

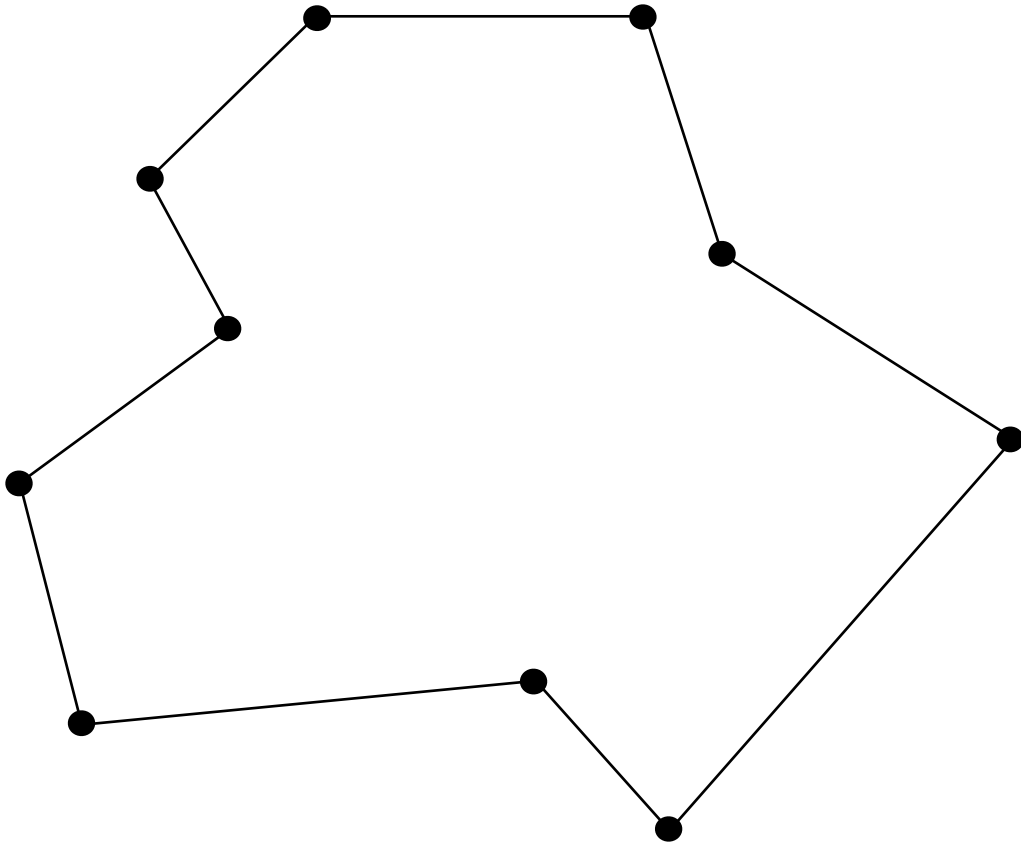
On a Linear Program for Minimum Weight Triangulation

