CrossMark

Four Decades of Mizar

Foreword

Adam Grabowski[1] · **Artur Korniłowicz**[1] ·
Adam Naumowicz[1]

Abstract This special issue is dedicated to works related to MIZAR, the theorem proving project started by Andrzej Trybulec in the 1970s, and other automated proof checking systems used for formalizing mathematics.

1 Introduction

The MIZAR project started under the leadership of Andrzej Trybulec, at the Płock Scientific Society, Poland. Currently, since 1976, it has been maintained at the University of Białystok (formerly the University of Warsaw, Białystok Branch), Poland. From the very beginnings Trybulec postulated a language and a computer system for recording mathematical papers such that [16]:

– The papers could be stored in a computer and later, at least partially, translated into natural languages,
– The papers would be formal and concise,
– It would form a basis for construction of an automated information system for mathematics,

---

✉ Adam Grabowski
  adam@math.uwb.edu.pl; adam@mizar.org

  Artur Korniłowicz
  arturk@mizar.org

  Adam Naumowicz
  adamn@mizar.org

[1] Institute of Informatics, University of Białystok, Ciołkowskiego 1M, 15-245 Białystok, Poland

- It would facilitate detection of errors, verification of references, elimination of repeated theorems, etc.,
- It would open a way to machine assisted education of the art of proving theorems,
- It would enable automated generation of input into typesetting systems.

MIZAR[1] is the name of both a formal language in which the mathematics is written, and of an entire software system that checks the texts for logical correctness and manages the data base of MIZAR articles (formal texts written in the MIZAR language [30]). The logical basis of MIZAR is a system of natural deduction close to the "composite system of logic" developed by Stanisław Jaśkowski [12].

In the period of over 40 years, a number of different versions of the MIZAR system have been investigated. The history of the first 30 years of MIZAR development has been described in "Mizar: the first 30 years" by Roman Matuszewski and Piotr Rudnicki [16]. Various linguistic constructions have been introduced that made the MIZAR language a more accurate reflection of the natural language of mathematics. Among the most important versions of the system we can distinguish: MIZAR PC—processing propositional calculus (around 1975) [26]; MIZAR QC—processing quantifier calculus (1977) [27,28]; MIZAR FC—supporting functional notation (1979); MIZAR-MSE—multi-sorted with equality (1982); MIZAR-2,3,4— direct predecessors of the current MIZAR.

2 Current Mizar

By current MIZAR [6,18,22] we understand subsequent versions of the system available for personal computers that got popular in 1980s. MIZAR articles were initially self-contained—in the preamble (an environment part) authors had the possibility to formulate their own axioms, definitions and theorems (without proofs), which were used in the main part of the article. In 1987–1988 this led to the collection of about 20 files. Around that time, in 1989, collecting of MIZAR articles started, and MML—the MIZAR Mathematical Library—was established with the principal aim of systematic building a centralized reusable knowledge base reflecting standard foundations of mathematics—classical first order logic and set theory.

The initial axiomatic article in the MML was Trybulec's "Tarski-Grothendieck Set Theory" [29], where the following axioms were postulated:

- Extensionality axiom,
- Axiom of pair,
- Axiom of union,
- Axiom of regularity,
- Replacement axiom, and
- Tarski's axiom.

Hence the set theory of the MIZAR library is the Tarski-Grothendieck system, which is basically the Zermelo–Fraenkel set theory with the axiom of infinity replaced by Tarski's axiom of existence of arbitrarily large, strongly inaccessible cardinals [25] of the form

For every set N there exists a system M of sets which satisfies the following conditions:

(i) $N \in M$;
(ii) If $X \in M$ and $Y \subseteq X$, then $Y \in M$;

(iii) If $X \in M$ and Z is the system of all subsets of X, then $Z \in M$;

(iv) If $X \subseteq M$ and X and M do not have the same potency, then $X \in M$.

