Initial Comparison of Formal Approaches to Fuzzy and Rough Sets

Adam Grabowski(✉) and Takashi Mitsuishi

[1] Institute of Informatics, University of Białystok
Ciołkowskiego 1M, 15-245 Białystok, Poland
adam@math.uwb.edu.pl
[2] University of Marketing and Distribution Sciences, Kobe
3-1 Gakuen Nishimachi, Nishi-ku, Kobe 655-2188, Japan
takashi_mitsuishi@red.umds.ac.jp

Abstract. Fuzzy sets and rough sets are well-known approaches to incomplete or imprecise data. In the paper we compare two formalizations of these sets within one of the largest repositories of computer-checked mathematical knowledge – the Mizar Mathematical Library. Although the motivation was quite similar in both developments, these approaches – proposed by us – vary significantly. Paradoxically, it appeared that fuzzy sets are much closer to the set theory implemented within the Mizar library, while in order to make more feasible view for rough sets we had to choose relational structures as a basic framework. The formal development, although counting approximately 15 thousand lines of source code, is by no means closed – it allows both for further generalizations, building on top of the existing knowledge, and even merging of these approaches. The paper is illustrated with selected examples of definitions, theorems, and proofs taken from rough and fuzzy set theory formulated in the Mizar language.

1 Introduction

Through the years, we can observe the evolution of mathematics from the classical paper-based model in the direction of use of computers. As fuzzy set theory proposed by Zadeh [24] 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 ordered 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. In the next section we describe a kind of scenario for our formalization. The third section is devoted specifically to the

L. Rutkowski et al. (Eds.): ICAISC 2015, Part I, LNAI 9119, pp. 160–171, 2015.
DOI: 10.1007/978-3-319-19324-3_15

situation in the area of computer-checked formalization of mathematics and contains a brief primer to formal fuzzy sets; in the fourth we gave a brief description of the alternative approach – rough set theory. The other two sections explain specific issues we met during our work with a more detailed study of dissimilarities between approaches, while the final brings some concluding remarks and the plans for future work.

2 The Motivation: Bridging a Route to OFNs

Usually, the formalization challenge should have a sort of a lighthouse – clearly visible target; sometimes, especially for a larger group of collaborators it is a tough theorem or chosen textbook. We aimed at getting to the OFN because only from broader perspective one can see if the developed approach was really flexible and reusable. Obviously, we aim also at the formal comparison of the two models. Ordered Fuzzy Numbers (OFN) were introduced by Kosiński, Prokopowicz and Ślęzak in 2002 [15,16] as the extension of the parametric representation of convex fuzzy numbers. Unquestionable advantage of OFN is that, in contrast to classical fuzzy numbers proposed by Dubois and Prade [2], they do not suffer from unexpected and uncontrollable results of repeatedly applied operations.

In this approach, an ordered fuzzy number is defined as a pair of continuous functions, (f, g) which domains are the intervals $[0, 1]$ and values are in $\mathbb{R}$. The continuity of both functions implies their images are bounded intervals, call them UP and $DOWN$, respectively. We can mark boundaries as $UP = [l_A, 1_A^-]$ and $DOWN = [1_A^+, p_A]$. In general, the functions f, g need not to be invertible, only continuity is required. If we add the constant function on the interval $[1_A^-, 1_A^+]$ with its value equal to 1, we might define the membership function

$$\begin{aligned} \mu(x) = \mu_{up}(x),\ \text{ if } x \in [l_A, 1_A^-] = [f(0), f(1)], \\ \mu(x) = \mu_{down}(x),\ \text{ if } x \in [1_A^+, p_A] = [g(1), g(0)] \ \text{ and} \\ \mu(x) = 1 \ \text{ when } x \in [1_A^-, 1_A^+] \end{aligned} \tag{1}$$

if

1. $f \leq g$ are both invertible, i.e. inverse functions $f^{-1} =: \mu_{up}$ and $g^{-1} =: \mu_{down}$ exist and
2. f is increasing, and g is decreasing.

The membership function obtained in this way $\mu(x), x \in \mathbb{R}$ represents a mathematical object which reminds a convex fuzzy number in the classical sense [12].

