THE TEACHING OF MATHEMATICS
2015, Vol. XVIII, 1, pp. 29–44

# ON-LINE COMPENDIUM OF TRIANGLE CONSTRUCTION PROBLEMS WITH AUTOMATICALLY GENERATED SOLUTIONS

**Vesna Marinković**

**Abstract.** In this paper an on-line compendium of solutions to a class of triangle construction problems using straightedge and compass is presented. It consists of: informal description of constructions in natural-language form, formal description of constructions, illustrations of generated constructions, as well as step-by-step animations of generated constructions. This collection was generated completely automatically, using the system for automated solving of construction problems ArgoTriCS. To our knowledge this is the first automatically generated compendium of construction problems which offers solutions to almost all solvable problems from one large corpus of problems.

*MathEduc Subject Classification*: P55

*MSC Subject Classification*: 97P50

*Key words and phrases*: Triangle construction; automatically generated solutions.

## 1. Introduction

A *construction problem* is a problem in which one has to, given a declarative specification of a figure, provide a corresponding—possibly equivalent—procedural specification of the figure based on available construction steps. In the text that follows only constructions using straightedge and compass will be considered.

By a *straightedge-and-compass construction* we will mean a sequence of the following primitive (or elementary) steps:

– construct an arbitrary point (possibly distinct from some given points);
– construct (by *straightedge*) the line passing through two given distinct points;
– construct (by *compass*) the circle centered at some point passing through another point;
– construct an intersection (if it exists) of two circles, two lines, or a line and a circle.

Construction problems in geometry are one of the oldest and the most studied problems in mathematical education. They have been studied since the time of ancient Greece and have always been a subject of study of geometry courses taught during primary and secondary education. Up to today they still present an area

This work was partially supported by the Serbian Ministry of Education, Science and Technological Development under Grant 174021.

full of challenges even for experienced mathematicians. It is so because of the huge search space which makes solving this type of problems very hard, both for human mathematicians and a computer. This huge search space originates from the fact that primitive construction steps can be applied in a number of ways, exploding further along the construction. Additional problem represents the fact that some of construction problems, for instance, circle squaring, cube duplication, and angle trisection are proved unsolvable using an algebraic argument [30]. Usually, these unconstructibility results are not taught during the geometry courses since these proofs involve algebraic knowledge.

### 1.1. Triangle construction problems

*Triangle location construction problems* are construction problems in which the task is to construct a triangle $ABC$, using straightedge and compass, given positions of three characteristic points of the triangle. The problems of this type have been studied for centuries, while in 1982 William Wernick presented a list of triangle location problems where three characteristic points are selected from the list of the following 16 points [32]:

- $A$, $B$, $C$, $O$: three vertices and circumcenter;
- $M_a$, $M_b$, $M_c$, $G$: side midpoints and centroid;
- $H_a$, $H_b$, $H_c$, $H$: three feet of altitudes and orthocenter;
- $T_a$, $T_b$, $T_c$, $I$: three feet of the internal angles bisectors and incenter.

In total, there are 560 triples of the above points, but the list presented by Wernick consisted of only 139 significantly different non-trivial problems.[1] Problems from the Wernick's list were divided in four different classes:

**redundant problems**: if there is a point in the triple such that it is uniquely determined and constructible using the remaining two points, the problem is considered *redundant* and denoted by **R** (for instance, the problem $\{A, B, M_c\}$ is redundant since given points $A$ and $B$, the point $M_c$ is uniquely determined).

**locus dependent problems**: if there exists the required triangle $ABC$ only if given points meet some constraints, then the problem is considered *locus dependent* and denoted by **L** (for instance, the problem $\{A, B, O\}$ is considered locus dependent, since the point $O$ has to belong to the perpendicular bisector of the segment $AB$ for the triangle $ABC$ to exist).

**solvable problems**: if there is a construction of the required triangle $ABC$ starting with the given points, the problem is considered *solvable* and denoted by **S** (for instance, the problem $\{A, B, M_a\}$ is solvable).

**unsolvable problems**: if for some given points the required triangle $ABC$ exists, but it is not constructible, then the problem is considered *unsolvable* and denoted by **U** (for instance, the problem $\{T_a, T_b, T_c\}$ is unsolvable).

[1] For instance, the triple $\{A, B, C\}$ is considered trivial and, for instance, the problems $\{A, B, M_a\}$, $\{A, B, M_b\}$, $\{B, C, M_b\}$, $\{B, C, M_c\}$, $\{A, C, M_a\}$, and $\{A, C, M_c\}$ are considered to be symmetric (i.e., analogous).

