ZESZYTY NAUKOWE POLITECHNIKI BIAŁOSTOCKIEJ 2007
Informatyka – Zeszyt ?

Grzegorz Bancerek[1]

Using evolved MML

Summary

The MIZAR language is a language used for such a formalization of mathematics that is close to the vernacular used in mathematical publications. An implemented MIZAR verifier is available for checking correctness of MIZAR texts according to Jaśkowski's natural deduction. The perpetual development of the MIZAR system (see [5]) has resulted in the MIZAR Mathematical Library (MML)—a centrally maintained library of formalized mathematics based on Tarski–Grothendieck set theory. MML is commonly considered the biggest library of computer proof-checked mathematics.

MML is continuously developed by MIZAR users and this development is managed by the Library Committee and the Development Committee of the Association of Mizar Users[2]. The tasks of the Library Committee concern the acceptance of new articles into MML and the distribution of MML. The policy of the Committee has been very liberal for a long time. It started from acceptance of correct MIZAR articles which satisfied the *non-triviality* criterion. The criterion was successively completed by software for recognition of unused elements, inaccessible fragments, irrelevant premises, etc. Recently, the policy is much stronger. A new MIZAR article must pass successfully the so-called revision software tests and additionally is must get 3 positive human reviews. The process of submission in some cases may take some time but in the result new articles are free from formalization mistakes which can be recognized by reviewers.

The Development Committee was created as a part of the Library Committee and now is a separate agenda (but the intersection is still not empty). The main aim of the Development Committee is to keep the MML in the state allowing for development. The organization of the material stored in MML is not fixed and the Committee is reorganizing it from time to time. Such reorganizations are called *revisions*. The essential goals of revisions are the quality and integrity of MML, [7] and [4].

[1] Faculty of Computer Science, Białystok Technical University, Białystok
[2] The Association of Mizar Users is the owner of MML

Revisions may change MML in a substantial or unsubstantial way. Substantial changes of the MML mean that some not yet submitted articles correct with regard to a previous version may now become incorrect. Substantial revisions change the database files including exportable elements. The exportable elements are presented in abstract files distributed with the system. Unsubstantial revisions do not change the content of abstract files. They concern editorial changes like rearrangement of labels, reformatting of formulae, or removing inaccessible fragments (which do not appear in abstracts). They concern also ameliorating changes done with revision software consisting of the removal of

- irrelevant references (premises) from inference steps,
- irrelevant inferences (irrelevant proof steps),
- irrelevant steps in iterative equality,
- trivial proofs, and
- irrelevant environment directives.

These ameliorations affect proofs (invisible in abstracts) but sometimes recognize unnecessary assumptions in theorems and strengthen them.

The aim of many revisions is to clean up the MML and change it towards a normalized database. The revisions are at least *conservative*, i.e., they do not reduce the content of MML and they may cause that some formalizations and reasoning could be done shorter and/or easier. It is obtained by the elimination of redundant definitions, correction and reformulation of definientia, generalization of arguments, etc. But, on the other hand, the elimination of obvious consequences which is mostly done automatically removes everything even if the obviousness is not so obvious to humans. It causes that the information retrieval in MML becomes harder and the fluent use of MML requires its better knowledge (which is not well-documented).

The obviousness of canceled theorems may be quite complicated as it involves the automation available in MIZAR. The automation depends on the environment of canceled theorem and it concerns

- special treatment of some constructors according to possessed properties [6],
- the addition of adjectives according to registrations,
- the unfolding of terms according to implicit functor definitions,
- the identification of terms and atomic formulae according to registrations.

For example, the predicate <= possesses the property `connectedness` what was the reason that the theorem

```
x < y & y < z implies x < z
```

has been canceled as obvious consequence[3] of

```
theorem  :: XXREAL_0:2
  x <= y & y <= x implies x <= z
```

It seems that reconstruction what should be used instead of a canceled theorem may be difficult and MIZAR users are left without documentation. However, the MML Query allow for solving some of those problems successfully. In the presentation we discuss queries aimed at *reconstruction of a canceled theorems* and give new features of MML Query prepared for those purposes.

References

[1] Andrea Asperti, Grzegorz Bancerek, and Andrzej Trybulec, editors. *Mathematical Knowledge Management, MKM'04*, volume 3119 of *LNCS*, Białowieża, 2004. Springer.

[2] Andrea Asperti, Bruno Buchberger, and James H. Davenport, editors. *Mathematical Knowledge Management, MKM'03*, volume 2594 of *LNCS*, Bertinoro, 2003. Springer.

[3] Grzegorz Bancerek. Information retrieval and rendering with MML Query. In Jonathan M. Borwein and William M. Farmer, editors, *Mathematical Knowledge Management, MKM'06*, volume 4108 of *LNAI*, pages 65–80, Wokingham, 2006. Springer.

[4] Adam Grabowski and Christoph Schwarzweller. Rough concept analysis – theory development in the Mizar system. In Asperti et al. [1], pages 130–144.

[5] Roman Matuszewski and Piotr Rudnicki. MIZAR: the first 30 years. *Mechanized Mathematics and Its Applications*, 4(1):3–24, 2005.

[6] Adam Numowicz and Czesław Byliński. Improving MIZAR texts with *properties* and *requirements*. In Asperti et al. [1], pages 290–301.

[7] Piotr Rudnicki and Andrzej Trybulec. On the integrity of a repository of formalized mathematics. In Asperti et al. [2], pages 162–174.

[3] It is identical modulo antonyms to the formula `z <= x & not y <= x implies x <= y` which must be a consequence of `XXREAL_0:2` because we know from `connectedness` that `x <= y or y <= x`