Arman Yousefi and Neal Young

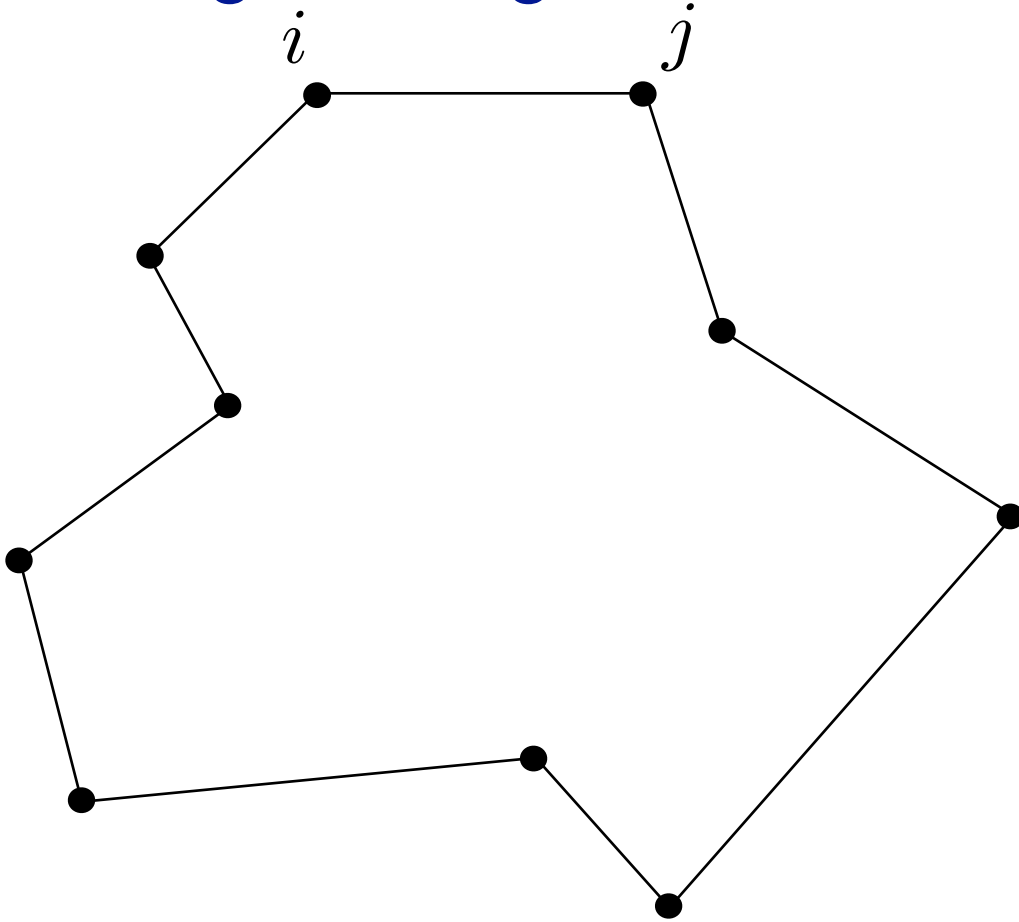
University of California, Riverside

full paper @ SODA 2012 / arxiv.org

min-weight triangulation of a **simple polygon**

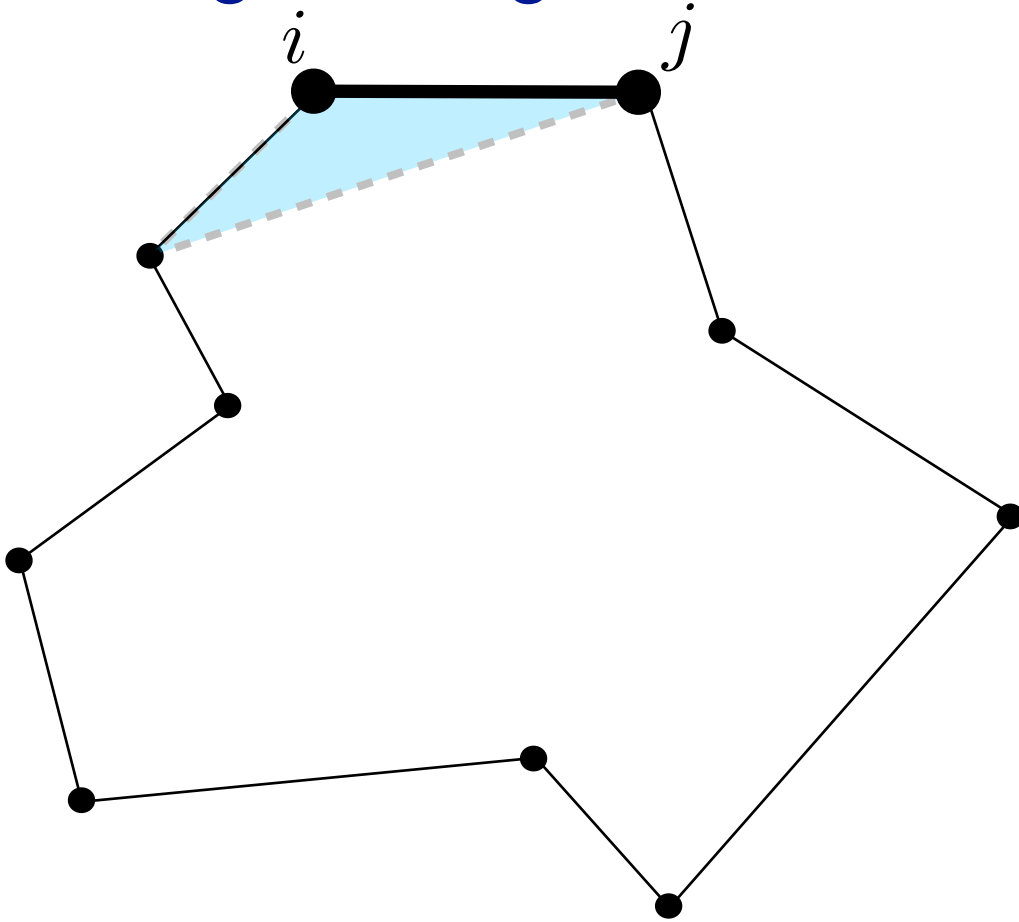


min-weight triangulation of a **simple polygon**



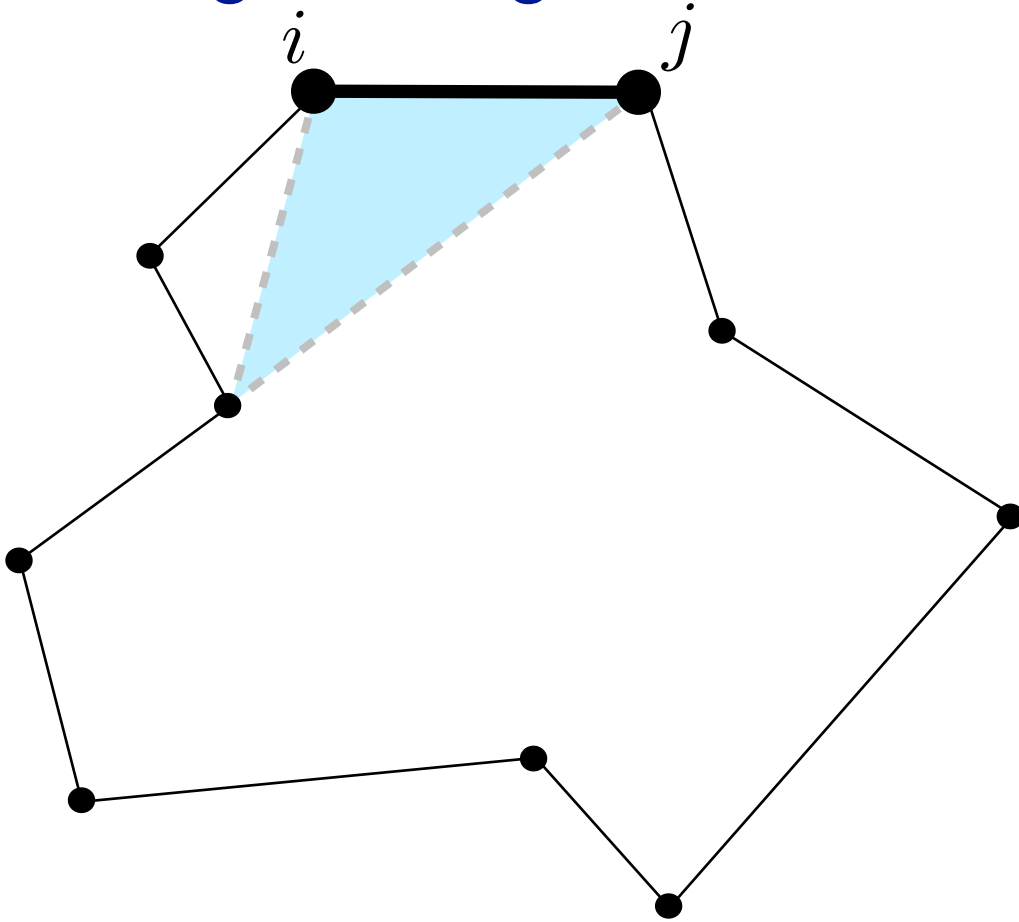
- dynamic programming
- $O(n^3)$ time

min-weight triangulation of a **simple polygon**



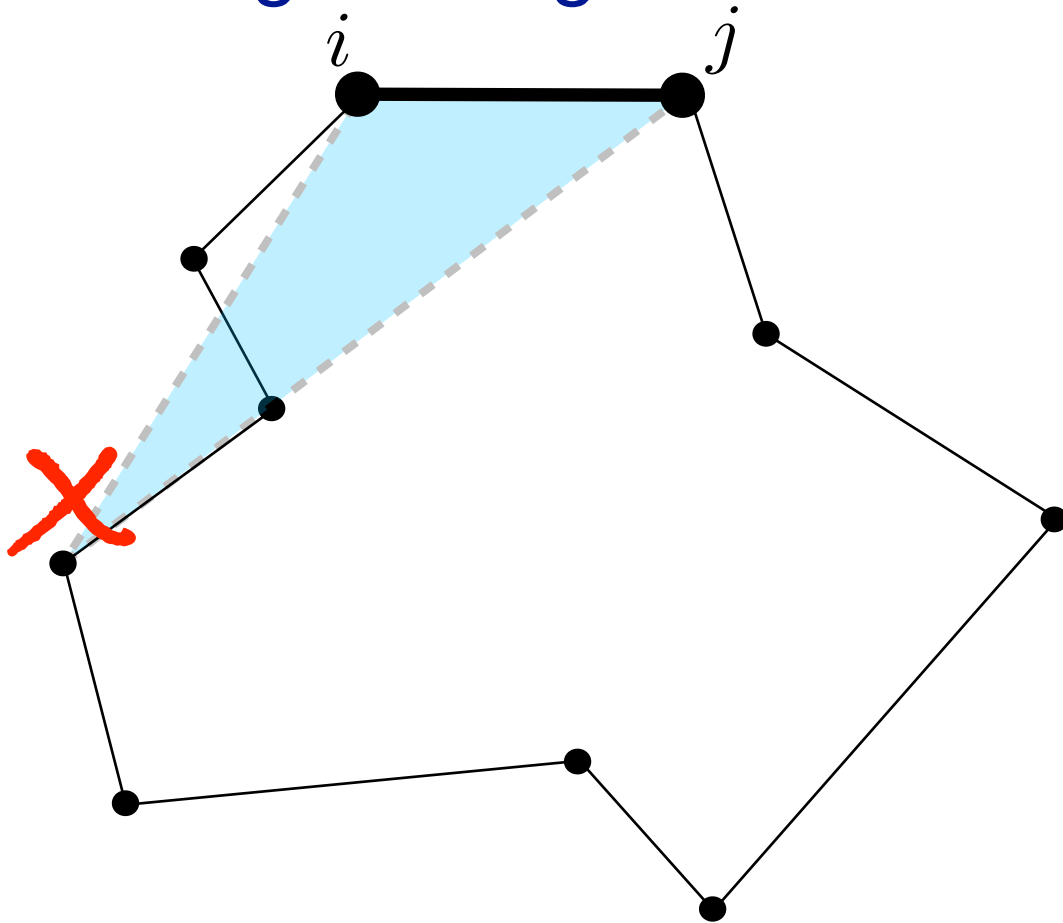
- dynamic programming
- $O(n^3)$ time

min-weight triangulation of a **simple polygon**



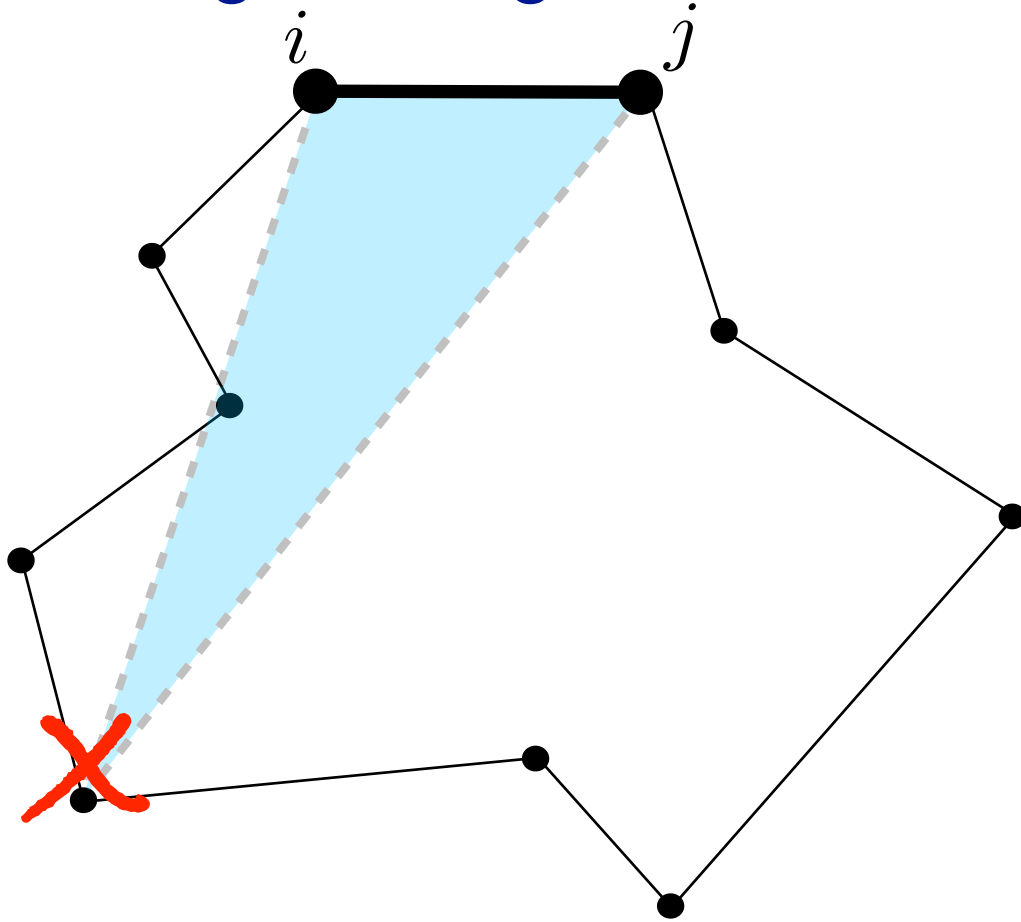
- dynamic programming
- $O(n^3)$ time

min-weight triangulation of a **simple polygon**



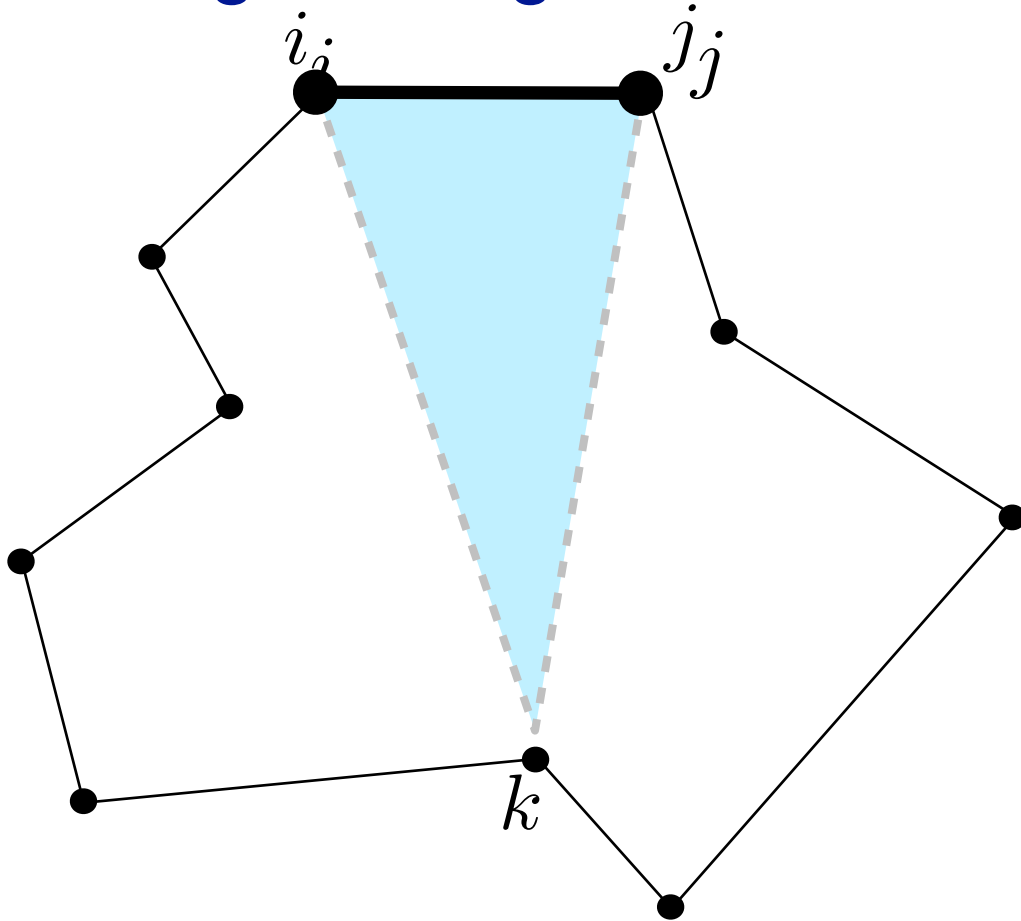
- dynamic programming
- $O(n^3)$ time

min-weight triangulation of a **simple polygon**



- dynamic programming