In the original version of the list, the status of 41 problems was unknown. In the meanwhile, many of the problems with unresolved status were analysed [1,5,6,23,29,31,33] and today, statuses of all problems from Wernick's list are determined and they are given in Table 1. There are 74 **S** problems, 39 **U** problems, 3 **R** problems, and 23 **L** problems.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1. $A$, $B$, $O$ | L | 36. $A$, $M_b$, $T_c$ | S | 71. $O$, $G$, $H$ | R | 106. $M_a$, $H_b$, $T_c$ | U |
| 2. $A$, $B$, $M_a$ | S | 37. $A$, $M_b$, $I$ | S | 72. $O$, $G$, $T_a$ | U | 107. $M_a$, $H_b$, $I$ | U |
| 3. $A$, $B$, $M_c$ | R | 38. $A$, $G$, $H_a$ | L | 73. $O$, $G$, $I$ | U | 108. $M_a$, $H$, $T_a$ | S |
| 4. $A$, $B$, $G$ | S | 39. $A$, $G$, $H_b$ | S | 74. $O$, $H_a$, $H_b$ | U | 109. $M_a$, $H$, $T_b$ | U |
| 5. $A$, $B$, $H_a$ | L | 40. $A$, $G$, $H$ | S | 75. $O$, $H_a$, $H$ | S | 110. $M_a$, $H$, $I$ | U |
| 6. $A$, $B$, $H_c$ | L | 41. $A$, $G$, $T_a$ | S | 76. $O$, $H_a$, $T_a$ | S | 111. $M_a$, $T_a$, $T_b$ | U |
| 7. $A$, $B$, $H$ | S | 42. $A$, $G$, $T_b$ | U | 77. $O$, $H_a$, $T_b$ | U | 112. $M_a$, $T_a$, $I$ | S |
| 8. $A$, $B$, $T_a$ | S | 43. $A$, $G$, $I$ | S | 78. $O$, $H_a$, $I$ | U | 113. $M_a$, $T_b$, $T_c$ | U |
| 9. $A$, $B$, $T_c$ | L | 44. $A$, $H_a$, $H_b$ | S | 79. $O$, $H$, $T_a$ | U | 114. $M_a$, $T_b$, $I$ | U |
| 10. $A$, $B$, $I$ | S | 45. $A$, $H_a$, $H$ | L | 80. $O$, $H$, $I$ | U | 115. $G$, $H_a$, $H_b$ | U |
| 11. $A$, $O$, $M_a$ | S | 46. $A$, $H_a$, $T_a$ | L | 81. $O$, $T_a$, $T_b$ | U | 116. $G$, $H_a$, $H$ | S |
| 12. $A$, $O$, $M_b$ | L | 47. $A$, $H_a$, $T_b$ | S | 82. $O$, $T_a$, $I$ | S | 117. $G$, $H_a$, $T_a$ | S |
| 13. $A$, $O$, $G$ | S | 48. $A$, $H_a$, $I$ | S | 83. $M_a$, $M_b$, $M_c$ | S | 118. $G$, $H_a$, $T_b$ | U |
| 14. $A$, $O$, $H_a$ | S | 49. $A$, $H_b$, $H_c$ | S | 84. $M_a$, $M_b$, $G$ | S | 119. $G$, $H_a$, $I$ | S |
| 15. $A$, $O$, $H_b$ | S | 50. $A$, $H_b$, $H$ | L | 85. $M_a$, $M_b$, $H_a$ | S | 120. $G$, $H$, $T_a$ | U |
| 16. $A$, $O$, $H$ | S | 51. $A$, $H_b$, $T_a$ | S | 86. $M_a$, $M_b$, $H_c$ | S | 121. $G$, $H$, $I$ | U |
| 17. $A$, $O$, $T_a$ | S | 52. $A$, $H_b$, $T_b$ | L | 87. $M_a$, $M_b$, $H$ | S | 122. $G$, $T_a$, $T_b$ | U |
| 18. $A$, $O$, $T_b$ | S | 53. $A$, $H_b$, $T_c$ | S | 88. $M_a$, $M_b$, $T_a$ | U | 123. $G$, $T_a$, $I$ | U |
| 19. $A$, $O$, $I$ | S | 54. $A$, $H_b$, $I$ | S | 89. $M_a$, $M_b$, $T_c$ | U | 124. $H_a$, $H_b$, $H_c$ | S |
| 20. $A$, $M_a$, $M_b$ | S | 55. $A$, $H$, $T_a$ | S | 90. $M_a$, $M_b$, $I$ | U | 125. $H_a$, $H_b$, $H$ | S |
| 21. $A$, $M_a$, $G$ | R | 56. $A$, $H$, $T_b$ | U | 91. $M_a$, $G$, $H_a$ | L | 126. $H_a$, $H_b$, $T_a$ | S |
| 22. $A$, $M_a$, $H_a$ | L | 57. $A$, $H$, $I$ | S | 92. $M_a$, $G$, $H_b$ | S | 127. $H_a$, $H_b$, $T_c$ | U |
| 23. $A$, $M_a$, $H_b$ | S | 58. $A$, $T_a$, $T_b$ | S | 93. $M_a$, $G$, $H$ | S | 128. $H_a$, $H_b$, $I$ | U |
| 24. $A$, $M_a$, $H$ | S | 59. $A$, $T_a$, $I$ | L | 94. $M_a$, $G$, $T_a$ | S | 129. $H_a$, $H$, $T_a$ | L |
| 25. $A$, $M_a$, $T_a$ | S | 60. $A$, $T_b$, $T_c$ | S | 95. $M_a$, $G$, $T_b$ | U | 130. $H_a$, $H$, $T_b$ | U |
| 26. $A$, $M_a$, $T_b$ | U | 61. $A$, $T_b$, $I$ | S | 96. $M_a$, $G$, $I$ | S | 131. $H_a$, $H$, $I$ | S |
| 27. $A$, $M_a$, $I$ | S | 62. $O$, $M_a$, $M_b$ | S | 97. $M_a$, $H_a$, $H_b$ | S | 132. $H_a$, $T_a$, $T_b$ | U |
| 28. $A$, $M_b$, $M_c$ | S | 63. $O$, $M_a$, $G$ | S | 98. $M_a$, $H_a$, $H$ | L | 133. $H_a$, $T_a$, $I$ | S |
| 29. $A$, $M_b$, $G$ | S | 64. $O$, $M_a$, $H_a$ | L | 99. $M_a$, $H_a$, $T_a$ | L | 134. $H_a$, $T_b$, $T_c$ | U |
| 30. $A$, $M_b$, $H_a$ | L | 65. $O$, $M_a$, $H_b$ | S | 100. $M_a$, $H_a$, $T_b$ | U | 135. $H_a$, $T_b$, $I$ | U |
| 31. $A$, $M_b$, $H_b$ | L | 66. $O$, $M_a$, $H$ | S | 101. $M_a$, $H_a$, $I$ | S | 136. $H$, $T_a$, $T_b$ | U |
| 32. $A$, $M_b$, $H_c$ | L | 67. $O$, $M_a$, $T_a$ | L | 102. $M_a$, $H_b$, $H_c$ | L | 137. $H$, $T_a$, $I$ | U |
| 33. $A$, $M_b$, $H$ | S | 68. $O$, $M_a$, $T_b$ | U | 103. $M_a$, $H_b$, $H$ | S | 138. $T_a$, $T_b$, $T_c$ | U |
| 34. $A$, $M_b$, $T_a$ | S | 69. $O$, $M_a$, $I$ | S | 104. $M_a$, $H_b$, $T_a$ | S | 139. $T_a$, $T_b$, $I$ | S |
| 35. $A$, $M_b$, $T_b$ | L | 70. $O$, $G$, $H_a$ | S | 105. $M_a$, $H_b$, $T_b$ | S | | |

Table 1. The final status of all Wernick's problems

Harold Connelly considered an extended Wernick's list [4], involving four additional points:

– $E_a$, $E_b$, $E_c$: three Euler points;

– $N$: center of the nine-point circle.

