the ordering of glueing counts.

```
definition let a,b,c be Real;
  assume a < b & b < c;
  func TriangularFS (a,b,c) -> FuzzySet of REAL
    equals
    AffineMap (0,0)
   +* (AffineMap (1/(b-a),-a/(b-a)) | [.a,b.])
   +* (AffineMap (-1/(c-b),c/(c-b)) | ].b,c.]);
end;
```

The assumptions on the ordering of real variables a, b, c are unnecessary here and will be removed; we kept this as needed to prove the continuity of this fuzzy set afterwards. It is worth mentioning here that the proof of correctness of the above definition is 40 lines long – surprisingly long comparing to the popular (of course, false) opinion that definitions don't need proofs. Continuity of this triangular fuzzy set needed much more lines in our Mizar script (90 lines in case of one-point glueing).

Remembering that a fuzzy number is a convex, normalized fuzzy set on the real line $\mathbb{R}$, with exactly one $x \in \mathbb{R}$ such that $\mu_A(x) = 1$ and μ_A is at least segmentally continuous, we defined it as the Mizar type:

```
mode FuzzyNumber is f-convex continuous
  strictly-normalized FuzzySet of REAL;
```

As all types are constructed as radix types with added optional adjectives, the generalization, especially that automateddriven (by cutting the adjectives in the assumptions), is possible and quite frequently used. Some of the adjectives are a little bit stronger that others, with the quoted below as example:

```
definition let C be non empty set;
           let F be FuzzySet of C;
  attr F is strictly-normalized means :SNDef:
    ex x being Element of C st
    F.x = 1  &  for y being Element of C st
      F.y = 1 holds y = x;
end;
```

Observe that this adjective means that a fuzzy set is also normalized in normal sense. Due to automatic clustering of attributes after *registering* this quite natural and easy property any additional reference won't be needed.

```
registration let C be non empty set;
  cluster strictly-normalized ->
    normalized for FuzzySet of C;
end;
```

In our opinion, we made some significant progress on the certification of fuzzy sets and numbers, but our primary aim was to get the formal net of notions correct and reusable and we hope to benefit from it in our future work.

VII. Conclusion and Further Work

The primary aim of using computers in the process of the formalization was to provide its undoubtful correctness. One can argue however that also careful human review should do the same work. The famous exception is the publication of the proof of the Kepler conjecture by Hales in "Annals of Mathematics"; referees cannot be fully sure of the correctness of computer programs and tedious, extremely long computer-driven calculations. But things are different when it comes to program themselves; hardly readable, looking like computer code, proof of Four Color Theorem is verified formally; it sheds some new light for the verification of program libraries – e.g. there is significant progress made with the computer certification of Java or C libraries or even compilers themselves. But here readability is of minor interest; also proofs and the content itself are rather routine. Once the topic is formalized in the machine-understandable language, automated provers can be applied to obtain new results automatically. Based on computer-certified content, further automatic semantical investigations can be made [5], as, for example, extracting lemmas, annotating technical proofs or investigating direct corollaries, automated translation, and fast unification. Furthermore, MML is a subject of continuous changes called *revisions* which can be the result of software upgrades, generalizations, theory merging, introducing new language constructions etc. Also the original first formal approach for fuzzy sets which is dated back to 2001 [6], was thoroughly revised by the author to improve its reuse (e.g., a fuzzy set was primarily defined as the Cartesian product of the set C and the image of the membership function applied to C).

Computer certification of proofs seems to be an emerging trend and some corresponding issues can be raised. We are assured that there are some visible pros of our approach, as for example, automated removal of repetitions, and also the need of writing a sort of preliminary section vanishes in the Mizar code. The type system enables us to search for possible generalizations (including a kind of reverse mathematics at the very end); the use of automated knowledge discovery tools is much easier due to internal information exchange format, which at the same time offers direct translations for a number of formats (e.g, close to the English-like human-oriented language), not limited to the Mizar source code. There are of course drawbacks we should remember of: first of all, the syntax. The Mizar language, although pretty close to natural language, is still an artificial language. Of course, main problem with the formalization is making proper formal background – lemmas and theorems – which can be really time-consuming, hence the stress on reusability of available knowledge.

We argue that the formalization itself can be very fruitful and creative as long as it extends the horizons of the research and make new results possible. Furthermore, the more the database larger is, the formalization can be more feasible. Even if the formalized content concerning fuzzy sets is not that big as of now (there is only about 9000 lines of Mizar code on fuzzy sets comparing with 2.5 million of lines in the whole MML), the basics are already done, and it can serve both as a good starting point for further development, including rough-fuzzy hybridization, as well as from translated existing content we can try to obtain new results. Regardless of the gains of the availability of the topic to majority of popular proof assistants one can ask a question of assurance of the correctness of the proofs; Urban's [8] tools translating Mizar language into the input of first-order theorem-provers or XML interface providing information exchange between various math-assistants are already in use, so not only proof-checkers other than the Mizar verifier can analyze it, but additionally it can allow for some "dirty work" to be done by computer.

References

[1] D. Dubois, H. Prade, Operations on fuzzy numbers, *International Journal of System Sciences,* 9(6), 613–626, 1978.

[2] A. Grabowski, The formal construction of some special fuzzy sets, to appear in *Formalized Mathematics,* 2013.

[3] A. Grabowski, M. Jastrzębska, Rough set theory from a math-assistant perspective, *Lecture Notes in Artificial Intelligence*, 4585, 152–161, 2007.

[4] A. Grabowski, A. Korniłowicz, A. Naumowicz, Mizar in a nutshell, *Journal of Formalized Reasoning,* 3(2), 153–245, 2010.

[5] A. Grabowski, Ch. Schwarzweller, Towards automatically categorizing mathematical knowledge, M. Ganzha, L. Maciaszek, and M. Paprzycki (Eds.), *FedCSIS 2012 Proceedings,* 63–68, 2012.

[6] T. Mitsuishi, N. Endou, Y. Shidama, The concept of fuzzy set and membership function and basic properties of fuzzy set operation, *Formalized Mathematics*, 9(2), 351–356, 2001.

[7] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data,* Kluwer, Dordrecht, 1991.

[8] J. Urban, G. Sutcliffe, Automated reasoning and presentation support for formalizing mathematics in Mizar, *LNCS,* 6167, 132–146, 2010.

[9] F. Wiedijk, Formal proof – getting started, *Notices of the American Mathematical Society,* 55(11), 1408–1414, 2008.

[10] Y.Y. Yao, A comparative study of fuzzy sets and rough sets, *Information Sciences*, 109(1-4), 227–242, 1998.

[11] L. Zadeh, Fuzzy sets, *Information and Control,* 8(3), 338-353, 1965.