The Geometrician: a computer model for problem solving in the field of geometry.

Edgar Acosta Villaseñor and **Rafael Pérez y Pérez**
Instituto de Investigación en Matemáticas Aplicadas y en Sistemas
Universidad Nacional Autónoma de México
Circuito Interior s/n, Ciudad Universitaria, Distrito Federal, México, 04510
Tel. (52 55) 5622 3619x44027
acosta.edgar@gmail.com rpyp@servidor.unam.mx

Keywords: engagement, reflection, creativity, problem solving, geometry.

Abstract

The Geometrician is a computer program that solves geometric construction problems based on a computer model of creativity known as Engagement-Reflection. Our goal is to demonstrate that this model can be employed in geometry problem-solving. This paper describes the general characteristics of The Geometrician and illustrates some examples of problems solved by the program. After testing our system we conclude that The Geometrician is able to solve new problems demostrating that the Engagement-Reflection approach to creativity can be applied to problem solving.

1 Introduction

The Geometrician is a computer program that solves geometric construction problems, i.e. problems that are solved employing only a straightedge and a compass. So, given some initial geometric objects (e.g. points, lines, circles and distances) the system builds new geometric objects that satisfy some given requirements. In this way, the solution of a problem consists on a sequence of drawings. A typical example of the kind of problems that The Geometrician solves is: given a point A and a distance d, construct a point B such that the distance between point A and point B is equal to d. As far as we know, there are not other computer programs that perform a similar task (c.f. with programs that perform geometry theorems demonstrations, e.g. [Tarski, 1951; Seidenberg, 1954; Collins, 1975; Gelernter, 1959; 1995; Gilmore, 1970; Nevins, 1975; McCharen *et al.*, 1976; Anderson, 1981; Coelho and Pereira, 1986; Koedinger and Anderson, 1990; Wen-Tsün, 1978; 1984; Chou *et al.*, 1994; 1996])

The system is based on the Engagement-Reflection computer model of creativity [Pérez y Pérez, 1999; Pérez y Pérez and Sharples, 2001]. During the engagement-mode the system produces material (sequences of actions to create geometric objects) driven by the constraints imposed by the context of the problem; as an important characteristic, during engagement the system does not perform any kind of analysis or evaluation on the material generated. During the reflection-mode the system breaks impasses originated during engagement, satisfies coherence-requirements, and evaluates the adequacy of the solution produced so far. The outcomes of the program are the consequence of the interaction between engagement and reflection. The system represents problem-stages as a structure formed by four lists: 1) a list of known geometric objects; 2) a list of unknown geometric objects; 3) a list of satisfied relations between geometric objects; 4) a list of unsatisfied relations between geometric objects. These four lists are referred to as the Context of the Problem (CP). When the system performs an action the CP is modified (new objects might be created or new relations established).

When The Geometrician starts it performs two main processes: 1) the construction of knowledge structures in memory; 2) the solution of a problem through an engagement-reflection cycle.

Construction of knowledge structures.
The user provides the system with a set of examples of how to solve geometric problems known as the *previous experience*. The system records the CP for each step in the solution of each problem in the previous experience and creates a group of knowledge structures in memory referred to as atoms.

Engagement-reflection cycle.
The system always starts in engagement. Given an initial CP the system looks for atoms in memory containing similar contexts in order to retrieve a possible next action to perform. In this way, the engagement-reflection cycle is triggered.

The following sections describe details of these processes. This paper is organised as follows: section 2 describes how knowledge structures are created in the system; section 3 describes how the system solves a problem; section 4 shows examples of problems that The Geometrician solves; section 5 evaluates the program; and section 6 offers some conclusions.

2 Knowledge Representation

The program makes use of knowledge structures in order to represent:

- geometric objects,
- relations between objects,
- description of problem-stages,
- actions that can be performed to solve a problem.

this section describes each of this structures and how they are employed.

2.1 Geometric Objects Representation

The Geometrician works with four types of geometric objects: points, lines, circles and distances. A circle C with center in point A and a distance r as radius is represented as follows:

$$\text{circle } C\ (A, r)$$

i.e. we represent an object with a sentence that begins with the type of the object (in this case a circle) followed by the name of the object (in this case C), and inside parenthesis the objects required for describing (and drawing) the object (in this case objects A and r).