There are 140 new significantly different problems. The current status of the problems from Connelly's list is given in Table 2: there are 73 **S** problems, 11 **U** problems, 5 **R** problems, 19 **L** problems, as well as 32 problems with unknown status.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1. $A$, $B$, $E_a$ | S | 36. $A$, $M_a$, $N$ | S | 71. $E_a$, $H$, $T_b$ | U | 106. $E_a$, $M_b$, $T_c$ | |
| 2. $A$, $B$, $E_c$ | S | 37. $A$, $M_b$, $N$ | S | 72. $E_a$, $H_a$, $H_b$ | S | 107. $E_a$, $N$, $O$ | S |
| 3. $A$, $B$, $N$ | S | 38. $A$, $N$, $O$ | S | 73. $E_a$, $H_a$, $I$ | S | 108. $E_a$, $N$, $T_a$ | S |
| 4. $A$, $E_a$, $E_b$ | S | 39. $A$, $N$, $T_a$ | | 74. $E_a$, $H_a$, $M_a$ | L | 109. $E_a$, $N$, $T_b$ | |
| 5. $A$, $E_a$, $G$ | S | 40. $A$, $N$, $T_b$ | | 75. $E_a$, $H_a$, $M_b$ | S | 110. $E_a$, $O$, $T_a$ | |
| 6. $A$, $E_a$, $H$ | R | 41. $E_a$, $E_b$, $E_c$ | S | 76. $E_a$, $H_a$, $N$ | L | 111. $E_a$, $O$, $T_b$ | |
| 7. $A$, $E_a$, $H_a$ | L | 42. $E_a$, $E_b$, $G$ | S | 77. $E_a$, $H_a$, $O$ | S | 112. $E_a$, $T_a$, $T_b$ | |
| 8. $A$, $E_a$, $H_b$ | L | 43. $E_a$, $E_b$, $H$ | S | 78. $E_a$, $H_a$, $T_a$ | L | 113. $E_a$, $T_b$, $T_c$ | |
| 9. $A$, $E_a$, $I$ | S | 44. $E_a$, $E_b$, $H_a$ | S | 79. $E_a$, $H_a$, $T_b$ | | 114. $G$, $H$, $N$ | R |
| 10. $A$, $E_a$, $M_a$ | S | 45. $E_a$, $E_b$, $H_c$ | S | 80. $E_a$, $H_b$, $H_c$ | L | 115. $G$, $H_a$, $N$ | S |
| 11. $A$, $E_a$, $M_b$ | S | 46. $E_a$, $E_b$, $I$ | U | 81. $E_a$, $H_b$, $I$ | | 116. $G$, $I$, $N$ | U |
| 12. $A$, $E_a$, $N$ | S | 47. $E_a$, $E_b$, $M_a$ | L | 82. $E_a$, $H_b$, $M_a$ | L | 117. $G$, $M_a$, $N$ | S |
| 13. $A$, $E_a$, $O$ | S | 48. $E_a$, $E_b$, $M_c$ | S | 83. $E_a$, $H_b$, $M_b$ | S | 118. $G$, $N$, $O$ | R |
| 14. $A$, $E_a$, $T_a$ | S | 49. $E_a$, $E_b$, $N$ | L | 84. $E_a$, $H_b$, $M_c$ | S | 119. $G$, $N$, $T_a$ | U |
| 15. $A$, $E_a$, $T_b$ | U | 50. $E_a$, $E_b$, $O$ | S | 85. $E_a$, $H_b$, $N$ | L | 120. $H$, $H_a$, $N$ | S |
| 16. $A$, $E_b$, $E_c$ | S | 51. $E_a$, $E_b$, $T_a$ | | 86. $E_a$, $H_b$, $O$ | S | 121. $H$, $I$, $N$ | U |
| 17. $A$, $E_b$, $G$ | S | 52. $E_a$, $E_b$, $T_c$ | U | 87. $E_a$, $H_b$, $T_a$ | | 122. $H$, $M_a$, $N$ | S |
| 18. $A$, $E_b$, $H$ | S | 53. $E_a$, $G$, $H$ | S | 88. $E_a$, $H_b$, $T_b$ | U | 123. $N$, $N$, $O$ | R |
| 19. $A$, $E_b$, $H_a$ | S | 54. $E_a$, $G$, $H_a$ | S | 89. $E_a$, $H_b$, $T_c$ | | 124. $N$, $N$, $T_a$ | U |
| 20. $A$, $E_b$, $H_b$ | L | 55. $E_a$, $G$, $H_b$ | S | 90. $E_a$, $I$, $M_a$ | S | 125. $H_a$, $H_b$, $N$ | L |
| 21. $A$, $E_b$, $H_c$ | S | $E_a$, $G$, $I$ | | 91. $E_a$, $I$, $M_b$ | | 126. $H_a$, $I$, $N$ | S |
| 22. $A$, $E_b$, $I$ | | 57. $E_a$, $G$, $M_a$ | S | 92. $E_a$, $I$, $N$ | S | 127. $H_a$, $M_a$, $N$ | L |
| 23. $A$, $E_b$, $M_a$ | S | 58. $E_a$, $G$, $M_b$ | S | 93. $E_a$, $I$, $O$ | | 128. $H_a$, $M_b$, $N$ | L |
| 24. $A$, $E_b$, $M_b$ | S | 59. $E_a$, $G$, $N$ | S | 94. $E_a$, $I$, $T_a$ | | 129. $H_a$, $N$, $O$ | S |
| 25. $A$, $E_b$, $M_c$ | S | 60. $E_a$, $G$, $O$ | S | 95. $E_a$, $I$, $T_b$ | | 130. $H_a$, $N$, $T_a$ | S |
| 26. $A$, $E_b$, $N$ | S | 61. $E_a$, $G$, $T_a$ | | 96. $E_a$, $M_a$, $M_b$ | L | 131. $H_a$, $N$, $T_b$ | |
| 27. $A$, $E_b$, $O$ | S | 62. $E_a$, $G$, $T_b$ | | 97. $E_a$, $M_a$, $N$ | R | 132. $I$, $M_a$, $N$ | S |
| 28. $A$, $E_b$, $T_a$ | | 63. $E_a$, $H$, $H_a$ | L | 98. $E_a$, $M_a$, $O$ | S | 133. $I$, $N$, $O$ | U |
| 29. $A$, $E_b$, $T_b$ | | 64. $E_a$, $H$, $H_b$ | L | 99. $E_a$, $M_a$, $T_a$ | S | 134. $I$, $N$, $T_a$ | |
| 30. $A$, $E_b$, $T_c$ | | 65. $E_a$, $H$, $I$ | S | 100. $E_a$, $M_a$, $T_b$ | | 135. $M_a$, $M_b$, $N$ | L |
| 31. $A$, $G$, $N$ | S | 66. $E_a$, $H$, $M_a$ | S | 101. $E_a$, $M_b$, $M_c$ | S | 136. $M_a$, $N$, $O$ | S |
| 32. $A$, $N$, $N$ | S | 67. $E_a$, $H$, $M_b$ | S | 102. $E_a$, $M_b$, $N$ | L | 137. $M_a$, $N$, $T_a$ | S |
| 33. $A$, $H_a$, $N$ | S | 68. $E_a$, $H$, $N$ | S | 103. $E_a$, $M_b$, $O$ | S | 138. $M_a$, $N$, $T_b$ | |
| 34. $A$, $H_b$, $N$ | S | 69. $E_a$, $H$, $O$ | S | 104. $E_a$, $M_b$, $T_a$ | | 139. $N$, $O$, $T_a$ | U |
| 35. $A$, $I$, $N$ | | 70. $E_a$, $H$, $T_a$ | S | 105. $E_a$, $M_b$, $T_b$ | | 140. $N$, $T_a$, $T_b$ | |

Table 2. Status of the problems from Connelly's list

Triangle location construction problems, like the ones from Wernick's and Connelly's corpora, are present in many (if not all) geometry courses taught during primary and secondary school. Also, since solving construction problems requests paradigmatic reasoning, they are often found in mathematical competitions.

### 1.2. Software for mathematical education

In recent years, development of new technologies has allowed new insights into the solving construction problems. With a help of computer tools, it is easier to work on these problems and to understand their solutions. Mathematical software have been widely used in education since early 1990's, primarily through dynamic geometry tools like Cabri, Sketchpad, Cinderella and GeoGebra [11,12,16,25,26]. They allow teachers and students to carry out geometric constructions in accurate way in a dynamic graphical environment. Also, in this way a user can explore geometric relationships of the objects dynamically and discover relationships that hold between some of the objects, which could be helpful for solving the problem. Teachers find it useful to spend some extra time to produce precise descriptions of constructions followed by the corresponding illustrations, since it could lead to better understanding of the construction problems and their solutions. The use of mathematical software tools have not only improved the relevance of constructions in teaching geometry, but it has also increased the number of applications of geometry to other fields. However, in most cases, the use of mathematical software was restricted to visualization and experimentation, without going into deeper issues of mathematical proving and solving processes. Only recently, advanced automated techniques like methods for automated theorem proving (that were earlier available only within academic tools) have been introduced in widely used educational tools like GeoGebra [2].