For years, the MIZAR Mathematical Library has been gradually developed by new submissions and the reorganization of existing MIZAR articles [9]. The MML continues to be one of the world's largest repositories of formalized and computer-checked mathematics. Over two hundred authors contributed to the MML, which at present includes over ten thousand definitions and over fifty thousand theorems with complete proofs in over one thousand and two hundred papers. 211 articles were written in first 2 years of the MML existence. Later, the growth stabilized at about 50 articles per year.

An interesting measure of the repository of MIZAR articles is the number of formalized items from "Top 100 mathematical theorems" list presented by Paul and Jack Abad and maintained by Freek Wiedijk.[2] MIZAR holds the second place with sixty four theorems completed so far. However, significant increase of the MIZAR repository was usually connected with big formalization projects, like the translation of "Compendium of Continuous Lattices" by Gierz et al. [2] or the formal proof of the Jordan Curve Theorem [13].

In the MML there are also systematically developed branches of mathematics [33], e.g., set theory [21], topology, functional analysis, abstract algebra [20], category theory, and lattice theory. Some of the results proven for lattices were obtained with the help of equational provers—like EQP/OTTER (now Prover9). The direction of the better integration of MIZAR with external systems significantly extends capabilities of the MIZAR checker. Among used systems we can point out such tools as SAT-solvers and systems computing Gröbner bases [17]. The cooperation with specialized tools helps both to obtain new results and to cross-validate already existing database. Efficient use of AI/ATP methods was possible via the translation from MIZAR logic into TPTP first order logic and dedicated XML format for information interchange. The need for such cooperation forced the developers of MIZAR to implement new intermediate dialects of the language carrying more information (and making it more explicit)—*weakly strict* and *more strict* MIZAR [3]. Also hyperlinked form of an article was reused [11,36].

Computerized proofs can be further improved in order to make their presentation more legible and accessible for human mathematicians [8]. Proof graphs measuring information flow available in the MML can be useful for that purpose as well [7,19]. Such enhancements are part of routine maintenance of MIZAR Mathematical Library that includes generalizing notions and theorems (and this new representation can help to obtain interesting mathematical results [5]), elimination of redundancy [9], preserving integrity [23], etc. These activities are performed by the Library Committee—an agency of the Association of Mizar Users[3] established to coordinate the overall maintenance of the centralized repository [1]. In 2002, first items in *Encyclopedia of Mathematics in* MIZAR started relatively revolutionary changes in the library, separating new files devoting to the same topic (now 14 monographic articles are available in this collection).

Even if the MIZAR language is relatively easily readable (and MIZAR mode for Emacs created by Josef Urban in 2000 is quite useful tool for tracking mathematical notions in the MML and supporting editorial work on articles), significant step towards better understanding by human mathematicians is the automatic translation of MIZAR articles into LaTeX format in the form of a journal. *Formalized Mathematics* has published 23 volumes up to now, and currently it is published both in the paper and electronic version. The representation of MIZAR

---

[2] http://www.cs.ru.nl/~freek/100/.

[3] http://mizar.org/sum/.

texts initialized by Trybulec and Rudnicki in *Collection of* T$_E$X *ed Mizar abstracts* [34] and reimplemented later by Grzegorz Bancerek (with Roman Matuszewski as Editor-in-Chief) contains e.g. partial translation of proofs of theorems. All submitted articles are subject to standard reviewing procedure.

After QED Workshop II in Warsaw, 1994, and significant support from the Office of Naval Research, USA, the paper version of *Formalized Mathematics* got new life—in years 1995–1997 *Journal of Formalized Mathematics* was published online as a hyperlinked version of *Formalized Mathematics* reflecting the current state of MML which evolved quite fast.

