International Journal of Computer Discovered Mathematics (IJCDM)
ISSN 2367-7775 ©IJCDM
Volume 9, 2024, pp. 259-263.
Received July 19, 2024. Published on-line December 31, 2024.
web: http://www.journal-1.eu/
©The Author(s) This article is published with open access[1].

Generalization of the Nagel Point

Benjamin Lee Warren[a] [2]
University of Binghamton, Mathematics,
4400 Vestel Pkwy E, Binghamton, NY 13902
e-mail: warrenbenjamin52@yahoo.com
web: https://orcid.org/0000-0002-2829-6953

Mathematics Subject Classification (2010). 51N20.

1. Introduction

The Nagel point [1] is the point of concurrence of the three splitters of a triangle, where a splitter is the line connecting a vertex to the point on the opposite side which requires traversing half the perimeter clockwise to reach it. Here, we will present a generalization of the Nagel point. Namely, we will prove that the cevian triangle of a cevian triangle of ABC is in perspective with ABC.

2. Results

Theorem 2.1. *Let A, B, and C be the vertices of a given triangle.*
Let D be some point in the plane.
Let lines AB and CD intersect at point E.
Let lines AC and BD intersect at point F.
Let lines AD and BC intersect at point G.
Let H be some point in the plane.
Let lines EF and GH intersect at point I.
Let lines EG and FH intersect at point J.
Let lines EH and FG intersect at point K.
Then lines AI, BJ, and CK are concurrent at point L.

[1]This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.
[2]Corresponding author

