Towards next-step guidance in triangle construction problems

Vesna Marinković, Filip Marić

Faculty of Mathematics, University of Belgrade, Serbia

ThEdu'21 11.7.2021.

Solving of construction problems in geometry

- One of the most studied problems in mathematical education
- Task: to describe a construction of geometrical figure which satisfies given set of constraints
  " construct $\triangle ABC$ given α, β and $|AB|$"
- Constructions are procedures
- Some instances are unsolvable (e.g. angle trisection)

Constructions using straightedge and compass

- Tools: straightedge and compass
- Elementary steps:
  - construction of an arbitrary point
  - construction of a line through two given points
  - construction of a circle centered at given point passing through another point
  - construction of an intersection of two circles, two lines, or a line and a circle
- We usually use compound construction steps
- The main problem in solving:
  combinatorial explosion – huge search space

Motivating example

Problem: *Construct a triangle* ABC *given vertices* A *and* B *and a centroid* G

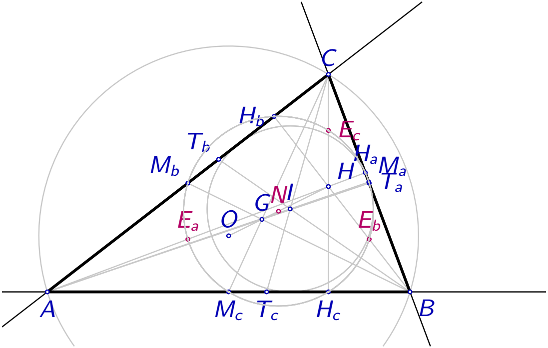
Solution: A, B, G

Overview of the goals

- systematization of the knowledge needed for solving one class of construction problems
- system for automated solving of location construction problems from the given corpus (ArgoTriCS, authors: V. Marinković, P. Janičić)
- adjusting the solver for usage in education is the subject of current work

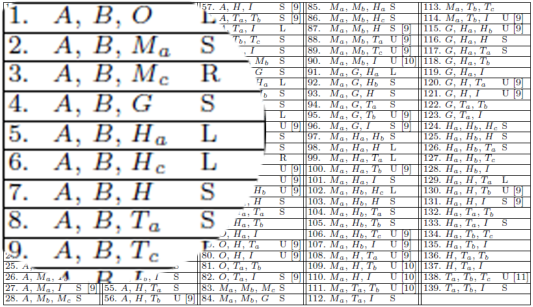
Corpora of construction problems

- Wernick's corpus (1982)
- Connelly's corpus (2009), an extension of Wernick's corpus
- Task: construct triangle ABC if locations of three significant points in the triangle are given



Corpora of construction problems (2)

Wernick's corpus: in total $\binom{16}{3} = 560$ instances, 139 non-trivial, significantly different problems; 3 redundant (R); 23 locus dependent (L); 74 solvable (S); 39 unsolvable (U)



1. A, B, O	L
2. A, B, M_a	S
3. A, B, M_c	R
4. A, B, G	S
5. A, B, H_a	L
6. A, B, H_c	L
7. A, B, H	S
8. A, B, T_a	S
9. A, B, T_c	…

