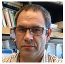
Supporting proving and discovering geometric inequalities in GeoGebra by using Tarski

Christopher W. Brown [1]   Zoltán Kovács [2]   Róbert Vajda [3]

[1]United States Naval Academy, Annapolis, MD, USA

[2]The Private University College of Education of the Diocese of Linz, Austria

[3]Bolyai Institute, Szeged, Hungary

ADG 2021, 16 September 2021

[2] Author is supported by the grant PID2020-113192GB-I00 from the Spanish MICINN.

[3] Author is supported by the EU-funded Hungarian grant EFOP-3.6.2-16-2017-00015.

Abstract

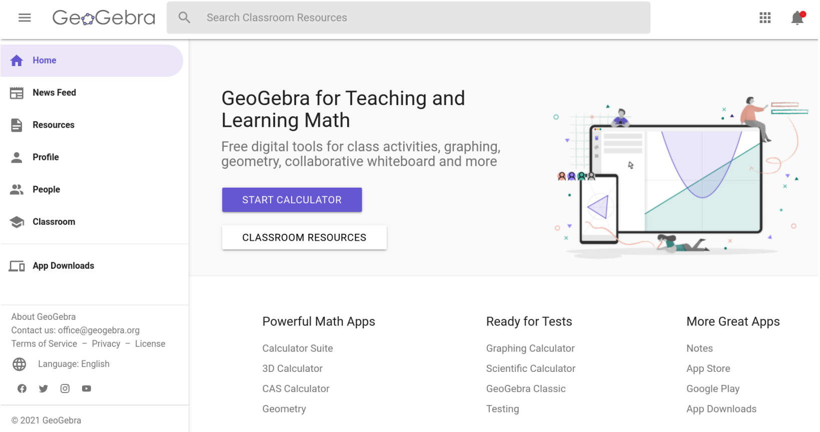
We introduce *GeoGebra Discovery* that can automatically prove or discover geometric inequalities. It consists of

- an extended version of *GeoGebra*,
- a controller web service *realgeom*,
- and the computational tool *Tarski* (with the extensive help of *QEPCAD B*).

We successfully solve several non-trivial problems in Euclidean planar geometry via a simple graphical user interface.

GeoGebra: open platform for teaching and learning math

geogebra.org



GeoGebra Discovery: an experimental version of GeoGebra

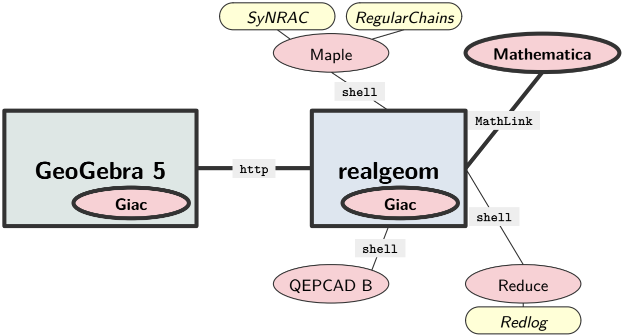
github.com/kovzol/geogebra-discovery

Feature	GeoGebra	GeoGebra Discovery	Next step
Discover tool/command	no	yes	Scheduled for merging into GeoGebra
Compare command	no	yes	GeoGebra Team: approve/update
IncircleCenter command	no	yes (with prover support)	GeoGebra Team: approve (discuss Center(Incircle) first)
Incircle tool	no	yes	GeoGebra Team: approve/update
IncircleCenter tool	no	yes	GeoGebra Team: approve/update
LocusEquation tool	no	yes	GeoGebra Team: approve/update
Envelope tool	no	yes	GeoGebra Team: approve/update
Raspberry Pi 3D View	no	yes	GeoGebra Team: approve/update
Java OpenGL	2.2	2.4	GeoGebra Team: approve/update
Giac: threads on Linux	no	yes	GeoGebra Team: approve/update
Same color for circles with the same radius	no	yes	GeoGebra Team: approve/update
Proving inequalities	no	yes	Use Tarski as a dynamic library
ApplyMap command	no	prototype	Fix bugs and make improvements