Proof. Without loss of generality, we will work in Cartesian coordinates.
A has coordinates $(0, 0)$.
B has coordinates $(1, 0)$.
C has coordinates (x_C, y_C).
D has coordinates (x_D, y_D).
E has coordinates $((x_C * y_D - x_D * y_C)/(y_D - y_C), 0)$.
F has coordinates $((x_C * y_D)/(x_C * y_D - x_D * y_C + y_C), (y_C * y_D)/(x_C * y_D - x_D * y_C + y_C))$.
G has coordinates $((x_D * y_C)/(-x_C * y_D + y_D + x_D * y_C), (y_C * y_D)/(-x_C * y_D + y_D + x_D * y_C))$.
H has coordinates (x_H, y_H).
I has coordinates
$x_I = -(y_C*(x_D*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_C*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H-2*(x_C-1)*x_C*x_H*y_D^3+(((4*x_C-1)*x_D-2*x_C)*x_H-x_C*x_D)*y_C*y_D^2-x_D*((2*x_D-1)*x_H-x_D)*y_C^2*y_D))/(((x_C-1)*y_D-x_D*y_C)*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_C*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H+(x_C-1)*(x_H+x_C)*y_C*y_D^3-((x_D+x_C-1)*x_H+(2*x_C-1)*x_D-2*x_C)*y_C^2*y_D^2+x_D*(x_H+x_D-2)*y_C^3*y_D)$
and
$y_I = (y_C*y_D*(((x_C-1)*x_C*y_D^2-2*(x_C-1)*x_D*y_C*y_D+(x_D-1)*x_D*y_C^2)*y_H-(x_H-x_C)*y_C*y_D^2+(x_H-x_D)*y_C^2*y_D))/(((x_C-1)*y_D-x_D*y_C)*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_C*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H+(x_C-1)*(x_H+x_C)*y_C*y_D^3-((x_D+x_C-1)*x_H+(2*x_C-1)*x_D-2*x_C)*y_C^2*y_D^2+x_D*(x_H+x_D-2)*y_C^3*y_D)$.
J has coordinates
$x_J = (y_D*(x_C*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_D*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H-x_C*((2*x_C-1)*x_H-x_C)*y_C*y_D^2+((2*(2*x_C-1)*x_D-x_C)*x_H-x_C*x_D)*y_C^2*y_D-2*(x_D-1)*x_D*x_H*y_C^3))/((x_C*y_D-(x_D-1)*y_C)*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_D*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H-x_C*(x_H+x_C-2)*y_C*y_D^3+((x_D+x_C-1)*x_H+2*(x_C-1)*x_D-x_C)*y_C^2*y_D^2-(x_D-1)*(x_H+x_D)*y_C^3*y_D)$
and
$y_J = -(y_C*y_D*(((x_C-1)*x_C*y_D^2-2*x_C*(x_D-1)*y_C*y_D+(x_D-1)*x_D*y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2-(x_H-x_D)*y_C^2*y_D))/((x_C*y_D-x_D*y_C+y_C)*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_D*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H-x_C*(x_H+x_C-2)*y_C*y_D^3+((x_D+x_C-1)*x_H+2*(x_C-1)*x_D-x_C)*y_C^2*y_D^2-(x_D-1)*(x_H+x_D)*y_C^3*y_D)$.
K has coordinates
$x_K = ((x_C * y_D - x_D * y_C) * (((x_C - 1) * x_C * y_D^2 - (x_D - 1) * x_D * y_C^2) * y_H - (x_H - x_C) * y_C * y_D^2 + (x_H - x_D) * y_C^2 * y_D))/((y_D - y_C) * ((x_C - 1) * x_C * y_D^2 - (x_D - 1) * x_D * y_C^2) * y_H - (2 * x_C - 1) * (x_H - x_C) * y_C * y_D^3 + (2 * (x_D + x_C - 1) * x_H - (4 * x_C - 1) * x_D + x_C) * y_C^2 * y_D^2 - (2 * x_D - 1) * (x_H - x_D) * y_C^3 * y_D)$
and
$y_K = (2*y_C*y_D*((x_C-1)*y_D-(x_D-1)*y_C)*(x_C*y_D-x_D*y_C)*y_H)/((y_D-y_C)*((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(2*x_C-1)*(x_H-x_C)*y_C*y_D^3+(2*(x_D+x_C-1)*x_H-(4*x_C-1)*x_D+x_C)*y_C^2*y_D^2-(2*x_D-1)*(x_H-x_D)*y_C^3*y_D)$.
Line AI is given by the equation
$y = -(y_D * (((x_C^2 - x_C) * y_D^2 + (2 - 2 * x_C) * x_D * y_C * y_D + (x_D^2 - x_D) * y_C^2) * y_H + (x_C - x_H) * y_C * y_D^2 + (x_H - x_D) * y_C^2 * y_D) * x)/(((x_C^2 - x_C) * x_D * y_D^2 + (2 * x_C * x_D - 2 * x_C * x_D^2) * y_C * y_D + (x_D^3 - x_D^2) * y_C^2) * y_H + (2 * x_C - 2 * x_C^2) * x_H * y_D^3 + (((4 * x_C - 1) * x_D - 2 * x_C) * x_H - x_C * x_D) * y_C * y_D^2 + ((x_D - 2 * x_D^2) * x_H + x_D^2) * y_C^2 * y_D)$.
Line BJ is given by the equation
$y = -(y_D * (((x_C - 1) * x_C * y_D^2 - 2 * x_C * (x_D - 1) * y_C * y_D + (x_D - 1) * x_D *$