- $O(n^3)$ time

min-weight triangulation of a **simple polygon**

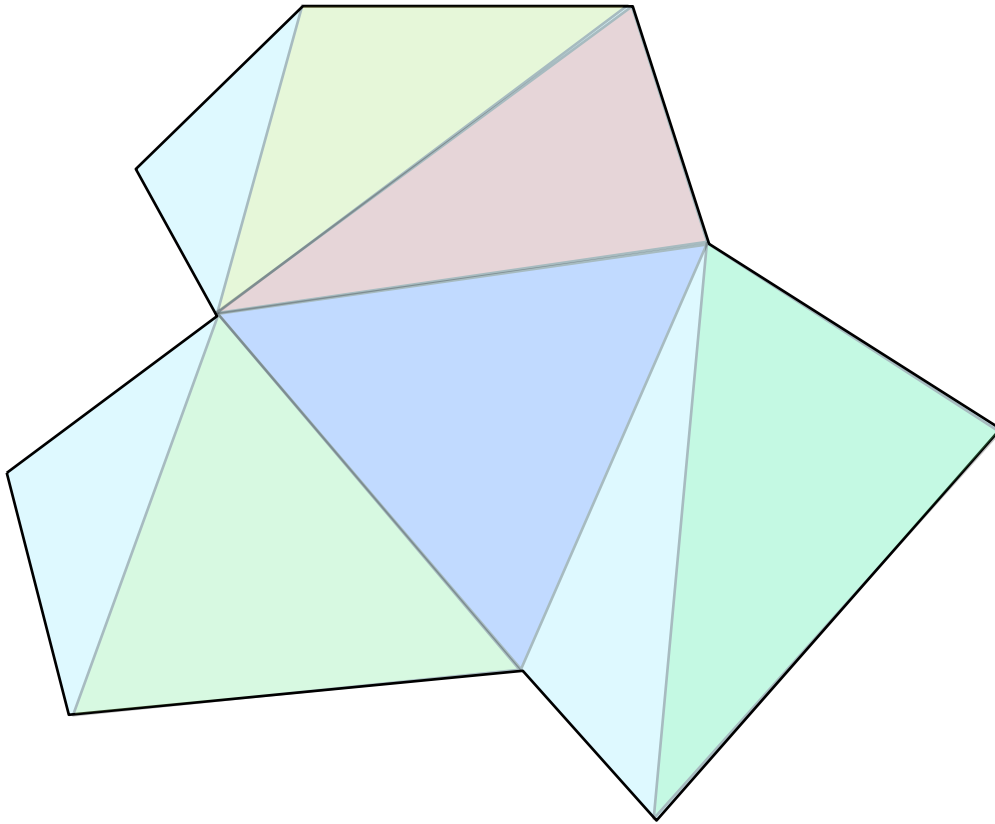


- dynamic programming
- $O(n^3)$ time

$$M[i, j] = \min \left\{ M[i, k] + M[k, j] + d(i, j) \mid i < k < j \right\}$$

$$M[i, i + 1] = d(i, i + 1)$$

min-weight triangulation of a **simple polygon**

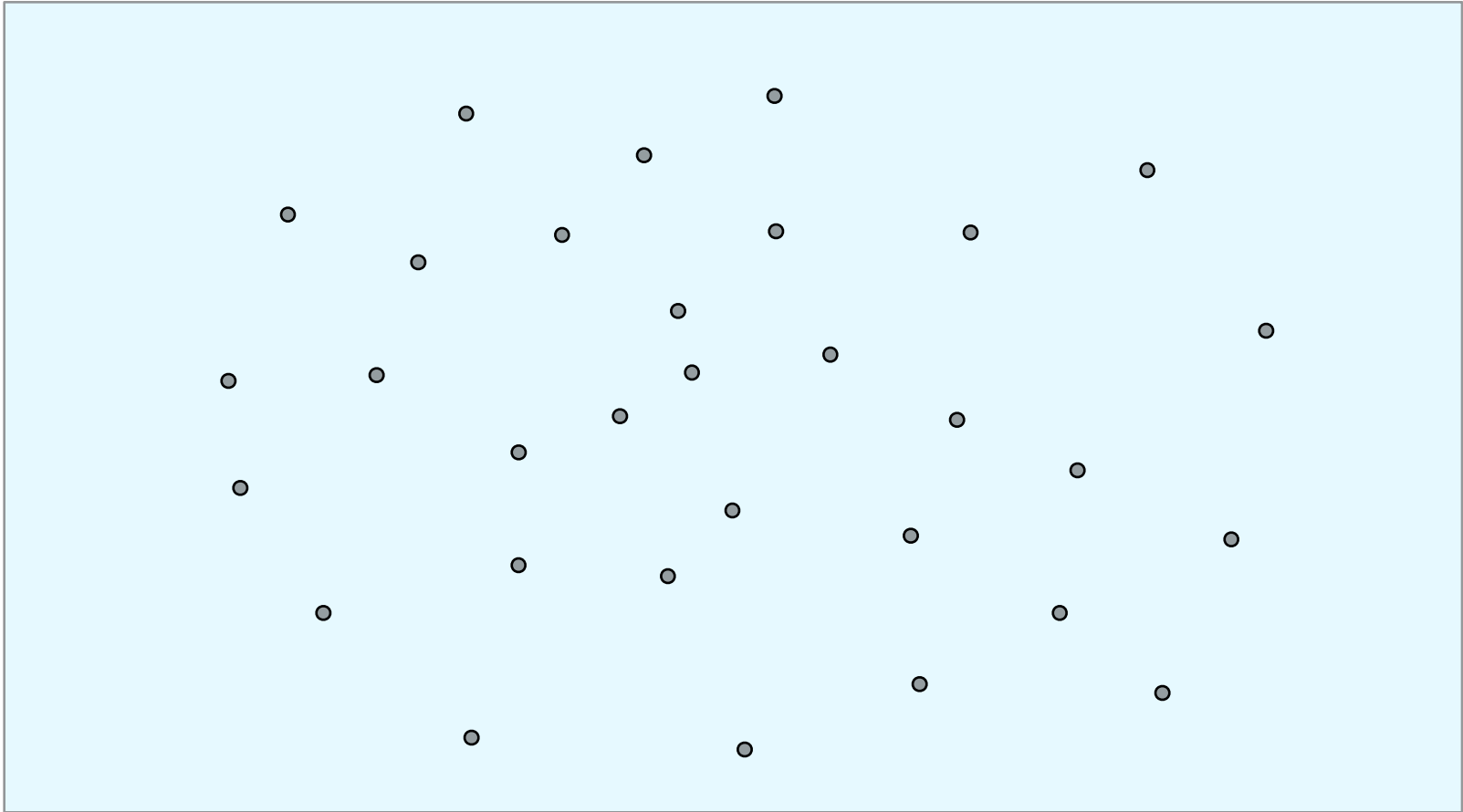


- dynamic programming
- $O(n^3)$ time

- [1979] Gilbert. *New results on planar triangulations.*
- [1980] Klincsek. *Minimal triangulations of polygonal domains.*

minimum weight triangulation (MWT)

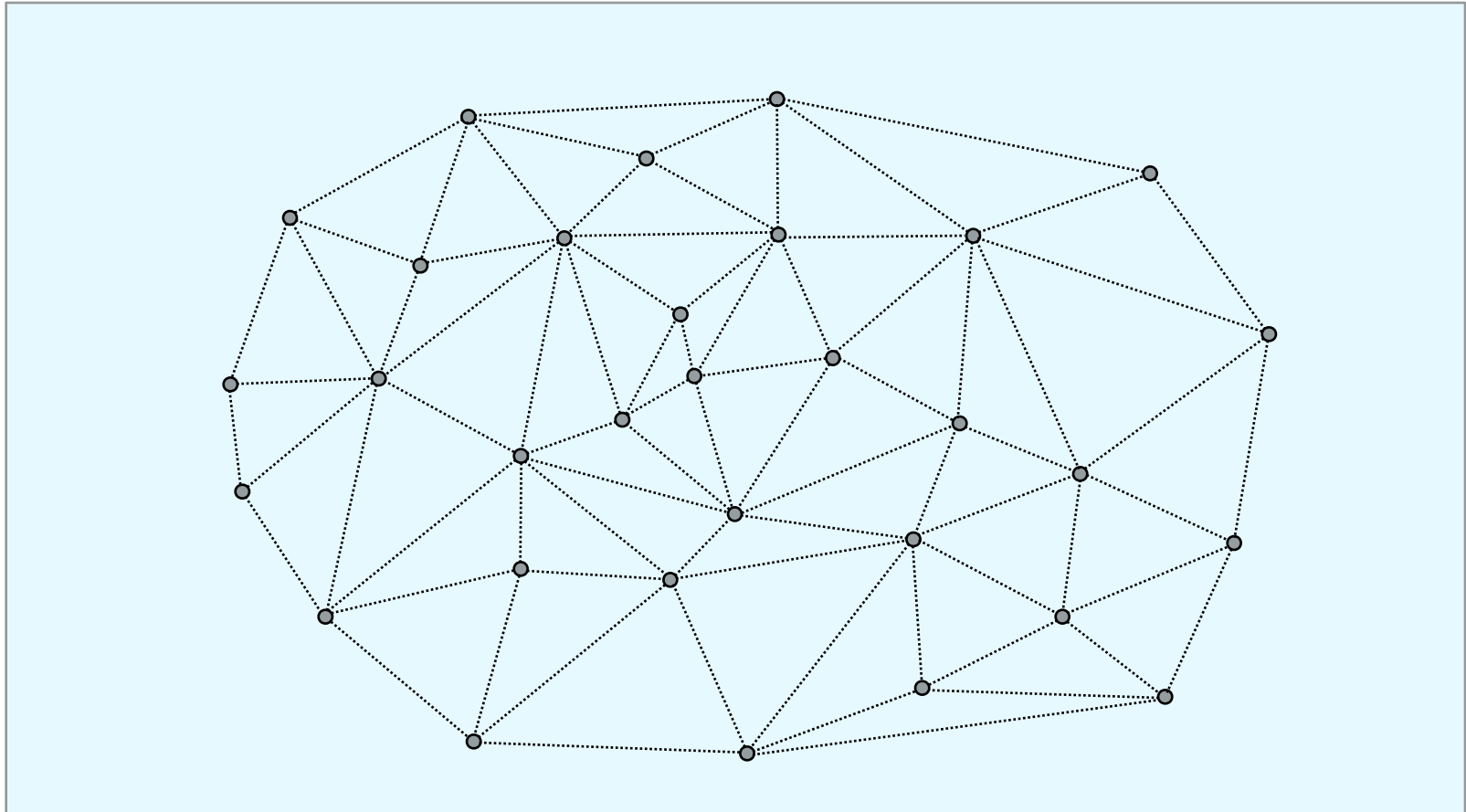
input: a set of points in the plane:



output:

minimum weight triangulation (MWT)

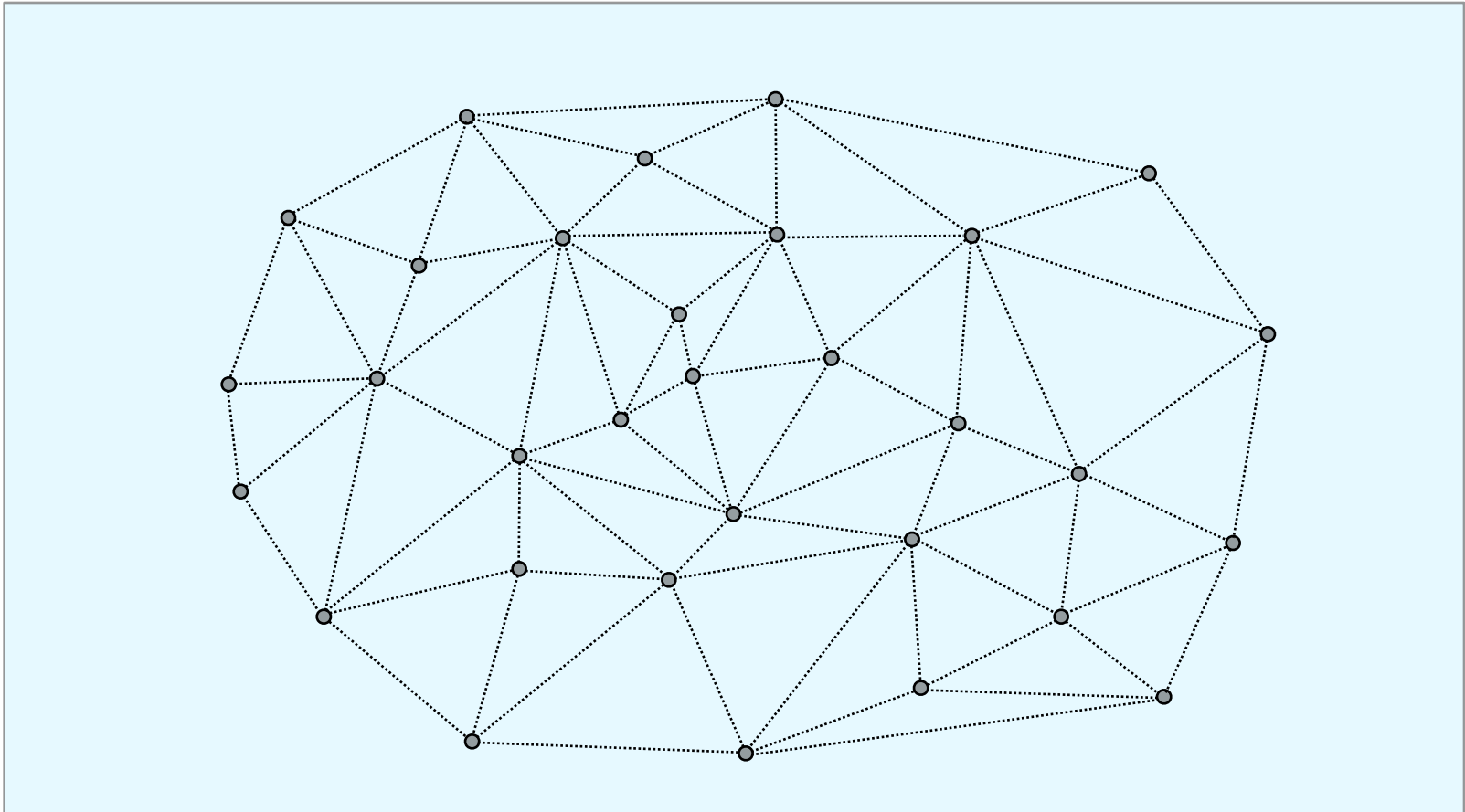
input: a set of points in the plane:



output: a triangulation T

minimum weight triangulation (MWT)