The verification system itself also evolves and the development concerns both the underlying language and the proof-checking software [32]. The evolution of the MIZAR language goes into the direction of improved expressiveness. Exemplary improvements include:

– The processing of attributes, which in the recent implementations can be expressed with their own visible arguments, e.g., n-dimensional, X-valued, etc.;
– Ellipsis, which introduces two logical connectives, flexary conjunction and flexary disjunction, which allow to write formulas more concisely [15];
– Synonyms, which allow to introduce different names for the same objects to imitate names used in traditional language of mathematics, e.g., elements of different structures can be named points in topological spaces and vectors in vector spaces;
– Antonyms to introduce antonymic names, e.g., odd integer numbers can be defined as non-even integers;
– Global choice construction with syntax `the Type`, which allows to introduce the unique constants for each well-defined type;
– Collective predicative formulas, which enable to write formulas consisting several (possibly different) predicate symbols, e.g., `{} c< A c= B`, with the obvious meaning that the empty set is strictly included in the set `A` and the set `A` is included in the set `B`;
– `Sethood` property, which is a way to say that all objects of a type for which the property is declared are elements of some set;
– Construction `the set of all`, which is an abbreviation for Fraenkel terms defining sets of terms, which terms do not have to satisfy any additional constraints, e.g. `the set of all n where n is natural number` defines the set of all natural numbers $\mathbb{N}$;
– Pragmas `::$EOF`, `::$P` and `::$V`, which help to control which parts of MIZAR texts should be verified during the process of writing articles, and pragma `::$N`, which serves to name important theorems included in MIZAR articles.

Although Andrzej Trybulec was the unquestionable leader of the project and the author of its most important ideas (and also the author of over 100 MIZAR articles), the main developer of the MIZAR software since 1976 and the person responsible for concrete implementations is Czesław Byliński. The changes in the language go hand in hand with developing more powerful automation techniques that make it possible to formalize many concepts in a more natural way. For example, the proof checker has recently been equipped with the following mechanisms:

– Reductions, which reduce terms to their proper sub-terms [14];
– Identifications, which identify equivalent notions from different theories;
– A built-in arithmetic of complex numbers.

The capabilities of the proof-checking software have also been strengthened by providing means for processing definitional expansions of formulas included in inferences being verified.

Nowadays there are a number of projects addressing problems related to computerized proof checking and theorem proving, and obviously MIZAR is listed among the basic ones [31]. The projects differ in their assumed theoretical basis (e.g., type theory or set theory, classical logic versus intuitionistic logic or higher order logic) and main goals, towards which the project is geared (formalization of mathematics, automated theorem proving, extracting programs from proofs, program verification, etc.).

There has been a significant interplay between MIZAR and other projects, especially ones developed in the tradition of early works of N.G. de Bruijn (the AUTOMATH system), L.A. Kaluzhnin (the idea of information language for mathematics which was discussed during Trybulec's visit at Institute of Scientific and Technical Information in Moscow in 1973), V.M. Glushkov (which gave rise to the system SAD), and R. Milner's LCF (a successor of which, HOL Light, is one of the most successful systems for formalizing mathematics). The contacts gain momentum with the discussions and research collaboration stemming from the QED initiative. To list the most important cases of MIZAR's influence on other systems we can mention the Declare system developed by Syme [24], the MIZAR mode for HOL by Harrison [10], the Isar language for Isabelle by Wenzel [37], MIZAR-light for HOL Light by Wiedijk [38], the declarative proof language (DPL) for Coq by Corbineau [4] and finally the Wiedijk's `miz3` proof interface for HOL Light [40] that combines both the procedural and declarative style of writing proofs. The MIZAR way of writing proofs was also the model for the notion of 'formal proof sketches' developed by Wiedijk [39].

From its beginnings, MIZAR was used as a tool for teaching mathematics [35]: initially, for teaching propositional logic, then, with the help of especially useful for educational purposes MIZAR-MSE, introduction to logic for secondary school students (this was also an example of a distant learning interactive course as it was published in Polish popular science monthly *Delta*). Then, in late 1980s other topics in mathematics education were covered—foundations of geometry, introduction to mathematics and lattice theory. After the MML was initialized, the stress was put on topology and then MIZAR was even an obligatory part of the curriculum (introduction to logic and set theory and formalization of mathematics) for all first-year students of computer science at University of Białystok. The MIZAR Mathematical Library is extensively tested by students—mainly as a knowledge base for mathematics. In the last 20 years numerous MIZAR articles were also written as a support for B.Sc., M.Sc., and Ph.D. theses (mainly by students in Poland, Canada, China, and Japan).

