Computer-Driven Searching for Axiomatization of Rough Sets

Adam Grabowski

Institute of Informatics
University of Białystok
ul. Akademicka 2
15-267 Białystok, Poland
adam@math.uwb.edu.pl

Abstract. The formalization of rough sets in a way understable by machines seems to be far beyond the test phase. For further research, we try to encode some problems within RST and as the testbed of already developed foundations – and in the same time as a payoff of the established framework, we shed some new light on the well-known question of generalization of rough sets and the axiomatization of approximation operators in terms of (various types of) relations.

1 Introduction

During the past decades, mathematics would evolve from the pen-and-paper model in the direction of use of computers. As rough set theory proposes new methods and tools which enhance the reasoning about uncertain or incomplete information, dealing mainly with those contained in digital repositories, it is not surprising that similar methods will be used in order to obtain the properties of objects within the theory itself.

Potentially, the methods of automated reasoning can be applied to any problems which can be reduced to the first order logic, but the application of existing provers to the questions which arose in rough set theory is not that straightforward as one can expect to be.

We deal with the Zhu's paper [13] on the axiomatization of various generalizations of rough sets with respect to ordinary binary relation properties. It was not suprising that the original Pawlak's approach [6] would soon be generalized, and so did Skowron and Stepaniuk [7] and Zhu [13], among others. But similar work, although precious and carefully driven, can be done by computers.

Rough sets deliver important tools to discover data from databases, it is now especially valuable taking into account the amount of stored information and the form of the records. Digitization of mathematical journals gets more and more popular, and it is often the case of:

– new material, when papers can be published faster, so information exchange, and hence research is more efficient and accessible – here the well-known example could be Springer's Online First;
– archival issues/journals.

Obviously, OCR is not the only activity in the latter case – at least bibliography section should be identified to count impact factors properly. We try to address some issues concerned with the digitization of the fragment of RST, representing a report from a certain part – so it can be considered as a case study in a knowledge management, being a work on rough sets in the same time.

The paper is organized as follows. In the next section we describe the current situation in the area of computer-checked formalization of mathematics. The third section is devoted specifically to the Mizar library, one of the leading mechanical repositories in the world, while in the fourth the outline of the existing formalization of rough sets is given. The other three sections contain possible (and already developed) models of rough sets and the draft of how properties of rough approximations can be mapped with those of binary relations. The final section brings some concluding remarks and the plans for future work.

2 State of the Art

"Formalization" is a term with a broad meaning which denoted rewriting the text in a specific manner, usually in a rigorous (i.e. strictly controlled by certain rules), although sometimes cryptic language. Obviously the notion itself is rather old, originated definitely from pre-computer era, and in the early years formalization was to ensure the correctness of the approach.

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, 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 RST can be also.

To give more or less formal definition, according to Wiedijk [10], *the formalization* can be seen presently as "the translation into a formal (i.e. rigorous) language so computers check this for correctness."

3 The Mizar Mathematical Library

The Mizar system [8] 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 bigest repositories of computer checked mathematical knowledge in the world.

The basic item in the MML is called 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, although the earlier needs some additional care.

For three years now, the most developed disciplines are general topology (steered by Trybulec, Białystok, Poland) and functional analysis (led by Shidama, Nagano, Japan). Also the author took a part in a large project of translating a compendium *Continuous Lattices and Domains* with a significant success. 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.

4 Rough Sets, Classically

As the notions of the upper and the lower approximations are expected to be well known, we cite here only the translation into the Mizar language of a single notion in the primary setting.

```
definition
  let A be Approximation_Space;
  let X be Subset of A;
  func LAp X -> Subset of A equals
:: ROUGHS_1:def 4
    { x where x is Element of A : Class (the InternalRel of A, x) c= X };
end;
```

All basic formalized definitions and theorems about rough sets can be tracked under the address `http://mizar.org/version/current/html/roughs_1.html`. This restricted A to be an approximation space, which is non empty relational structure with the internal relation to be just equivalence. Note that equivalence relations are defined via Mizar attributes (lexically, adjectives), which reflect natural adjectives corresponding to the properties of relations. This offers simple mechanism of the (very basic level of) generalization: we get a variable under universal quantifier and from its type[1] we cut off adjectives, one by one. If the inference is still accepted, the assumption hidden under this adjective is unnecessary, otherwise – it isn't. Of course, the generalization can force another way of proving things, but we do not care about this at this time.

[1] On the contrary to the most provers, the Mizar language is typed.

5 A Summary of the Formalization of Zhu

In this section we focus on the details contained in [13].[2] We noticed that the two notions of a general character quoted by Zhu, i.e. seriality and mediateness of a binary relation, were not defined in MML, so it had to be introduced by us – of course, this time we deal only with relations; no structures were directly involved.