3 Fuzzy Sets Treated Formally

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 first 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 [8] was dropped as completely useless here. Surprisingly to us, [11] used term "formalization" to describe algebraic model of fuzzy relations – and this representation strongly depends on relational structures. It should be remembered that we do much more – the language we use should be understandable by computers, hence the choice of the approach is determined by other means. 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. All the operations on fuzzy sets actually are concerned with the membership functions, which initially seemed to be a little bit controversial, but we found it rather useful.

"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. Hence such activity 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 [23], *the formalization* can be seen presently as "the translation into a formal (i.e. rigorous) language so computers check this for correctness."

Among many available systems which serve as a proof-assistant we have chosen Mizar. The Mizar system [9] 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 theorems are stated and the other one – proofs. Naturally, although the latter is the larger, the earlier needs some additional care.

As far as we know, this is the first attempt to formalize fuzzy sets in such extent using any popular computerized proof assistant. The formalization work started as described in [5] was completed by the first author and accepted to Mizar database under the name `FUZNUM_1` at the end of 2014 [4]. Earlier development – by the second author – was accepted under MML identifier `FUZZY_1` in 2001 [17] and passed the massive redesign to follow new expressive power of the Mizar language.

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 the Tarski-Grothendieck set theory, which is basically Zermelo-Frankel set theory with the Axiom of Choice where the axiom of infinity is replaced by Tarski's axiom of existence of arbitrarily large, strongly inaccessible cardinals) of the form: functions $\rightarrow$ relations = subsets of Cartesian product $\rightarrow$ sets, so it is a binary relation, i.e. subset of Cartesian product $X \times X$.

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 FuzzySet of C is Membership_Func of C;
end;
```

Of course, `Membership_Func` is not uniquely determined for C – the keyword `mode` starts the shorthand for a type[1] in Mizar, that is, in fact C variable can be read from the corresponding function rather than vice versa.

We collected translations of selected formalized notions in Table 1.

Table 1. 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`

As we can read from Table 1, 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). The support of a membership function shouldn't be defined because it was used already by the theory of formal power series.

[1] Most theorem provers are untyped, the Mizar language has types.

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. Of course all such automation techniques are turned on after proving corresponding properties formally.

4 Rough Sets as the Structural Counterpart

Another extension of classical set theory, dealing with the situation of incomplete or uncertain information, is the rough set theory. There no space here to discuss pros and cons of both approaches, although both can be feasible in the same situation (rough-fuzzy hybridization). As the origins seem to very similar, at least at the very first sight, we could expect also the same lines of formalizing code. Much to our surprise it wasn't so.

The literature of machine-aided formalization of mathematics makes a clear distinction between classical (sometimes called also *concrete*) mathematics and abstract one. The difference is in dealing with the notion of a structure (the latter one uses such notion while the earlier doesn't) – pretty technical notion which can be viewed as partial function.

Both fuzzy and rough set theory extends the ordinary set theory in the direction of partial membership; clearly in rough case it is more probability-oriented. Even in very informal form, one can see clear correspondence between membership function and basic notions of ZF – essentially membership function *per se* can be expressed in this language.

The Bayesian content in RST can be easily observed under formal code of

```
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
:: ROUGHS_1:def 9
   for x being Element of A holds
     it.x = card (X /\ Class (the InternalRel of A, x)) /
       (card Class (the InternalRel of A, x));
  correctness;
end;
```

which informally can be read as

$$\mu_X(x) = \frac{|X \cap [x]_R|}{|[x]_R|}$$

In other words, the primitive notion is the indiscernibility relation establishing a relational structure, and the membership function is build by counting cardinalities of corresponding classes of abstraction (remember Pawlak's original idea was to have equivalence relation for that). Of course, given a tolerance space, its μ_X is determined uniquely, hence contrary to the `mode` as in fuzzy case, here we have keyword `func` which is a (language) function – both its existence and uniqueness can be proven.

5 Extending the Ordinary Set Theory

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. The classical example is the parallel treatment of lattices – both in the sense of relational structures equipped by the partial ordering and corresponding algebraic structure – tools which offer automated discovery of facts are usually specialized (including those based on the equational calculus).

In the fuzzy set theory similar dualism can be noticed at the very beginning – some people treat fuzzy sets as the pair of the set and corresponding membership function. On the other hand, also axiomatical approaches to obtain fuzzy sets without ZF as an intermediate step are known; it is a kind of interesting research direction but we decide to put the stress on the reusability of knowledge.

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;
```

After this, we can choose from all fuzzy sets in this very broad sense those ones which membership function has values only 0 or 1 – i.e. crisp sets (which can be also mapped back to classical bivalent sets by appropriate α-cuts).