$y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2-(x_H-x_D)*y_C^2*y_D)*(x-1))/((x_D-1)*(x_C^2*y_D^2-x_C*y_D^2-2*x_C*x_D*y_C*y_D+2*x_D*y_C*y_D+x_D^2*y_C^2-x_D*y_C^2)*y_H-2*(x_C-1)*x_C*(x_H-1)*y_D^3+(((4*x_C-3)*x_D-2*x_C+1)*x_H-(3*x_C-2)*x_D+x_C)*y_C*y_D^2-(x_D-1)*((2*x_D-1)*x_H-x_D)*y_C^2*y_D)$.
Line CK is given by the equation
$y=(((2*y_C*y_D*((x_C-1)*y_D-(x_D-1)*y_C)*(x_C*y_D-x_D*y_C)*y_H)/((y_D-y_C)*((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(2*x_C-1)*(x_H-x_C)*y_C*y_D^3+(2*(x_D+x_C-1)*x_H-(4*x_C-1)*x_D+x_C)*y_C^2*y_D^2+(1-2*x_D)*(x_H-x_D)*y_C^3*y_D)-y_C)*(x-x_C))/(((x_C*y_D-x_D*y_C)*(((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(x_H-x_C)*y_C*y_D^2+(x_H-x_D)*y_C^2*y_D))/((y_D-y_C)*((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(2*x_C-1)*(x_H-x_C)*y_C*y_D^3+(2*(x_D+x_C-1)*x_H-(4*x_C-1)*x_D+x_C)*y_C^2*y_D^2+(1-2*x_D)*(x_H-x_D)*y_C^3*y_D)-y_C)*(x-x_C))+y_C$.
Lines AI and BJ intersect at point L whose coordinates are given by
$x_L=((((x_C^2-x_C)*y_D^2+(2*x_C-2*x_C*x_D)*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2+(x_D-x_H)*y_C^2*y_D)*(((x_C^2-x_C)*x_D*y_D^2+(2*x_C*x_D-2*x_C*x_D^2)*y_C*y_D+(x_D^3-x_D^2)*y_C^2)*y_H+(2*x_C-2*x_C^2)*x_H*y_D^3+(((4*x_C-1)*x_D-2*x_C)*x_H-x_C*x_D)*y_C*y_D^2+((x_D-2*x_D^2)*x_H+x_D^2)*y_C^2*y_D))/(((x_C^4-2*x_C^3+x_C^2)*y_D^4+((4*x_C-4*x_C^2)*x_D^2+(4*x_C^2-4*x_C)*x_D)*y_C*y_D^3+((8*x_C-4)*x_D^3+(-2*x_C^2-10*x_C+4)*x_D^2+(2*x_C^2+2*x_C)*x_D)*y_C^2*y_D^2+(-4*x_D^4+8*x_D^3-4*x_D^2)*y_C^3*y_D+(x_D^4-2*x_D^3+x_D^2)*y_C^4)*y_H^2+((-2*x_C^4+4*x_C^3-2*x_C^2)*y_D^5+(((8*x_C^2-8*x_C)*x_D-4*x_C^3+2*x_C^2+2*x_C)*x_H+(4*x_C^3-10*x_C^2+6*x_C)*x_D)*y_C*y_D^4+(((8-16*x_C)*x_D^2+(4*x_C^2+12*x_C-4)*x_D-2*x_C^2-2*x_C)*x_H+(8*x_C-4)*x_D^2+(-2*x_C^2-2*x_C)*x_D)*y_C^2*y_D^3+((8*x_D^3+(4*x_C-14)*x_D^2+(6-4*x_C)*x_D)*x_H+(-4*x_C-2)*x_D^3+(4*x_C+2)*x_D^2)*y_C^3*y_D^2+((-4*x_D^3+6*x_D^2-2*x_D)*x_H+2*x_D^4-2*x_D^3)*y_C^4*y_D)*y_H+((4*x_C-4*x_C^2)*x_H^2+(4*x_C^3-2*x_C^2-