Automated Euclidean Plane Geometry Problem Solving with Neuro-Symbolic Reasoning

——Introduction of AlphaGeometry and Optimization Attempts on it

HAN YAOTING

3220101611@zju.edu.cn
State Key Laboratory of Computer Aided Design and Graphics (CAD&CG)

School of Mathematical Sciences
Zhejiang University

May 14 2025

Contents

Computer-Aided Mathematical Research

- Data Generation: Generate large data sets of mathematical objects to perform computations and to make conjectures.
  - Legendre and Gauss conjecture ”Prime Number Theorem” by using extensive tables of prime numbers;
  - Birch and Swinnerton-Dyer propose conjectures on elliptic curves over finite fields by generating enough data on it.
- Scientific Computation: Widely used to solve differential equations, dynamical systems and statistics of large matrices
  - Hendrik Lorentz assembled a team of human computers to model the fluid flow around the Afsluitdijk;
  - Numerical analysis of Navier–Stokes equations

Computer-Aided Mathematical Theorem Proving

Four color theorem

Given any separation of a plane into contiguous regions, the regions can be colored using at most four colors so that no two adjacent regions have the same color. (intuitive statement)
For a loopless planar graph G, its chromatic number is $\chi(G) \leq 4$. (graph-theoretic statement)

- The first major theorem mainly proved with the help of a computer and computers check the "reducibility" and "unavoidability." of almost 2000 configurations. [1]
- The proof was not accepted initially because it's lack of logical consistency and hard for a human to check by hand.

[1] Appel and Haken, "Every Planar Map Is Four Colorable", 1976

Computer-Aided Mathematical Theorem Proving

Boolean Pythagorean triples theorem

The set $\{1, \dots, 7824\}$ can be partitioned into two classes, neither of which contains a Pythagorean triple (a, b, c) with $a^2 + b^2 = c^2$; however, this is not possible for $\{1, \dots, 7825\}$.

- The proof required four CPU-years of computation and generated a 200 terabyte propositional proof, which was later compressed to 68 gigabytes.[2]
- It is a typical example of a result proved using a satisfiability (SAT) solvers, both which and satisfiability modulo theories (SMT) solvers can perform complex logical deductions from certain restricted sets of hypotheses.

[2] Heule, Marijn J. H.; Kullmann, Oliver; Marek, Victor W., "Solving and Verifying the Boolean Pythagorean Triples problem via Cube-and-Conquer", 2016

Two types of Automated Reasoning and Formal Proofs

- Interactive theorem proving:
  - Proof assistants such as Coq, Isabelle, and Lean.
  - Expressive logic[a], e.g., dependent type theory and is successfully used in large formalization projects; [b]
  - Needs lots of efforts from humans to write formalized proof.

- Automated theorem proving:
  - SMT solvers, model checkers, ATP systems in first-order logic, etc.
  - Minimal efforts from humans;
  - Limited expressiveness and difficult to scale.

[a] Leroy, "Formal Verification of a Realistic Compiler", 2008.

[b] Hales et al., "A Formal Proof of the Kepler Conjecture", 2017.

Proof Assistants

- Goal for Formalization: verify the correctness of the conclusion of a logical or mathematical argument.
- Modern proof assistants: Coq, Isabelle, or Lean,

An Example of Theorem Proof in Lean

- natural statement:
  $\forall n \in \mathbb{N}$, the greatest common divisor of n and itself is n itself.
- Lean statement:

```
Theorem gcd_self (n : nat) : gcd n n = n :=
begin
    cases n,
    { unfold gcd },
    unfold gcd,
    rewrite mod_self,
    apply gcd_zero_left
end
```

Proof Assistants

- Characteristics of Formalization:
  - Code only compiles if the proof is valid;
  - Task of formalization on a natural proof that compiles in a formal proof assistant is quite time consuming.
- Values of Formalization:
  - provides an extremely high level of confidence that a given result is correct;
  - uncovers minor issues in the human proof, and sometimes reveals simplifications or strengthenings of the argument;
  - Contributes many basic mathematical results to a mathematical library during the process;
  - Offers a chance that more researchers can be allowed to participate in a project without considering the accuracy of their work.