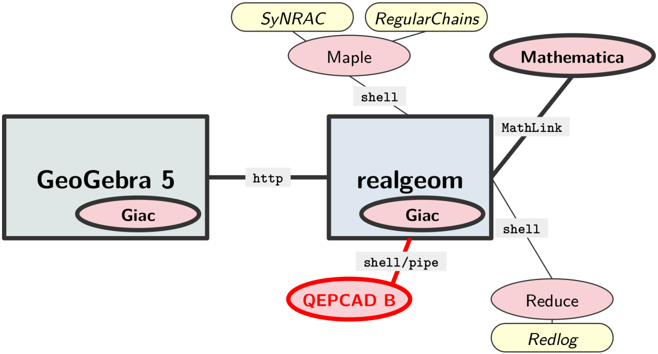
Feature	GeoGebra	GeoGebra Discovery	Next step
Discover tool/command	no	yes	Scheduled for merging into GeoGebra
Compare command	no	yes	GeoGebra Team: approve/update
IncircleCenter command	no	yes (with prover support)	GeoGebra Team: approve (discuss Center(Incircle) first)
Incircle tool	no	yes	GeoGebra Team: approve/update
IncircleCenter tool	no	yes	GeoGebra Team: approve/update
LocusEquation tool	no	yes	GeoGebra Team: approve/update
Envelope tool	no	yes	GeoGebra Team: approve/update
Raspberry Pi 3D View	no	yes	GeoGebra Team: approve/update
Java OpenGL	2.2	2.4	GeoGebra Team: approve/update
Giac: threads on Linux	no	yes	GeoGebra Team: approve/update
Same color for circles with the same radius	no	yes	GeoGebra Team: approve/update
Proving inequalities	no	yes	Use Tarski as a dynamic library
ApplyMap command	no	prototype	Fix bugs and make improvements

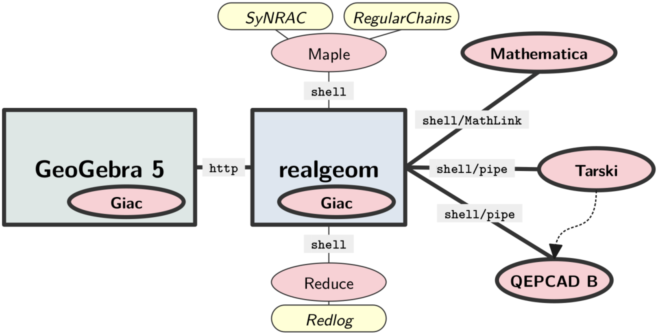
Implementation: System layout of GeoGebra Discovery

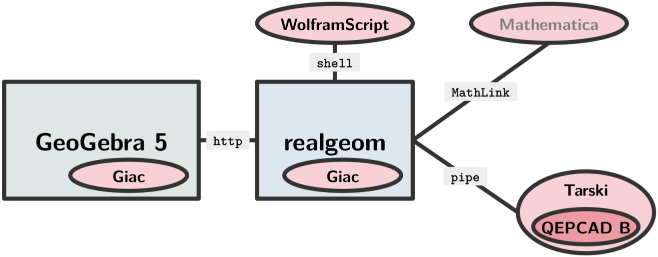
September 2020



- R. Vajda and Z. Kovács, "GeoGebra and the *realgeom* reasoning tool," in *PAAR+SC-Square 2020. Workshop on Practical Aspects of Automated Reasoning and Satisfiability Checking and Symbolic Computation Workshop 2020*, P. Fontaine, K. Korovin, I. S. Kotsireas, *et al.*, Eds., 2752 vols., Nov. 28, 2020, pp. 204–219. eprint: http://ceur-ws.org/Vol-2752/paper15.pdf. [Online]. Available: https://doi.org/urn:nbn:de:0074-2752-0

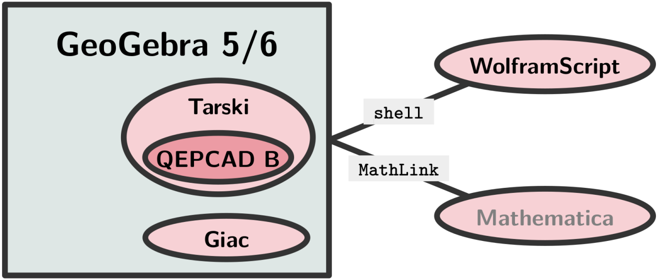

SyNRAC
RegularChains
Maple
Mathematica
shell
MathLink
GeoGebra 5
Giac
http
realgeom
Giac
shell
shell/pipe
QEPCAD B
Reduce
Redlog


SyNRAC
RegularChains
Maple
Mathematica
shell
shell/MathLink
GeoGebra 5
Giac
http
realgeom
Giac
shell/pipe
Tarski
shell/pipe
shell
QEPCAD B
Reduce
Redlog