2*x_C)*x_H-2*x_C^3+2*x_C^2)*y_C*y_D^5+(((8*x_C-4)*x_D+4*x_C^2-8*x_C+1)*x_H^2+((-12*x_C^2+4*x_C+2)*x_D+2*x_C^2+2*x_C)*x_H+(6*x_C^2-4*x_C)*x_D-x_C^2)*y_C^2*y_D^4+((-4*x_D^2+(8-8*x_C)*x_D+4*x_C-2)*x_H^2+((12*x_C-2)*x_D^2+(-4*x_C-2)*x_D)*x_H+(2-6*x_C)*x_D^2+2*x_C*x_D)*y_C^3*y_D^3+((4*x_D^2-4*x_D+1)*x_H^2+(2*x_D^2-4*x_D^3)*x_H+2*x_D^3-x_D^2)*y_C^4*y_D^2)$
and
$y_L=-(y_D*(((x_C^2-x_C)*y_D^2+(2-2*x_C)*x_D*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_C-x_H)*y_C*y_D^2+(x_H-x_D)*y_C^2*y_D)*(((x_C^2-x_C)*y_D^2+(2*x_C-2*x_C*x_D)*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2+(x_D-x_H)*y_C^2*y_D))/(((x_C^4-2*x_C^3+x_C^2)*y_D^4+((4*x_C-4*x_C^2)*x_D^2+(4*x_C^2-4*x_C)*x_D)*y_C*y_D^3+((8*x_C-4)*x_D^3+(-2*x_C^2-10*x_C+4)*x_D^2+(2*x_C^2+2*x_C)*x_D)*y_C^2*y_D^2+(-4*x_D^4+8*x_D^3-4*x_D^2)*y_C^3*y_D+(x_D^4-2*x_D^3+x_D^2)*y_C^4)*y_H^2+((-2*x_C^4+4*x_C^3-2*x_C^2)*y_D^5+(((8*x_C^2-8*x_C)*x_D-4*x_C^3+2*x_C^2+2*x_C)*x_H+(4*x_C^3-10*x_C^2+6*x_C)*x_D)*y_C*y_D^4+(((8-16*x_C)*x_D^2+(4*x_C^2+12*x_C-4)*x_D-2*x_C^2-2*x_C)*x_H+(8*x_C-4)*x_D^2+(-2*x_C^2-2*x_C)*x_D)*y_C^2*y_D^3+((8*x_D^3+(4*x_C-14)*x_D^2+(6-4*x_C)*x_D)*x_H+(-4*x_C-2)*x_D^3+(4*x_C+2)*x_D^2)*y_C^3*y_D^2+((-4*x_D^3+6*x_D^2-2*x_D)*x_H+2*x_D^4-2*x_D^3)*y_C^4*y_D)*y_H+((4*x_C-4*x_C^2)*x_H^2+(4*x_C^3-2*x_C^2-2*x_C)*x_H-2*x_C^3+2*x_C^2)*y_C*y_D^5+(((8*x_C-4)*x_D+4*x_C^2-8*x_C+1)*x_H^2+((-12*x_C^2+4*x_C+2)*x_D+2*x_C^2+2*x_C)*x_H+(6*x_C^2-4*x_C)*x_D-x_C^2)*y_C^2*y_D^4+((-4*x_D^2+(8-8*x_C)*x_D+4*x_C-2)*x_H^2+((12*x_C-2)*x_D^2+(-4*x_C-2)*x_D)*x_H+(2-6*x_C)*x_D^2+2*x_C*x_D)*y_C^3*y_D^3+((4*x_D^2-4*x_D+1)*x_H^2+(2*x_D^2-4*x_D^3)*x_H+2*x_D^3-x_D^2)*y_C^4*y_D^2)$.
We then verify that L lies on line CK since we calculate that
$(((2*y_C*y_D*((x_C-1)*y_D-(x_D-1)*y_C)*(x_C*y_D-x_D*y_C)*y_H)/((y_D-y_C)*((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(2*x_C-1)*(x_H-x_C)*y_C*y_D^3+(2*(x_D+x_C-1)*x_H-(4*x_C-1)*x_D+x_C)*y_C^2*y_D^2+(1-2*x_D)*(x_H-x_D)*y_C^3*y_D)-y_C)*((((((x_C^2-$