		57. A, H, I S [9]	85. M_a, M_b, H_a S	113. M_a, T_b, T_c
		…A, T_a, T_b S [9]	86. M_a, M_b, H_c S	114. M_a, T_b, I U [9]
		…T_a, I L	87. M_a, M_b, H S [9]	115. G, H_a, H_b U [9]
		…T_b, T_c S	88. M_a, M_b, T_a U [9]	116. G, H_a, H S
		…I S	89. M_a, M_b, T_c U [9]	117. G, H_a, T_a S
		…M_b S	90. M_a, M_b, I U [10]	118. G, H_a, T_b
		…G S	91. M_a, G, H_a L	119. G, H_a, I
		…H_a L	92. M_a, G, H_b S	120. G, H, T_a U [9]
		…H_b S	93. M_a, G, H S	121. G, H, I U [9]
		… S	94. M_a, G, T_a S	122. G, T_a, T_b
		… L	95. M_a, G, T_b U [9]	123. G, T_a, I
		… U [9]	96. M_a, G, I S [9]	124. H_a, H_b, H_c S
		… S	97. M_a, H_a, H_b S	125. H_a, H_b, H S
		… S	98. M_a, H_a, H L	126. H_a, H_b, T_a S
		… R	99. M_a, H_a, T_a L	127. H_a, H_b, T_c
		… U [9]	100. M_a, H_a, T_b U [9]	128. H_a, H_b, I
		… U [9]	101. M_a, H_a, I S	129. H_a, H, T_a L
		…H_b U [9]	102. M_a, H_b, H_c L	130. H_a, H, T_b U [9]
		…H S	103. M_a, H_b, H S	131. H_a, H, I S [9]
		…T_a S	104. M_a, H_b, T_a S	132. H_a, T_a, T_b
		…H_a, T_b	105. M_a, H_b, T_b S	133. H_a, T_a, I S
		…H_a, I	106. M_a, H_b, T_c U [9]	134. H_a, T_b, T_c
		…O, H, T_a U [9]	107. M_a, H_b, I U [9]	135. H_a, T_b, I
24…		80. O, H, I U [9]	108. M_a, H, T_a U [9]	136. H, T_a, T_b
25. $A,$ …		81. O, T_a, T_b	109. M_a, H, T_b U [10]	137. H, T_a, I
26. $A, M_a,$ …	…I S	82. O, T_a, I S [9]	110. M_a, H, I U [10]	138. T_a, T_b, T_c U [11]
27. A, M_a, I S [9]	55. A, H, T_a S	83. M_a, M_b, M_c S	111. M_a, T_a, T_b U [10]	139. T_a, T_b, I S
28. A, M_b, M_c S	56. A, H, T_b U [9]	84. M_a, M_b, G S	112. M_a, T_a, I S	

Connelly's corpus: 580 new instances, 140 significantly different problems; 5 R; 19 L; 74 S; 42 U

Knowledge representation

Motivating example: *Construct the midpoint M_c of the segment AB, and then construct a point C such that it holds $\overrightarrow{M_cC}/\overrightarrow{M_cG} = 3$*
uses following knowledge:

- M_c is the midpoint of the segment AB (definition of the point M_c)
- G is the centroid of the triangle ABC (definition of the point G)
- it holds: $\overrightarrow{M_cG} = 1/3\overrightarrow{M_cC}$ (lemma)
- given points X and Y, one can construct the midpoint of the segment XY (primitive construction)
- given points X and Y, one can construct a point Z, such that it holds: $\overrightarrow{XZ}/\overrightarrow{XY} = m/n$, for integers m, n (primitive construction)

Definitions

- Instantiated definitions:
  - Centroid G is the intersection point of medians t_b and t_c
  - Points N_a, N_b, and N_c are intersection points of circumcircle with internal angle bisectors
  - Perpendicular bisectors of triangle m_a, m_b, and m_c are lines perpendicular to lines BC, AC, and AB and incident to side midpoints M_a, M_b, and M_c
- General definitions:
  - $\mathcal{C}(X, Y)$ is the circle centered at X passing through Y
  - $l(X, Y)$ is the line passing through points X and Y