WolframScript
Mathematica
shell
MathLink
GeoGebra 5
Giac
http
realgeom
Giac
pipe
Tarski
QEPCAD B

Implementation: System layout of GeoGebra Discovery

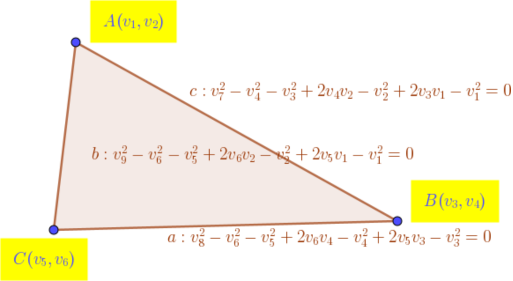
Planned, on-going work



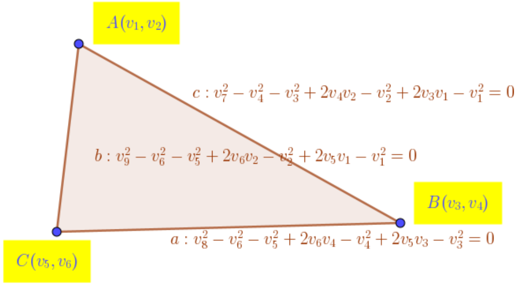
Motivation

A generalization of the Pythagorean Theorem

1. Equational hypotheses:



1. Equational hypotheses:

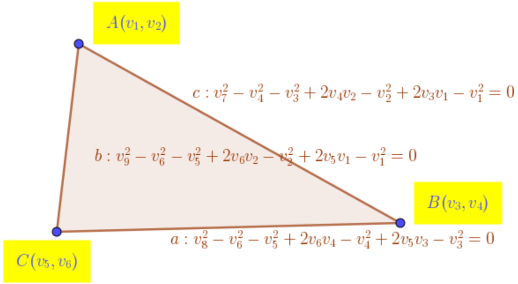


2. Non-degeneracy condition:

$$v_{10} \cdot (v_5 \cdot v_4 - v_6 \cdot v_3 - v_5 \cdot v_2 + v_3 \cdot v_2 + v_6 \cdot v_1 - v_4 \cdot v_1) = 1$$

3. Exploration related equation: $\mu \cdot v_7^2 = v_8^2 + v_9^2$
4. Non-equational assumptions: $v_7 > 0 \ \wedge \ v_8 > 0 \ \wedge \ v_9 > 0$

1. Equational hypotheses:



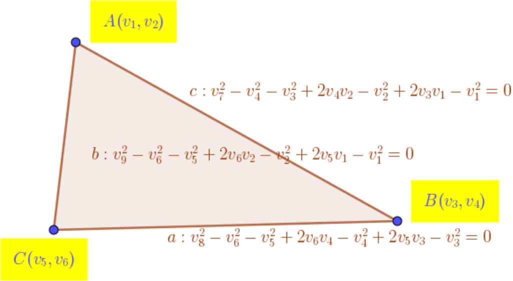
2. Non-degeneracy condition:

$$v_{10} \cdot (v_5 \cdot v_4 - v_6 \cdot v_3 - v_5 \cdot v_2 + v_3 \cdot v_2 + v_6 \cdot v_1 - v_4 \cdot v_1) = 1$$

3. Exploration related equation: $\mu \cdot v_7^2 = v_8^2 + v_9^2$
4. Non-equational assumptions: $v_7 > 0 \ \wedge \ v_8 > 0 \ \wedge \ v_9 > 0$

$\Rightarrow$

1. Equational hypotheses:



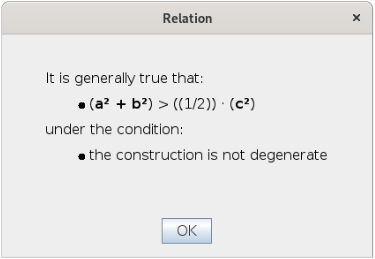
2. Non-degeneracy condition:

$$v_{10} \cdot (v_5 \cdot v_4 - v_6 \cdot v_3 - v_5 \cdot v_2 + v_3 \cdot v_2 + v_6 \cdot v_1 - v_4 \cdot v_1) = 1$$

3. Exploration related equation: $\mu \cdot v_7^2 = v_8^2 + v_9^2$
4. Non-equational assumptions: $v_7 > 0 \ \wedge \ v_8 > 0 \ \wedge \ v_9 > 0$

$\Rightarrow \mu > 1/2$

Symbolic check in GeoGebra (via `Relation`$(a^2 + b^2, c^2)$):



Applied methods

1. Exploration related equation:

$$Q_1 = \mu \cdot Q_2$$

   where Q_1 and Q_2 are the geometric quantities to compare and $\mu \in \mathbb{R}$ is a new variable ("proportion" or "ratio").