We believe that computer-based methods are now in mature enough stage to provide a substantial steps forward in mathematical educations. There are automated theorem provers capable of proving highly complex geometry theorems and solvers capable of solving highly complex construction problems. Such computer-based systems can, further, help systematic coverage of large portions of school geometry. However, none of geometric construction tools such as Cabri, Sketchpad, Cinderella and GeoGebra offer a system for automated solving of construction problems or a systematic set of solved construction problems, which could be useful for students.

In this paper we present on-line compendiums of automatically generated solutions to construction problems. They consist of hundreds of interesting problems from Wernick's and Connelly's corpora. For solving these construction problems our system for automated solving of construction problems ArgoTriCS was used [20]. For each problem, a construction in natural-language form, a formal description of the construction in GCLC language, an illustration of the construction, as well as a step-by-step animation of the construction are generated. The compiled on-line compendium contains many interesting problems which should be useful in education, in the teaching process, for million of students solving this type of problems all over the world, especially in situations when a student does not have his/her teacher nearby to help with a specific problem. The generated step-by-step constructions could guide the students during the search for a construction.

*Overview of the paper.* In Section 2, a brief overview of the ArgoTriCS system is given, with some examples of generated constructions. In Section 3, a description of automatically generated step-by-step constructions, accompanied by animated figures is given. In Section 4, the sets of problems that the compendiums consist of are described and the generated compendiums, with a couple of examples are presented. In Section 5, plans for the future work are described and final conclusions are drawn.

## 2. Overview of the system ArgoTriCS

In this section we give a brief overview of the system ArgoTriCS [20,21,22]. It is a system that, given some background geometrical knowledge, solves automatically a construction problem. Its main features include:

– detection of relevant geometry knowledge and its division into the sets of definitions, lemmas and primitive constructions;[2]
– guided search which restricts construction of objects only to those objects that could be relevant for a construction;
– appropriate handling of redundant and locus dependent problems.

Solving of the problems from one given corpus consists of preprocessing phase, used for gathering all relevant instantiated knowledge available, and the solving phase. The solving phase consists of firstly testing if the problem is symmetric to some other already solved problem, then testing if the problem is redundant, otherwise, testing if the problem is locus dependent, and finally, if neither of these two is true – a generation of construction. If the problem is solved, the obtained construction trace is simplified by keeping only relevant construction steps in the construction [20].

Each solution of a construction problem, automatically generated by the system ArgoTriCS, consists of:

– informal description of construction in natural language form;
– formal specification of construction using the GCLC language [13,14];
– corresponding illustration;
– step-by-step animation of the generated figure, showed in synchronization with step-by-step textual description of the construction;
– *non-degeneracy conditions* (NDG conditions) which guarantee that the solution exists and *determination conditions* (DET conditions) which state when the solution is uniquely determined;
– proofs of correctness of generated construction.

A construction in natural language form is given in English language, in LaTeX format and it consists of the list of construction steps. Each construction step is

[2]The detected knowledge consists of around 90 instantiated definitions and 2 general definitions, around 120 instantiated lemmas and 20 general lemmas, as well as around 20 primitive constructions.

accompanied by the set of non-degeneracy conditions and the set of determination conditions (if they are present).

Construction in GCLC language represents a way to give the construction in more formal way, and to generate a corresponding illustration of the construction. For this purpose the GCLC tool is used [13,14,15].

Apart from the construction, a specification of the input to the provers available within the GCLC tool and OpenGeoProver [19] can be automatically generated. In this way, for problems where automated theorem provers succeed in proving these conjectures, a correctness of generated construction is automatically obtained. Proofs of correctness will not be discussed in the further text and they will not be a part of the generated compendiums.

Step-by-step animations of constructions are a new feature of the system, first presented in this paper (as well as the compendium itself). Step-by-step animations are described in more details in the next section.

The following example shows an automatically generated construction in natural language form for Problem 47 : $\{A, M_a, T_a\}$ (along with non-degenerate conditions and determination conditions).

EXAMPLE 1. Given points $A$, $M_a$, and $T_a$, construct the triangle $ABC$.

1. *Using the point $A$ and the point $T_a$, construct a line $s_a$ (rule W02);*
   % DET: points $A$ and $T_a$ are not the same;
2. *Using the point $M_a$ and the point $T_a$, construct a line $a$ (rule W02);*
   % DET: points $M_a$ and $T_a$ are not the same;
3. *Using the point $M_a$ and the line $a$, construct a line $m_a$ (rule W10b);*
4. *Using the line $m_a$ and the line $s_a$, construct a point $N_a$ (rule W03);*
   % NDG: lines $m_a$ and $s_a$ are not parallel
   % DET: lines $m_a$ and $s_a$ are not the same;
5. *Using the point $A$ and the point $N_a$, construct a line $m(AN_a)$ (rule W14);*
   % DET: points $A$ and $N_a$ are not the same;
6. *Using the line $m(AN_a)$ and the line $m_a$, construct a point $O$ (rule W03);*
   % NDG: lines $m(AN_a)$ and $m_a$ are not parallel
   % DET: lines $m(AN_a)$ and $m_a$ are not the same;
7. *Using the point $A$ and the point $O$, construct a circle $k(O, C)$ (rule W06);*
   % NDG: points $A$ and $O$ are not the same;
8. *Using the circle $k(O, C)$ and the line $a$, construct a point $C$ and a point $B$ (rule W04).*
   % NDG: line $a$ and circle $k(O, C)$ intersect.

*Non-degenerate conditions: line $a$ and circle $k(O, C)$ intersect; points $A$ and $O$ are not the same; lines $m(AN_a)$ and $m_a$ are not parallel; lines $m_a$ and $s_a$ are not parallel.*

*Determination conditions: lines $m(AN_a)$ and $m_a$ are not the same; points $A$ and $N_a$ are not the same; lines $m_a$ and $s_a$ are not the same; points $M_a$ and $T_a$ are not the same; points $A$ and $T_a$ are not the same.*