$x_C)*y_D^2+(2*x_C-2*x_C*x_D)*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2+(x_D-x_H)*y_C^2*y_D)*(((x_C^2-x_C)*x_D*y_D^2+(2*x_C*x_D-2*x_C*x_D^2)*y_C*y_D+(x_D^3-x_D^2)*y_C^2)*y_H+(2*x_C-2*x_C^2)*x_H*y_D^3+(((4*x_C-1)*x_D-2*x_C)*x_H-x_C*x_D)*y_C*y_D^2+((x_D-2*x_D^2)*x_H+x_D^2)*y_C^2*y_D))/(((x_C^4-2*x_C^3+x_C^2)*y_D^4+((4*x_C-4*x_C^2)*x_D^2+(4*x_C^2-4*x_C)*x_D)*y_C*y_D^3+((8*x_C-4)*x_D^3+(-2*x_C^2-10*x_C+4)*x_D^2+(2*x_C^2+2*x_C)*x_D)*y_C^2*y_D^2+(-4*x_D^4+8*x_D^3-4*x_D^2)*y_C^3*y_D+(x_D^4-2*x_D^3+x_D^2)*y_C^4)*y_H^2+((-2*x_C^4+4*x_C^3-2*x_C^2)*y_D^5+(((8*x_C^2-8*x_C)*x_D-4*x_C^3+2*x_C^2+2*x_C)*x_H+(4*x_C^3-10*x_C^2+6*x_C)*x_D)*y_C*y_D^4+(((8-16*x_C)*x_D^2+(4*x_C^2+12*x_C-4)*x_D-2*x_C^2-2*x_C)*x_H+(8*x_C-4)*x_D^2+(-2*x_C^2-2*x_C)*x_D)*y_C^2*y_D^3+((8*x_D^3+(4*x_C-14)*x_D^2+(6-4*x_C)*x_D)*x_H+(-4*x_C-2)*x_D^3+(4*x_C+2)*x_D^2)*y_C^3*y_D^2+((-4*x_D^3+6*x_D^2-2*x_D)*x_H+2*x_D^4-2*x_D^3)*y_C^4*y_D)*y_H+((4*x_C-4*x_C^2)*x_H^2+(4*x_C^3-2*x_C^2-2*x_C)*x_H-2*x_C^3+2*x_C^2)*y_C*y_D^5+(((8*x_C-4)*x_D+4*x_C^2-8*x_C+1)*x_H^2+((-12*x_C^2+4*x_C+2)*x_D+2*x_C^2+2*x_C)*x_H+(6*x_C^2-4*x_C)*x_D-x_C^2)*y_C^2*y_D^4+((-4*x_D^2+(8-8*x_C)*x_D+4*x_C-2)*x_H^2+((12*x_C-2)*x_D^2+(-4*x_C-2)*x_D)*x_H+(2-6*x_C)*x_D^2+2*x_C*x_D)*y_C^3*y_D^3+((4*x_D^2-4*x_D+1)*x_H^2+(2*x_D^2-4*x_D^3)*x_H+2*x_D^3-x_D^2)*y_C^4*y_D^2))-x_C))/(((x_C*y_D-x_D*y_C)*(((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(x_H-x_C)*y_C*y_D^2+(x_H-x_D)*y_C^2*y_D))/((y_D-y_C)*((x_C-1)*x_C*y_D^2-(x_D-1)*x_D*y_C^2)*y_H-(2*x_C-1)*(x_H-x_C)*y_C*y_D^3+(2*(x_D+x_C-1)*x_H-(4*x_C-1)*x_D+x_C)*y_C^2*y_D^2+(1-2*x_D)*(x_H-x_D)*y_C^3*y_D)-x_C)+y_C-(-(y_D*(((x_C^2-x_C)*y_D^2+(2-2*x_C)*x_D*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_C-x_H)*y_C*y_D^2+(x_H-x_D)*y_C^2*y_D)*(((x_C^2-x_C)*y_D^2+(2*x_C-2*x_C*x_D)*y_C*y_D+(x_D^2-x_D)*y_C^2)*y_H+(x_H-x_C)*y_C*y_D^2+(x_D-x_H)*y_C^2*y_D))/(((x_C^4-2*x_C^3+x_C^2)*y_D^4+((4*x_C-4*x_C^2)*x_D^2+(4*x_C^2-4*x_C)*x_D)*y_C*y_D^3+((8*x_C-4)*x_D^3+(-2*x_C^2-10*x_C+4)*x_D^2+(2*x_C^2+2*x_C)*x_D)*y_C^2*y_D^2+(-4*x_D^4+8*x_D^3-4*x_D^2)*y_C^3*y_D+(x_D^4-2*x_D^3+x_D^2)*y_C^4)*y_H^2+((-2*x_C^4+4*x_C^3-2*x_C^2)*y_D^5+(((8*x_C^2-8*x_C)*x_D-4*x_C^3+2*x_C^2+2*x_C)*x_H+(4*x_C^3-10*x_C^2+6*x_C)*x_D)*y_C*y_D^4+(((8-16*x_C)*x_D^2+(4*x_C^2+12*x_C-4)*x_D-2*x_C^2-2*x_C)*x_H+(8*x_C-4)*x_D^2+(-2*x_C^2-2*x_C)*x_D)*y_C^2*y_D^3+((8*x_D^3+(4*x_C-14)*x_D^2+(6-4*x_C)*x_D)*x_H+(-4*x_C-2)*x_D^3+(4*x_C+2)*x_D^2)*y_C^3*y_D^2+((-4*x_D^3+6*x_D^2-2*x_D)*x_H+2*x_D^4-2*x_D^3)*y_C^4*y_D)*y_H+((4*x_C-4*x_C^2)*x_H^2+(4*x_C^3-2*x_C^2-2*x_C)*x_H-2*x_C^3+2*x_C^2)*y_C*y_D^5+(((8*x_C-4)*x_D+4*x_C^2-8*x_C+1)*x_H^2+((-12*x_C^2+4*x_C+2)*x_D+2*x_C^2+2*x_C)*x_H+(6*x_C^2-4*x_C)*x_D-x_C^2)*y_C^2*y_D^4+((-4*x_D^2+(8-8*x_C)*x_D+4*x_C-2)*x_H^2+((12*x_C-2)*x_D^2+(-4*x_C-2)*x_D)*x_H+(2-6*x_C)*x_D^2+2*x_C*x_D)*y_C^3*y_D^3+((4*x_D^2-4*x_D+1)*x_H^2+(2*x_D^2-4*x_D^3)*x_H+2*x_D^3-x_D^2)*y_C^4*y_D^2)) = 0$
as desired.

