

Challenge to the proving of theorems of Euclid's elements of as ATP

Fumiya Iwama*¹, Tadashi Takahashi²

¹ Graduate School of Natural Science, Konan University, Japan.

² Department of Intelligence and Informatics, Konan University, Japan.

Abstract: Background: Automated theorem proving (ATP) is a subfield of automated reasoning and mathematical logic dealing with proving mathematical theorems by computer programs. An Isabelle/HOL is a generic proof assistant.

Objective: We perform the challenge for the proving theorems of Euclid's elements of geometry.

Results: We could prove some theorems of Euclid's elements of geometry.

Conclusion: Technique of programming and mental conception interact. The mathematics education which prove theorems of the Euclidean geometry by using the Isabelle/HOL can correct the present weak point

Key words: *ATP, Mathematics education, Isabelle/HOL*

INTRODUCTION

Automated theorem proving (ATP) is a subfield of automated reasoning and mathematical logic dealing with proving mathematical theorems by computer programs [1]. By using theorem prover, we can reduce the proofs of complicated theorems to simple steps. The development of theorem prover is an exciting study of mathematics and computer science. Creating the mathematical technology (creating the algorithm) proceed by thinking and applying the results. There are many software for proving mathematical theorems in automatic mode. We will show the approach using the Isabelle/HOL.

Isabelle/HOL is a generic proof assistant [2]. Using an Isabelle/HOL requires insight into procedures as well as into the concepts involved. In addition, the way computer manages the procedures can affect mathematical concepts. In cognitive science, use of the Isabelle/HOL with a theorem prover can correct the weakness in mathematical studies. We can clearly understand mathematical concepts and can minimize the burden of operation opportunities. The use of Isabelle/HOL bring about serious changes for the proving theorems of the Euclidean geometry. Research on formalizing abstract algebra in Isabelle/HOL is based

on work by Hidetsune Kobayashi [3,4]. This study focuses on researching mathematics, and in particular,

on training researchers in the technics of proving. In the area of mechanical theorem proving, Kobayashi gave a decision procedure for what he called abstract algebra, based on algebraic method. It is really surprise to prove many abstract algebra theorems whose traditional proofs need enormous amounts of human intelligence. One of the key observations of Kobayashi is that theorems in abstract algebra can be relatively easily dealt with by a lot of lemmata, completely from former methods. The power of the method can be shown by experiments on computers in which many abstract algebra theorems were proved. The success of Kobayashi's method stimulated researchers to apply the connection of lemmata images. This research on formalizing abstract algebra in Isabelle/HOL is being conducted in order to develop a computer algebra system that supports mathematical study focused on "abstract algebra".

The system combines methods of automated theorem proving and also integrates programming in a natural way. It is interesting to note that theorems have been verified by this method. In a limited sense, this "theorem prover" is capable of "reasoning" about algebraic conjectures, an area often considered to be solely the domain of human intelligence. This research aims at

Corresponding Author: Fumiya Iwama. Konan University, Graduate School of Natural Science. Kobe. Japan, 8-9-1 Okamoto, Higashinada, Kobe Hyogo 658-8501, JAPAN, +81-78-435-2533, e-mail: d1623001@s.konan-u.ac.jp

extending current computer systems using facilities for supporting mathematical proving. The system consists of a general higher-order predicate logic prover and a collection of special provers. The individual provers imitate the proof style of human mathematicians and produce human-readable proofs in natural language presented in nested cells. The long-term goal of this research is to produce a complete system, which supports mathematicians.

IMPLEMENTATION

First, declare the necessary types.

```
datatype point = "char"
      datatype segment = Se "point" "point"
      datatype seg_list = Sel "segment" "segment"
      datatype circle = Ci "point" "point"
```

