Two Formal Approaches to Rough Sets

Adam Grabowski and Magdalena Jastrzębska

Institute of Mathematics
University of Białystok
ul. Akademicka 2
15-267 Białystok, Poland
adam@math.uwb.edu.pl, magja@osiedle.net.pl

Abstract. The formalization of rough sets in a way understable to machines seems to be far beyond the test phase. For further research, we try to encode the bunch of classical papers within RST and as the testbed of already developed foundations of the theory we try to adopt the interval set model to put it within the existing set-theory machinery in the Mizar computer-checked repository.

1 Preliminaries

During the past decades, mathematics evolved from the pen-and-paper model in the direction of extensive use of computers. Digitization of mathematical journals gets more and more popular, and it is often the case of

- the 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 of journals.

Obviously, simple optical character recognition (OCR) is not the only activity in the latter case – at least the bibliography section could be identified to calculate impact factors properly. These activities can however be done independently, unlike the formalization efforts. If we try to reach the research frontier, i.e. to formalize new results, either solid background should be provided, or the discipline should be relatively new.

We try to address some issues concerning the formalization of a fragment of rough set theory using the Mizar language, presenting a report on the current state of the work. RST delivers 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. The discipline is rather an emerging trend, and however the stress is put on applications, some valuable results are already available.

The main pros of the use of the Mizar system are as follows:

- repetitions are no longer justified;

ISBN 978-83-7431-229-5 ISSN 0860–150X

– possible generalizations, even those computer-driven;
– possibility of the automatic obtaining new results – the area of knowledge discovery.

The paper is organized as follows. The next section is devoted specifically to the Mizar library, one of the leading mechanical repositories in the world, then follows an outline of one of the most popular models for rough sets is given – namely interval sets. Finally we describe our formalization efforts. The final section brings some concluding remarks and the plans for future work.

2 The Repository

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 an oracle to check if the text is correct.

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 is too high, only software engineers will be attracted and all the usefulness for an ordinary mathematician will be lost.

The 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, called a Mizar article, 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 part, the former needs some additional care – the submission will be accepted for inclusion into the MML if the approach is correct and the topic is not already present there.

In recent years, the most intensively developed disciplines were general topology (steered by Trybulec, Białystok, Poland) and functional analysis (led by Shidama, Nagano, Japan). The first author took part in a large project of translating a compendium *Continuous Lattices and Domains* with a significant success. As a by-product of this encoding, apart from quite readable Mizar scripts, also the presentation of the source which is accessible to ordinary mathematicians and pure HTML form with clickable links to notions and theorems are available.

Since mathematics in MML is based on ZFC set theory, and the notion of structures is of somewhat other character, the basic division of the repository into two parts is provided: the articles which do not use the notion of a structure (forming the so called concrete part) and the remaining, abstract part of MML. This division stimulates the enhancement of the library, forcing the movement of preliminary concrete items from the abstract part of the library.

This can be considered a drawback of the MML – obviously the chosen set theory cannot be changed easily, and the formalization of work which strongly depends (or just proposes another axiomatics) on a set theory other than ZFC, e.g. Bryniarski's work [1] is not that straightforward.

3 The Theory, Informally

There are two popular extensions of the classical set theory – one of them, Zadeh's fuzzy sets, are already present in the MML [7]. The rough-set (and corresponding interval-set) model is another important extension for modelling vagueness, where information is incomplete or imprecise. One of the views for rough sets is operator-oriented, with approximations defined as two additional set-theoretic objects. The other is set-oriented view, taking a rough set as the family of sets having the same upper and lower approximations. In the interval-set model, the range of the unknown set is given as a pair of its bounds. Hence virtually any member of such family can be the considered set. Although both models have strong logical foundations, we will focus here rather on the construction of appropriate algebraic models.