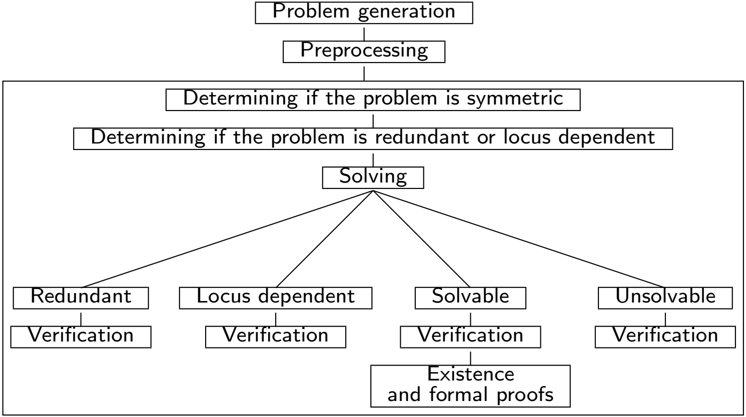
Lemmas

- Instantiated lemmas:
  - G belongs to median t_a
  - A and B belongs to the circle $\mathcal{C}(O, C)$
  - N_a, N_b, and N_c belong to bisectors of sides AB, BC, AC
  - $\overrightarrow{AG}/\overrightarrow{AM_a} = 2/3$
  - H, G, and O are collinear and it holds: $\overrightarrow{HG}/\overrightarrow{HO} = 2/3$
  - altitutes of $\triangle ABC$ are internal angle bisectors of $\triangle H_aH_bH_c$
- General lemmas:
  - the center of an arbitrary circle belongs to the perpendicular bisector of its arbitrary chord
  - $\overrightarrow{XY}/\overrightarrow{ZW} = r \implies \overrightarrow{ZW}/\overrightarrow{XY} = 1/r$

Primitive constructions

- Given in an uninstantiated form
- Accompanied by a set of NDG conditions and DET conditions
    1. Given points X and Y one can construct a line XY
    2. Given two distinct points X and Y one can construct a circle $\mathcal{C}(X, Y)$ centered at X which passes through Y
    3. Given a point X and a line p one can construct a line q which passes through X and which is perpendicular to p
    4. Given points X and Y one can construct perpendicular bisector of the segment $\overline{XY}$
    5. Given points X and Y one can construct a circle with diameter $\overline{XY}$
    6. Given points X, Z, and W, and a rational number r one can construct a point Y for which holds: $\overrightarrow{XY}/\overrightarrow{ZW} = r$

Algorithm



Problem generation
Preprocessing
Determining if the problem is symmetric
Determining if the problem is redundant or locus dependent
Solving
Redundant
Locus dependent
Solvable
Unsolvable
Verification
Verification
Verification
Verification
Existence
and formal proofs

Algorithm - solving phase

- search starts from the objects given and stops once all vertices are constructed or no more applicable primitive construction
- iterative procedure with forward chaining
- primitive constructions are applied in a waterfall manner
- objects that can be constructed are only relevant ones
- definitions and lemmas guide the search process
- early pruning of inapplicable primitive constructions

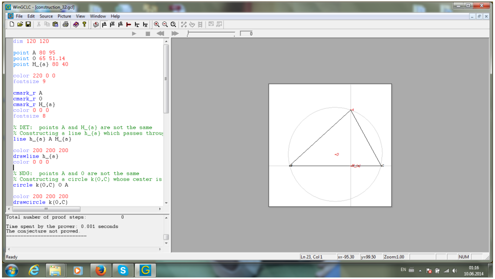
Construction in natural language form and in GCLC

Problem 32: Given points A, O, and H_a construct the triangle ABC.

Construction:

1. Construct the line h_a through points A and H_a;
2. Construct the circle $\mathcal{C}(O, C)$ with center at point O passing through point A;
3. Construct the perpendicular a to line h_a through point H_a;
4. Construct the intersection points C and B of circle $\mathcal{C}(O, C)$ and line a.

NDG conditions: line a and circle $\mathcal{C}(O, C)$ intersect; A and O are not the same.
Determination conditions: points A and H_a are not the same.