Formal specification of generated construction, which follows, is generated automatically using the ArgoTriCS tool, in geometrical language GCLC.

```
point A 80 95
point M_a 65 40
point T_a 70.86 40

cmark_t A
cmark_b M_a
cmark_b T_a

% DET: points A and T_a are not the same
% Constructing a line s_a which passes through point A and point T_a
line s_a A T_a
drawline s_a

% DET: points M_a and T_a are not the same
% Constructing a line a which passes through point M_a and point T_a
line a M_a T_a
drawline a

% Constructing a line m_a which is perpendicular to line a
% and which passes through point M_a
perp m_a M_a a
drawline m_a

% NDG: lines m_a and s_a are not parallel
% DET: lines m_a and s_a are not the same
% Constructing a point N_a which belongs to line m_a and line s_a
intersec N_a m_a s_a
cmark_b N_a

% DET: points A and N_a are not the same
% Constructing bisector m(AN_a) of the segment AN_a
med m(AN_a) A N_a
drawline m(AN_a)
drawsegment A N_a

% NDG: lines m(AN_a) and m_a are not parallel
% DET: lines m(AN_a) and m_a are not the same
% Constructing a point O which belongs to line m(AN_a) and line m_a
intersec O m(AN_a) m_a
cmark_t O

% NDG: points A and O are not the same
% Constructing a circle k(O,C) whose center is at point O
% and which passes through point A
circle k(O,C) O A
drawcircle k(O,C)

% NDG: line a and circle k(O,C) intersect
% Constructing points C and B which are in intersection of k(O,C) and a
intersec2 C B k(O,C) a
cmark_b C
cmark_b B

drawsegment A B
drawsegment A C
drawsegment B C
```

The corresponding automatically generated illustration is given in Figure 1. The points given by the problem setting are given in different color than the points introduced during the construction.

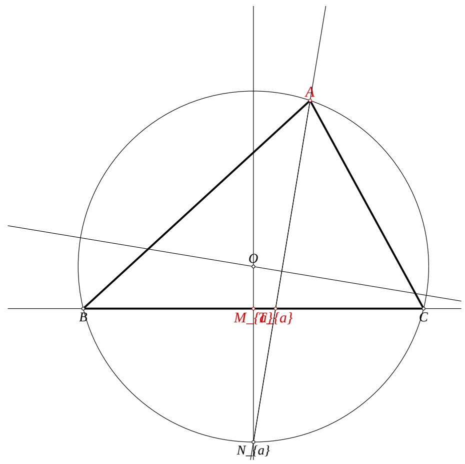



Fig. 1. Illustration for Problem 47

## 3. Animated step-by-step constructions

In order to make the generated constructions more comprehensible to students a step-by-step illustrations of constructions are produced. They are important since in this way students do not only see the final construction and the illustration, but can also follow the construction step-by-step.

These animated step-by-step illustrations are generated using the GCLC tool. As we already mentioned, ArgoTriCS can automatically export a formal description of the generated construction in GCLC language. This formal specification can be exported into a sequence of *bmp* figures which can be further used for generation of the animated figure in *gif* format. One such animated figure is illustrated in Figure 2.

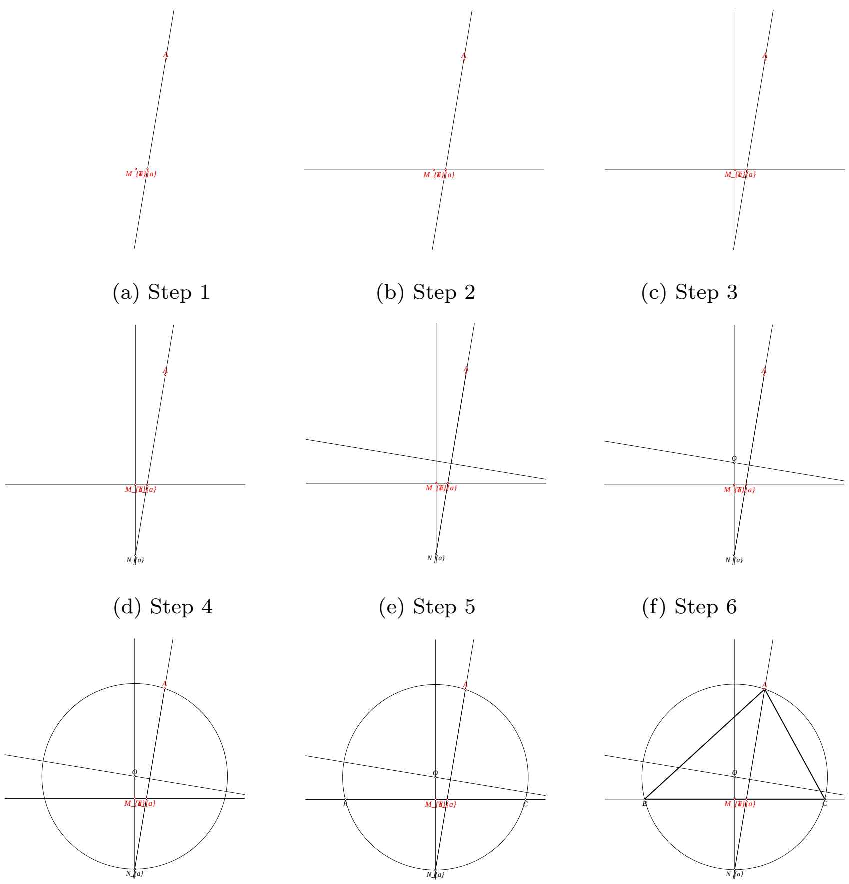
Beside the illustrations, a step-by-step textual descriptions of the constructions are provided. They are generated in HTML using jQuery, by showing an item by item of the list of construction steps (given in natural language form), in synchronization with the animation.

An example of the page showing the first five construction steps, followed by the partially generated illustration is shown in Figure 3.

## 4. On-line compendium

The solutions to construction problems obtained automatically by the system ArgoTriCS are organized within compendiums of construction problems, while the solving system ArgoTriCS is also publicly available. Two compendiums have been generated – one for problems belonging to Wernick's corpus and the other for the problems from Connelly's corpus.

These compendiums can be beneficial both to students and teachers. Teachers could enrich their lectures by showing these step-by-step generated solutions of



(a) Step 1 (b) Step 2 (c) Step 3

(d) Step 4 (e) Step 5 (f) Step 6

(g) Step 7 (h) Step 8 (i) Step 9

Fig. 2. Step-by-step animation of generated figure

construction problems. For students, this would present a big corpus of solutions to many interesting problems, containing many different ideas, and since it is put on-line it is highly accessible. This present a sort of encyclopedia knowledge, available to everyone who needs help with solving this kind of problems.

### 4.1. Set of problems

In Wernick's list, the problems $\{A, B, G\}$ and $\{A, C, G\}$ are considered analogous and therefore the second one is not given in the list (see Table 1). In contrast to this, in the method used by ArgoTriCS the centroid is not defined as an intersection point of three medians of the triangle, but as an intersection point of two medians, and the property that the third one passes through the centroid is considered a lemma. So, in contrast to Wernick's list, we will consider the problems $\{A, B, G\}$ and $\{A, C, G\}$ distinct. According to this, the set of problems that have

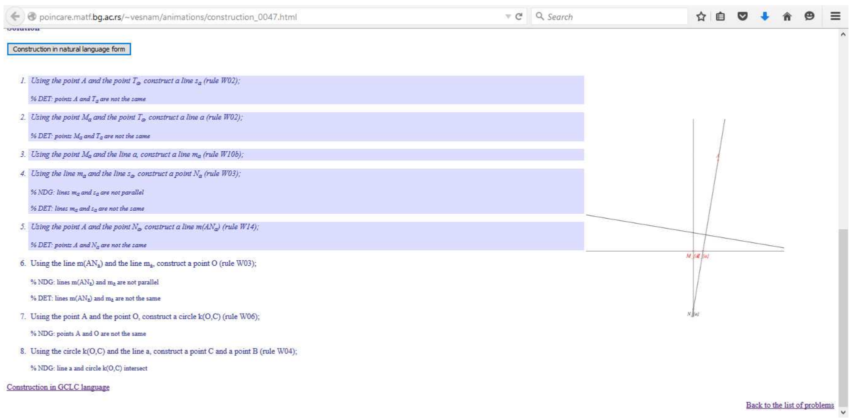



Fig. 3. Automatically generated HTML page showing the partial construction for Problem 47 (consisting of first five construction steps)

been solved using the ArgoTriCS are the problems from extended Wernick's and Connelly's corpora.