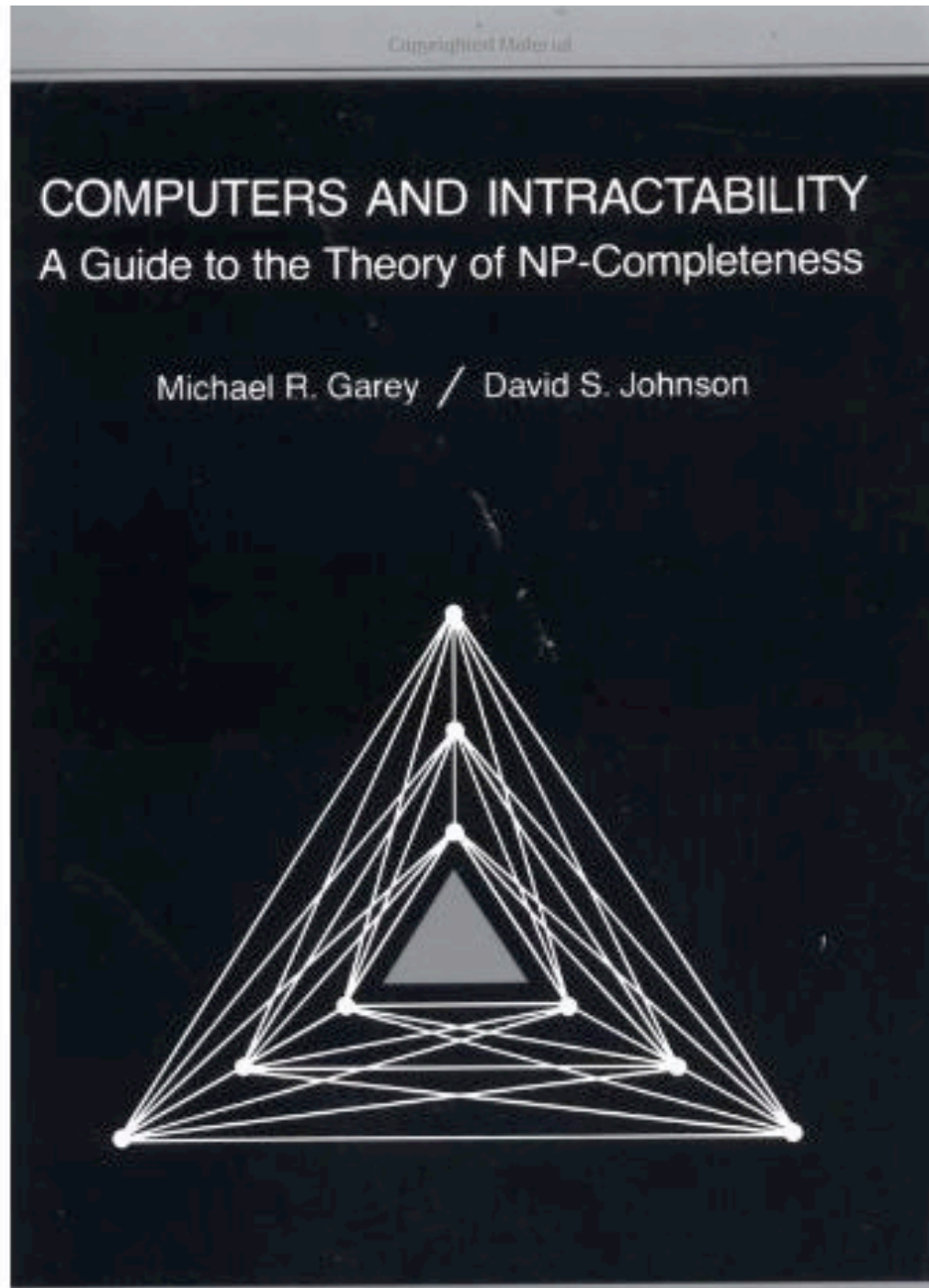
input: a set of points in the plane:



output: a triangulation T of minimum weight, $\sum_{e \in T} |e|$

THE BIBLE (1979)

THE BIBLE (1979)



THE BIBLE (1979)

Contents

Preface	ix
1 Computers, Complexity, and Intractability	1
1.1 Introduction	1
1.2 Problems, Algorithms, and Complexity	4
1.3 Polynomial Time Algorithms and Intractable Problems	6
1.4 Provably Intractable Problems	11
1.5 NP-Complete Problems	13
1.6 An Outline of the Book	14
2 The Theory of NP-Completeness	17
2.1 Decision Problems, Languages, and Encoding Schemes	18
2.2 Deterministic Turing Machines and the Class P	23
2.3 Nondeterministic Computation and the Class NP	27
2.4 The Relationship Between P and NP	32
2.5 Polynomial Transformations and NP-Completeness	34
2.6 Cook's Theorem	38
3 Proving NP-Completeness Results	45
3.1 Six Basic NP-Complete Problems	46
3.1.1 3-SATISFIABILITY	48
3.1.2 3-DIMENSIONAL MATCHING	50
3.1.3 VERTEX COVER and CLIQUE	53
3.1.4 HAMILTONIAN CIRCUIT	56
3.1.5 PARTITION	60
3.2 Some Techniques for Proving NP-Completeness	63
3.2.1 Restriction	63
3.2.2 Local Replacement	66
3.2.3 Component Design	72
3.3 Some Suggested Exercises	74
4 Using NP-Completeness to Analyze Problems	77
4.1 Analyzing Subproblems	80
4.2 Number Problems and Strong NP-Completeness	90
4.2.1 Some Additional Definitions	92
4.2.2 Proving Strong NP-Completeness Results	95
4.3 Time Complexity as a Function of Natural Parameters	106
5 NP-Hardness	109
5.1 Turing Reducibility and NP-Hard Problems	109
5.2 A Terminological History	118
6 Coping with NP-Complete Problems	121
6.1 Performance Guarantees for Approximation Algorithms	123
6.2 Applying NP-Completeness to Approximation Problems	137
6.3 Performance Guarantees and Behavior "In Practice"	148
7 Beyond NP-Completeness	153
7.1 The Structure of NP	154
7.2 The Polynomial Hierarchy	161
7.3 The Complexity of Enumeration Problems	167
7.4 Polynomial Space Completeness	170
7.5 Logarithmic Space	177
7.6 Proofs of Intractability and P vs. NP	181
Appendix: A List of NP-Complete Problems	187
A1 Graph Theory	190
A1.1 Covering and Partitioning	190
A1.2 Subgraphs and Supergraphs	194
A1.3 Vertex Ordering	199
A1.4 Iso- and Other Morphisms	202
A1.5 Miscellaneous	203
A2 Network Design	206
A2.1 Spanning Trees	206
A2.2 Cuts and Connectivity	209
A2.3 Routing Problems	211
A2.4 Flow Problems	214
A2.5 Miscellaneous	218
A3 Sets and Partitions	221
A3.1 Covering, Hitting, and Splitting	221
A3.2 Weighted Set Problems	223
A4 Storage and Retrieval	226
A4.1 Data Storage	226
A4.2 Compression and Representation	228
A4.3 Database Problems	232

THE BIBLE (1979)

CONTENTS

vii

A5	Sequencing and Scheduling	236
A5.1	Sequencing on One Processor	236
A5.2	Multiprocessor Scheduling	238
A5.3	Shop Scheduling	241
A5.4	Miscellaneous	243
A6	Mathematical Programming	245
A7	Algebra and Number Theory	249
A7.1	Divisibility Problems	249
A7.2	Solvability of Equations	250
A7.3	Miscellaneous	252
A8	Games and Puzzles	254
A9	Logic	259
A9.1	Propositional Logic	259
A9.2	Miscellaneous	261
A10	Automata and Language Theory	265
A10.1	Automata Theory	265
A10.2	Formal Languages	267
A11	Program Optimization	272
A11.1	Code Generation	272
A11.2	Programs and Schemes	275
A12	Miscellaneous	279
A13	Open Problems	285
	Symbol Index	289
	Reference and Author Index	291
	Subject Index	327
	Update for the Current Printing	339

**MWT
NP-Hard?
In P?**



approximation algorithms

- [1987] Plaisted and Hong. O(log n)-approx
A heuristic triangulation algorithm.
- ★ [1996] Levcopoulos and Krznaric. O(1)-approx
Quasi-greedy triangulations approximating the minimum weight triangulation.
- [2006] Remy and Steger. QPTAS
A quasi-polynomial time approximation scheme for minimum weight triangulation.

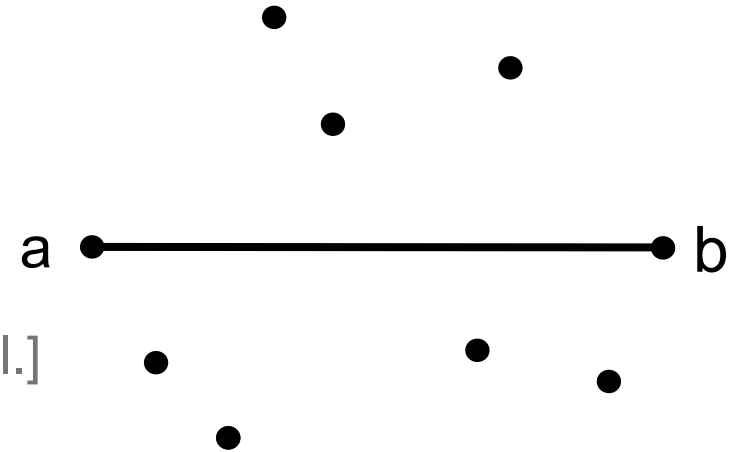
hardness result

- [2006] Mulzer and Rote. 25 years after G+J! NP-HARD
Minimum weight triangulation is NP-hard.

heuristics!

- edges that can't be in any MWT:
 - diamond test

[1989 Das and Joseph; 2001 Drysdale et al.]

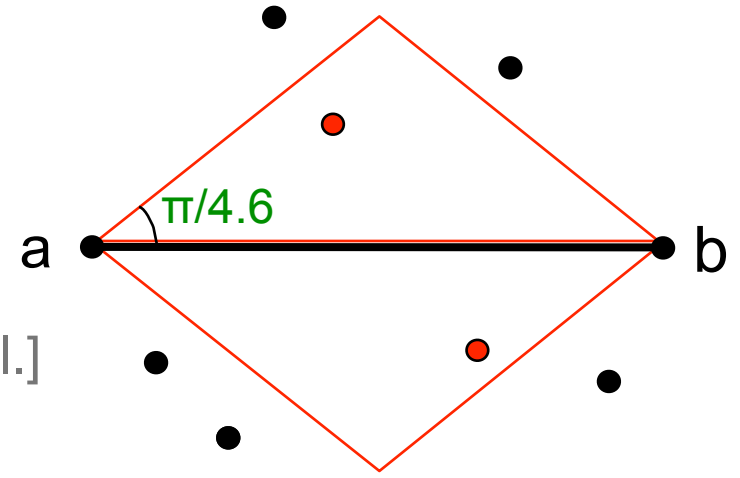


heuristics!

- edges that can't be in any MWT:

- diamond test

[1989 Das and Joseph; 2001 Drysdale et al.]

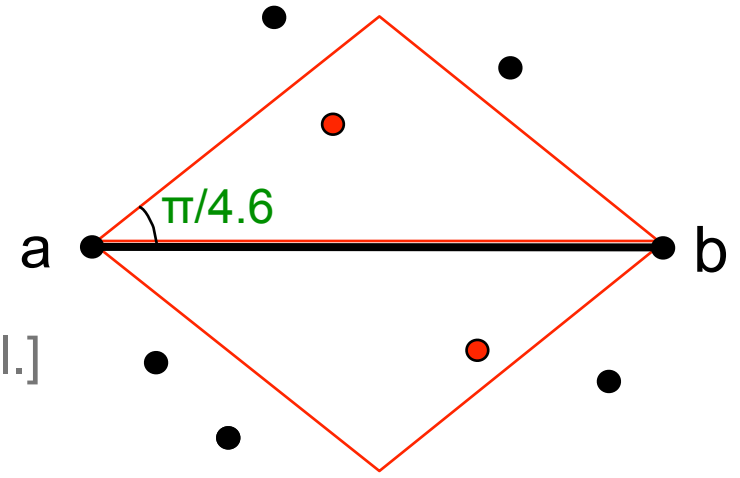


heuristics!