In order to represent a point situated on another object we employ the same notation followed by the word *on*. For example, a point P on circle K is represented by:

$$\text{point } P\ (K) \text{ on}$$

In the same way we employ

$$\text{point } P\ (l, m) \text{ intersection.}$$

to represent a point situated on the intersection of two geometric objects l and m.

2.2 Relations between Objects

In this paper we represent the relations with the usual mathematical notation. For example, two equal distances d and r are represented by $d = r$.

2.3 Problem-stages Representation

The structure used to represent a problem-stage is called **Context of the Problem (CP)** and consists of four lists:

- list of known geometric objects,
- list of unknown objects,
- list of satisfied relations between objects, and
- list of unsatisfied relations between geometric objects.

The next structure represents the problem: *Given the point A and the distance d construct a point P such that $|\overline{AP}| = d$.*

Known Objects	**Satisfied Relations**
point A distance d	
Unknown Objects	**Unsatisfied Relations**
point P distance $d'(A, P)$	$d' = d$

Each time an action is performed the CP is updated, i. e. objects and relations might be added or eliminated in the four lists. So, a problem is solved when the lists of unknown objects and unsatisfied relations are empty.

2.4 Actions

All actions in The Geometrician have associated a set of consequences and preconditions. The system includes three types of consequences:

1. drawing a geometric object
2. establishing relations between objects
3. relocating unknown objects to the list of known objects.

Preconditions are formed by all those geometric objects needed to construct a new object. Therefore, in order to perform an action, all its preconditions must be part of the known objects list in the CP. Notice that during Engagement the system ignores preconditions; only during Reflection The Geometrician verifies that they are satisfied.

Actions are represented by the name of the action and its list of consequences:

Action i
List of consequences

For example, given the following CP:

Known Objects	**Satisfied Relations**
point A distance d circle C	
Unknown Objects	**Unsatisfied Relations**
point P distance $r(A, P)$	$r = d$

A possible action to solve this problem might be:

Action ii
point $P(C)$ on
satisfies: $P \neq A$, $r = d$
moves: r

where the first line in the description of consequences indicates that the system must draw a point P on circle C; the second line establishes the relations $P \neq A$ and $r = d$; finally, the third line indicates that the system moves the distance r to the list of known objects.

2.5 Building knowledge structures in memory

The Geometrician builds its knowledge from a set of solved problems given by the user of the system. We refer to this set as *previous experience*.

To build its knowledge structures The Geometrician takes the CP that represents the initial stage of a problem in the previous experiences. This CP is modified each time an action is executed by the system. For example, let us assume that the solution for a given problem is the sequence of actions $A_1, A_2, \ldots, A_k$ and C_0 represents the initial CP. i.e. the description of the problem. After executing the first action, C_0 is modified by the consequences of the action A_1 and a new context C_1 is created.

$$C_0 + A_1 \Longrightarrow C_1$$

We can repeat this process for each action in the solution.

$$\mathbf{C_0} + A_1 \Longrightarrow \mathbf{C_1} + A_2 \Longrightarrow \mathbf{C_2} + A_3 \Longrightarrow \ldots \Longrightarrow \mathbf{C_{k-1}} + A_k \Longrightarrow \mathbf{C_k}$$

At the end we obtain a collection of contexts and logical actions to perform. The Geometrician copies each of this contexts $(C_0, C_1, C_2, \ldots, C_k)$ and its following action to memory (see fig. 1). In this paper we refer to each context copied to memory as an atom. So, knowledge structures are formed by a set of atoms and their associated actions to perform

$$C_0 \longrightarrow \{A_1\} \quad C1 \longrightarrow \{A_2\} \quad C2 \longrightarrow \{A_3\}$$
$$\ldots$$
$$C_{k-2} \longrightarrow \{A_{k-1}\} \quad C_{k-1} \longrightarrow \{A_k\}$$

Figure 1: **The knowledge structures in memory are associations between atoms and sets of actions**

3 Solving a problem

The system starts solving a problem once a CP, representing the initial stage of the problem, is provided.