> Let R be relation on U. If $\forall_{x \in U} \exists_{y \in U} \; xRy$, then we say R is a serial relation.

```
definition let R be Relation, X be set;
  pred R is_serial_in X means :Def0:
    x in X implies ex y st y in X & [x,y] in R;
end;
```

These newly defined properties of relations will probably be moved to the part of MML devoted to relations. In classical setting however, as it can be tracked in [2], we based our development on relational structures rather than relations. So the structure is meant to be serial if its internal relation is serial in the sense of `Def0`. It appeared to be pretty feasible, because we could formalize natural properties in a rather compact way, as shown below:

```
theorem TwPro2L: :: Prop2 2L
  for R being non empty total serial RelStr holds
    LAp {}R = {}R
  proof
    let R be non empty total serial RelStr;
    {x where x is Element of R :
      Class (the InternalRel of R, x) c= {}R} c= {}R
    proof
      let y be set;
      assume A1: y in { x where x is Element of R :
        Class (the InternalRel of R, x) c= {}R}; then
      consider z being Element of R such that
A2:   z = y & Class (the InternalRel of R,z) c= {}R;
A3:   Class (the InternalRel of R,z) = {}R by A2;
      Class (the InternalRel of R,z) <> {}R by help2;
      hence thesis by A2;
    end;
    hence thesis by ROUGHS_1:def 4;
  end;
```

The proof of `TwPro2L` contains implicitly only one of the inclusions, because the system knows that the empty set is contained in any set. Observe that `help2` is a reference for the previously stated lemma:

[2] We hope to succeed with full certification of this paper soon, but as of now we did not the work completely. As of now, our Mizar source code is about 2000 lines long.

```
theorem help2:
  for R being non empty total serial RelStr, x being Element of R holds
    Class (the InternalRel of R,x) <> {};
```

Note that it should be formulated, but not necessarily proven before. Although the error flagged "This inference is not accepted" is called, the verification doesn't stop (on the contrary to usual compilers).

Based on the simple example, we can show can many inferences can be taken into account automatically. It is a simple observation that all binary relations which are symmetric, transitive, and serial are also reflexive. This can actually be proven as the lemma, but to be exported to the Mizar database, it should be labelled as theorem.

```
theorem Lemma1:
  (R is symmetric & R is transitive & R is serial) implies
    R is reflexive;
```

This implication can be expressed in a bit more compact way (in the same time however, this way is less readable for the mathematician not acquainted with the syntax of the Mizar language). Obviously, referencing for the simple technical lemma above works.

```
registration
  cluster symmetric transitive serial -> reflexive for Relation;
  coherence by Lemma1;
end;
```

After this registration of a cluster the Mizar verifier every time when the binary relation R has the three properties written on the left hand side of the sign `->` adds also the property on the right hand side of the arrow to R, and after that reflexivity is obtained without any ingerence of the user. This mechanism allows us to simplify the proofs, not much affecting its understanding for human.

6 Functional Apparatus at Work

Let us take into account another property.

```
theorem :: Prop2 4L
  for R being non empty RelStr, X, Y being Subset of R holds
    LAp (X /\ Y) = LAp (X) /\ LAp (Y);
```

Consequently, we defined the map which for every subset A of the approximation space R returns its upper approximation (because the `UAp` is a Mizar *functor*, i.e. operator of linguistic, not of the mathematical character.

```
definition let R be non empty RelStr;
  func UAp R -> Function of bool the carrier of R, bool the carrier of R
  means :DefU:
  for X being Subset of R holds it.X = UAp X;
end;
```

Unfortunately, we had to pay for the use of the Mizar notion of a relational structure – in the correspondence of the existence of the upper approximation operator and the function with the properties as shown below we had to propose a lifting – after we construct the appropriate relation, we had to define underlying structure, which was technically standard, but it needed additional proof step in every single Zhu's proposition:

> If an operation $H : P(U) \to P(U)$ satisfies the following properties: $H(\emptyset) = \emptyset$ and $H(X \cup Y) = H(X) \cup H(Y)$, then there exists a relation R on U such that $H = H(R)$.

In fact, every time we proven that there exists not *a relation*, but *a relational structure* – for human it doesn't make much difference, for computer – it is just another object.

```
theorem :: Prop1
  for A being non empty finite set,
      U being Function of bool A, bool A st
  U.{} = {} &
  (for X, Y being Subset of A holds U.(X \/ Y) = U.X \/ U.Y) holds
  ex R being non empty finite RelStr st
  the carrier of R = A & U = UAp R;
```

This is really the place where much work can still be done; although many generalizations are already present in the literature, taking the aforementioned [7] and [13] as simple examples, the knowledge should be still gathered across many various papers.

7 Possibility of Other Approaches

Although the repository of mathematical facts expressed in the Mizar language as a rule is rather uniform, additional background knowledge can be applied to shed some new light on old topics. Such disciplines as general topology and lattice theory can be reused and extended. We can cite here interval sets [3] and merging with lattices [4] as examples of such successful reuse (in the spirit of [5]).