The system ArgoTriCS is able to solve almost all solvable problems from Wernick's and Connelly's list: 66 out of 74 solvable problems from Wernick's list and 62 out of 73 solvable problems from Connelly's list [20,21,22]. The system also successfully detects all redundant and locus dependent problems, as well as problems symmetric to another ones.

Considering the extended versions of the corpora, ArgoTriCS is used for solving all 560 triples of points from Wernick's list and currently it identified 268 of them solvable, 93 locus dependent, 7 redundant, while currently it cannot solve 192 triples. Out of these 192 problems, 166 were proved to be unsolvable [28], while for the remaining ones ArgoTriCS requires some additional knowledge to be added. ArgoTriCS is also used for solving 580 triples of points from Connelly's corpus and currently it can solve 223 triples of points, it identified 84 as locus dependent, 9 of them as redundant, while 264 of them the system can not solve using the current background knowledge.

The times needed for solving problems presented in the compendiums span from a couple of milliseconds to more than an hour. The longest generated construction consists of 19 construction steps and it is reached for problem $\{E_a, I, M_a\}$ from Connelly's corpus.

### 4.2. Description of the compendium

There are 560 problems in Wernick's corpus and 580 problems in Connelly's corpus. Overall, the compendiums contain 491 solvable problems with automatically generated solutions, 177 locus dependent problems and 16 redundant problems. The compendiums are available from `http://poincare.matf.bg.ac.rs/~vesnap/animations/compendiums.html`. They are organized as follows: firstly, an HTML page which contains some basic informations about these two corpora is shown. It contains a short description of each corpus with the illustration of the points used in these two corpora, links to two generated compendiums and accompanying bibliography (Figures 4 and 5).

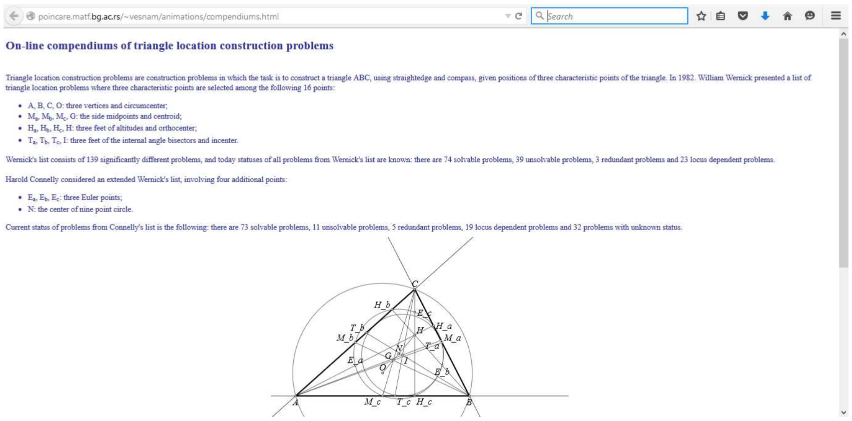



Fig. 4. Automatically generated HTML page containing the description of the compendiums (1)

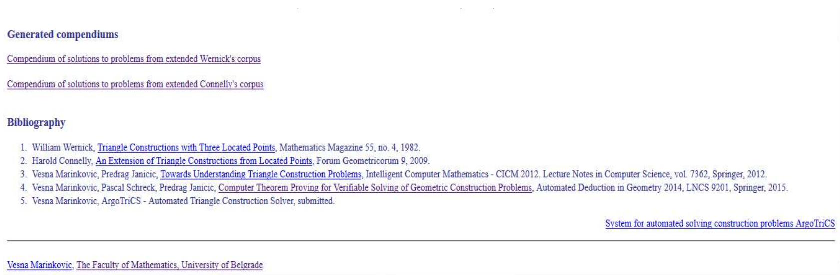



Fig. 5. Automatically generated HTML page containing the description of the compendiums (2)

Each compendium is presented by a table containing settings of all problems from that corpus, along with a status of that problem (Figure 6). Each table entry represents a link and by clicking the link a page containing a detailed description of the solution of that problem can be accessed.

The page with a solution contains:

- the problem description given in natural language form;
- the status of the problem determined by the system ArgoTriCS – it can be solvable, redundant, locus dependent or its status could not be determined by the system (and in this case the actual status of the problem is given);
- the illustration of the construction;
- the button which enables displaying of the construction in natural language form and a corresponding animated figure, displayed step by step;

poincare.matf.bg.ac.rs/~vesnam/animations/compendium_wernick.html

**The list of Wernick's problems**

Statuses of all problems: *S (s)* denotes that the problem is solvable and its solution is given, *S (ns)* denotes that the problem is proved solvable but its solution is not given, *L* that the problem is locus dependent, *R* that the problem is redundant, while *U* denotes that the problem is unsolvable.