3 Contents of the Special Issue

The papers contained in this volume are closely related to areas important for the MIZAR system for 40 years—computer formalization of mathematics. The continuation of the research by V.M. Glushkov gave rise to the system described in the paper "Evidence Algorithm and Inference Search in First-Order Logics" by Alexander Lyaletski. HOL Light is one of the most successful systems for formalizing mathematics—an example formalization is presented in the paper "Formal Proofs of Hypergeometric Sums" by John Harrison.

An interesting issue of representations of formal approaches to ordered pairs and functions within ZF without the axiom of infinity and its formalization in the proof assistant Coq using only the simply typed features of Coq is presented by Chad E. Brown in his paper "Reconsidering Pairs and Functions as Sets" (a similar problem in MIZAR was addressed before in [23]).

One of the dynamically developed branches in the MML, lattice theory, was described in the paper "Mechanizing Complemented Lattices Within Mizar Type System" by Adam

Grabowski. Karol Pąk in his submission "Improving Legibility of Formal Proofs Based on the Close Reference Principle is NP-Hard" investigates proof graphs available in the MML measuring information flow.

The paper "Miz𝔸ℝ 40 for Mizar 40" by Cezary Kaliszyk and Josef Urban presents efficient use of AI/ATP methods which was possible via the translation from MIZAR logic into TPTP first order logic and dedicated XML format for information interchange.

Adam Naumowicz in "Automating Boolean Set Operations in Mizar Proof Checking with the Aid of an External SAT Solver" described the possibilities of the better integration of MIZAR with external systems as SAT-solvers. Another way for strengthening capabilities of the proof-checking software is providing means for processing definitional expansions of formulas included in inferences being verified is described in the paper "Definitional Expansions in Mizar" by Artur Korniłowicz.

This special issue was intended as a celebration of 40 years of MIZAR, and it is a natural continuation of [16] by Matuszewski and Rudnicki written a decade earlier. Andrzej Trybulec, who co-edited this volume together with Krystyna Trybulec Kuperberg, passed away on September 11, 2013.

References

1. Alama, J., Kohlhase, M., Naumowicz, A., Rudnicki, P., Urban, J.: Licensing the Mizar Mathematical Library. In: Davenport, J.H., Farmer, W.M., Urban, J., Rabe, F. (eds.) Proceedings of the 18th Calculemus and 10th International Conference on Intelligent Computer Mathematics, MKM'11, *Lecture Notes in Computer Science*, vol. 6824, pp. 149–163. Springer, Berlin (2011). doi:10.1007/978-3-642-22673-1_11

2. Bancerek, G., Rudnicki, P.: A Compendium of Continuous Lattices in Mizar: formalizing recent mathematics. J. Autom. Reason. **29**(3–4), 189–224 (2002)

3. Byliński, C., Alama, J.: New developments in parsing Mizar. In: Jeuring, J., Campbell, J.A., Carette, J., Dos Reis, G., Sojka, P., Wenzel, M., Sorge, V. (eds.) Intelligent Computer Mathematics 11th International Conference, AISC 2012, 19th Symposium, Calculemus 2012, 5th International Workshop, DML 2012, 11th International Conference, MKM 2012, Systems and Projects, *Lecture Notes in Artificial Intelligence*, vol. 7362, pp. 427–431. Springer, Berlin (2012). doi:10.1007/978-3-642-31374-5_30

4. Corbineau, P.: A declarative language for the Coq proof assistant. In: Miculan, M., Scagnetto, I., Honsell, F. (eds.) Types for Proofs and Programs, International Conference, TYPES 2007, Cividale del Friuli, Italy, May 2–5, 2007, Revised Selected Papers, *Lecture Notes in Computer Science*, vol. 4941, pp. 69–84. Springer, New York (2007). doi:10.1007/978-3-540-68103-8_5