As we noted in the previous section, Zhu in [13] dealt with selected properties of binary relations.

Although formal approach tends to be rather unique, multiple views for the same notion (treated informally) are also possible.

Here we can point out two issues: the notion of relational structure vs. relation and the possibility of defining things in parallel ways. As the example of the place where two various approached can meet, we will list the following:

```
definition let R be non empty reflexive RelStr;
  redefine attr R is serial means
    for x being Element of R ex y being Element of R st x <= y;
  compatibility;
end;
```

So, the seriality expressed in the fifth section with the help of a bit confusing syntax, can now be understood without a pain (but the `compatibility` property that these two notions coincide, should be technically justified).

Of course, one can ask whether all this relational structures' apparatus should be reused. The answer is naturally negative. But then we couldn't benefit from the aforementioned simple poset-like description.

We can freely develop the generalization work reusing the available knowledge of topology (as, e.g. [14]) because general topology is pretty well represented in the repository of Mizar texts and there are feasible mechanisms allowing to merge structures (in our case, relational structures with topological ones).

8 Conclusion and Further Work

One can ask a question of why the generalization of rough sets can be discussed again and again as there are so many approaches to do so in the literature. Computer certification of proofs seems to be an emerging trend and some corresponding issues can be raised. We see some pros of our approach:

- repetitions are no longer justified (as they can be discovered automatically);
- possible generalizations, even those computer-driven;
- possibility of the automatic obtaining of new results – knowledge discovery;
- available via translation interfaces for other math-assistants;
- understandable and unified;
- automatically translated for human-oriented language;
- students can play with it with not much support from the teacher's side;
- machine can be better reviewer than human, potentially.

But some things are not that beautiful:

- although syntax is close to natural language and the number of keywords is rather limited, at least basic knowledge of computer tools is needed;
- this can take a lot of time to formalize a hard problem;
- in the real mathematics, many problems are solved based on theorems from other disciplines; even if MML is built on the axioms of set theory, one can imagine e.g. building a part of MML under the assumption of the axiom of determinacy without any discussion of the inconsistency of such system; but then any theorem proven with the help of AD will have explicit assumption of AD. As a rule, to be accepted for inclusion into MML, all such preliminary facts have to proven as well.

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.

Although existing provers are best known in the area of finding short axiomatizations of various logical systems (yet classical problem of Robbins algebras), the other possibilities can enhance this framework. Urban's [9] tools translating Mizar language into the input of first-order theorem-provers (TPTP – Thousands of Problems for Theorem Provers) or XML interface providing information exchange between various math-assistants are already in use. We expect also that it can allow for some “mechanical work” as in [1] did by computer. But earlier careful background should be prepared to obtain the results and interpret them in a proper way.

References

1. A. Gomolińska, On certain rough inclusion functions, *Transactions on Rough Sets,* 9, 35–55, 2008.
2. A. Grabowski, Basic properties of rough sets and rough membership function, *Formalized Mathematics,* 12(1), 21–28, 2004; can be tracked also under `http://mizar.org/version/current/html/roughs_1.html`.
3. A. Grabowski, M. Jastrzębska, On the lattice of intervals and rough sets, *Formalized Mathematics,* 17(4), 237–244, 2009.
4. A. Grabowski, Ch. Schwarzweller, Rough Concept Analysis – theory development in the Mizar system, *Lecture Notes in Computer Science,* 3119, 130–144, 2004.
5. J. Järvinen, Lattice theory for rough sets, *Transactions on Rough Sets VI,* 400–498, 2007.
6. Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data,* Kluwer, Dordrecht, 1991.
7. A. Skowron, J. Stepaniuk, Tolerance approximation spaces, *Fundamenta Informaticae*, 27, 245–253, 1996.
8. The Mizar Home Page, `http://mizar.org/`.
9. J. Urban, G. Sutcliffe, Automated reasoning and presentation support for formalizing mathematics in Mizar, *Lecture Notes in Computer Science,* 6167, 132–146, 2010.
10. F. Wiedijk, Formal proof – getting started, *Notices of the AMS,* 55(11), 1408–1414, 2008.
11. Y.Y. Yao, Constructive and algebraic methods of theory of rough sets, *Information Sciences*, 109, 21–47, 1998.
12. Y.Y. Yao, X. Li, Comparison of rough-set and interval-set models for uncertain reasoning, *Fundamenta Informaticae*, 27, 289–298, 1996.
13. W. Zhu, Generalized rough sets based on relations, *Information Sciences,* 177, 4997–5011, 2007.
14. W. Zhu, Relationship between generalized rough sets based on binary relations and covering, *Information Sciences,* 178(21), 4105–4113, 2008.