2. Derivation of an equivalent form of the (semi-)algebraic system:
   1. elimination via Gröbner bases, for algebraic systems,
   2. cylindrical algebraic decomposition (CAD) and real quantifier elimination (RQE), for semi-algebraic systems.

1. Exploration related equation:

$$Q_1 = \mu \cdot Q_2$$

   where Q_1 and Q_2 are the geometric quantities to compare and $\mu \in \mathbb{R}$ is a new variable ("proportion" or "ratio").

2. Derivation of an equivalent form of the (semi-)algebraic system:
   1. elimination via Gröbner bases, for algebraic systems,
   2. cylindrical algebraic decomposition (CAD) and real quantifier elimination (RQE), for semi-algebraic systems.

$\Rightarrow$

Applied methods

1. Exploration related equation:

$$Q_1 = \mu \cdot Q_2$$

   where Q_1 and Q_2 are the geometric quantities to compare and $\mu \in \mathbb{R}$ is a new variable ("proportion" or "ratio").

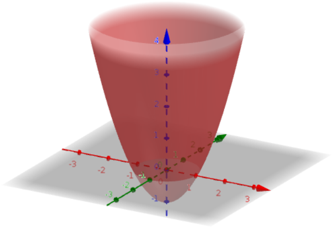
2. Derivation of an equivalent form of the (semi-)algebraic system:
   1. elimination via Gröbner bases, for algebraic systems,
   2. cylindrical algebraic decomposition (CAD) and real quantifier elimination (RQE), for semi-algebraic systems.

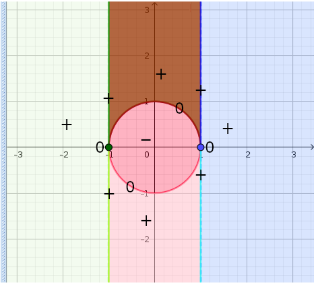
$$\Rightarrow m \cdot Q_2 \underset{(=)}{<} Q_1 \underset{(=)}{<} M \cdot Q_2$$

where $m, M \in \mathbb{R}_0^+$ are sharp constants.

Definition

Given a set S of polynomials in $\mathbb{Z}[x_1, x_2, \ldots, x_n]$, a CAD is a decomposition of $\mathbb{R}^n$ into special connected semi-algebraic sets, on which each polynomial has constant sign, either $+$, $-$ or 0.





Example: $S = \{x_1^2 + x_2^2 - 1\}$ and a CAD of it. Here $\mathbb{R}^2$ can be decomposed into 13 semi-algebraic sets ($13 = 1 + 3 + 5 + 3 + 1$).

The quantified formula (after simplifying):

$$\exists_{v_{10},v_5,v_6,v_7,v_8,v_9\in\mathbb{R}} v_7 > 0 \ \wedge \ v_8 > 0 \ \wedge \ v_9 > 0 \wedge$$
$$v_{10}v_6 = 1 \ \wedge \ -v_5^2 + 2v_5 - v_6^2 + v_8^2 = 1 \ \wedge \ v_5^2 + v_6^2 = v_9^2 \ \wedge$$
$$v7 = 1 \ \wedge \ \mu = v_8^2 + v_9^2.$$

The quantified formula (after simplifying):

$$\exists_{v_{10},v_5,v_6,v_7,v_8,v_9\in\mathbb{R}} v_7 > 0 \land v_8 > 0 \land v_9 > 0 \land$$
$$v_{10}v_6 = 1 \land -v_5^2 + 2v_5 - v_6^2 + v_8^2 = 1 \land v_5^2 + v_6^2 = v_9^2 \land$$
$$v7 = 1 \land \mu = v_8^2 + v_9^2.$$

$\Rightarrow \mu > 1/2$ (a quantifier-free formula).