- edges that *can't* be in any MWT:

- **diamond test**

[1989 Das and Joseph; 2001 Drysdale et al.]



- edges that *have to be* in every MWT:

- **mutual nearest neighbors**

[1979 Gilbert; 1994 Yang et al]

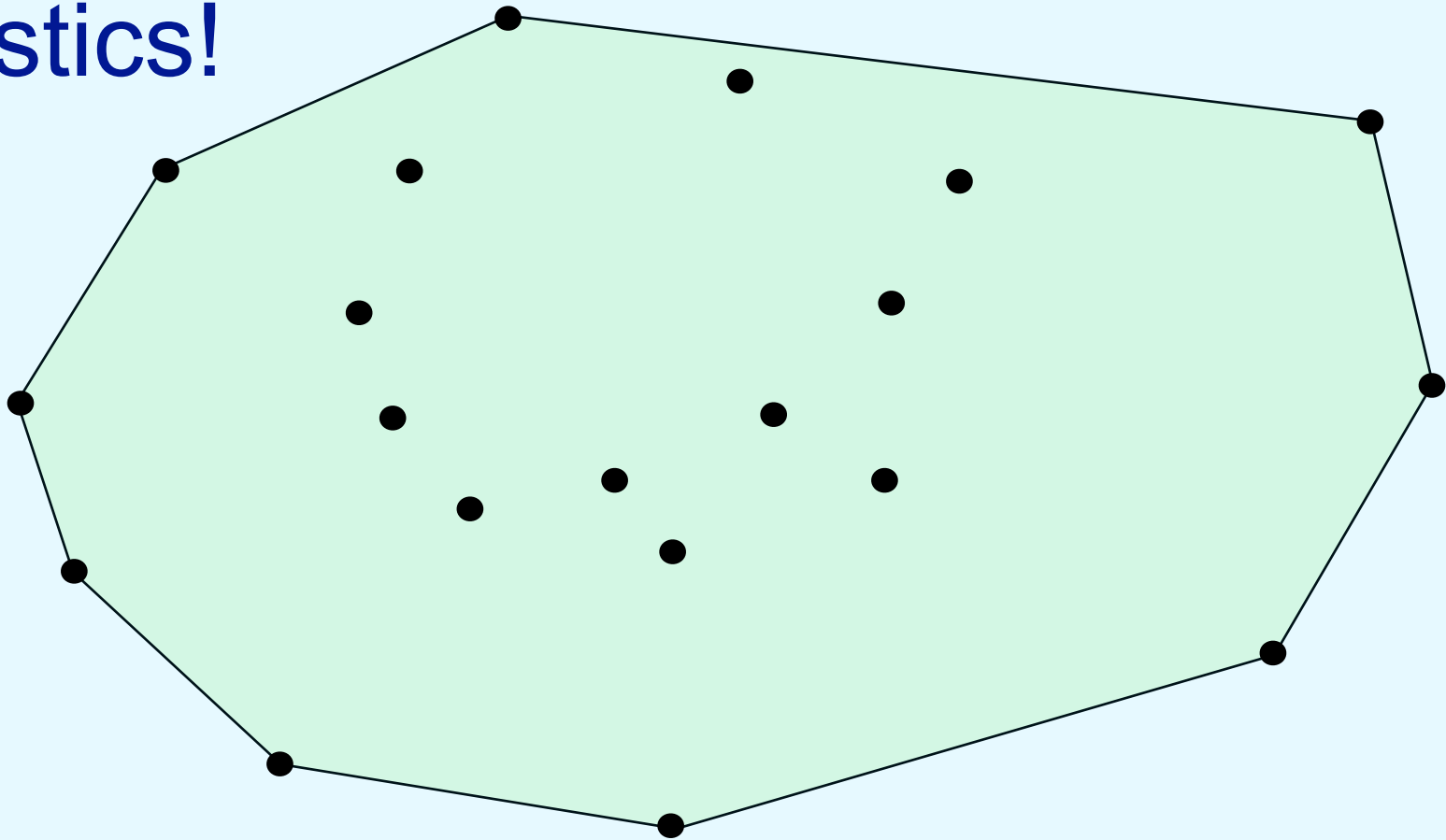
- **β -skeleton**

[1993 Keil; 1995 Yang; 1996 Cheng and Xu]

- **locally minimal triangulation (“LMT-skeleton”)**

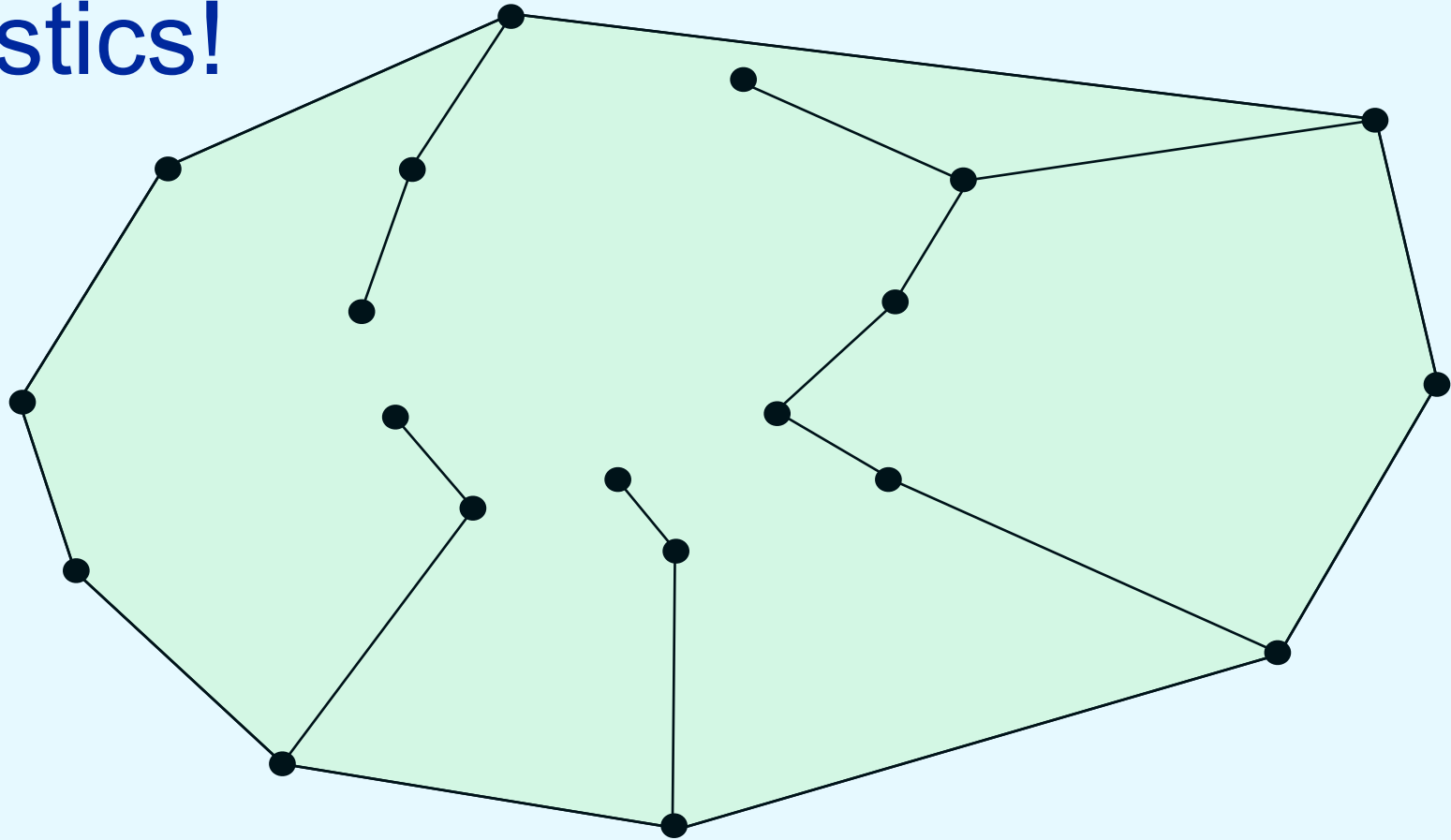
[1997 Dickerson et al; 1998 Beirouti and Snoeyink; 1996 Cheng et al;
1999 Aichholzer et al; 1996 Belleville et al; 2002 Bose et al]

heuristics!



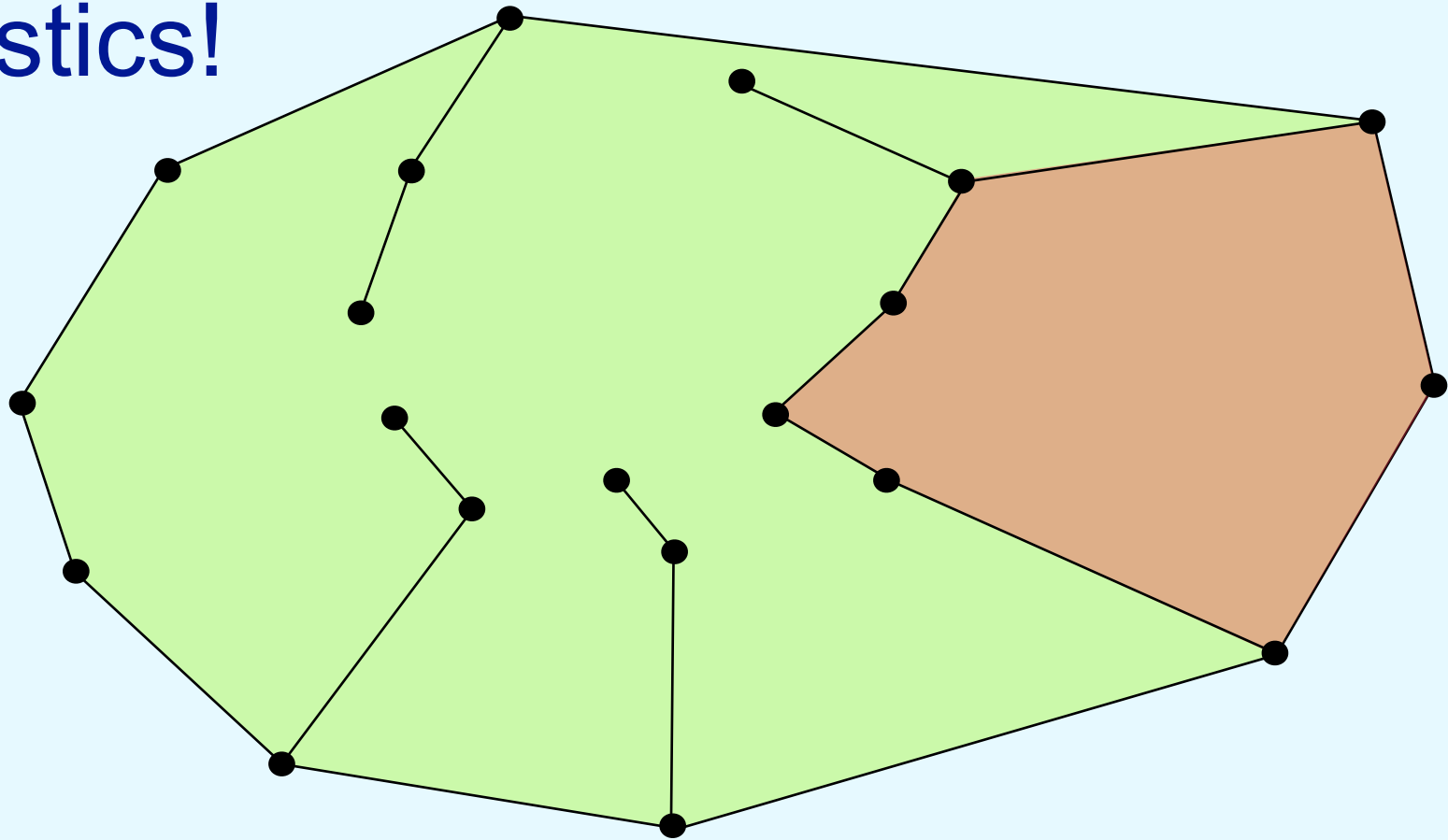
1. The boundary edges have to be in the MWT.

heuristics!