3.1 Rough sets

Rough sets, which were introduced by Pawlak [10], are often viewed through the prism of applications, especially when the information is incomplete or uncertain. Given a finite non-empty universe U and the relation R defined on U which in the original Pawlak's approach is an equivalence relation, i.e. is reflexive, symmetric and transitive, a partition U/R of U into equivalence classes E_i, called elementary sets, is given. The pair $\langle U, R\rangle$ is called an approximation space. From the agent's point of view, elements which belong to the same class of R are indiscernible.

In the approximation space $\langle U, R\rangle$ one can define for an arbitrary set $A \subseteq U$ the two operators of the lower and upper approximation.

$$A_\star = \bigcup_{E_i \subseteq A} E_i = \{x \in U : [x]_R \subseteq A\}$$

$$A^\star = \bigcup_{E_i \cap A \neq \emptyset} E_i = \{x \in U : [x]_R \cap A \neq \emptyset\}$$

Potentially, we can consider elementary sets as just elements of a partition of U with R as a hidden argument. It is clear that if equivalence classes are singletons, then the rough set, treated as a pair of $A_\star$ and $A^\star$ reduces to the ordinary set A. Lattices of rough sets were studied primarily by Iwiński in [5] in the original Pawlak's setting. In our case we drop the assumption of reflexivity of R, i.e. in case of symmetric and transitive relations, or in case of tolerance relations, the lattice retains some basic properties. This generalization can go even further, with the obvious change from $[x]_R$ into the image $R(x)$.

3.2 Interval sets

According to the interval-set approach, over a finite non-empty universe U we can think of the interval set $\mathcal{A}$ as a pair of two sets, $A_1, A_2 \in 2^U$, where $A_1 \subseteq A_2$ as follows:

$$\mathcal{A} = [A_1, A_2] = \{X \in 2^U : A_1 \subseteq X \subseteq A_2\}.$$

Given two intervals, $\mathcal{A} = [A_1, A_2]$ and $\mathcal{B} = [B_1, B_2]$. We define the interval-set intersection, union and difference as

$$\mathcal{A} \sqcap \mathcal{B} = \{A \cap B : A \in \mathcal{A}, B \in \mathcal{B}\}$$

$$\mathcal{A} \sqcup \mathcal{B} = \{A \cup B : A \in \mathcal{A}, B \in \mathcal{B}\}$$

$$\mathcal{A} \setminus \mathcal{B} = \{A \setminus B : A \in \mathcal{A}, B \in \mathcal{B}\}$$

Further on, we can define the interval-set complement by putting

$$\neg[A_1, A_2] = [U, U] \setminus [A_1, A_2].$$

It is clear that $I(2^U)$, the set of all intervals over U, together with the above operations, forms a completely distributive lattice which is not a Boolean algebra, since for an interval set A, $A \sqcap \neg A$ is not necessarily equal to $[\emptyset, \emptyset]$, similarly $A \sqcup \neg A$ is not necessarily equal to $[U, U]$.

4 Formalization of Intervals

In this section we describe briefly main points of the formalization of (lattices of) intervals we recently completed in [4].

Tolerance approximation spaces, i.e. the framework in which rough sets are defined, are relational structures. An important fact here is that the formalization of interval sets does not depend on the notion of a structure (essentially the operations on such sets can be viewed as the operations applied to ordered pairs, or as collective operations on families of subsets).

4.1 A mixture of modes and functors

Since appropriate correctness conditions should be proven for all definitions (the existence for both modes and functors and additionally the uniqueness for functors), the Library Committee suggests to use the functors instead of modes when possible to prove that they are uniquely determined (one of the exceptions is given in Section 5.1). According to this policy, we are forced to use the notion of the interval set parametrized by its ends.

```
definition let U be set;
           let X, Y be Subset of U;
  func Inter (X,Y) -> Subset-Family of U equals
:: INTERVA1:def 1
    { A where A is Subset of U : X c= A & A c= Y };
end;
```

But we needed also more general notion, the class of all intervals over given universe.

```
definition let U be set;
  mode IntervalSet of U -> Subset-Family of U means
:: INTERVA1:def 2
    ex A, B be Subset of U st it = Inter (A, B);
end;
```