Math Problems Solving with LLMs

- The performance of logical reasoning, particularly in math, is now becoming one of the main metrics for evaluating the capabilities of LLMs;
- Apply such general-purpose LLMs to attack mathematical problems directly may get the wrong result, which is called "Hallucination";

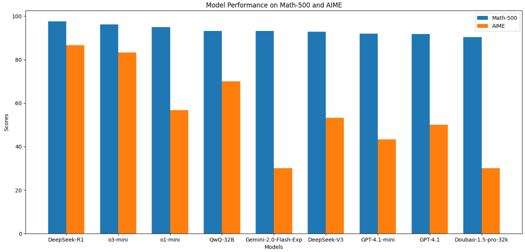
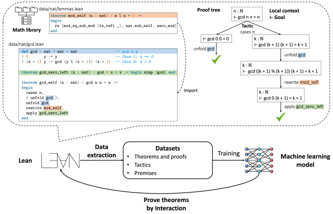


Fig: Performance of several models on Math-550 and AIME[3]

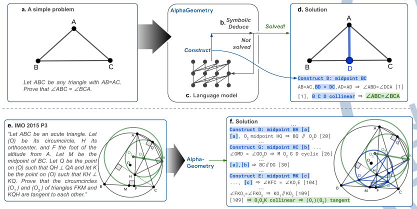
[3]Data comes from https://opencompass.org.cn

Math Problems Solving with LLMs

- Two ways to generate reliable results using LLMs:
  - Ask LLMs to generate a step of proof including static and possible premises rather than to try building the whole proof directly, and then executing that program in formalized language to reliably verify the quality of the output.[4]
  - Use LLMs to enhance existing symbolic proof engines to attack narrow classes of mathematics problems, such as Olympiad geometry problems.[5]



(a) Diagram of Leandojo



(b) Diagram ofAlphaGeometry

[4] Kaiyu Yang et al., "LeanDojo: Theorem Proving with Retrieval-Augmented Language Models" ,2023.

[5] Trieu H. Trinh et al., "Solving olympiad geometry without human demonstrations" ,2024.

Reasons for Choosing Euclidean Plane Geometry

- Clear and axiomatized system allows for the formalization and systematization of geometric theorem proofs, making them easier for computer to perform symbolic reasoning and verify;
- The proof often requires creative construction of auxiliary points, which can be given by LLMs;
- It would be possible to attempt to have LLMs derive most of the theorems in Euclidean plane geometry symbolically from the most basic axioms and compare them with the content constructed by humans.

Wu's Method

- Core Idea: Transform geometric problems into algebraic problems and then prove them mechanically through pseudodivision of polynomials.
- The form of the Transcribed Problem:

$$\forall x, y, z, \ldots I(x, y, z, \ldots) \implies f(x, y, z, \ldots)$$

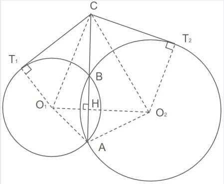
  where f is a polynomial equation and I is a conjunction of polynomial equations. The algorithm is complete for such problems over the complex domain.
- Brief Introduction of Algorithm: For an ideal I in the ring $K[x_1, x_2, \ldots, x_n]$ over a field K, a (Ritt) characteristic set C of I is composed of a set of polynomials in I, which is in triangular shape. Given a characteristic set C of I, one can decide if a polynomial f is zero modulo I.

Wu's Method

An Easy Example

Problem

As shown in figure, let circles O_1 and O_2 intersect at points A and B. Take any point C on the line containing A and B. Draw tangents CT_1 and CT_2 from C to circles O_1 and O_2 respectively. Show that $|CT_1| = |CT_2|$.