2. Use the heuristics to find more edges that have to be in the MWT.

heuristics!

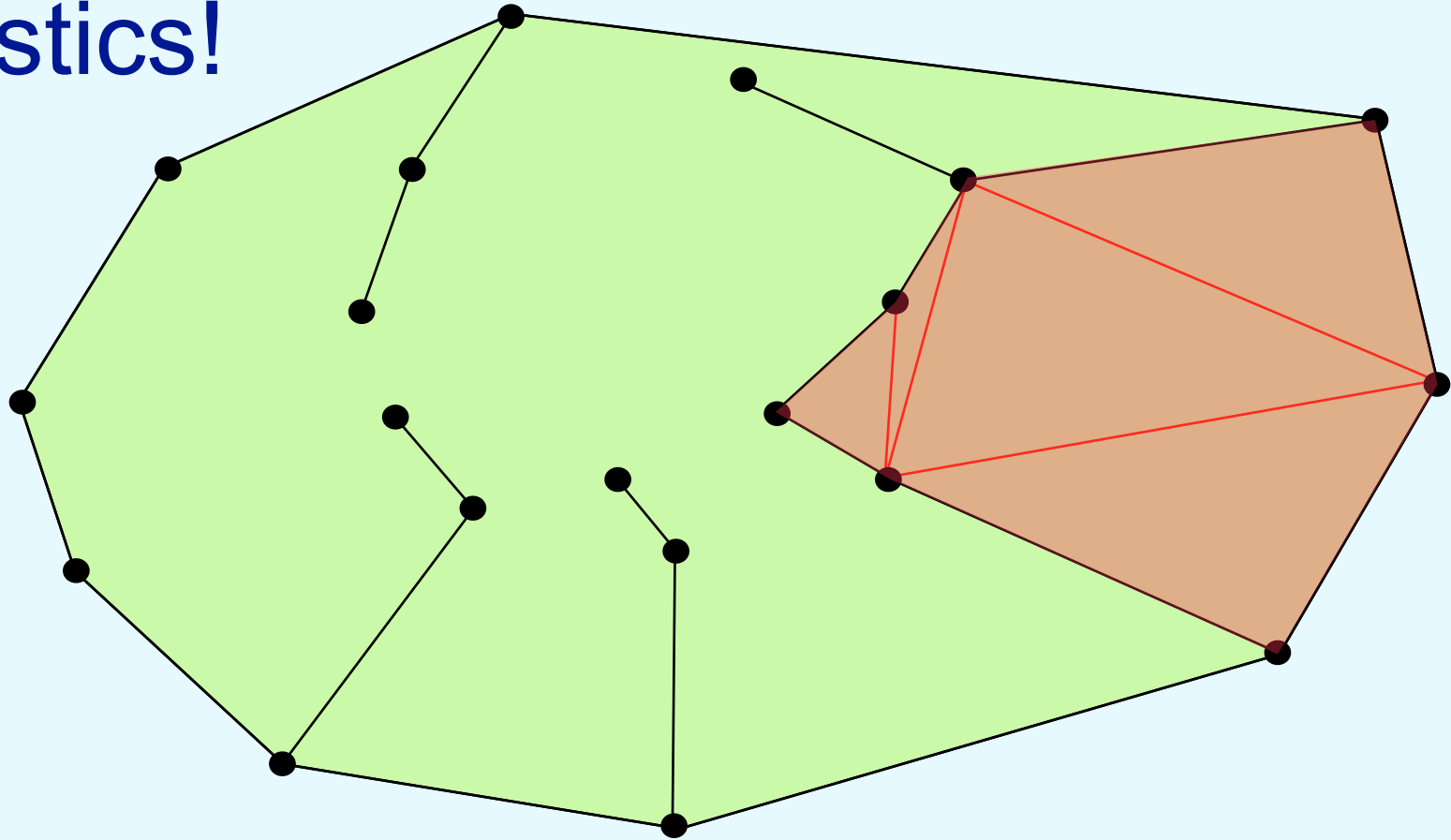


if you're lucky...

found edges connect all points to boundary.

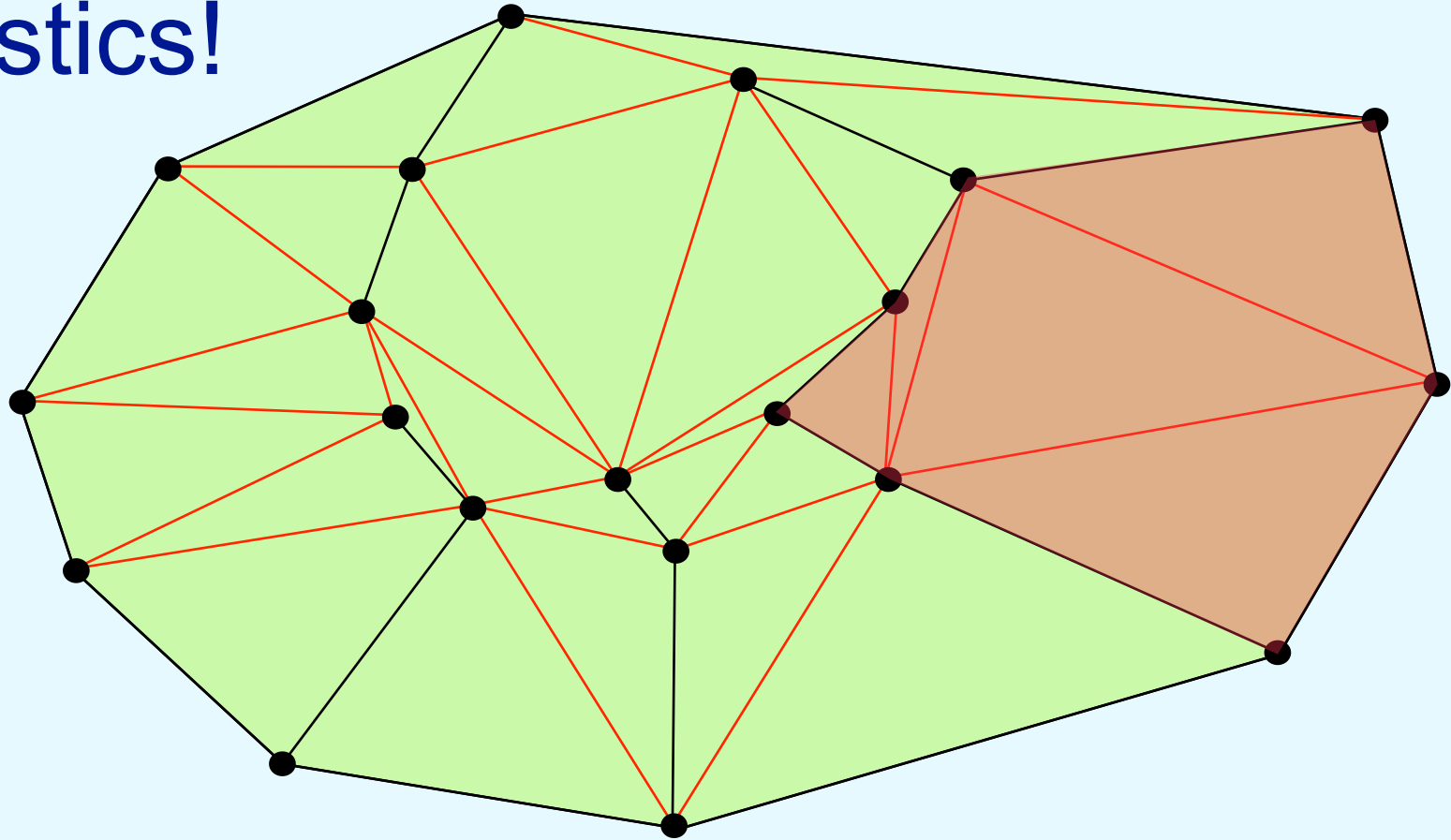
Then remaining regions are simple polygons.

heuristics!



3. Triangulate each remaining region optimally using the dynamic-programming algorithm.

heuristics!



3. Triangulate each remaining region optimally using the dynamic-programming algorithm.

heuristics

- This approach solves most random 40,000-point instances. [Dickerson et al. '97]
- But.. for random instances, heuristics leave (in expectation) $\Omega(n)$ internal components (but hidden constant is astronomically small, 10^{-51}).

[Bose et al. '02]

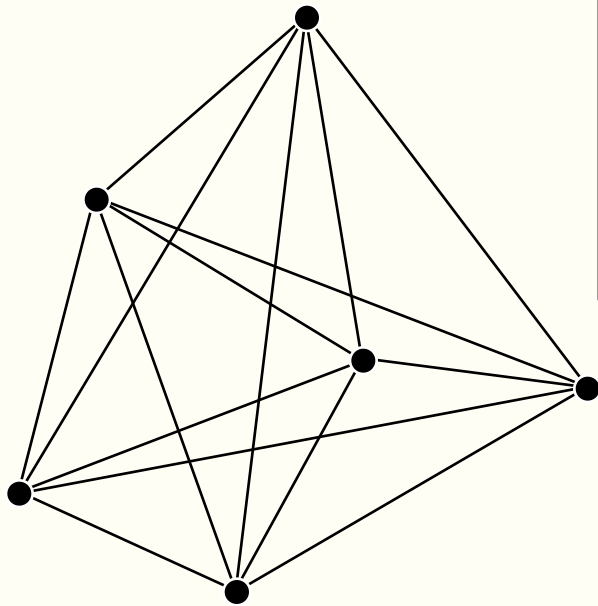
linear programs for MWT

- [1985] Dantzig et al.
Triangulations (tilings) and certain block triangular matrices.
- Subsequently studied in [1996 Loera et al; 2004 Kirsanov, etc...]

edge-based linear programs:

- [1997] Kyoda et al.
A branch-and-cut approach for minimum weight triangulation.
- [1996] Kyoda.
A study of generating minimum weight triangulation within practical time.
- [1996] Ono et al. *A package for triangulations.*
- [1998] Tajima. *Optimality and integer programming formulations of triangulations in general dimension.*
- [2000] Aurenhammer and Xu. *Optimal triangulations.*

Dantzig et al's triangle-based LP [1985]



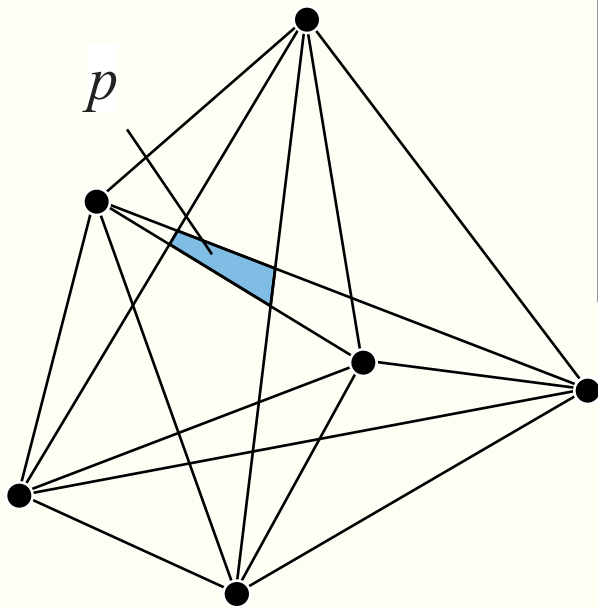
minimize $\sum_{\text{triangles } t} |t| X_t$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

Dantzig et al's triangle-based LP [1985]