| | | | | |
|---|---|---|---|---|
| 1. A, B, C *S (s)* | 2. A, B, O *L* | 3. A, B, $M_a$ *S (s)* | 4. A, B, $M_b$ *S (s)* | 5. A, B, $M_c$ *R* |
| 6. A, B, G *S (s)* | 7. A, B, $H_a$ *L* | 8. A, B, $H_b$ *L* | 9. A, B, $H_c$ *L* | 10. A, B, H *S (s)* |
| 11. A, B, $T_a$ *S (s)* | 12. A, B, $T_b$ *S (s)* | 13. A, B, $T_c$ *L* | 14. A, B, I *S (s)* | 15. A, C, O *L* |
| 16. A, C, $M_a$ *S (s)* | 17. A, C, $M_b$ *R* | 18. A, C, $M_c$ *S (s)* | 19. A, C, G *S (s)* | 20. A, C, $H_a$ *L* |
| 21. A, C, $H_b$ *L* | 22. A, C, $H_c$ *L* | 23. A, C, H *S (s)* | 24. A, C, $T_a$ *S (s)* | 25. A, C, $T_b$ *L* |
| 26. A, C, $T_c$ *S (s)* | 27. A, C, I *S (s)* | 28. A, O, $M_a$ *S (s)* | 29. A, O, $M_b$ *L* | 30. A, O, $M_c$ *L* |
| 31. A, O, G *S (s)* | 32. A, O, $H_a$ *S (s)* | 33. A, O, $H_b$ *S (s)* | 34. A, O, $H_c$ *S (s)* | 35. A, O, H *S (s)* |
| 36. A, O, $T_a$ *S (s)* | 37. A, O, $T_b$ *S (s)* | 38. A, O, $T_c$ *S (s)* | 39. A, O, I *S (s)* | 40. A, $M_a$, $M_b$ *S (s)* |
| 41. A, $M_a$, $M_c$ *S (s)* | 42. A, $M_a$, G *R* | 43. A, $M_a$, $H_a$ *L* | 44. A, $M_a$, $H_b$ *S (s)* | 45. A, $M_a$, $H_c$ *S (s)* |
| 46. A, $M_a$, H *S (s)* | 47. A, $M_a$, $T_a$ *S (s)* | 48. A, $M_a$, $T_b$ *U* | 49. A, $M_a$, $T_c$ *U* | 50. A, $M_a$, I *S (s)* |
| 51. A, $M_b$, $M_c$ *S (s)* | 52. A, $M_b$, G *S (s)* | 53. A, $M_b$, $H_a$ *L* | 54. A, $M_b$, $H_b$ *L* | 55. A, $M_b$, $H_c$ *L* |
| 56. A, $M_b$, H *S (s)* | 57. A, $M_b$, $T_a$ *S (s)* | 58. A, $M_b$, $T_b$ *L* | 59. A, $M_b$, $T_c$ *S (s)* | 60. A, $M_b$, I *S (s)* |
| 61. A, $M_c$, G *S (s)* | 62. A, $M_c$, $H_a$ *L* | 63. A, $M_c$, $H_b$ *L* | 64. A, $M_c$, $H_c$ *L* | 65. A, $M_c$, H *S (s)* |
| 66. A, $M_c$, $T_a$ *S (s)* | 67. A, $M_c$, $T_b$ *S (s)* | 68. A, $M_c$, $T_c$ *L* | 69. A, $M_c$, I *S (s)* | 70. A, G, $H_a$ *L* |
| 71. A, G, $H_b$ *S (s)* | 72. A, G, $H_c$ *S (s)* | 73. A, G, H *S (s)* | 74. A, G, $T_a$ *S (s)* | 75. A, G, $T_b$ *U* |
| 76. A, G, $T_c$ *U* | 77. A, G, I *S (s)* | 78. A, $H_a$, $H_b$ *S (s)* | 79. A, $H_a$, $H_c$ *S (s)* | 80. A, $H_a$, H *L* |
| 81. A, $H_a$, $T_a$ *L* | 82. A, $H_a$, $T_b$ *S (s)* | 83. A, $H_a$, $T_c$ *S (s)* | 84. A, $H_a$, I *S (s)* | 85. A, $H_b$, $H_c$ *S (s)* |

Fig. 6. Automatically generated HTML page containing the table with settings of problems from Wernick's corpus

– the link to the construction in GCLC language.

An example of the page containing a solution to Problem 47 : $\{A, M_a, T_a\}$ is presented in Figures 7 and 8.

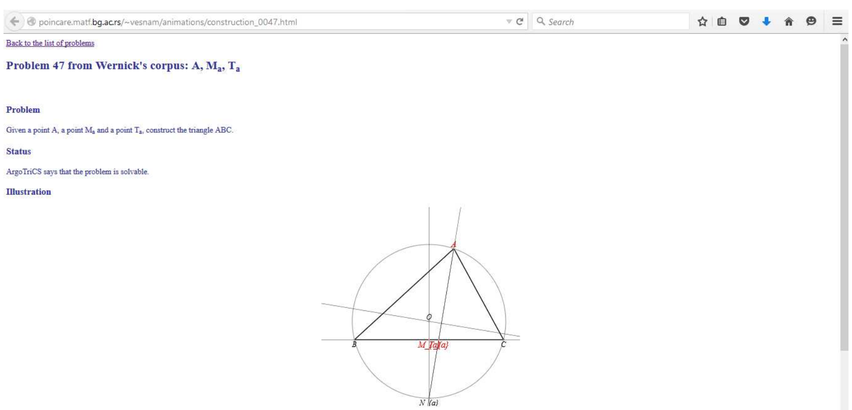



Fig. 7. Automatically generated HTML page containing the solution of Problem 47 with full construction given (1)

### 4.3. Related work

In the near past, some efforts were already put into compiling collections of solutions to triangle construction problems.

Fursenko considered a somewhat different set of triangle construction problems [18] (which involves size of the angles in triangle, lengths of sides, altitudes, medians, radius of inscribed circle and circumcircle, etc.) and published (in Russian) the

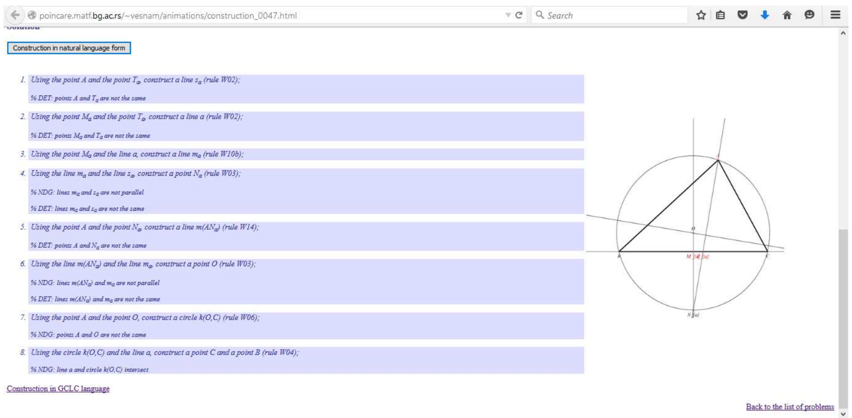



Fig. 8. Automatically generated HTML page containing the solution of Problem 47 with full construction given (2)

results consisting a brief analysis of 350 essentially different problems sometimes followed by an accompanying illustration [7,8]. However, all solutions were manually made and there is neither informal nor formal description of constructions.

Lopes considered more-or-less the same corpus of problems and published some of these results on-line (in Spanish) [17]. However, these solutions offer only the analysis of the solution (which is sometimes obtained by some algebraic arguments), there is no description of the construction, nor the according illustration and all presented solutions were made by hand.

Montesdeoca published in Spanish an on-line compendium consisting of around 200 interesting triangle construction problems [24]. For each of the problems, a link to detailed analysis is given, accompanied by the illustration in Cabri. Also, the number of solutions is discussed. All of these solutions were made by hand, with no automation involved.

Specht created an on-line compendium in German [29] which offers solutions to 62 (out of 74) solvable problems from Wernick's list. However it deals only with original Wernick's list (and not Connelly's list or their extensions) and all presented solutions were made by hand. Also, there are no formal specifications of the constructions.

None of these compendiums do not offer step-by-step animations. Also, only the compendium proposed in this paper is generated automatically, in a systematic manner, which makes it more reliable compared to the other systems.

There are also another approaches to automatically solving construction problems in geometry [3,9,10,27], but they were not used for generation of big, on-line compendiums of construction problems.