They are type declarations of point, line segment, circle, line segment + line segment, respectively. Next, declare that these are data types with properties. The locale function “dist” for implementing the properties of line segments is shown below.

```
locale dist = fixes ldist :: "segment => segment => bool"
(infixl "[@]" 50)
  and llist :: "seg_list => seg_list => bool" (infixl
"[@@]" 50)
  and ldist_list :: "segment => seg_list => bool" (infixl
"[@-@]" 50)
      assumes dist_ref [simp,intro] : "s1 [@] s1"
      and dist_rev1 : "[s1 [@] s2] ==> s2 [@] s1"
      and dist_trans : "[s1 [@] s2; s2 [@] s3] ==> s1 [@]
s3"
      and dist_rev2 : "[ (Se x1 y1) [@] (Se x2 y2) ] ==>
(Se y1 x1) [@] (Se y2 x2)"
      and dist_rev3 : "[ (Se x1 y1) [@] (Se x2 y2) ] ==>
(Se x1 y1) [@] (Se y2 x2)"
      and dist_list : "[s1 [@] s2] ==> (Sel s1 b s3) [@@]
(Sel s2 b s3)"
      and list_ref [simp,intro] : "s1 [@@] s1"
      and list_trans : "[s1 [@@] s2; s2 [@@] s3] ==>
s1 [@@] s3"
      and list_dist1 : "( (Sel s1 s3) [@@] (Sel s2 s3) ) <-->
s1 [@] s2"
      and list_dist2 : "[ (Sel s1 s3) [@@] (Sel s2 s4); s3 [@]
s4 ] ==> s1 [@] s2"
      and list_dist3 : "[ (Sel s1 s3) [@@] (Sel s2 s4); s1 [@]
s2 ] ==> s3 [@] s4"
      and list_rev1 : "[s1 [@@] s2] ==> s2 [@@] s1"
```

```
and list_rev2 : "[ (Sel s1 s3) [@@] (Sel s2 s4) ] ==>
(Sel s3 s1) [@@] (Sel s4 s2)"
and list_rev3 : "[ (Sel s1 s3) [@@] (Sel s2 s4) ] ==>
(Sel s1 s3) [@@] (Sel s4 s2)"
and dist_list_dist : "[s1 [@-@] s1; s2 [@-@] s2; s1
[@@] s2] ==> s1 [@] s2"
and dist_list_list : "[s1 [@-@] s1; s2 [@-@] s2; s1
[@@] s2] ==> s1 [@@] s2"
```

What is declared here is the relationship that arises between the two types under the condition that "given two line segments are equal", "given two sets of line segments + line segments are equal" and "One given line segment and one set of line segment + line segment are equal". The relations declared here (each “assume” statements) can be used for proof of lemma and theorem specifying the function “dist” (and locale function including it), respectively. To give a brief description of each \$assume\$ statements, it is as follows.

1. $AB = AB$
2. $AB = CD \rightarrow CD = AB$
3. $AB = BC \ \& \ BC = CD \rightarrow AB = CD$
4. $AB = CD \rightarrow BA = DC$
5. $AB = CD \rightarrow AB = DC$
6. $AB = CD \rightarrow AB+EF = CD+EF$
7. $AB+CD = AB+CD$
8. $AB+CD = EF+GH \ \& \ EF+GH = IJ+KL \rightarrow AB+CD = IJ+KL$
9. $AB+EF = CD+EF \leftrightarrow AB = CD$
10. $AB+EF = CD+GH \ \& \ EF = GH \rightarrow AB = CD$
11. $AB+EF = CD+GH \ \& \ AB = CD \rightarrow EF = GH$
12. $AB+CD = EF+GH \rightarrow EF+GH = AB+CD$
13. $AB+CD = EF+GH \rightarrow CD+AB = GH+EF$
14. $AB+CD = EF+GH \rightarrow AB+CD = GH+EF$
15. $AD = AB+CD \ \& \ EH = EF+GH \ \& \ AB+CD = EF+GH \rightarrow AD = EH$
16. $AD = AB+CD \ \& \ EH = EF+GH \ \& \ AB = EH \rightarrow AB+CD = EF+GH$

Here are some examples of simple proof using "dist".

Ex1 : If $AB = AC$, prove that $AB = BC$, $BC = AC$.

```
lemma (in dist) ex1:
  fixes AB BC AC :: segment
  assumes "AB [@] BC" "BC [@] AC"
  shows "AB [@] AC"
  using assms
  apply (blast intro:dist_trans)
```

done

The meaning of each “assume” sentence is as follows.

Ex2: Prove that $AB = BA$.

```
lemma (in dist) ex2:
  fixes AB BA :: segment
  assumes "AB = Se A B" "BA = Se B A"
  shows "AB [@] BA"
  using assms
  apply (simp add:dist_rev2 dist_rev3)
  done
```

1. When P is a point on the circumference of a circle with a center A and a radius AB, $AB = AP$.
2. When P and Q are the points on the circumference of the circle with the center A and the radius AB respectively, $PA = QA$.
3. When P is a point on the circumference of a circle with a center A and a radius AB and a point on the circumference of a circle with a center B and a radius BA, $PA = PB$.

Ex3 : If $BC = BD$, prove that $AB + BC = AB + BD$.

```
lemma (in dist) ex3:
  fixes A B C D :: point
  and AB BC BD :: segment
  and AB_BC AB_BD :: seg_list
  assumes "AB = Se A B" "BC = Se B C" "BD = Se B D"
  "AB_BC = Sel BC AB" "AB_BD = Sel BD AB"
  "BC [@] BD"
  shows "AB_BC [@@] AB_BD"
  using assms
  apply (blast intro:dist_list list_rev2)
  done
```

PROVING

We prove the following problem on Isabelle. These are the proof questions stated in [5].

Prove1 : Make an equilateral triangle on a given line segment.

The proof process shown in [5] becomes as follows.