Additional ways for users to enter input

...instead of using `Relation`$(a^2 + b^2, c^2)$

1. Direct proof by typing `Prove`$(a^2 + b^2 > c^2/2)$, or by trial-and-error:
   - e.g. `Prove`$(a^2 + b^2 > c^2)$

Additional ways for users to enter input

...instead of using `Relation`($a^2 + b^2$, c^2)

1. Direct proof by typing `Prove`($a^2 + b^2 > c^2/2$), or by trial-and-error:
   - e.g. `Prove`($a^2 + b^2 > c^2$) $\rightarrow$

Additional ways for users to enter input

...instead of using `Relation`(a^2+b^2,c^2)

1. Direct proof by typing `Prove`($a^2+b^2>c^2/2$), or by trial-and-error:
   - e.g. `Prove`($a^2+b^2>c^2$) $\rightarrow$ false,

Additional ways for users to enter input

...instead of using `Relation`$(a^2 + b^2, c^2)$

1. Direct proof by typing `Prove`$(a^2 + b^2 > c^2/2)$, or by trial-and-error:
    - e.g. `Prove`$(a^2 + b^2 > c^2) \rightarrow$ false,
    - e.g. `Prove`$(a^2 + b^2 > c^2/3)$

Additional ways for users to enter input

...instead of using `Relation(`$a^2 + b^2$`,`c^2`)`

1. Direct proof by typing `Prove(`$a^2 + b^2 > c^2/2$`)`, or by trial-and-error:
    - e.g. `Prove(`$a^2 + b^2 > c^2$`)` $\rightarrow$ false,
    - e.g. `Prove(`$a^2 + b^2 > c^2/3$`)` $\rightarrow$

Additional ways for users to enter input

...instead of using `Relation`$(a^2 + b^2, c^2)$

1. Direct proof by typing `Prove`$(a^2 + b^2 > c^2/2)$, or by trial-and-error:
   - e.g. `Prove`$(a^2 + b^2 > c^2)$ $\to$ false,
   - e.g. `Prove`$(a^2 + b^2 > c^2/3)$ $\to$ true,

Additional ways for users to enter input

...instead of using `Relation`($a^2 + b^2$, c^2)

1. Direct proof by typing `Prove`($a^2 + b^2 > c^2/2$), or by trial-and-error:
   - e.g. `Prove`($a^2 + b^2 > c^2$) $\rightarrow$ false,
   - e.g. `Prove`($a^2 + b^2 > c^2/3$) $\rightarrow$ true,
   - ...

Additional ways for users to enter input

...instead of using `Relation(`a^2+b^2, c^2`)`

1. Direct proof by typing `Prove(`$a^2+b^2>c^2/2$`)`, or by trial-and-error:
   - e.g. `Prove(`$a^2+b^2>c^2$`)` → false,
   - e.g. `Prove(`$a^2+b^2>c^2/3$`)` → true,
   - ...
2. Low-level command `Compare(`a^2+b^2, c^2`)` to get direct result (→ JavaScript API)

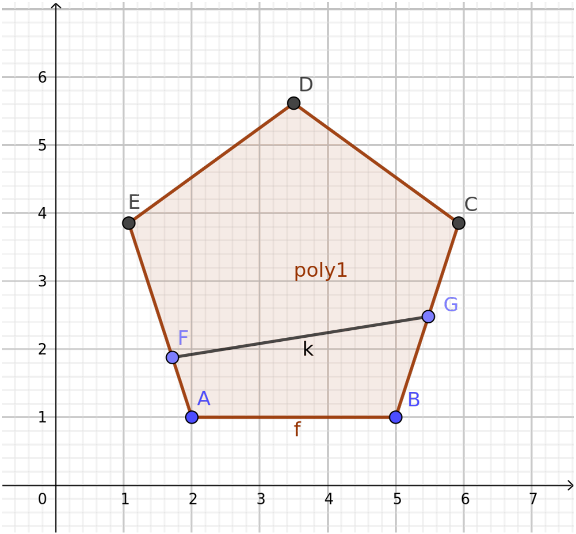
Additional ways for users to enter input

. . .instead of using `Relation`$(a^2 + b^2, c^2)$

1. Direct proof by typing `Prove`$(a^2 + b^2 > c^2/2)$, or by trial-and-error:
   - e.g. `Prove`$(a^2 + b^2 > c^2)$ $\rightarrow$ false,
   - e.g. `Prove`$(a^2 + b^2 > c^2/3)$ $\rightarrow$ true,
   - . . .
2. Low-level command `Compare`$(a^2 + b^2, c^2)$ to get direct result ($\rightarrow$ JavaScript API)
3. In simpler cases: point-and-click (via the Relation tool)

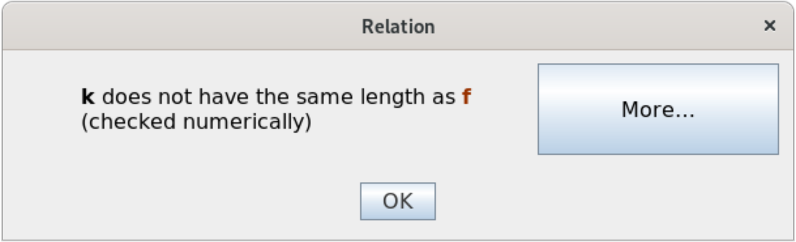
Shortest path between two sides of a regular pentagon?