The last redefinition is to restrict the result type of the functor `Inter` and to ensure that we can use operations defined on itervals to concrete intervals determined by its ends.

```
definition let U; let A, B be Subset of U;
  redefine func Inter (A, B) -> IntervalSet of U;
end;
```

4.2 Operations on intervals

On the one hand, treating interval sets as families of subsets, we can consider lattice operations on them as corresponding set-theoretical collective operations.

```
definition let U be non empty set,
               A, B be non empty IntervalSet of U;
  func A _/\_ B -> IntervalSet of U equals
:: INTERVA1:def 3
    INTERSECTION (A, B);
end;
```

Using the "equals" construction both constructors (i.e. INTERSECTION and "`_/\_`" operation) are automatically unified.

```
definition let SFX, SFY be set;
  func INTERSECTION (SFX,SFY) means
:: SETFAM_1:def 5
     Z in it iff ex X,Y st X in SFX & Y in SFY & Z = X /\ Y;
end;
```

Note that these operations, named INTERSECTION and UNION in the Mizar formalism, in fact were originally defined on families of subsets, and later the types of arguments were generalized just into sets. Of course, basic translation lemmas are provided.

```
theorem :: INTERVA1:12
  for U being non empty set,
      A1, A2, B1, B2 being Subset of U st
        A1 c= A2 & B1 c= B2 holds
    INTERSECTION (Inter (A1,A2), Inter (B1,B2)) =
      { C where C is Subset of U : A1 /\ B1 c= C & C c= A2 /\ B2 };
```

We proved some necessary lemmas – properties of intervals, and instead of using concrete functors (`Inter`), we introduced an attribute with a similar meaning.

```
definition let X be set; let F be Subset-Family of X;
  attr F is ordered means
:: INTERVA1:def 8
    ex A, B being set st
      A in F & B in F & for Y being set holds
        Y in F iff A c= Y & Y c= B;
end;
```

We were surprised why the following was not proven before – Mizar article SETFAM_1 [9] is dated back to 1989; apparently for twenty years nobody had needed this elementary fact of distributivity or just the proof is hidden too deeply (simple searching for simultaneous use of both functors was unsuccessful).

```
theorem :: INTERVA1:30
  for X being set, A,B,C being non empty ordered Subset-Family of X holds
    UNION (A, INTERSECTION (B,C)) =
      INTERSECTION (UNION (A,B), UNION (A,C));
```

4.3 Linking the classical set theory

Note that min and max synonyms were introduced to set-theoretical meet and union, respectively, because interval's ends can be calculated so.

```
theorem :: INTERVA1:27
  for A,B being Subset of U,
      F being ordered non empty Subset-Family of U st
      F = Inter (A,B) holds
      min F = A & max F = B;
```

Speaking informally about the extension of the classical set theory we can avoid some real formal difficulties (as in the case of the set of complex numbers which is an extension of the set of all reals, but was originally defined in the MML as the Cartesian square $\mathbb{R}^2$). Hence, trivial intervals determined by A correspond to the singletons of A.

4.4 The complementation operator

The complementation has a slightly different notation than the ordinary set-theoretic one in order to avoid the overloading (remember A is a family of subsets of the universe U, and hence a subset of 2^U):

```
definition let U be non empty set, A be non empty IntervalSet of U;
  func A ^ -> non empty IntervalSet of U equals
:: INTERVA1:def 10
    Inter ([#]U,[#]U) _\_ A;
end;
```

Similar overloading would be dangerous in the case of projections `‘1` and `‘2` – originally these were just the coordinates of the Cartesian product. We defined the `A‘‘1` and `A‘‘1` as the left-hand-side and the right-hand-side ends of the interval A (or the meet and the union of A).

```
theorem :: INTERVA1:46
  for U being non empty set, A being non empty IntervalSet of U holds
    A^ = Inter ((A‘‘2)‘,(A‘‘1)‘);
```