Original approach was not very carefully chosen but we can revise articles (this is the policy like Wikipedia has however the process of submissions and revisions is supervised by the Library Committee of the Association of Mizar Users). Of course, from purely formal point of view, the Cartesian product of the set and a singleton cannot be equal to the original set regardless of the approach we will choose.

6 Attributes Defining Fuzzy Numbers

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:

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

As all types are constructed as radix types with added optional adjectives, the generalization, especially that automated-driven (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 this won't need any additional reference.

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

The proof of the above registration is trivial (note `coherence;` without any additional explanations the Mizar checker knows it is logically true taking into account only definitional expansions of the corresponding definitions and first order logic). As all Mizar types should have non-empty denotation (i.e. an example object should be constructed), it would force us to define both triangular and trapezoidal fuzzy sets. The natural definition is usually written in form similar to Eq. (1), i.e. 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 gluing counts.

```
definition let a,b,c be Real;
  assume a < b & b < c;
  func TriangularFS (a,b,c) -> FuzzySet of REAL equals
:: FUZNUM_1:def 7
    AffineMap (0,0) | (REAL \ ].a,c.[)
      +* (AffineMap (1/(b-a),-a/(b-a)) | [.a,b.])
      +* (AffineMap (-1/(c-b),c/(c-b)) | [.b,c.]);
  correctness;
end;
```

The assumptions on the ordering of real variables a, b, c seem to be unnecessary here and potentially they can be removed (as if a is not less than c, then the first part is just the affine map equal to zero for any real argument, hence fuzzy set, but not of triangular shape); we kept this as needed to prove the continuity of this fuzzy set afterwards. However, we can define this in a more natural way, avoiding somewhat cryptic `AffineMap` – per cases, but of course the compatibility property should be proved (to show that both definienses describe the same notion); such construction is called *redefinition.*

```
definition
  let a,b,c be Real;
  assume a < b & b < c;
  redefine func TriangularFS (a,b,c) -> FuzzySet of REAL means
  for x being Real holds
    ((x <= a or c <= x) implies it.x = 0) &
    (a <= x & x <= b implies it.x = (x-a)/(b-a)) &
    (b <= x & x <= c implies it.x = (c-x)/(c-b));
  correctness;
  compatibility;
end;
```

Note that we can define this more generally, without assuming that the universe is just the set of all real numbers. Many properties of fuzzy numbers are just properties of fuzzy sets over the real universe and they are automatically recognized so. Within such framework, the target type of OFN was introduced as

```
definition
  mode OrderedFuzzyNumber -> element means :OFNDef:
    ex f, g being continuous PartFunc of REAL, REAL st
    dom f = [. 0,1 .] & dom g = [. 0,1 .] & it = [f,g];
  existence;
end;
```

and to encode Eq. (1) we applied similar techniques as in defining triangular fuzzy sets; essentially we used three intervals taking on the intermediate the constant map with the value 1.

- `[.inf UP OFN, sup UP OFN.]`,
- `[.inf DOWN OFN, sup DOWN OFN.]`,
- `AffineMap (0, 1) | [.sup UP OFN, inf DOWN OFN.]`.

Then the corresponding membership function should be again the result of the gluing operation applied to inverse functions limited to these intervals where instead of writing f^{-1} we used `(up-part OFN)"`. Significant issue is that the converse of the function always exists as it is just a relation; in case of invertible function the converse is also a function.

As we noticed right under the definition of triangular fuzzy set, some restrictions (i.e. assumptions) are artificial. To show something more meaningful however, they are really needed within the proofs of the uniqueness of the peak and the continuity of the resulting fuzzy set as a whole.

```
theorem :: FUZNUM_1:29
  for a,b,c being Real st a < b & b < c holds
    TriangularFS (a,b,c) is strictly-normalized;
```

Paradoxically, most of the real technical obstacles we met during proving the continuity of the map glued from segmentally continuous parts, i.e. in the theorem

```
theorem :: FUZNUM_1:30
  for a,b,c being Real st a < b & b < c holds
    TriangularFS (a,b,c) is continuous;
```

where the main step was the following lemma:

```
theorem :: FUZNUM_1:23
  for f,g being PartFunc of REAL, REAL st
  f is continuous & g is continuous &
  ex x being object st dom f /\ dom g = {x} &
  for x being object st x in dom f /\ dom g holds f.x = g.x holds
    ex h being PartFunc of REAL, REAL st
    h = f +* g & for x being Real st
    x in dom f /\ dom g holds h is_continuous_in x;
```

which can be understood as the fact that gluing of two continuous real functions which coincide at a singleton returns again a continuous real function – pretty straightforward at first sight.