Different solutions

- Order in which background knowledge is given sometimes make the difference
- Sometimes for symmetric problems different, non-symmetric solutions are generated

Next step guidance

Work in progress

Aim

Help student once he gets stuck and does not know how to proceed with the construction

How to help:

- suggest the next construction step
- suggest an object (point, line, circle) that should be constructed
- suggest a lemma that the construction relies on

Target construction

- There can be many possible constructions for the same problem
- Example: A, B, H
  - A, B, H - solution 1
  - A, B, H - solution 2
  - A, B, H - solution 3
- Next step guidance assumes that we select one target construction

Partial construction

- The target construction should extend the partial construction given by the student
- Example: partial construction for A, B, H
  - A, B, H - partial construction

Select target from the list of all available constructions

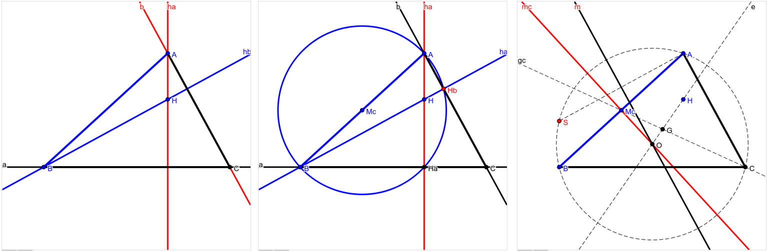
- How to choose the target construction that should be used for suggestions?
- ArgoTriCS can find several possible constructions (in advance) and they can all be compared to the partial construction
- The construction that is "the most adequate" for the partial construction is chosen for the target construction

Comparing constructions

- How to compare constructions and choose the one that is “the most adequate”?
  - the number of shared (*important*) objects should be maximized
  - the number of (*important*) objects from the partial construction that are not used in the target construction should be minimized
  - the number of construction steps that can immediately be applied should be maximized
  - the number of remaining construction steps should be minimized

Comparing constructions: example

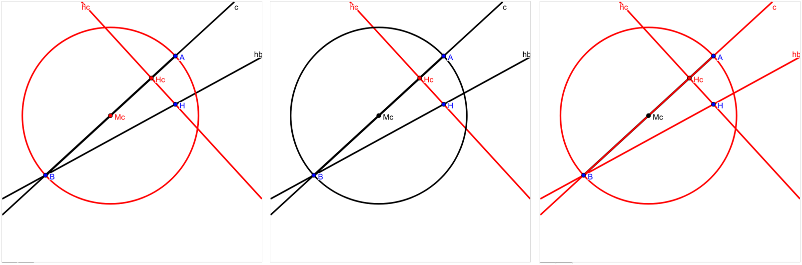
Example: A, B, H



- Comparing partial construction with three available solutions
- Blue - already constructed objects
- Red - objects that can be constructed next

Comparing constructions: example

Example: A, B, H



- Comparing partial construction with three available solutions
- Red - objects that would be unused

Comparing constructions: example

Example: A, B, H

	Shared objects	Unused objects	Available steps	Remaining steps
Construction 1	2	4	2	6
Construction 2	4	2	2	8
Construction 3	2	5	2	11

Construction 2 should be chosen as the target construction!

Restarting the solver

- Alternative approach: restart ArgoTriCS given all the objects constructed by the partial construction
- Modify the search heuristics so that it prioritizes construction steps that use more input objects

Target construction – resolving ambiguity

- What if there is no partial construction or if more than one target construction matches the partial construction?
- How to choose the target construction that should be suggested in that case?
- choose the simplest construction
  - constructions with the least construction steps
  - constructions that use simpler construction steps
  - constructions that use simpler (well-known) lemmas
  - a simplicity measure can combine all those criteria

Target construction – resolving ambiguity

If no partial construction is given

- A, B, H – choose construction 1
- A, B, T_a – choose construction 1 over construction 2 (it is longer, but uses simpler construction steps)

Suggestions

- Once the target construction is fixed, what should be suggested?
  - the next construction step
  - an important auxiliary object (point, line, circle)
  - an important lemma used in construction correctness proof

Suggesting the next construction step

- this strategy is simplest to implement
- suggest the first available step in the target construction that is not already present in the partial construction
- suggest an available step in the target construction that opens the most possibilities for the next step