Quick answer by using the Relation $\overset{?}{a=b}$ tool



Shortest path between two sides of a regular pentagon?

First attempt: a numerical comparison (no result)



Shortest path between two sides of a regular pentagon?

Second-third attempts: symbolic comparisons with proportions

Relation

It is generally true that:

- **f** ≤ **k** ≤ (((√5+1)/2) · **f**)

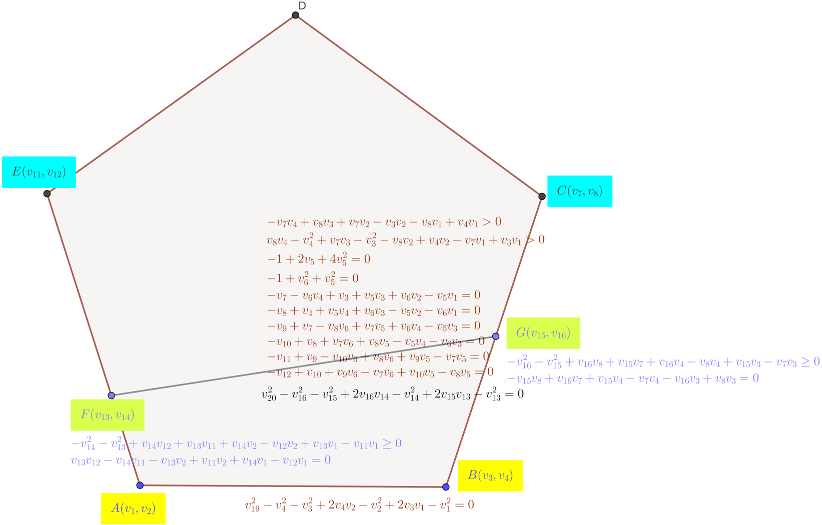
under the condition:

- the construction is not degenerate

OK

Shortest path between two sides of a regular pentagon?

The (semi-)algebraic translation of the geomeric setup



Shortest path between two sides of a regular pentagon?

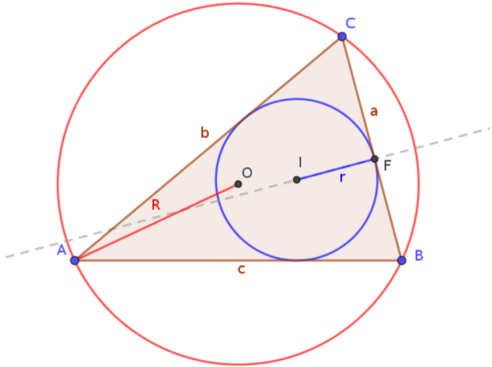
Final input for Tarski (after delineraization) ⇒ output