Within the Mizar library some useful special objects are constructed when needed to register some existential clusters, and counterexamples are considered usually to illustrate the topic to students. Since interval sets act in some cases not as classical crisp sets, we find it useful to also include them in the article [4].

```
theorem :: INTERVA1:54
  for A being non empty IntervalSet of U holds
    {} in A _/\_ (A^);
```

5 Rough Sets Revisited

The problem with the rough sets is that in order to fully reuse the expressive power of the Mizar language and to reflect the current state of the MML, unlike the intervals, rough sets should be defined using the abstract part of the Mizar library. On the other hand, to benefit from the generalized approach to sets via rough sets, it would be necessary to have them in the concrete part. This, however, is impossible, because the rough sets were defined in Mizar over the tolerance space, i.e. the structure belonging to the latter part of MML.

5.1 Pairs vs. subsets

We will not describe here the basic notions of the formalized approximations, the details can be found in [2] and [3]. They are defined in Mizar pretty closely to the natural language according to Section 3.1. The only doubt was whether to choose the definition of a rough set as a union of elementary sets or the latter one, in terms of equivalence classes. The former can omit the notion of indiscernibility relation, but we decided to follow the latter way as it is used more often. Obviously, none of the properties of indiscernibility relation, neither transitivity, nor even symmetry is assumed. The properties are added later to show essential properties of the operators, only where needed.

For example, the lower approximation $X_\star$ of a rough set X in the approximation space A is given by

```
definition let A be non empty RelStr;
           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;
```

where `Class` is just another name for the image of a relation (as a result of a revision, originally the class of abstraction of an equivalence relation).

```
definition
  let A be Approximation_Space;
  let X be Subset of A;
  mode RoughSet of X means
:: ROUGHS_1:def 8
    it = [LAp X, UAp X];
end;
```

Potentially, this mode can be defined as a functor (it is unique), but we found it useful to have both views for rough sets formalized verbatim (with the other representation just as the subset of the tolerance approximation space, being rough in the case of the different lower and upper approximations, and exact or crisp, otherwise).

```
definition let X be Tolerance_Space;
  mode RoughSet of X ->
    Element of [:bool the carrier of X, bool the carrier of X:] means
:: INTERVA1:def 13
    not contradiction;
end;
```

As it can be easily seen, both notions coincide, but to ensure the correspondence between the two models, the following operator which converts a subset of approximation space (with the tolerance relation as a hidden argument) into the pair of sets was introduced:

```
definition let X be Tolerance_Space, A be Subset of X;
  func RS A -> RoughSet of X equals
:: INTERVA1:def 14
    [LAp A, UAp A];
end;
```

5.2 Lattice-theoretical approach

In many real-life applications, lattice theory usually serves well as the source of appropriate models, hence many examples of lattices can be found in the MML, e.g. lattices of subgroups, subspaces of a vector space, real numbers or topological domains. Also lattices of fuzzy sets are formalized there.

```
definition let X be Tolerance_Space;
  func RSLattice X -> strict LattStr means
:: INTERVA1:def 23
    the carrier of it = RoughSets X &
    for A, B being Element of RoughSets X,
        A', B' being RoughSet of X st A = A' & B = B' holds
     (the L_join of it).(A,B) = A' _\/_ B' &
     (the L_meet of it).(A,B) = A' _/\_ B';
end;
```

We have shown basic properties of the lattice of rough sets, such as distributivity, boundedness and completeness (Stone algebras are not yet defined in the MML), most of them formulated as functorial registrations of adjective clusters of the form

```
registration let X be Tolerance_Space;
  cluster RSLattice X -> complete;
end;
```

Besides the obvious pros of using a computer math-assistant we can point out here two main gains of this formalization. From the viewpoint of rough-set theory, it is the inheritance – in Mizar the structures can be freely extended, e.g. to apply to the lattice structures given a topology, or an ordering relation (since lattices and posets are developed in some sense independently, or in parallel).

From the viewpoint of the Mizar community the gain is that some evident gaps were identified and filled in, like the aforementioned distributivity of collective intersection and union.