1. Draws a circle whose radius is AB with each point of the given line segment AB as the center.
2. Let Γ be the intersection (upper side of the line) of the two circles, connect each of the points A and B with the line segments, and make the line segments ΓA and ΓB .
3. By definition, $\Gamma A = AB$ & $\Gamma B = AB$, so $\Delta \Gamma AB$ is an equilateral triangle.

Next we implement the properties of the circle. As shown at the beginning, the circle type variable (type of circle) is defined by two points. For example, "CircAB = Ci A B" means a circle whose radius is the length of the segment AB with respect to the point A.

The following locale function sets the relationship between that point and the circle (two points defining it) when "the given point is on the circumference of the given circle".

```
locale circledf = dist +
  fixes lcircle :: "point => circle =>bool" (infixl "[on]" 50)
  assumes circle1 : "[| p [on] (Ci c r) |] ==> (Se c r) [ @ ] (Se c p)"
  and circle2 : "[| p1 [on] (Ci c r); p2 [on] (Ci c r) |] ==> (Se p1 c) [ @ ] (Se p2 c)"
  and circle3 : "[| p [on] (Ci c r); p [on] (Ci r c) |] ==> (Se p c) [ @ ] (Se p r)"
```

At the beginning of the function, we specified to add the previously declared locale function "dist" so that we can use functions declared with "dist" ([@], etc.).

Implementation procedure on Isabelle is as follows.

- 1*. Prepare points A, B, Γ , line segment AB, ΓA , ΓB , circle AB, BA.
- 2*. Let Γ be a point on the circumference of each of the circles AB and BA.
- 3*. Prove that line segments AB, ΓA , ΓB are equal.

The full text of proof by Isabelle is shown below.

```
theorem (in circledf) Prove1:
  fixes A B  $\Gamma$  :: point
  and AB  $\Gamma A$   $\Gamma B$  :: segment
  and CircAB CircBA :: circle
  assumes "AB = Se A B" "TA = Se  $\Gamma$  A" "TB = Se  $\Gamma$  B"
  "CircAB = Ci A B" "CircBA = Ci B A" "T [on] CircAB" "T [on] CircBA"
  shows "AB [ @ ]  $\Gamma A$   $\wedge$  AB [ @ ]  $\Gamma B$   $\wedge$   $\Gamma B$  [ @ ]  $\Gamma A$ "
  using assms
```

apply (simp add:dist_rev1 dist_rev3 circle_dist1
circle_dist3)

done

Prove2 : Create a line segment starting from a given point and equal to the given line segment.

The proof process shown in [5] becomes as follows.

1. Given a given point as A, a given line segment as BΓ, connect line segment AB.
2. Create an equilateral triangle ΔAB on line AB (Prove 1).
3. Let line segments AE and BZ be line segments extending ΔA and ΔB.
4. Draw a circle with a radius BΓ around B and let the point H be the intersection of the circle and BZ.
5. Draw a circle with a radius ΔH around Δ and let the point λ be the intersection of the circle and AE.
6. By definition BΓ = BH and Δλ = ΔH, ΔA = ΔB, so Aλ = BH, ie Aλ = BΓ.
7. A line segment Aλ equal to the given line segment BΓ at a given point A was created.

Implementation procedure on Isabelle is as follows.

- 1*. Prepare points A, B, Γ, Δ, λ, H, line segments ΔA, Aλ, ΔB, BΓ, AB, BH, Δλ, ΔH, line segments + line segments ΔA_Aλ, ΔB_BΓ, ΔB_BH, circles BΓ and ΔH.
- 2*. AB = ΔA, ΔA = ΔB, ΔB = AB (Prove 1).
- 3*. Let points H and Γ be the points on the circumference of the circle BΓ.
- 4*. Let points λ and H be the points on the circumference of the circle ΔH.
- 5*. Δλ = ΔA_Aλ, ΔH = ΔB_BH.
- 6*. Prove ΔA_Aλ = ΔB_BH.
- 7*. Prove ΔB_BH = ΔB_BΓ.
- 8*. From 6* and 7*, we prove ΔA_Aλ = ΔB_BΓ.
- 9*. From 8*, we prove Aλ = BΓ.

