UniTriSat

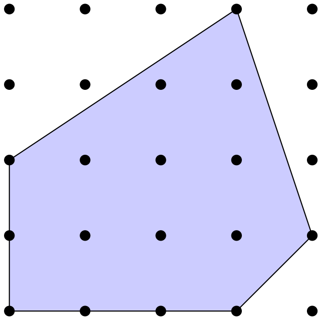
Unimodular Triangulations via SATISFIABILITY

Kyle Huang, Robert Lauff, Charles Zhang

BTU Cottbus-Senftenberg, TU Berlin, University of Vienna

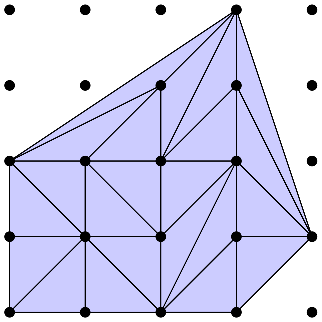
July 13th, 2026

Unimodular Triangulations



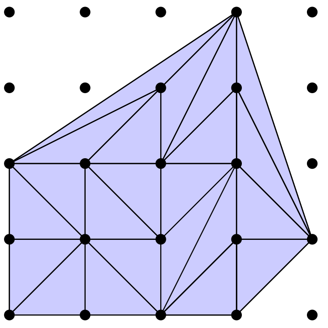
- P a lattice polytope

Unimodular Triangulations



- P a lattice polytope
- A *unimodular triagulation* of P is a triangulation of $P \cap \mathbb{Z}^d$ with simplices of normalized volume 1

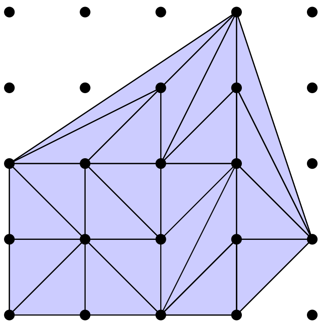
Unimodular Triangulations



- P a lattice polytope
- A *unimodular triagulation* of P is a triangulation of $P \cap \mathbb{Z}^d$ with simplices of normalized volume 1

Question: Given a lattice polytope, does it have a (flag, regular) unimodular triangulation?

Unimodular Triangulations



- P a lattice polytope
- A *unimodular triagulation* of P is a triangulation of $P \cap \mathbb{Z}^d$ with simplices of normalized volume 1

Question: Given a lattice polytope, does it have a (flag, regular) unimodular triangulation?

Open for smooth polytopes, lattice parallelipipeds, s-Lecture Hall simplices...

via SATISFIABILITY

via SATISFIABILITY

Input: A formula Φ of boolean variables $a_1, ..., a_n$, with $\neg, \vee$ and $\wedge$.

Output: Does there exist an assigment

$$a_1, ..., a_n \mapsto \{T, F\}$$

such that Φ evaluates to `true`?

via SATISFIABILITY

Input: A formula Φ of boolean variables $a_1, ..., a_n$, with $\neg, \vee$ and $\wedge$.

Output: Does there exist an assigment

$$a_1, ..., a_n \mapsto \{T, F\}$$

such that Φ evaluates to `true`?

$$(a_1 \vee a_2 \vee \neg a_3) \wedge (\neg a_1 \vee \neg a_2) \wedge (a_1 \vee a_3) \longrightarrow \text{SAT solver} \longrightarrow \texttt{True}$$

$$a_1, a_2, a_3 \mapsto T, F, T$$

via SATISFIABILITY

Input: A formula Φ of boolean variables $a_1, ..., a_n$, with $\neg, \vee$ and $\wedge$.

Output: Does there exist an assigment

$$a_1, ..., a_n \mapsto \{T, F\}$$

such that Φ evaluates to `true`?

$$(a_1 \vee a_2 \vee \neg a_3) \wedge (\neg a_1 \vee \neg a_2) \wedge (a_1 \vee a_3) \longrightarrow \text{SAT solver} \longrightarrow \texttt{True}$$

$$a_1, a_2, a_3 \mapsto T, F, T$$

NP-Hard, but due to significant development in the last 20 years, "real world problems with >1,000,000 variables/clauses are OK."[1]

[1] CSE417: Algorithms and Computational Complexity at University of Washington

via SATISFIABILITY

Given a lattice polytope P, we construct a boolean formula Φ_P

via SATISFIABILITY

Given a lattice polytope P, we construct a boolean formula Φ_P

- Enumerate all unimodular simplices $S \subseteq P$, and give them boolean variables
- For S_i, S_j with non-trivial intersection, add the clause

$$\neg S_i \vee \neg S_j$$

- For any simplex S, and F a facet of S interior to P, add the clause

$$\neg S \vee \left(\bigvee S'\right)$$

  where S' iterates over other simplices with F as a facet
- At least one simplex is present:

$$\bigvee S$$

QuickStart Guide

```
julia> using Pkg; Pkg.add(url="https://github.com/krhuang/UniTriSat")

julia> using UniTriSat

julia> P = [0 0 0; 1 0 0; 1 2 0; 0 01; 2 1 -1; 1 2 -1; 2 3 1]

julia> triangulate(P)
    ---------------------------------------
    Run Summary
    ---------------------------------------
    Total Polytopes Processed:     1
    Triangulatable:                1
    Non-Triangulatable:            0
    Total Run Time:                 00:00:00
    ---------------------------------------

julia> triangulate(P, intersection_backend="cpu",unimodular=true,regular=false,
flag_triangulation=false,find_all=false,log_file="",terminal_output="final",
plot=false,use_normaliz=false,return_triangulations="first",solver="cadical",
incremental_solving=false,enable_parallel=true)
```

THANK YOU