The number of actions produced during Engagement is determined by a variable that can be modified by the user. With a value of one in this variable, The Geometrician enters to the Reflection-mode after each action is produced. In the other hand, with a high value in this variable, the system produce many actions before evaluating them in the Reflection-mode.

No matter what value this variable has, whenever The Geometrician is unable to produce an action, an impasse is declared and the system enters to Reflection.

3.1 Engagement

During Engagement the system employs the current CP as cue to probe memory and retrieve actions to solve the problem.

The Geometrician follows three search strategies. The first, called exact-strategy, tries to match an atom in memory identical to the CP, i.e. they are composed by exactly the same elements (the same type of objects and relations).

The second search strategy, called inclusive-strategy, consists of searching memory for atoms similar to the CP such that contains most of the CP as part of its organization.

The third search strategy, called dynamic-strategy, consists of eliminating some elements from the CP and triggering a new search in memory. The logic behind this strategy is that as long as the essential features of the CP are conserved, the system is able to retrieve logical actions to solve the problem.

The process for retrieving possible next actions from memory works as follows:

1. The CP represents the current state of the problem in progress.
2. The Geometrician searches memory applying the exact-strategy.
3. If no structure is matched; The Geometrician applies the inclusive-strategy.
4. If no structure is matched and the system has not employed the third search strategy yet, it eliminates some elements from the CP and goes back to step 2. Otherwise an impasse is declared.

These search strategies provide the system with an important flexibility during the retrieval process.

3.2 Reflection

During Reflection the system performs two main processes: it verifies that preconditions of the retrieved actions are fulfilled and breaks impasses.

Action's preconditions are not verified during Engagement, therefore, sequences of not coherent actions might be produced. During Reflection, if an unsatisfied precondition is found The Geometrician inserts those actions necessary to build the geometric objects needed to satisfy the preconditions.

The Geometrician breaks impasses with the help of three heuristics. The first determines if it is possible to build new geometric objects: if all the preconditions of an unknown object X are known objects and there is no unsatisfied relations in which X participates, then X can be constructed. The second heuristic atempts to merge two or more actions in the solution in progress. Two actions can be merged if their consequences independently build a point P on object X_i; in this way, the new merged action has as consequence the construction of a point P on $\cap\{X_i\}$ (the intersection of all the objects X_i). The third heuristic looks in the CP for Unsatisfied relations that can be satisfied by a Self-Generated Action (see below). The goal of these heuristics is to modify the CP in order to switch to Engagement and then try again to match an atom in memory. If the impasse cannot be broken the system declares itself unable to solve the problem.

Self-Generated Actions (SGA)

The Geometrician makes use of a special kind of actions called Self-generated actions. SGA groups sequences of actions into a single command. In this way, the steps performed by The Geometrician to solve a problem can become a new action into the system.

For example, The Geometrician does not know how to reflect a point through a line, but the user can ask the system to solve a problem whose goal is to construct a point that has to meet all the conditions of the reflect of another point through a line. Once the system sorts it out, the user can indicate The Geometrician to generate a SGA establishing that the constructed point is known as the *reflect* of the given point through the given line. Now, the program can use this new SGA (reflect) as another action to solve new problems.

3.3 Interaction between E & R modes

The program starts producing actions in the Engagement-mode. Once the system triggers an engagement-reflection cycle, it only stops when:

1. The lists of unknown objects and unsatisfied relations of the CP are empty.
2. An unbreakable impasse is declared.
3. An action's precondition cannot be satisfied.

After the engagement-reflection cycle ends, The Geometrician performs a final analysis. Its purpose is to try to produce a shorter solution. The Geometrician accomplishes this with the use of the second heuristic used to break impasses.

The last step is to print the final solution.

4 Examples

The Geometrician is able to solve new problems, i.e. problems different to the problems for which the system knows a solution.

In a series of experiments The Geometrician was provided with solutions to the following problems:

Problem 1 *Given a point A and distance d, construct a point B such that $|\overline{AB}| = d$ (the distance between points A and B is equal to d).*

Problem 2 *Given the line l, construct a point P on l.*

Problem 3 *Given two known points A and B, build a distance d.*

The solutions of this problems are very short. For example, solution to problem 1 has only two actions:

Action A

circle $C(A, d)$	Draw a circle C with center A and radius d.
satisfies: $B \in C$	Point B is on C.