realgeom -s -g

File Edit Tabs Help

```
-v15*v8+v8+v16*v7-v16,-v10+2*v7*v8-v8,v7^2-v7-v8^2-v9+1,-v11*v14+v12*v13,-v13^2+2*v13*v15-v15^2-v14^
2+2*v14*v16-v16^2+v21^2,-m+v21,-v22+1]) -> [v10,v7,v8,v12,v9,v11,v15,v16,v14,v13,v21,m,v22]
2021-09-13 15:42:30.621 LOG: after removing unnecessary variables, vars=[v10,v7,v8,v12,v9,v11,v15,v1
6,v14,v13,v21,m,v22]
2021-09-13 15:42:30.621 LOG: after removing m, vars=v10,v7,v8,v12,v9,v11,v15,v16,v14,v13,v21,v22
2021-09-13 15:42:30.621 LOG: code=(def process (lambda (F) (def L (getargs F)) (def V (get L 0 0 1))
 (def B (bbwb (get L 1))) (if (equal? (get B 0) 'UNSAT) [false] ((lambda () (def G (qfr (t-ex V (get
 B 1)))) (if (equal? (t-type G) 1) G (if (equal? (t-type G) 6) (qepcad-api-call G 'T) (if (equal? (t
-type G) 5) (qepcad-api-call (bin-reduce t-or (map (lambda (H) (qepcad-api-call (exclose H '(m)) 'T)
) (getargs G))) 'T) (qepcad-api-call G 'T))))))))) (def expand (lambda (F)      (def A (getargs F))
     (def V (get A 0 0 1))      (def G (get A 1))      (def X (dnf G))      (def L (if (equal? (t-ty
pe X) 5) (getargs X) (list X)))      (map (lambda (f) (exclose f '(m))) L) ))(def epc (lambda (F) (n
ormalize (bin-reduce t-or (map (lambda (G) (if (equal? (t-type G) 6) (process G) G)) (expand F))))))
(epc [ ex v10,v11,v12,v13,v14,v15,v16,v21,v22,v7,v8,v9 [v21>0 /\ v22>0 /\ v10 v7-2 v7 v8-v12+v8 v9+v
8=0 /\ -v10 v8+v8^2-v11-v7^2+v7 v9+v7=0 /\ v7^2-2 v7+v8^2=0 /\ 4 v7^2-6 v7+1=0 /\ -v15 v8+v8+v16 v7-
v16=0 /\ -v10+2 v7 v8-v8=0 /\ v7^2-v7-v8^2-v9+1=0 /\ -v11 v14+v12 v13=0 /\ -v13^2+2 v13 v15-v15^2-v1
4^2+2 v14 v16-v16^2+v21^2=0 /\ -m+v21=0 /\ -v22+1=0 /\ (-v14^2-v13^2+v14 v12+v13 v11)>=0 /\ (-v16^2-
v15^2+v16 v8+v15 v7+v15-v7)>=0 /\ v8>0 /\ (v7-1)>0]])
[m - 1 >= 0 /\ m^2 - m - 1 <= 0]:tar
2021-09-13 15:42:32.8 LOG: result=m - 1 >= 0 /\ m^2 - m - 1 <= 0
2021-09-13 15:42:32.8 LOG: mathcode=solve([(m - 1 >= 0),(m^2 - m - 1 <= 0),(m>0)],m)
GIAC: solve([(m - 1 >= 0),(m^2 - m - 1 <= 0),(m>0)],m) -> list[((m>=1) and (m<=((√5+1)/2)))]
2021-09-13 15:42:32.801 ((m>=1) and (m<=((sqrt5+1)/2)))
```

Euler's Inequality

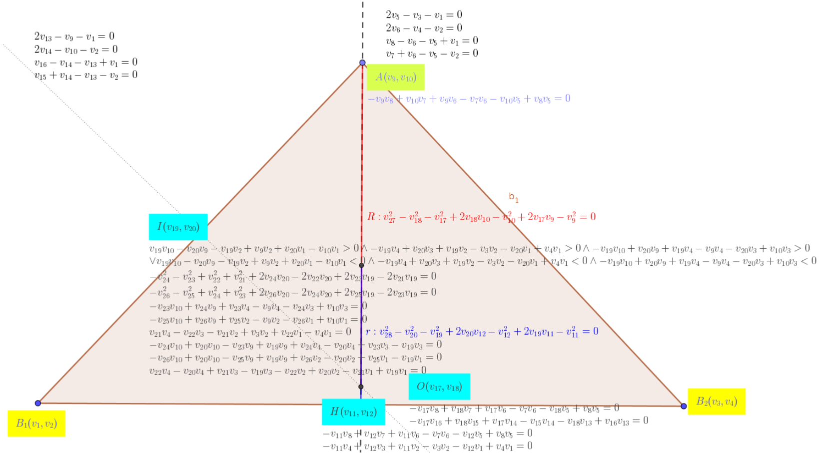


Theorem (Euler 1765, Chapple 1746)

In all triangle it holds that $R \geq 2 \cdot r$ where R is the circumradius and r is the inradius of the triangle.

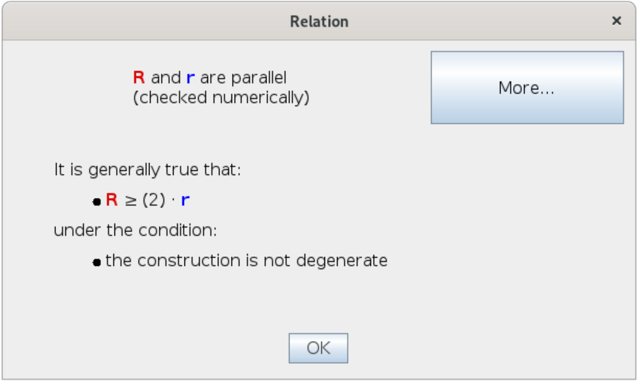
Euler's Inequality in an isosceles triangle