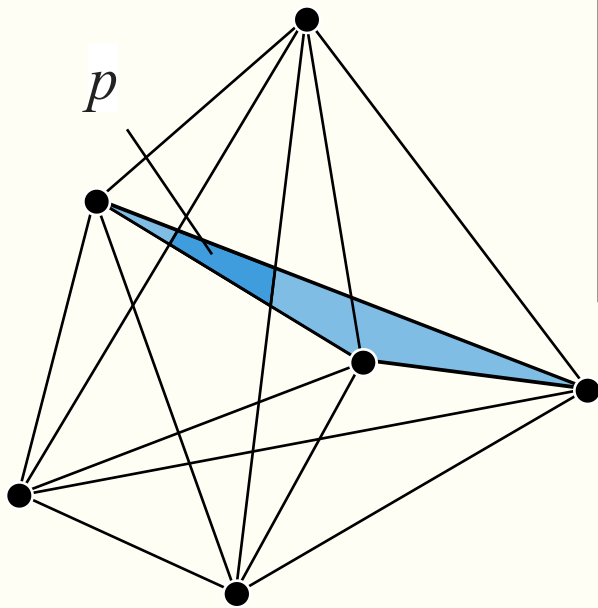
minimize $\sum_{\text{triangles } t} |t| X_t$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

Dantzig et al's triangle-based LP [1985]



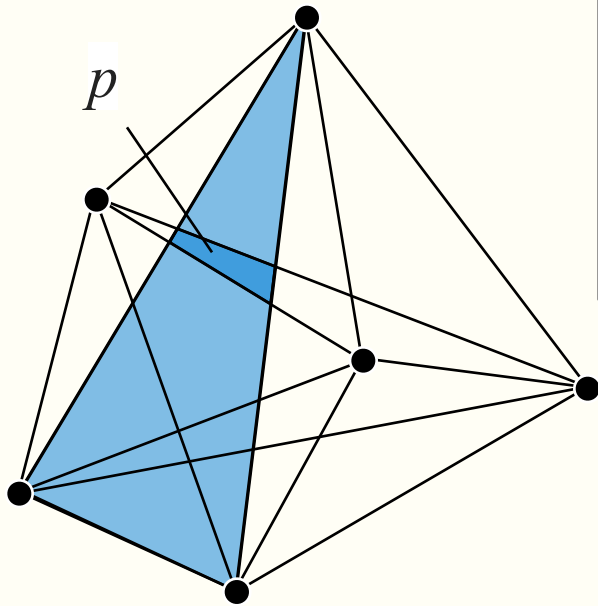
$$\text{minimize } \sum_{\text{triangles } t} |t| X_t$$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

Dantzig et al's triangle-based LP [1985]



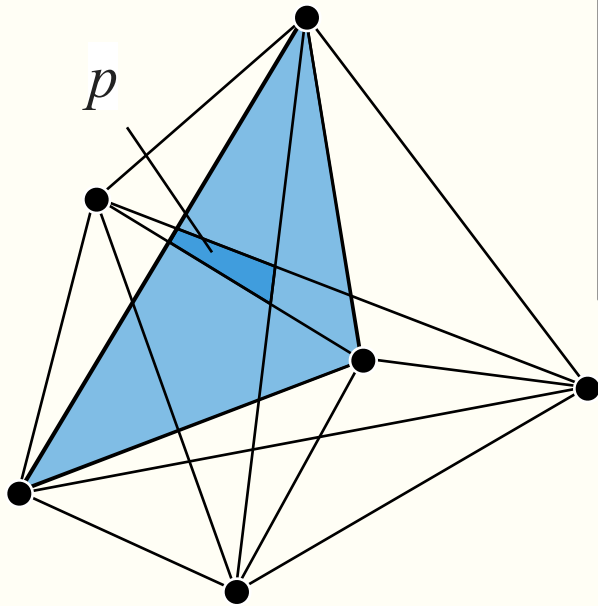
minimize $\sum_{\text{triangles } t} |t| X_t$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

Dantzig et al's triangle-based LP [1985]



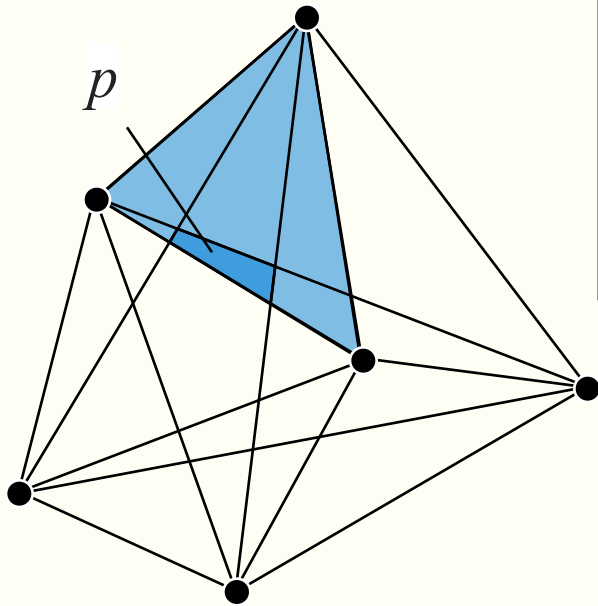
minimize $\sum_{\text{triangles } t} |t| X_t$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

Dantzig et al's triangle-based LP [1985]



$$\text{minimize } \sum_{\text{triangles } t} |t| X_t$$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

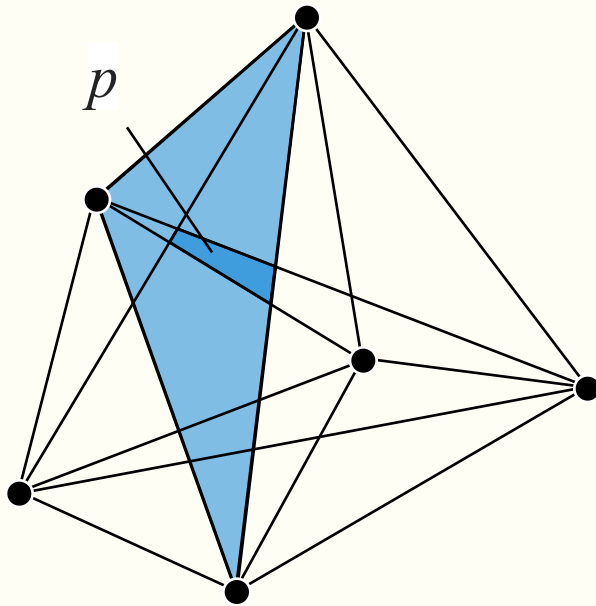
Dantzig et al's triangle-based LP [1985]

$$\text{minimize } \sum_{\text{triangles } t} |t| X_t$$

subject to

$$\sum_{t \ni p} X_t = 1 \quad (\forall \text{ points } p)$$

$$X_t \geq 0 \quad (\forall \text{ triangles } t)$$

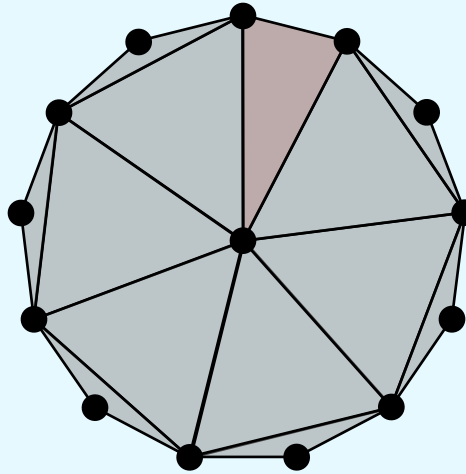


Exactly one of these triangles must be in the triangulation.

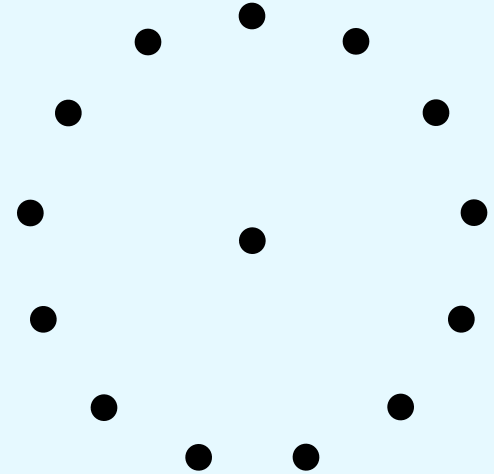
“Exact cover by triangles.”