Suggesting an important auxiliary object

- How to estimate the importance of an object?
  - prioritize "well-known" objects
    - A, B, C
    - I, O, G, H
    - M_a, M_b, M_c, H_a, H_b, H_c, T_a, T_b, T_c, P_a, P_b, P_c
    - perpendicular bisector, angle bisector, median, altitude
    - incircle, circumcirle
    - Euler line, Euler circle, ...
  - prioritize objects that are "close" to the the partial construction i.e. that require only a few construction steps
  - prioritize objects constructable using simple constrution steps, relying on simple lemmas
  - prioritize unusual objects, rarely used and occuring only in specific lemmas
  - prioritize objects shared by the target construction and many other available solutions

Suggesting an important auxiliary object

- A, B, H - solution 2
  Suggestion: “construct circle over segment AB“
- A, B, H - solution 3
  Suggestion: “construct O, then construct G“
- A, I, O - solution 1
  Suggestion: “construct N_b and N_c“
- A, Mb, H - solution 3
  Suggestion: “construct altitude midpoints H'_a, H'_c, then altitude feet H_a, H_c“
- A, Mb, I - solution 2
  Suggestion: “construct P_a, P_b, P_c“

Suggesting an important lemma

- suggest lemmas that are used in the construction, but are not well-known and contain more advanced geometric knowledge
- suggest non-trivial lemmas that are applied several times in the target construction

Suggesting an important lemma

- A, B, H - solution 1
  "altitudes are orthogonal to triangle sides ($h_a \perp a$, $h_b \perp b$)".
- A, B, H - solution 2
  "altitude feet are incident to circle over a triangle side ($H_a \in \mathcal{C}_{AB}$, $H_b \in \mathcal{C}_{AB}$)".
- A, M_b, T_a - solution 1
  "Triangle sides are symmetric wrt. the angle bisector ($sym_{s_a}(b, c)$)".
- A, M_b, T_a - solution 2
  "Altitude from A and line that connects A and circumcenter O are symmetric over angle bisector at A ($sym_{s_a}(h_a, OA)$)".

DG.js - a JavaScript library for dynamic visualization

- DG.js
- construction is specified as a linear program in JavaScript

Constructions are linear programs

A, *B*, *H* - solution 1

```
A = point(80, 95).color("blue").label("A");
B = point(20, 40).color("blue").label("B");
H = point(80, 72.73).color("blue").label("H");

AB = segment(A, B).color("red").width(2);

ha = line(A, H).color("magenta").label("ha");
hb = line(B, H).color("magenta").label("hb");
a = drop_perp(ha, B).color("red").label("a");
b = drop_perp(hb, A).color("red").label("b");
C = intersectLL(a, b).color("red").label("C");

AC = segment(A, C).color("red").width(2);
BC = segment(B, C).color("red").width(2);
```

Construction steps are defined as functions

```
function median(A, B) {
  c1 = circle(A, B); c2 = circle(B, A);
  [M1, M2] = intersectCC(c1, c2).both();
  return line(M1, M2);
}

function midpoint(A, B) {
  return intersectLL(line(A, B), median(A, B));
}
```

Predicates

- Predicates are used to select objects and enable branching constructions.

Example: Constraining random points

```
function drop_perpendicular(A, l) {
   B = l.randomPoint(B => B != A);
   c = circle(A, B);
   [X1, X2] = intersectCL(c, l);
   return median(X1, X2);
}
```

Example: choosing intersection points

the point within the Poincaré disc is always choosen

```
abs = circle(O, I); // absolute of the Poincare disc
X = intersectCC(l1, l2).select(P => abs.inDisc(P));
```

Checking construction correctness

- A precise construction specification enables checking correctness of a construction
- A preliminary work towards this goal has been made in Marinković's PhD thesis
- Link to algebraic provers
- Link to interactive theorem provers (e.g., Isabelle/HOL)

Conclusions

- ArgoTriCS – a system for automated triangle constructions
- Next-step guidance feature under way (hope to have it fully available by the end of this summer)
- DG.js – a lightweight JavaScript library for dynamic visualisation of geometric constructions
- Link to algebraic and interactive theorem provers