Action B

point B (C) on	Draw point B on C.
satisfies: $|\overline{AB}| = d$	In this way $|\overline{AB}| = d$ and
move: $B, |\overline{AB}|$	point B and distance $|\overline{AB}|$ are known objects.

Let us see how this solution is used to build knowledge structures in memory: the inital CP of problem 1 is the CP A

CP A

Known Objects	Satisfied Relations
point A distance d	
Unknown Objects	**Unsatisfied Relations**
point B distance $r(A, B)$	$r = d$

This CP is copied into memory, becoming an atom, and action A becomes one of its associated actions.

After executing action A, their consequences modifies CP A to obtain CP B.

CP B

Known Objects	Satisfied Relations
point A distance d circle $C(A, d)$	$B \in C$
Unknown Objects	**Unsatisfied Relations**
point B distance $r(A, B)$	$r = d$

Now CP B is copied into memory as an atom and action B becomes one of its associated actions.

Then, action B is executed, thus the CP B is modified obtaining CP C.

CP C

Known Objects	Satisfied Relations
point A distance d circle $C(A, d)$ point $B(C)$ on distance $r(A, B)$	$B \in C$ $r = d$
Unknown Objects	**Unsatisfied Relations**

Since the lists of unknown objects and unsatisfied relations of the CP C are empty, the problem is solved.

With this previous experience The Geometrician is able to solve problems 4, 5, 6 and 7 among other problems.

Problem 4 (Reflect of a point through a line) *Given points A and B on line l and point C not on line l, draw a point P different to C such that $|\overline{AP}| = |\overline{AC}|$ and $|\overline{BP}| = |\overline{BC}|$.*

Problem 5 *Given a line l, a point A on l and any point P. Draw a point B on l such that $|\overline{BP}| = |\overline{AP}|$.*

Problem 6 *Given the distance d and points A and B, construct a point C such that $|\overline{AC}| = |\overline{BC}| = d$.*

Problem 7 (perpendicular line) *Given a line l and a point D not on l, draw a line m through points D and D' (the reflection of D through l).*

In order to generate a solution for problem 4, The Geometrician first takes de initial CP of his problem (CP 4) and then enters to the Engagement-mode.

CP 1

Known Objects	Satisfied Relations
line l point A point B point C distance $d(A, C)$ distance $e(B, C)$	$A \in l$ $B \in l$ $C \notin l$
Unknown Objects	**Unsatisfied Relations**
point P distance $d'(A, P)$ distance $e'(B.P)$	$P \neq C$ $d' = d$ $e' = e$

To produce a first action, the system tries to match atoms in memory applying the exact-strategy without success. Then the system employs the inclusive-strategy and again it is unable to match atoms in memory. Then The Geometrician applies the dynamic-startegy in which produces (by deleting elements in the CP 4) the following three alternative structures (AS) to match atoms:

AS 1.1

Known Objects	Satisfied Relations
point C	
Unknown Objects	**Unsatisfied Relations**
point P	$P \neq C$

AS 1.2

Known Objects	**Satisfied Relations**
point A point C distance $d(A,C)$	
Unknown Objects	**Unsatisfied Relations**
point P distance $d'(A,P)$	$d' = d$

AS 1.3

Known Objects	**Satisfied Relations**
point B point C distance $e(B,C)$	
Unknown Objects	**Unsatisfied Relations**
point P distance $e'(B,P)$	$e' = e$

Now the system launches a search for each of these AS and finds that AS 1.2 and AS 1.3 are similar to atom CP A, which is associated to action A, so the system retrieves the action A two times as possible action to perform next:

Action iii
circle W (A,d)
satisfies: $P \in W$

Action iv
circle K (B,e)
satisfies: $P \in K$

Finally, from these two actions, action iv is randomly selected to become the first action to solve the problem:

Action 1

circle K (B,e)	Draw circle K with center B and radius $\lvert\overline{BC}\rvert$.
satisfies: $P \in K$	Point P is on circle K.