Natural Proof

The length of the tangent are $CT_1^2 = CO_1^2 - O_1T_1^2 = CO_1^2 - O_1A^2$ and $CT_2^2 = CO_2^2 - O_2A^2$.
Connect O_1O_2 and let it intersect AB at point H. Since A and B are the intersection points of the two circles, $O_1O_2 \perp AB$.
Thus we have $CO_1^2 - O_1A^2 = CO_2^2 - O_2A^2$, which implies $CT_1 = CT_2$.

Wu's Method

An Easy Example

Wu's Method Proof

Assume the plane is the complex plane, the coordinates of O_1, O_2 are z_1, z_2 and the radii are r_1, r_2. Then solve for the coordinates of A, B and the equation of that line by combining the equations of the circles:

$$\begin{aligned} O_1 : \quad & (z - z_1)(\bar{z} - \bar{z}_1) = r_1^2 \\ O_2 : \quad & (z - z_2)(\bar{z} - \bar{z}_2) = r_2^2 \end{aligned}$$

The result of subtracting the two equations is the equation of the line $AB : z(\bar{z}_2 - \bar{z}_1) + \bar{z}(z_2 - z_1) + (z_1\bar{z}_1 - z_2\bar{z}_2) = r_1^2 - r_2^2$.

The equality of the lengths of the tangents is equivalent to verifying the polynomial:

$$f(z) = (|z - z_1|^2 - r_1^2) - (|z - z_2|^2 - r_2^2)$$

is equal to 0 when $z \in AB$.

Wu's Method

- Characteristics:
  - The proof process of Wu's method is clearly algorithmic and can be easily implemented on a computer, e.g. mathematica;
  - Automatically identify the non-degenerate conditions of the problems.
- Disadvantages:
  - For enough complex problem, the scale and degrees of polynomials are large and high , which consume a large amount of computational resources and time;
  - Its highly mechanized process makes it difficult for human to read and understand;
  - When condition comes to inequality, it struggles to handle them effectively.

AlphaGeometry

- An AI system developed by Google DeepMind to solve complex geometry problems.
- In a benchmark test of 30 Olympiad-level geometry problems, AlphaGeometry1 solved 25, compared to the human gold - medalists' average of 25.9. [6]. Now AlphaGeometry2 solved 42 on a benchmark test of 50 Olympiad-level geometry problems. [7]

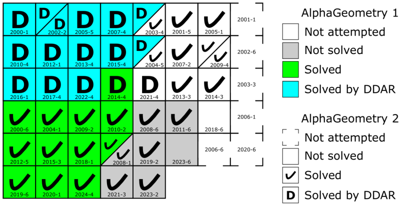


Fig: Results on all 2000-2024 IMO geometry problems

[6] Trieu H. Trinh et al., "Solving olympiad geometry without human demonstrations" ,2024.
[7] Yuri Chervonyi et al., "Gold-medalist Performance in Solving Olympiad Geometry with AlphaGeometry2",2025

AlphaGeometry

Process of problems solving in AG

1. Input a formalized geometry problem;
2. Do Deductive Database (DD) and Algebraic Reasoning (AR) to gain a deduction closure and new algebraic relationships respectively;
3. If the target conclusion is in the deduction closure, the problem is proven.
4. If not, feed the new algebraic relationships given by AR back to DD to further expand the deduction closure (step 3).
5. If the target conclusion is still not in the new deduction closure, then use LM to generate auxiliary constructions (points) and back to step 2.

Formalization

In AlphaGeometry (AG), Below are the main components and features of the formalization language:

- Basic Elements
  - Points: Represented by single lowercase letters, e.g., a.
  - Lines: Represented by two points, e.g., line a b denotes the line passing through points a and b.
  - Circles: Represented by a center and a point on the circumference, e.g., circle a b denotes a circle centered at point a and passing through point b.
- Relations and Assertions
  The AG language uses a set of relations and assertions to describe the properties and interactions between geometric objects.
  - Collinearity: coll a b c indicates that points a, b, and c are collinear.
  - Concyclicity: cyclic a b c d indicates that points a, b, c, and d are concyclic.
  - Equality of Distances: cong a b c d indicates that line segment ab is equal to line segment cd.
  - ...