6 Conclusions and Further Work

Regardless of the concrete formal definition of rough sets chosen (either as a pair of approximations or as the equivalence class of indiscernibility relation), many interesting algebraic models of rough sets are presented, see e.g. Pomykała [11] or the aforementioned Iwiński [5]. Also Bryniarski [1] offers a formal approach which is pretty close to the Mizar formalism in its style, although the motivation is somewhat different (and also computers were not used there).

Based on the MML, we can be sure that a thorough exploration of the theory, like the formalization of "Continuous Lattices and Domains" or general topology points out much better existing gaps and possible improvements of this repository. Unlike RST, these disciplines have standard textbooks, and this can make this work harder. Our goal is to formalize (or, more precisely, to map, because many of the facts are already available in the MML) Järvinen's paper [6]. Having constructed the structure of intervals and rough sets, further research will be continued. As a by-product, we can also reuse general topology which is the area where roughness could also be studied – with the obvious example of the upper and lower approximation operators as the topological closure and interior.

In parallel, the authors from China (not directly involved in this project) wrote an article about the properties of rough subgroups, already accepted for inclusion into the MML, as an extension of the existing group theory corpus.

Strenghtening of the Mizar checker will hopefully decrease the de Bruijn factor (the quotient of the size of a formalization of a mathematical text and the size of its informal original), and hence, the proofs will be more compact, with the readability unaffected.

Although existing provers are best known in the area of finding short axiomatizations of various logical systems (like the classical problem of Robbins algebras), other possibilities can enhance this framework. Urban's [13] tools translating the

Mizar language into the input of first-order theorem provers or XML interface providing information exchange between various math-assistants are already in use. Here we can foresee exploration of the properties of certain lattices.

References

1. Bryniarski, E.: *Formal conception of rough sets,* Fundamenta Informaticae 27(2–3), pp. 109–136, 1996.
2. Grabowski, A.: *Basic properties of rough sets and rough membership function,* Formalized Mathematics, 12(1), pp. 21–28, 2004.
3. Grabowski, A.: *On the computer-assisted reasoning about rough sets,* in B. Kęplicz et al. (Eds.), Monitoring, Security and Rescue Techniques in Multiagent Systems, *Advances in Soft Computing,* Springer, pp. 215–226, 2005.
4. Grabowski, A. and Jastrzębska, M.: *On the lattice of intervals and rough sets*, to appear in Formalized Mathematics, 2009.
5. Iwiński, T.: *Algebraic approach to rough sets,* Bulletin of the Polish Academy of Sciences. Mathematics, 35, pp. 673–683, 1987.
6. Järvinen, J.: *Lattice theory for rough sets,* in J.F. Peters et al. (Eds.), Transactions on Rough Sets VI, LNCS 4374, pp. 400–498, 2007.
7. Mitsuishi, T., Endou, N. and Shidama, Y.: *The concept of fuzzy set and membership function and basic properties of fuzzy set operation,* Formalized Mathematics, 9(2), pp. 351–356, 2001.
8. Mizar Home Page, `http://mizar.org/`.
9. Padlewska, B.: *Families of sets*, Formalized Mathematics, 1(1), pp. 147–152, 1990.
10. Pawlak, Z.: *Rough sets,* International Journal of Computer and Information Sciences, 11, pp. 341–356, 1982.
11. Pomykała, J. and Pomykała, J.A.: *The Stone algebra of rough sets,* Bulletin of the Polish Academy of Sciences. Mathematics, 36(7–8), pp. 495–508, 1988.
12. Trybulec, A.: Some Features of the Mizar Language. In Proceedings of ESPRIT Workshop, Torino, 1993, available at `http://mizar.uwb.edu.pl/project/trybulec93.ps`.
13. Urban, J.: *Translating Mizar for first order theorem provers,* LNCS 2594, pp. 203–215, 2003.
14. Yao, Y.Y. and Li, X.: *Comparison of rough-set and interval-set models for uncertain reasoning,* Fundamenta Informaticae, 27(2–3), pp. 289–298, 1996.