5. Grabowski, A.: Automated discovery of properties of rough sets. Fundam. Inform. **128**(1–2), 65–79 (2013). doi:10.3233/FI-2013-933

6. Grabowski, A., Korniłowicz, A., Naumowicz, A.: Mizar in a nutshell. J. Formaliz. Reason. Spec. Issue User Tutor. I **3**(2), 153–245 (2010)

7. Grabowski, A., Schwarzweller, C.: Towards automatically categorizing mathematical knowledge. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) Proceedings of Federated Conference on Computer Science and Information Systems, FedCSIS 2012, pp. 63–68. Wrocław, Poland (2012)

8. Grabowski, A., Schwarzweller, C.: Translating mathematical vernacular into knowledge repositories. In: Kohlhase, M. (ed.) Proceedings of the 4th International Conference on Mathematical Knowledge Management, MKM'05, *Lecture Notes in Computer Science*, vol. 3863, pp. 49–64. Springer, Berlin (2006). doi:10.1007/11618027_4

9. Grabowski, A., Schwarzweller, C.: Revisions as an essential tool to maintain mathematical repositories. In: Kauers, M., Kerber, M., Miner, R., Windsteiger, W. (eds.) Proceedings of the 14th Symposium on Towards Mechanized Mathematical Assistants: 6th International Conference, Calculemus '07 / MKM

'07, *Lecture Notes in Computer Science*, vol. 4573, pp. 235–249. Springer, Berlin (2007). doi:10.1007/978-3-540-73086-6_20

10. Harrison, J.: A Mizar mode for HOL. Proceedings of the 9th International Conference on Theorem Proving in Higher Order Logics. TPHOLs'96, Lecture Notes in Computer Science, vol. 1125, pp. 203–220. Springer, London, UK (1996)

11. Iancu, M., Kohlhase, M., Rabe, F., Urban, J.: The Mizar Mathematical Library in OMDoc: translation and applications. J. Autom. Reason. **50**(2), 191–202 (2013). doi:10.1007/s10817-012-9271-4

12. Jaśkowski, S.: On the Rules of Suppositions in Formal Logic. Studia Logica. Nakładem Seminarjum Filozoficznego Wydziału Matematyczno-Przyrodniczego Uniwersytetu Warszawskiego (1934)

13. Korniłowicz, A.: Jordan Curve Theorem. Formaliz. Math. **13**(4), 481–491 (2005). http://fm.mizar.org/2005-13/pdf13-4/jordan.pdf

14. Korniłowicz, A.: On rewriting rules in Mizar. J. Autom. Reason. **50**(2), 203–210 (2013). doi:10.1007/s10817-012-9261-6

15. Korniłowicz, A.: Flexary connectives in Mizar. Computer Languages, Systems & Structures (2015). doi:10.1016/j.cl.2015.07.002

16. Matuszewski, R., Rudnicki, P.: Mizar: The first 30 years. Mech. Math. Appl. Special Issue on 30 Years of Mizar **4**(1), 3–24 (2005)

17. Naumowicz, A.: Interfacing external CA systems for Gröbner bases computation in Mizar proof checking. Int. J. Comput. Math. **87**(1), 1–11 (2010). doi:10.1080/00207160701864459

18. Naumowicz, A., Korniłowicz, A.: A brief overview of Mizar. In: Berghofer, S., Nipkow, T., Urban, C., Wenzel, M. (eds.) Proceedings of the 22nd International Conference on Theorem Proving in Higher Order Logics, *TPHOLs'09, Lecture Notes in Computer Science*, vol. 5674, pp. 67–72. Springer, Berlin (2009). doi:10.1007/978-3-642-03359-9_5

19. Pąk, K.: Methods of lemma extraction in natural deduction proofs. J. Autom. Reason. **50**(2), 217–228 (2013). doi:10.1007/s10817-012-9267-0

20. Rudnicki, P., Schwarzweller, C., Trybulec, A.: Commutative algebra in the Mizar system. J. Symb. Comput. **32**, 143–169 (2001)