During Engagement another 2 actions are produced in a similar way:

Action 2

circle W (A,d)	Draw circle W with center A and radius $\lvert\overline{AC}\rvert$.
satisfies: $P \in W$	Point P is on circle W.

Action 3

point $P(W)$ select satisfies: $d' = d$ move: P,d'	Draw point P on circle W. In this way $\lvert\overline{AP}\rvert = \lvert\overline{AC}\rvert$ and point P and distance $\lvert\overline{AP}\rvert$ are known objects.

Then the system enters to the Reflection-mode where preconditions of actions are verified. In this case all preconditions are satisfied. Now the program returns to the Engagement-mode to produce more actions:

Action 4

point $P(K)$ on satisfies: $e' = e$ move: P,e'	Draw point P on circle K. In this way $\lvert\overline{BP}\rvert = \lvert\overline{BC}\rvert$ and point P and distance $\lvert\overline{BP}\rvert$ are known objects.

After producing action 4, the program cannot produce more actions so an impasse is declared, switching to the Reflection-mode. In the Reflection-mode preconditions of action 4 are verified. Since all of them are fulfilled, the system tries to break the impasse. This is accomplished with the use of the second heuristic to break impasses which produces action 5 to substitute actions 3 and 4.

Action 5

point P (K,W) intersection	Draw point P on the intersection of circles K and W.
satisfies: $d' = d$, $e' = e$, $P \neq C$	In this way $\lvert\overline{AP}\rvert = \lvert\overline{AC}\rvert$, $\lvert\overline{BP}\rvert = \lvert\overline{BC}\rvert$, $P \neq C$
move: d',e'	and distances $\lvert\overline{AP}\rvert$ and $\lvert\overline{BP}\rvert$ are known objects.

After performing action 5, the lists of unknown objects and unsatisfied relations in the CP are empty so the problem is solved. In this way the solution to problem 4 is composed by actions 1, 2 and 5.

As we can see, The Geometrician solves problems by means of the interaction between the engagement and reflection modes and thanks to knowledge embodied in the previous experience. Previous experience determines which problems can be solved by the system.

5 Evaluation

The Geometrician's solutions were evaluated by means of an Internet questionnaire. Twenty six subjects answered it: 24 of them were Mexicans, 1 from U.S.A. and 1 from Roumania. Of them 42% were undergraduates, 58% were graduates and 38% possesed postgraduate degrees.

The questionnaire included solutions to problems 4 and 5. For each problem it was provided a solution by The Geometrician, a solution by an expert and at least two solutions by children 10 years old. For each solution the subjects were asked to:

1. evaluate if the solution was correct.
2. indicate if the problem was solved by a beginner or by an expert.
3. indicate the age of the person that solved the problem.

It was hypothesized that solutions proposed by The Geometrician would have lower rates than the solutions proposed by an expert and similar rates to the solutions proposed by children. The results are shown in tables 1 and 2.

The Geometrician solutions obtained higher values than the best child solution; compared to the expert solutions, The Geometrician got lower values in correctness, but higher values for expertise and age.

Solution by	is correct?	is an expert?	age
The Geometrician	79.31%	96.55%	17.76
an expert	96.77%	61.29%	13.58
a child	67.74%	29.03%	13.61

Table 1: **Results of the evaluation to solutions to problem 4.**

Solution by	is correct?	is an expert?	age
The Geometrician	76.92%	96.15%	17.23
an expert	96.15%	80.77%	15.69
a child	46.15%	0%	12.92

Table 2: **Results of the evaluation to solutions to problem 5.**

Evaluation of Creativity

We consider The Geometrician creative if it is capable to solve novel problems. To evaluate the novelty of the problems, the unknown objects and unsatisfied relations of the problems solved by The Geometrician are compared with those of the problems for which the system knows its solution. Table 3 shows this comparison. As can be observed, The Geometrician is capable of solving novel problems.