Of course, once the continuity of glued maps is established, defining another Mizar functor `TrapezoidalFS` to construct trapezoidal fuzzy set is a similar routine work.

7 Remarks on Approaches' Dissimilarities

From the viewpoint of automated proof-assistant, we were surprised that the basic difference between two approaches can be found even at the very first stage: the definitions of types *rough set* and *fuzzy set*. In fact, as we noticed before in Sec. 3, rough sets in Mizar can be treated either as a pair of the two approximation operators or the classes of abstraction with the respect to the given indiscernibility relation. The formal definition of a fuzzy set is even simpler that the classical one (see Sec. 3) – just certain membership function. Rough sets are more abstract while fuzzy sets are strongly tied with the set theory, specifically Zermelo-Fraenkel in the case of our framework (as many mathematicians claim).

Rough sets have more purely algebraic flavour, it is quite natural that they use in a significant way the automation provided by the Mizar system. In case of fuzzy sets, where concrete computations are more important [13], it is not so.

Some natural questions concerning these concrete pieces of Mizar code devoted to extensions of ordinary set theory may arise:

– Is there any other (closer to set-theoretical origins) formal definition of rough sets? On the one hand, fuzzy sets seem naturally extending classical set theory – contrary to approximation spaces. As far as we know, nobody formalized them in any other leading proof assistants (although set-theoretical axiomatic formalism is given by Bryniarski [1]);

- How to cope with the universe on which fuzzy sets are defined? According to Mizar Tarski-Grothendieck formalism, the function's value outside its domain is equal to the empty set (or zero, which is the same taking into account von Neumann definition of naturals), but we can aim at the clear distinction between a fuzzy set and the other one defined only on the support of the previous one;
- The transition from classical part to structural formalism is relatively easy (in other proof-assistants they may be called *setoids*), but the opposite direction is not that straightforward; recall that groups can be defined as ordered tuples, but in real mathematical practice this is not the case;
- It is interesting that – contrary to rough sets [7] – we don't see a clear way of generalization of the classical notion of a fuzzy set, while there are lots of such within rough set theory: from equivalence relations to tolerances, partitions vs. coverings, etc. Even mereology as a theory of a part of a whole as a primitive notion instead of an element is quite feasible in case of Pawlak's sets, so maybe because both approaches are formalized in a highly structured database, automated provers can discover new results and mark already existing similarities automatically [6];
- As the gluing of maps defined on unit interval into topological space (called *paths*) is the very basic notion in algebraic topology and homotopy theory, it would be interesting to explore the correspondence between these two important topics and OFNs.

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. It is worth noticing here that although in the current paper we focus on declarative part – definitions and theorems, all proofs can be tracked either at the home page of the Mizar project (http://mizar.org) or offline after the installation of the Mizar system.

Although it is possible in Mizar to prove theorems without proving lemmas they depend on before, we aimed at the formalization of all the content just from its basics. The other issue is that the theorems which are "conditionally" proven will not be accepted as a part of the Mizar Mathematical Library, so they are formally useless in some sense. Such approach can be useful, however, from the viewpoint of more advanced topics (preliminaries can be just stated without proving).

8 Conclusion and Further Work

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. Also as we explained before, 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 (including 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 (a kind of programming) 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. Good example here (other than the aforementioned) are BCK algebras which are already formalized – this can start the development of fuzzy logic within MML. Even if the formalized content concerning fuzzy sets is not that big as of now (there is about 15 thousand lines of Mizar code on fuzzy and rough 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 [3], 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 [22] tools translating Mizar language into the input of first-order theorem-provers (TPTP – Thousands of Problems for Theorem Provers), proof simplification via lemma extraction [21] 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 time-consuming proofs to be done by computer. Of course, still all these achievements are subject to careful human supervising in order to provide the proper research background.

