COMPUTER ASSISTED REASONING WITH MIZAR

Andrzej Trybulec & Howard Blair

EECS U-157, University of Connecticut
Storrs CT, 06268, USA

ABSTRACT

We describe the MIZAR Computer Assisted Reasoning system. Several versions of MIZAR are currently implemented with varying powers of expressiveness and proof analysis. The main use of MIZAR has thus far been in mathematics education. Evolving versions of MIZAR have been under development for the past decade; consequently the system has been subjected to extensive testing. Several experiments connected with the preparation of mathematics research papers have also been carried out. The naturalness and readability of MIZAR texts is demonstrated by example. A proof scanning facility that modularly splits into a proof structure analyzer and inference checker is described. We discuss our use of the facility in teaching undergraduate discrete mathematics for computer science.

1.Introduction

A computer assisted reasoning (CAR) system is nontrivial if the level of detail with which the user must be concerned corresponds well with the level of detail required for clear, realistic mathematical presentation. Mathematical vernacular is characterized in part by an open ended system of standardized notation. A writer of mathematics is not free to fill his text with an undisciplined proliferation of freely invented notation. If a standard notation is adequate for the purpose, the writer is well-advised to use it. Only the rarest circumstances permit a relaxation of this practice. Nevertheless, excessive formalism should be avoided since it invites a level of detail simply too distracting, indeed boring, from the main themes of the argument.

The standardization of notation permits the possibility of a formal reconstruction of mathematical vernacular. The requirement that the mathematical presentation be not too formally detailed, but nevertheless clear in a step-wise fashion, permits the use of some aspects of automated reasoning in a reasoning assistance system. A single human oriented step in a mathematical argument is viewed as a small, quickly solvable, automated reasoning task. The ultimate success and value of such a system is determined by how useful a tool it proves to be in practice and not by how well it is alleged to embody a particular pedagogical ideology. A number of attempts to carry out such a development have been tried: in particular, the earlier system, AUTOMATH, of N.G.de Bruijn [dB70], FEA of S. Postma [Po78], QUIP of R.L.Smith [Sm75], EXCHECK of L.Blaine [B181], PL/CV2 of R.L.Constable [Co82].

Basically, most of the systems mentioned are based on classical logic. An Important exception is AUTOMATH. This was an attempt to construct a system more fundamental than logic, but classical logic can be embedded in it. The principal concern here is to describe and report on the achievements up to the present of one such line of development, namely the MIZAR family of CAR systems that have been under development for the past decade. MIZAR has enjoyed the opportunity to be extensively tested in a variety of mathematics educational settings as well as, in one case, intensive referee-like circumstances in mathematics research (Homotopy theory). The concept of proof is stressed in MIZAR. Some of the other systems we mentioned view the role of proof differently.

The MIZAR system takes a traditional approach. A proof is a documentation of the validity of a theorem and its explanatory function is secondary. We believe we are near to AUTOMATH, where proof is the basic concept, rather than theorems.

We argue here that MIZAR captures the balance between formalism and human-oriented mathematical expression. In addition, since the experience of preparing a MIZAR text is similar to composing a structured program by step-wise refinement, MIZAR facilitates the ability to produce a proof with its main ideas clearly discernible. A well organized proof is often enough a mathematical explanation.

There are several versions of MIZAR with varying levels of logical richness and inference checking power. Two interesting experiments are worth noting. In a version of MIZAR oalled MIZAR-HPF, fundamental axioms for a fragment of category theory were prepared in an environment (see the "environ" segment in the example given below) and a series of exercises were soanned. The axiomatization strategy was carried through, in effect, by employing sorts for higher order objects. In MIZAR-2, a paper of Karol Borsuk [Bo70], in Homotopy Theory, was thoroughly transcribed. This resulted in a text approximately twice the size, in lines/words, as the original text. The task of transcription for suoh a paper is rather similar to writing a detailed exegesis.

2. An Example.

We present in this section as an example. The example is interesting for several reasons. First, and most importantly, it is representative of MIZAR texts, although quite small.

The example conforms to the restrictions of MIZAR-MSE ("Many-Sorted first-order logic with Equality", but without function symbols). This is the simplest MIZAR and yet as we will discuss in the following sections has a degree of power making it useful in a wide range of topics in discrete mathematics. It will be seen by inspection that the text we present is as readable as any mathematical text. The experience of preparing a MIZAR text is analogous to composing a structured program, and this is often visible in the result.

MIZAR reads the text checking for mistakes. Apart from syntactic errors the mistakes the currently implemented version of MIZAR detects are of two kinds: inferences that are invalid or too intricate, and inappropriate assumptions or non-concluded lines of reasoning. Most of the time what is too intricate for the MSE inference checker is indeed too intricate for a human as well to see immediately. Students, in particular, when they attempt to exceed the power of the checker on a single step often do not really explicitly see their proposed reasoning step. This is confirmed by the difficulty they frequently have in providing the intermediate steps. We claim that the power of the MSE inference checker matches well the power of an intelligent student to explicitly and effectively see his way through a chain of deductions. That is to say, the attention to detail is about at the right level. We do not claim that this level of detail and formality in MSE is appropriate for the pursuit of real mathematical discovery; we do claim, however, that the power afforded by the MSE checker is appropriate for developing the skills required for composing mathematical demonstrations.

What follows is an example of a text that can be submitted to MIZAR for checking.

```
environ let x denote human;
  Ax:(for x being human holds works[x])
     implies WFS

begin
== Comments are set off by '=s' at the beginnings
=s of lines. The environment presents an axiom
== characterizing a WelFare State. We prove the
■» following correot consequence.  It is
== essentially the
== result of a nonintuitionistic prenex
== operation.
  ex x st works[x] implies WFS
proof
1: now assume  for x holds works[x]; then WFS
    by Ax; henoe thesis
   end;
   now given x such that 3: not works[x];
   thus thesis by 3 end;
   henoe thesis by 1
end
```

This text may be input to MIZAR for checking and MIZAR will respond (properly) that the argument is correct.