(Semi-)algebraic translation



Euler's Inequality in an isosceles triangle

Output in GeoGebra Discovery



Euler's Inequality

Benchmarking (outputs in seconds, timeout: 30 secs, Intel Xeon CPU X5675 @ 3.07GHz)

			CAD backend
Case	Result	*Mathematica*	*Tarski + QEPCAD B*
Isosceles	$R \geq 2 \cdot r$	1.2	8.7
Right	$R \geq (\sqrt{2}+1) \cdot r$	2.1	4.3
General	$R \geq 2 \cdot r$	timeout	21.5

Euler's Inequality

Benchmarking (outputs in seconds, timeout: 30 secs, Intel Xeon CPU X5675 @ 3.07GHz)

			CAD backend
Case	Result	*Mathematica*	*Tarski + QEPCAD B*
Isosceles	$R \geq 2 \cdot r$	1.2	8.7
Right	$R \geq (\sqrt{2} + 1) \cdot r$	2.1	4.3
General	$R \geq 2 \cdot r$	timeout	21.5

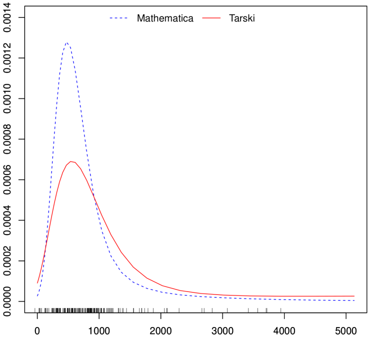
- A. Strzeboński, "Cylindrical algebraic decomposition using local projections," *Journal of Symbolic Computation*, vol. 76, pp. 36–64, Sep. 2016
- F. Vale-Enriquez and C. Brown, "Polynomial constraints and unsat cores in TARSKI," in *Mathematical Software – ICMS 2018. LNCS, vol. 10931*, Springer, Cham, 2018, pp. 466–474

Benchmarks

- 131 simple/moderate tests Database
  - 117/116 can be successfully solved (Mathematica/Tarski) within 30 seconds

Benchmarks

- 131 simple/moderate tests Database
  - 117/116 can be successfully solved (Mathematica/Tarski) within 30 seconds

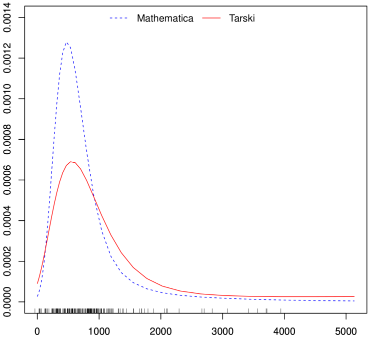


Density estimate on 103 tests that work uniformly (timing in ms),

$\mu_M = 1361,\ \mu_T = 2841,\ \sigma_M = 3379,\ \sigma_T = 4616$

Benchmarks

- 131 simple/moderate tests Database
  - 117/116 can be successfully solved (Mathematica/Tarski) within 30 seconds

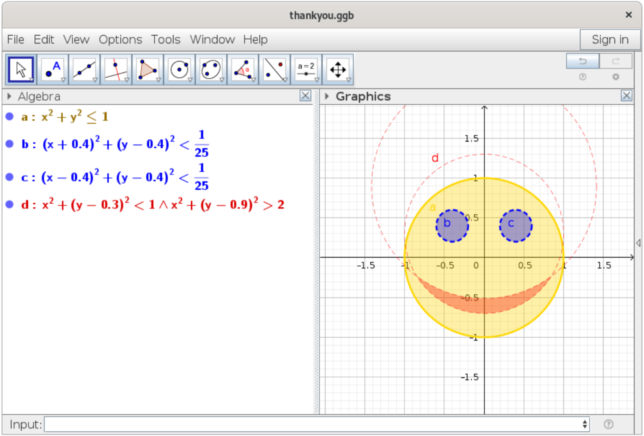


Density estimate on 103 tests that work uniformly (timing in ms),

$$\mu_M = 1361,\ \mu_T = 2841,\ \sigma_M = 3379,\ \sigma_T = 4616$$

- 46 additional tests to prove a given conjecture Database
  - 33/35 can be successfully proven (Mathematica/Tarski) within 40 seconds

Thank you!



The yellow region corresponds to a semi-algebraic set!