## 5. Conclusions and future work

In this paper we have presented the on-line compendiums consisting of automatically generated solutions to most of the problems from extended Wernick's and Connelly's corpora. To our knowledge, these are the first on-line compendiums which were completely automatically generated and which provide step-by-step animated constructions. They also contain elegant constructions in natural language form, as well as formal descriptions of constructions, with associated illustrations.

For the future work, we plan to consider other corpora of triangle construction problems based on various geometrical quantities [7,8,18] and to generate similar on-line compendiums of solved problems from these corpora.

Also, we plan to incorporate the new levels of interaction with users into the compendiums, thereby making them more interactive.

It would be also useful to provide to students one-step guidance, by showing step by step of the analysis phase of the solution of construction problem. For instance, some properties which follow from the specification of the problem could be given, but not enough for performing the construction, and student should find the rest of the properties which enable a construction.

Acknowledgment. I would like to thank Andjelka Zečević for her help about using jQuery and Predrag Janičić for many useful comments.

### REFERENCES

[1] J. Anglesio and V. Schindler, *Solution to problem 10719*, Amer. Math. Monthly **107** (2000), 952–954.

[2] F. Botana, M. Hohenwarter, P. Janičić, Z. Kovács, I. Petrović, T. Recio, S. Weitzhofer, *Automated theorem proving in GeoGebra: Current achievements*, J. Automated Reasoning **55**, 1 (2015), 39–59.

[3] S.C. Chou, X.S. Gao, J.Z. Zhang, *Machine Proofs in Geometry*, World Scientific, Singapore, 1994.

[4] H. Connelly, *An extension of triangle constructions from located points*, Forum Geometricorum, **9** (2009), 109–112.

[5] H. Connelly, N. Dergiades, J.-P. Ehrmann, *Construction of triangle from a vertex and the feet of two angle bisectors*, Forum Geometricorum, **7** (2007), 103–106.

[6] E. Danneels, *A simple construction of a triangle from its centroid, incenter, and a vertex*, Forum Geometricorum, **5** (2005), 53–56.

[7] V. B. Fursenko, *Lexicographic account of triangle construction problems (part I)*, Mathematics in schools **5** (1937), 4–30 [In Russian].

[8] V. B. Fursenko, *Lexicographic account of triangle construction problems (part II)*, Mathematics in schools **6** (1937), 21–45 [In Russian].

[9] Xiao-Shan Gao, Shang-Ching Chou, Solving geometric constraint systems. II. A symbolic approach and decision of rc-constructibility, Computer Aided Design **30** (1998), 115–122.

[10] S. Gulwani, V. A. Korthikanti, A. Tiwari, *Synthesizing geometry constructions*, in: *Programming Language Design and Implementation, PLDI 2011*, pp. 50–61, ACM, 2011.

[11] M. Hohenwarter, K. Fuchs, *Combination of dynamic geometry, algebra and calculus in the software system GeoGebra*, Proceedings of Computer Algebra Systems and Dynamic Geometry Systems in Mathematics Teaching Conference, 2004.

[12] N. R. Jackiw, W. F. Finzer, *The Geometer's Sketchpad: Programming by Geometry*, in: *Watch What I Do: Programming by Demonstration*, MIT Press, Cambridge, USA, 1993.

[13] P. Janičić, *GCLC – a tool for constructive Euclidean geometry and more than that*, in: *Proceedings of International Congress of Mathematical Software (ICMS 2006)* (N. Takayama, A. Iglesias, J. Gutierrez, eds.), Lecture Notes in Computer Science, Vol. 4151, pp. 58–73, Springer-Verlag, 2006.

[14] P. Janičić, P. Quaresma, *System description: Gclcprover + GeoThms*, in: *International Joint Conference on Automated Reasoning (IJCAR-2006)* (U. Furbach, N. Shankar, eds.), Lecture Notes in Artificial Intelligence, Vol. 4130, pp. 145–150, Springer-Verlag, 2006.

[15] P. Janičić, *Geometry constructions language*, J. Automated Reasoning **4**, 1-2 (2010), 3–24.

[16] J.-M. Laborde, R. Strasser, *Cabri-géométre: A microworld of geometry for guided discovery learning*, Zentrablatt für Didactic der Matematik **22**, 5 (1990), 171–177.

[17] L. Lopes, *Report on Construction Problems*, on-line at: `http://www.escolademestres.com/images/stories/artigos/pdf/construcoes-geometricas-parte-8-luis-lopes.pdf`

[18] L. Lopes, *Manuel de Construction de Triangles*, QED Texte, 1996.

[19] F. Marić, I. Petrović, D. Petrović, P. Janičić, *Formalization and implementation of algebraic methods in geometry*, in: *Proceedings First Workshop on CTP Components for Educational Software, Wrocław, Poland, 31th July 2011* (P. Quaresma, R.-J. Back, eds.), Electronic Proceedings in Theoretical Computer Science, Vol. 79, pp. 63–81, Open Publishing Association, 2012.

[20] V. Marinković, *ArgoTriCS – Automated Triangle Construction Solver*, submitted.

[21] V. Marinković, P. Janičić, *Towards understanding triangle construction problems*, in: *Intelligent Computer Mathematics – CICM 2012* (J. Jeuring et al., eds.), Lecture Notes in Computer Science, Vol. 7362, Springer, 2012.

[22] V. Marinković, P. Janičić, P. Schreck, *Solving geometric construction problems supported by theorem proving*, in: *Proceedings of the 10th International Workshop on Automated Deduction in Geometry (ADG 2014)* CISUC Technical Report TR 2014/01, University of Coimbra, pp. 121–146, 2014.

[23] L. F. Meyers, *Update on William Wernick's triangle constructions with three located points*, Math. Magazine **69**, 1 (1996), 46–49.

[24] A. Montesdeoca, *Resólucion de triángulos*, Technical report, 2015.

[25] J. Richter-Gebert, U. Kortenkamp, *Cinderella – The interactive geometry software*, Springer, 1993.

[26] D. Scher, *Lifting the curtain: The evolution of the Geometer's Sketchpad*, Math. Educator **10**, 2 (2000).

[27] P. Schreck, *Constructions à la règle et au compas*, PhD Thesis, University of Strasbourg, 1993.

[28] P. Schreck, P. Mathis, *RC-constructibility of problems in Wernick's list*, in: *Proceedings of the 10th International Workshop on Automated Deduction in Geometry (ADG 2014)*, pp. 85–104, CISUC Technical Report TR 2014/01, University of Coimbra, 2014.

[29] E. Specht, *Wernicks Liste* [in German], `http://hydra.nat.uni-magdeburg.de/wernick/`

[30] I. Stewart, *Galois Theory*, Chapman and Hall Ltd., 1973.

[31] A. V. Ustinov, *On the construction of a triangle from the feet of its angle bisectors*, Forum Geometricorum **9** (2009), 279–280.

[32] W. Wernick, *Triangle constructions with three located points*, Math. Magazine **55**, 4 (1982), 227–230.

[33] P. Yiu, *Elegant geometric constructions*, Forum Geometricorum **5** (2005), 75–96.



Faculty of Mathematics, University of Belgrade, Serbia

*E-mail*: `vesnap@matf.bg.ac.rs`