The full text of proof by isabelle is shown below.

```
theorem (in circleddef) Prove2:
  fixes A B Γ Δ λ H :: point
  and ΔA Aλ ΔB BΓ AB BH Δλ ΔH :: segment
  and ΔA_Aλ ΔB_BΓ ΔB_BH :: seg_list
  and CircBΓ CircΔH :: circle
  assumes "ΔA = Se Δ A" "Aλ = Se A λ" "ΔB = Se Δ
  B"
  "BΓ = Se B Γ" "AB = Se A B" "BH = Se B H"
```

```
"Δλ = Se Δ λ" "ΔH = Se Δ H"
"ΔA_Aλ = Se1 ΔA Aλ" "ΔB_BΓ = Se1 ΔB BΓ"
"ΔB_BH = Se1 ΔB BH"
"Δλ [@-@] ΔA_Aλ" "ΔH [@-@] ΔB_BH"
"CircBΓ = Ci B Γ" "CircΔH = Ci Δ H"
"λ [on] CircΔH" "H [on] CircBΓ" "H [on] CircΔH"
"Γ [on] CircBΓ"
"AB [@] ΔA" "ΔA [@] ΔB" "ΔB [@] AB"
(*Prove1*)
shows "Aλ [@] BΓ"
proof -
  from assms have P1:"ΔA_Aλ [@@] ΔB_BH"
  by (simp add:circle_dist2 dist_rev2 dist_list_list)
  from assms have P2:"ΔB_BH [@@] ΔB_BΓ"
  by (simp add:dist_list list_rev2 circle_dist2
  dist_rev2)
  from assms P1 P2 have P3:"ΔA_Aλ [@@] ΔB_BΓ"
  by (blast intro:list_trans)
  from assms P3 show "Aλ [@] BΓ" by (simp
  add:list_dist3)
qed
```

Prove3 : Suppose there are two unequal line segments. Cut out the line segment equal to the smaller one from the larger one.

The proof process shown in [5] becomes as follows.

1. Let AB and Γ are unequal line segments given, and let AB > Γ.
2. At point A, we create a line segment AΔ equal to line segment Γ (Prove 2).
3. Draws a circle with a radius AΔ around A and the intersection of the circle and AB is the point E.
4. By definition, AE = AΔ and AΔ = Γ so AE = Γ.
5. AE equal to the smaller Γ from the larger AB was cutted out.

Implementation procedure on Isabelle is as follows.

- 1*. Prepare points A, B, Δ, E, line segments AB, AΔ, AE, Γ, line segments + line segments AE_EB and circle AΔ.
- 2*. Γ = AE (Prove 2).
- 3*. Let E be a point on the circumference of the circle AΔ.
- 4*. AB = AE_EB.
- 5*. Prove Γ = AE.

The full text of proof by isabelle is shown below.

```

theorem (in circleddef) Prove3:
  fixes A B Δ E :: point
  and AB AΔ AE Γ :: segment
  and AE_EB :: seg_list
  and CircAΔ :: circle
  assumes "AB = Se A B" "AΔ = Se A Δ" "AE = Se A
E"
  "CircAΔ = Ci A Δ"
  "E [on] CircAΔ"
  "AB [@-@] AE_EB"
  "Γ [@] AΔ" (*Prove2*)
  shows "Γ [@] AE"
  using assms
  apply (simp add:circle_dist1)
  done
    
```

DISCUSSION AND CONCLUSION

In the 1990s, there were computers for introducing computer-based mathematics education, and the automated theorem proving was part of this attempt. These proposals were expected to develop ways in which to use computer effectively as "tools" for mathematics education and to achieve good results as a teaching method for mathematics education. The automated theorem proving was not such an easy thing. Researchers were uncertain as to what kind of basic principles the utilization of computer for mathematics education stood for, or what goals they were trying to achieve. For this reason, only a relatively small number of enthusiastic mathematics educators were involved in the research. Strategic use requires the student to think about the problem and the tool. To reach this stage, students must encourage to develop the habits of discerning and strategic use of theorem prover along with the elements of algebraic insight required. In this way, theorem prover gives many opportunities for rich mathematical learning.

Technology in mathematics education is the fulfilment of the aspiration of mathematics education in the computer age. By recent advances in computational logic, computational algebra and software technology, the automation of reasoning promises to become practical feasible and useful for mathematics education.

ACKNOWLEDGEMENTS

This work is supported in part by MEXT, Japan.

REFERENCES

- [1] Wikipedia, Automated theorem proving, https://en.wikipedia.org/wiki/Automated_theorem_proving
- [2] Isabelle/HOL— Higher-Order Logic, <https://www.cl.cam.ac.uk/research/hvg/Isabelle/dist/library/HOL/HOL/document.pdf#search=%27Isabelle%2FHOL%27>
- [3] Kobayashi H, Suzuki H and Ono Y. 2005. Formalization of Henzel's Lemma, 18th International Conference, TPHOLs, Oxford, UK, Emerging Trends Proceedings Oxford Research Report : 924-927.
- [4] He C, Kobayashi H, Suzuki H. 2006. An implementation of a database system for theorem search, ICCSE'2006: Proceedings of the first international conference on computer science & education: 287-288.
- [5] Fitzpatrick R, EUCLID'S ELEMENTS OF GEOMETRY, <http://farside.ph.utexas.edu/Books/Euclid/Elements.pdf#search=%27Euclid+Elements%27>