Deductive Database

- Definition:
  DD is a rule-based reasoning system that derives new geometric facts from given premises. It expands the set of known facts by applying a series of predefined geometric rules until the target conclusion is reached or no new facts can be derived.[8]
- Working Principle:
  - Input: A set of geometric premises, for example, "points A, B, C are collinear".
  - Rule Application: Derives new geometric facts based on predefined geometric rules, for example, "if points A, B, C are collinear, and points B, C, D are collinear, then points A, B, C, D are collinear".
  - Output: A set of all derived facts, known as the deduction closure. If the target conclusion is in the deduction closure, the problem is proven.

[8] Ye, Z., Chou,"Automated Deduction in Geometry", 2011

Algebraic Reasoning

- Definition:
  AR is an algebra-based reasoning engine used to handle algebraic relationships in geometric problems, such as calculations involving angles, ratios, and distances.
- Working Principle:
  - Input: A set of geometric premises, which can be equations or inequalities involving angles, ratios, and distances, for example, $\angle ABC = \angle DEF$.
  - Rule Application: processes the input equations using algebraic methods (such as Gaussian elimination) to derive new algebraic relationships.
  - Output: A set of new algebraic relationships

Algebraic Reasoning

- Details:
  In geometry, any equality is of the form $a - b = c - d \Leftrightarrow a - b - c + d = 0$, so AR populate the row for each equality at columns of matrix A, then Gaussian elimination on A returns a new matrix with leading 1s at each of the columns, essentially representing each variable as a unique linear combination of all remaining variables.
  As an example, suppose we have $a - b = b - c$, $d - c = a - d$ and $b - c = c - e$, then AR can get:

$$\begin{pmatrix} a & b & c & d & e \\ 1 & -2 & 1 & 0 & 0 \\ -1 & 0 & -1 & 2 & 0 \\ 0 & 1 & -2 & 0 & 1 \end{pmatrix} \xrightarrow{\text{GE}} \begin{pmatrix} a & b & c & d & e \\ 1 & 0 & 0 & -1.5 & 0.5 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & -0.5 & -0.5 \end{pmatrix} \Rightarrow \begin{cases} a = 1.5d - 0.5e \\ b = d \\ c = 0.5d + 0.5e \end{cases}$$

  From this result, AR can deterministically and exhaustively deduce all new equalities.

Auxiliary Constructions Given by LM

- In many complex geometric problems, symbolic reasoning engines (DD and AR) alone may not be able to directly prove the target conclusion. Auxiliary constructions are often needed to help derive the target conclusion.
- The LM gets a formalized problem and then generates one or more descriptions of auxiliary points.
- The number of problems requiring auxiliary points reaches 26 in AG2.

Ongoing Work and Future Plans

References

- Kaiyu Yang, Aidan M. Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan Prenger, and Anima Anandkumar, LeanDojo: Theorem Proving with Retrieval-Augmented Language Models, Oct. 27, 2023
- Trieu H. Trinh, Yuhuai Wu, Quoc V. Le et al., Solving olympiad geometry without human demonstrations, Jan. 18, 2024
- Yuri Chervonyi, Trieu H. Trinh, Miroslav Olšák, Xiaomeng Yang, Hoang Nguyen, Marcelo Menegali, Junehyuk Jung, Vikas Verma, Quoc V. Le, and Thang Luong, Gold-medalist Performance in Solving Olympiad Geometry with AlphaGeometry2, Feb. 28, 2025
- Vladmir Sicca, Tianxiang Xia, Mathïs Fédérico, Philip John Gorinski, Simon Frieder, and Shangling Jui, Newclid: A User-Friendly Replacement for AlphaGeometry, Nov. 20, 2024