21. Rudnicki, P., Trybulec, A.: On equivalents of well-foundedness. An experiment in Mizar. J. Autom. Reason. **23**, 197–234 (1999)

22. Rudnicki, P., Trybulec, A.: Mathematical knowledge management in Mizar. In: Buchberger, B. Caprotti, O. (eds.) Procedings of the First International Workshop on Mathematical Knowledge Management, Linz, Austria (2001)

23. Rudnicki, P., Trybulec, A.: On the integrity of a repository of formal mathematics. In: Asperti, A., Buchberger, B., Davenport, J.H. (eds.) Proceedings of MKM 2003, *Lecture Notes in Computer Science*, vol. 2594, pp. 162–174. Springer, Berlin (2003). doi:10.1007/3-540-36469-2_13

24. Syme, D.: DECLARE: A prototype declarative proof system for higher order logic. Technicl Report. University of Cambridge, Cambridge (1997)

25. Tarski, A.: On well-ordered subsets of any set. Fundam. Math. **32**, 176–183 (1939)

26. Trybulec, A.: Informationslogische sprache Mizar, Dokumentation–Information, Heft 33, pp. 46–53. Ilmenau (1977)

27. Trybulec, A.: The Mizar-QC/6000 logic information language. ALLC Bull. **6**(2), 136–140 (1978)

28. Trybulec, A.: The Mizar logic information language. Stud. Log. Gramm. Rhetor. **1**, 127–136 (1980)

29. Trybulec, A.: Tarski Grothendieck set theory. Form. Math. **1**(1), 9–11 (1990). http://fm.mizar.org/1990-1/pdf1-1/tarski.pdf

30. Trybulec, A.: Some features of the Mizar language. In: Proceedings of ESPRIT Workshop. Torino (1993)

31. Trybulec, A.: Mizar. In: Wiedijk, F. (ed.) The Seventeen Provers of the World, *Lecture Notes in Computer Science*, vol. 3600, pp. 20–23. Springer, Berlin (2006). doi:10.1007/11542384_4

32. Trybulec, A., Blair, H.A.: Computer aided reasoning. In: Proceedings of the Conference Logic of Programs, *Lecture Notes in Computer Science,* vol. 193, pp. 406–412. Springer, New York (1985)

33. Trybulec, A., Korniłowicz, A., Naumowicz, A., Kuperberg, K.: Formal mathematics for mathematicians. J. Autom. Reason. **50**(2), 119–121 (2013). doi:10.1007/s10817-012-9268-z

34. Trybulec, A., Rudnicki, P.: A Collection of TEXed Mizar Abstracts. University of Alberta, Canada (1989)

35. Trybulec, A., Rudnicki, P.: Using Mizar in computer aided instruction of mathematics. In: Norvegian-French Conference of CAI in Mathematics. Oslo (1993)

36. Urban, J., Rudnicki, P., Sutcliffe, G.: ATP and presentation service for Mizar formalizations. J. Autom. Reason. **50**(2), 229–241 (2013). doi:10.1007/s10817-012-9269-y

37. Wenzel, M., Wiedijk, F.: A comparison of Mizar and Isar. J. Autom. Reason. **29**(3–4), 389–411 (2003). doi:10.1023/A:1021935419355

38. Wiedijk, F.: Mizar Light for HOL Light. In: Boulton, R.J., Jackson, P.B. (eds.) Proceedings of 14th International Conference TPHOLs, *Lecture Notes in Computer Science*, vol. 2152, pp. 378–393. Springer, New York (2001)
39. Wiedijk, F.: Formal proof sketches. In: Berardi, S., Coppo, M., Damiani, F. (eds.) TYPES, *Lecture Notes in Computer Science*, vol. 3085, pp. 378–393. Springer, New York (2004). doi:10.1007/978-3-540-24849-1_24
40. Wiedijk, F.: A synthesis of the procedural and declarative styles of interactive theorem proving. Log. Methods Comput. Sci. **8**(1:30), 1–26 (2012)