□

The point L becomes the Nagel point when D is taken to be the centroid and ΔIJK is taken to be the intouch triangle of ΔEFG.

REFERENCES

[1] Weisstein, Eric W. "Nagel Point." From MathWorld–A Wolfram Web Resource. https://mathworld.wolfram.com/NagelPoint.html

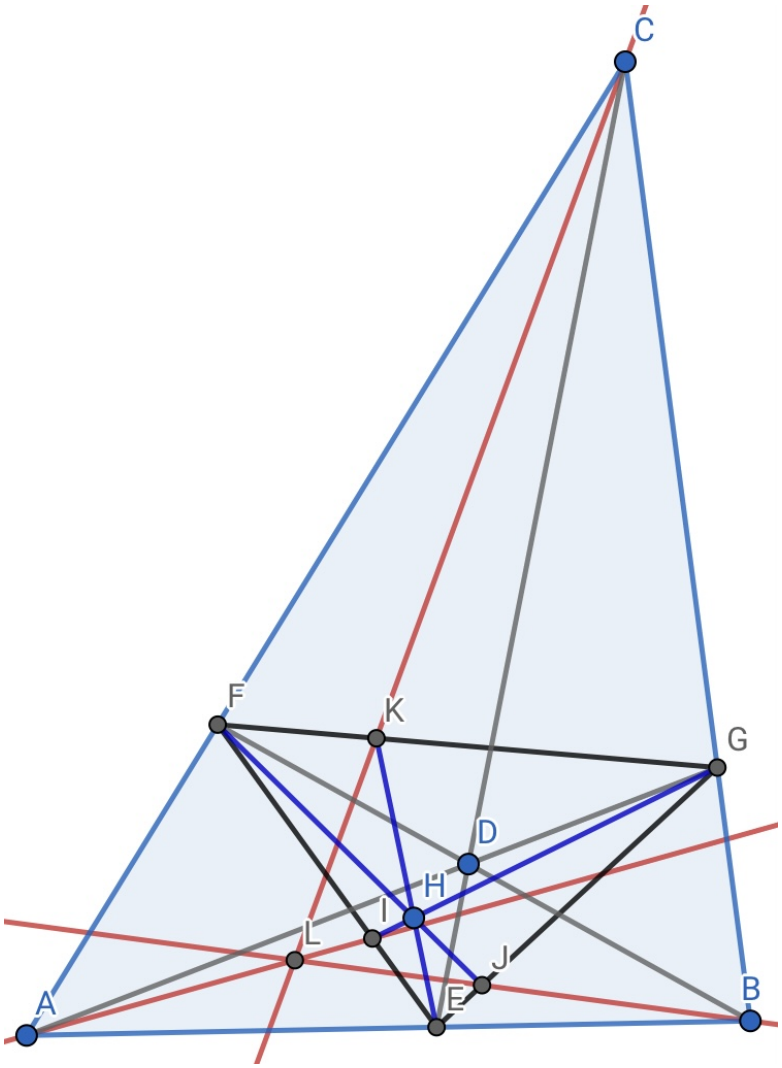

Figure 1. The cevian triangle of a cevian triangle is in perspective with the original triangle.