The following example is that of a text output by MIZAR. This output resulted from submitting an input text of the preceding proof at an intermediate stage of development.

```
     environ let x denote human;
       (for x being human holds works[x])

       implies WFS
     begin
       ex x st works[x] implies WFS
     proof
     1:now assume  for x holds works[x];
       then WFS by Ax; hence thesis end;
••**                 *73
       now given x such that 3: not works[x];
       thus thesis end;
**** *103
     end
••••   *81

 sorry

errors explanation

 53  wrong beginning of the item in environ part,
     only axioms, i.e. labeled sentences,
     predeclarations (starting with "let") and
     oonstant declarations (starting with "given")
     are allowed. (*1)

 73  no sentence is designated by this label, the
     label was not used to label any of the
     previous sentences.

 81  reasoning or proof is not concluded.

103  your inference is not accepted by the cheoker.

 remarks

 (*1) due to this error a portion of the text
      usually until "begin", "end", "environ",
      "now","proof" or semicolon has been skipped
      in the analysis.
```

Both authors prepare MIZAR texts in an iterative fashion by first preparing proof skeletons. The practice is derived naturally enough from structured programming. Whatever its merits or demerits in prinoiple, the result in fact is nearly always a more rapid, more elegantly structured proof than we obtain when we yield to the temptation of discarding this discipline in favor of a 'linearly' composed proof.

3. Proof Scanning

i. Preprocessing

Sentences that occur in the text are preprocessed by the MIZAR system to forms contained in the fragment of logic generated from atomic sentences by negation, conjunction, and universal quantification. Double negation and associativity of conjunction are used to obtain simplified forms.

Ii. Justifications.

There are two different possibilities to justify sentences in MIZAR-MSE that enable two opposite tasks. The first is straightforward inference checking by referring to earlier sentences (i.e. axioms, Statements or assumptions) that constitute sufficient conditions for the sentence to be justified.

The second task concerns the proof structure. The user's attention to structuring the proof enables him to divide his main goal (justifying the sentence) into subgoals. In MIZAR this is done explicitly by the user as it would naturally be done in writing a mathematical text. The importance of such structuring was recognized at the very beginning of developing MIZAR [Tr78]. M. Davis in [Da8l] proposes a formal characterization of obviousness, regarding inference checking. We are experimenting with his particular characterization in a version of MIZAR-MSE with an altered inference checker, [cf. TrB85]. The point here is that since the inference checker is a separate module, the exploration of the appropriate inference power of a checking facility is easily implemented without altering the entire system, which would have the effect of leading a user to reorganize his proofs if he were to redo them. This has particularly aided the development of MIZAR while simultaneously encouraging users to set down their MIZAR texts with a carefully, and naturally, organized structure.

4. MIZAR In Use.

MIZAR-MSE (on Vax/780 under Unix) was used at the University of Connecticut in the undergraduate Introduction to Discrete Systems course. Students could elect to participate in a unit on MIZAR in lieu of another unit. Because results are satisfactory we have decided to introduce MIZAR as a required component of the course. Our subjective impression is that students that participated in the MIZAR unit did learn how to develop proofs. MIZAR is used at Warsaw University, Poland, at University of Alberta, Canada, and at University of Louvain, Belgium. It was recently introduced also at Washington State University at Pullman. A version of MIZAR-MSE has also been implemented for Apple-II.

REFERENCES

[B181] Blaine, L., in Patrick Suppes, University level computer assisted instruction at Stanford 1968-1980, Institute for Math. Studies in Social Science, Stanford, California, 1981.

[Bo70] Borsuk, Karol, On the homotopy types of some decomposition spaces, Bull.Acad.Pol.Sci., ser.math, astr. et phys. no.5, 18(1970)

[dB70] de Bruijn, N.G., The mathematical language Automath, its usage and extensions, Lecture Notes in Mathematics, Springer Verlag, 125(1970)

[Co82] Constable, R.L., An introduction to the PL/CV2 programming logic, Lecture Notes in Computer Science, Springer Verlag, 135(1982)

[Da8l] Davis, Martin, Obvious logical inferences, Proc. of IJCAI-81 Vancouver,BC, August 1981,530-531

[GMW79] Gordon, M., Milner, R., Wadsworth, C., Edinburgh LCF, Lecture Notes in Computer Science, Springer Verlag, 78(1980)

[MoTr85] Mostowski, Marcin; Trybulec, Zinaida, A certain experimental computer aided course of logic in Poland, Proc. of World Conference on Computers in Education, Norfolk, VA, July-August 1985, North Holland, to appear

[Po78] Postma, Stef W., FEA - a formal English subset for algebra/assertions, ACM Sigplan Notices, 13(1978), no.7

[PrRu83] Prazmowski, Krzysztof; Rudnicki, Piotr, Mizar MSE primer, ICS PAS Reports, no.529, Warsaw, 1983

[RuDr85] Rudnicki, Piotr; Drabent, Wlodzimierz, Proving properties of Pascal programs in Mizar-2, Acta Informatica, Springer Verlag, to appear

[Sm75] Smith, R.L., et al. Computer assisted axiomatic mathematics: informal rigor, Computer in Education, IFIP, North Holland 1975

[Tr78] Trybulec, Andrzej, The MIZAR-QC/6000 logic information language, Association for Literary and Linguistic Computing, Bulletin, 6(1978),136-140

[TrB85] Trybulec, A. & Blair, H. Computer Assisted Reasoning. Logics of Programs 1985. Springer-Verlag. (to appear)