Problem	unk. objects	uns. relations
Problem 1*	point B distance $d_1(A,B)$	$d_1 = d$
Problem 2*	point A	$A \in l$
Problem 3*	distance d	
Problem 4	point P distance $d_1(A,P)$ distance $e_1(B,P)$	$d_1 = d$ $e_1 = e$ $C \neq P$
Problem 5	point B distance $e(B,P)$	$B \in l$ $e = d$
Problem 6	point C distance $d_1(A,C)$ distance $d_2(B,C)$	$d_1 = d$ $d_2 = d$
Problem 7	point D' line $m(D,D')$	D' is the reflect of D through line l

Table 3: **Unknown objects and unsatisfied relations in the previous experience* and in the problems that The Geometrician solves.**

6 Conclusions

The Geometrician is capable to solve novel problems by means of the interaction between the Engagement & Reflection modes. This shows that the Engagement-Reflection computer model of creativity can be applied in problem solving.

References

[Anderson, 1981] J. R. Anderson. Tuning of search of the problem space for geometry proofs. In *Proc. of Int, Joint Conf. Artificial Intelligence (IJCAI)*, pages 165–170, Vancouver, 1981.

[Chou *et al.*, 1994] Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang. *Machine Proofs in Geometry. Automated Production of Readable Proofs for Geometry Theorems*, volume 6 of *Applied Mathematics*. 1994.

[Chou *et al.*, 1996] Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang. An introduction to geometry expert. In Michael A. McRobbie and John K. Slaney, editors, *CADE*, volume 1104 of *Lecture Notes in Computer Science*, pages 235–239. Springer, 1996.

[Coelho and Pereira, 1986] H. Coelho and L.M. Pereira. Automated reasoning in geometry with prolog. *J. of Automated Reasoning*, 2(4):329–390, 1986.

[Collins, 1975] George E. Collins. Hauptvortrag: Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In H. Barkhage, editor, *Automata Theory and Formal Languages*, volume 33 of *Lecture Notes in Computer Science*, pages 134–183. Springer, 1975.

[Gelernter, 1959] H. Gelernter. Realization of a geometry theorem proving machine. In *IFIP Congress*, pages 273–281, 1959.

[Gelernter, 1995] H. Gelernter. Realization of a geometry-theorem proving machine. In E. A. Feigenbaum and J. Feldman, editors, *Computers and Thought*, pages 134–152. MIT Press, 1995.

[Gilmore, 1970] P. C. Gilmore. An examination of the geometry theorem proving machine. *Artificial Intelligence*, 1(3):171–187, 1970.

[Koedinger and Anderson, 1990] K. R. Koedinger and J. R. Anderson. Abstract planning and perceptual chunks: Elements of expertise in geometry. *Cognitive Science*, 14:511–550, 1990.

[McCharen *et al.*, 1976] John D. McCharen, Ross A. Overbeek, and Larry Wos. Problems and experiments for and with automated theorem-proving programs. *IEEE Transactions on Computers*, 25(8):773–782, 1976.

[Nevins, 1975] Arthur J. Nevins. Plane geometry theorem proving using forward chaining. *Artif. Intell.*, 6(1):1–23, 1975.

[Pérez y Pérez and Sharples, 2001] Rafael Pérez y Pérez and Mike Sharples. Mexica: A computer model of a cognitive account of creative writing. *J. Expt. Theor. Artif. Intell.*, 13:119–139, 2001.

[Pérez y Pérez, 1999] Rafael Pérez y Pérez. *MEXICA: a Computer Model of Creativity in Writing*. PhD thesis, University of Sussex, England, 1999.

[Seidenberg, 1954] A. Seidenberg. A new decision method for elementary algebra. *Annals of Math.(2)*, 60:365–374, 1954.

[Tarski, 1951] A. Tarski. *A Decision Method for Elementary Algebra and Geometry*. Univ. of California Press, Berkeley, California, 1951.

[Wen-Tsün, 1978] Wu Wen-Tsün. On the decision problem and the mechanization of theorem in elementary geometry. *Scientia Sinica*, 2:159–172, 1978.

[Wen-Tsün, 1984] Wu Wen-Tsün. On the decision problem and the mechanization of theorem in elementary geometry. In Amer. Math. Soc., editor, *Automated Theorem Proving: After 25 years*, volume 29 of *Contemprorary Mathematics*, pages 213–234, 1984.