References

1. Bryniarski, E.: Formal conception of rough sets. Fundamenta Informaticae 27(2/3), 109–136 (1996)
2. Dubois, D., Prade, H.: Operations on fuzzy numbers. International Journal of System Sciences 9(6), 613–626 (1978)
3. Dubois, D., Prade, H.: Rough fuzzy sets and fuzzy rough sets. International Journal of General Systems 17(2–3), 191–209 (1990)
4. Grabowski, A.: The formal construction of fuzzy numbers. Formalized Mathematics 22(4), 313–319 (2014)
5. Grabowski, A.: On the computer certification of fuzzy numbers. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) Proceedings of Federated Conference on Computer Science and Information Systems, FedCSIS 2013, pp. 51–54 (2013)
6. Grabowski, A.: Automated discovery of properties of rough sets. Fundamenta Informaticae 128(1-2), 65–79 (2013)

7. Grabowski, A.: On the computer-assisted reasoning about rough sets. In: Dunin-Kęplicz, B., Jankowski, A., Szczuka, M. (eds.) Monitoring, Security and Rescue Techniques in Multiagent Systems. Advances in Soft Computing, vol. 28, pp. 215–226 (2005)
8. Grabowski, A., Jastrzębska, M.: Rough set theory from a math-assistant perspective. In: Kryszkiewicz, M., Peters, J.F., Rybiński, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 152–161. Springer, Heidelberg (2007)
9. Grabowski, A., Korniłowicz, A., Naumowicz, A.: Mizar in a nutshell. Journal of Formalized Reasoning 3(2), 153–245 (2010)
10. Kacprzak, M., Kosiński, W.: On lattice structure and implications on ordered fuzzy numbers. In: Proc. of EUSFLAT 2011, pp. 267–274 (2011)
11. Kawahara, Y., Furusawa, H.: An algebraic formalization of fuzzy relations. Fuzzy Sets and Systems 101, 125–135 (1999)
12. Klir, G.J.: Fuzzy arithmetic with requisite constraints. Fuzzy Sets and Systems 91, 165–175 (1997)
13. Koleśnik, R., Prokopowicz, P., Kosiński, W.: Fuzzy calculator – useful tool for programming with fuzzy algebra. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 320–325. Springer, Heidelberg (2004)
14. Korniłowicz, A.: On rewriting rules in Mizar. Journal of Automated Reasoning 50(2), 203–210 (2013)
15. Kosiński, W., Prokopowicz, P., Ślęzak, D.: Fuzzy numbers with algebraic operations: algorithmic approach. In: Kłopotek, M., Wierzchoń, S.T., Michalewicz, M. (eds.) Intelligent Information Systems, IIS 2002, Poland, pp. 311–320 (2002)
16. Kosiński, W., Prokopowicz, P., Ślęzak, D.: Ordered fuzzy numbers. Bulletin of the Polish Academy of Sciences, Sér. Sci. Math. 51(3), 327–338 (2003)
17. Mitsuishi, T., Endou, N., Shidama, Y.: The concept of fuzzy set and membership function and basic properties of fuzzy set operation. Formalized Mathematics 9(2), 351–356 (2001)
18. Moore, R., Lodwick, W.: Interval analysis and fuzzy set theory. Fuzzy Sets and Systems 135(1), 5–9 (2003)
19. Naumowicz, A., Korniłowicz, A.: A brief overview of Mizar. In: Berghofer, S., Nipkow, T., Urban, C., Wenzel, M. (eds.) TPHOLs 2009. LNCS, vol. 5674, pp. 67–72. Springer, Heidelberg (2009)
20. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer, Dordrecht (1991)
21. Pąk, K.: Methods of lemma extraction in natural deduction proofs. Journal of Automated Reasoning 50(2), 217–228 (2013)
22. Urban, J., Sutcliffe, G.: Automated reasoning and presentation support for formalizing mathematics in Mizar. In: Autexier, S., Calmet, J., Delahaye, D., Ion, P.D.F., Rideau, L., Rioboo, R., Sexton, A.P. (eds.) AISC 2010. LNCS, vol. 6167, pp. 132–146. Springer, Heidelberg (2010)
23. Wiedijk, F.: Formal proof – getting started. Notices of the AMS 55(11), 1408–1414 (2008)
24. Zadeh, L.: Fuzzy sets. Information and Control 8(3), 338–353 (1965)