Integer vs. fractional MWT

Integer MWT

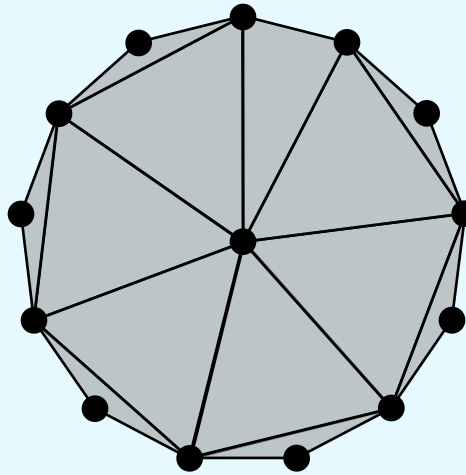


Fractional MWT
Each triangle has weight 1/2

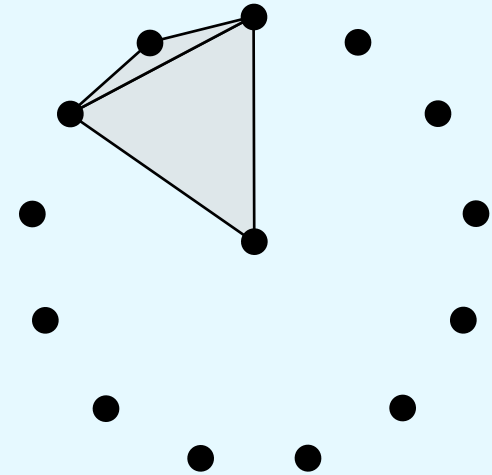


Integer vs. fractional MWT

Integer MWT

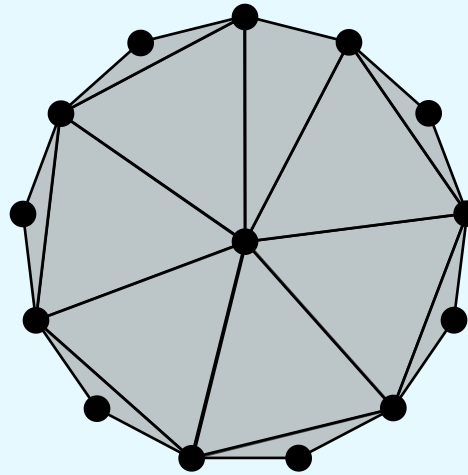


Fractional MWT
Each triangle has weight $1/2$

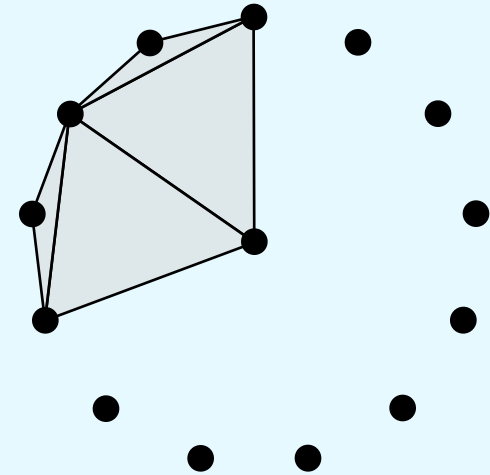


Integer vs. fractional MWT

Integer MWT

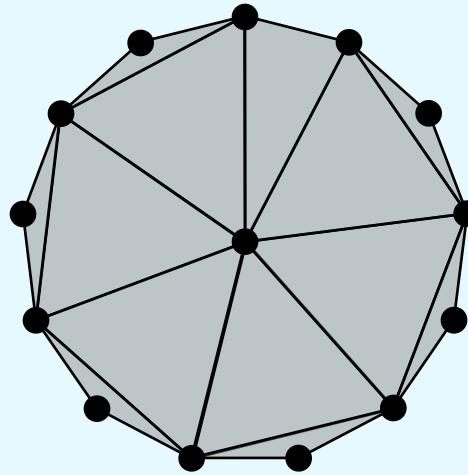


Fractional MWT
Each triangle has weight 1/2

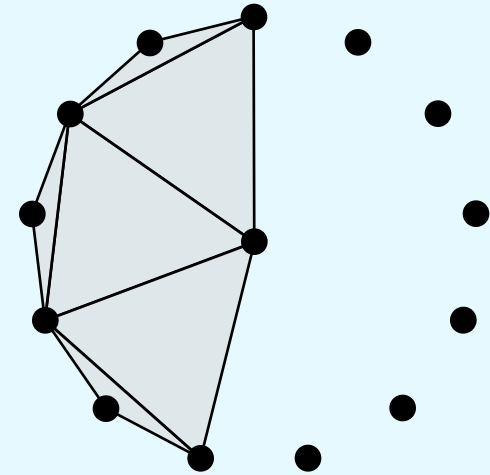


Integer vs. fractional MWT

Integer MWT

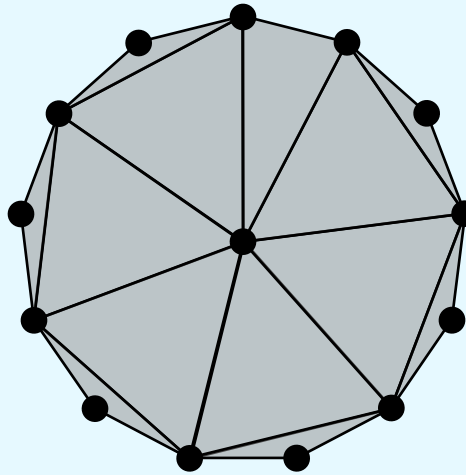


Fractional MWT
Each triangle has weight $1/2$

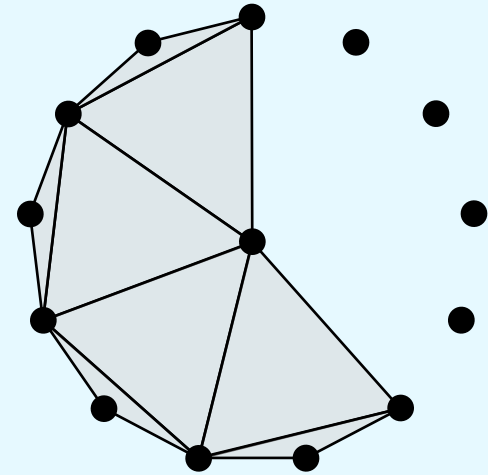


Integer vs. fractional MWT

Integer MWT

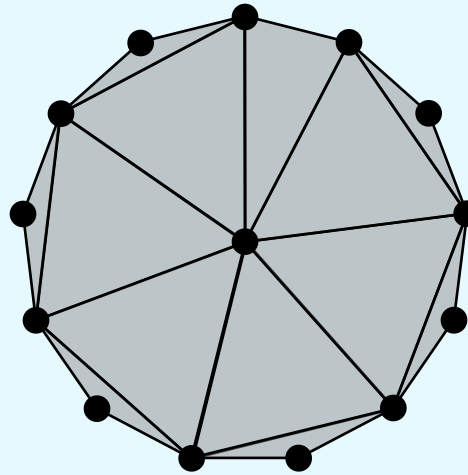


Fractional MWT
Each triangle has weight $1/2$

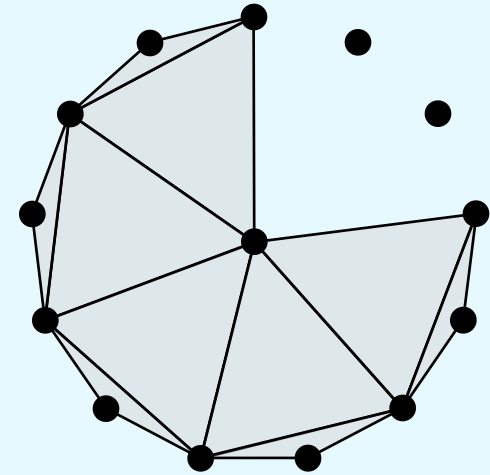


Integer vs. fractional MWT

Integer MWT

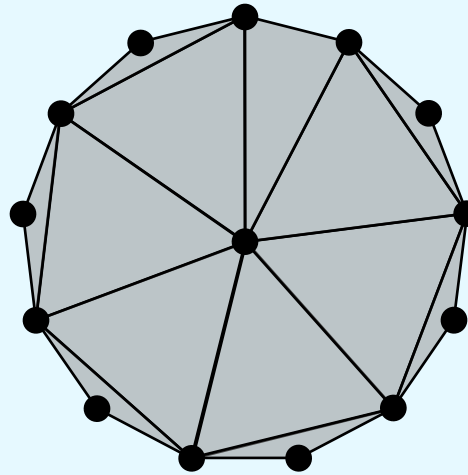


Fractional MWT
Each triangle has weight $1/2$

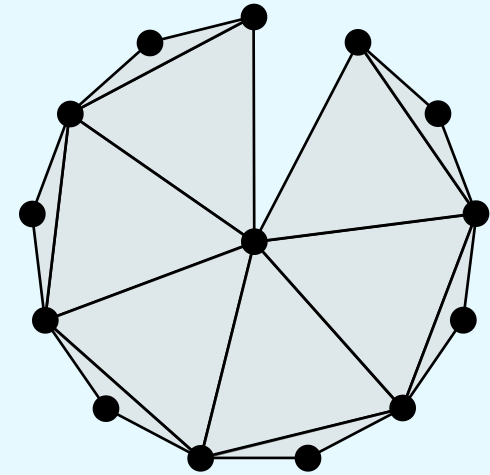


Integer vs. fractional MWT

Integer MWT

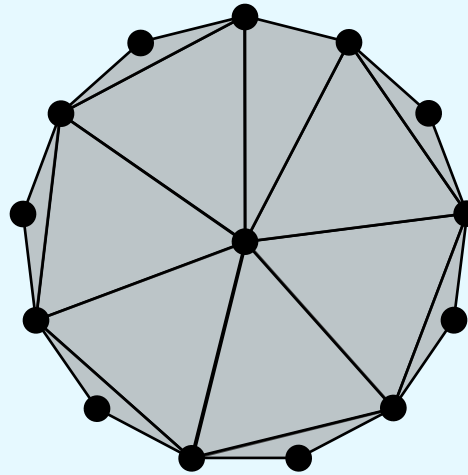


Fractional MWT
Each triangle has weight 1/2

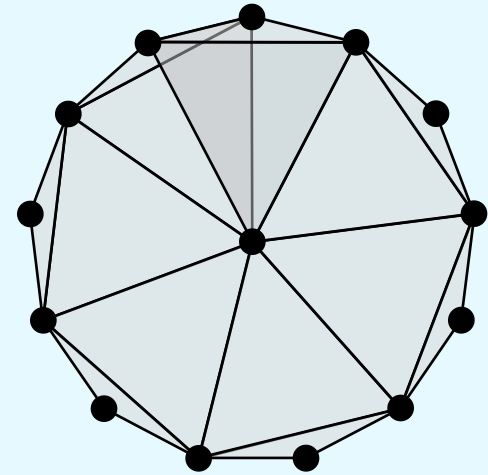


Integer vs. fractional MWT

Integer MWT

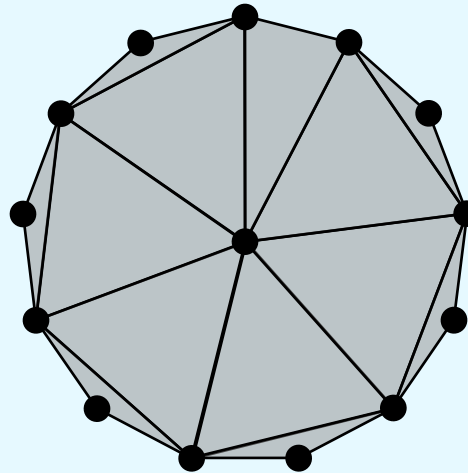


Fractional MWT
Each triangle has weight 1/2

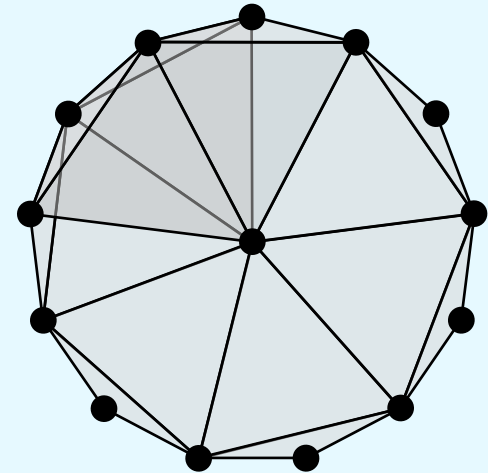


Integer vs. fractional MWT

Integer MWT

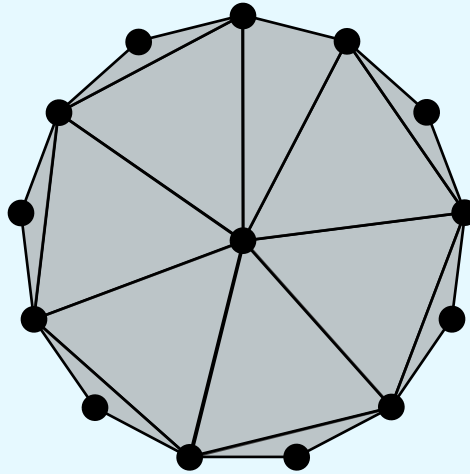


Fractional MWT
Each triangle has weight 1/2

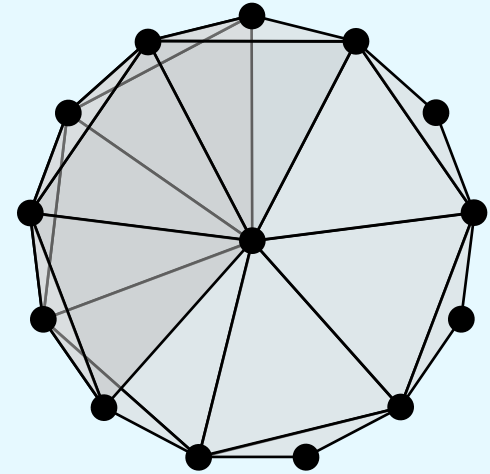


Integer vs. fractional MWT

Integer MWT

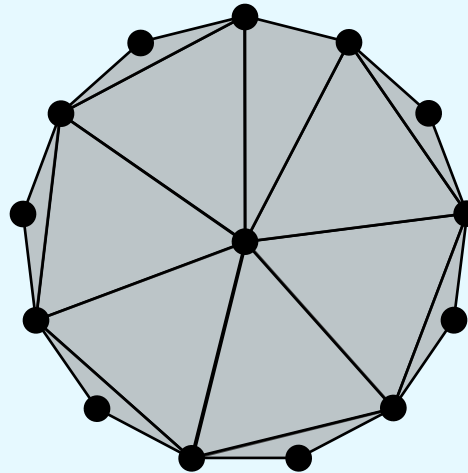


Fractional MWT
Each triangle has weight 1/2

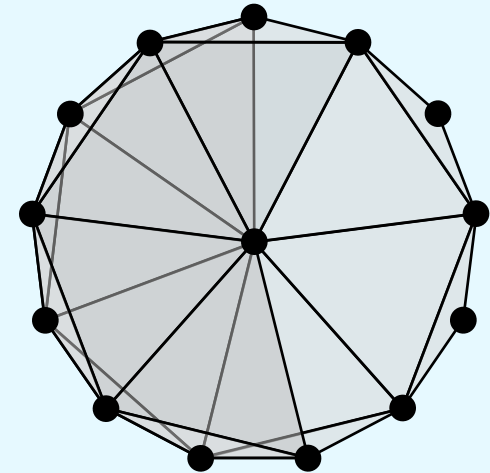


Integer vs. fractional MWT

Integer MWT

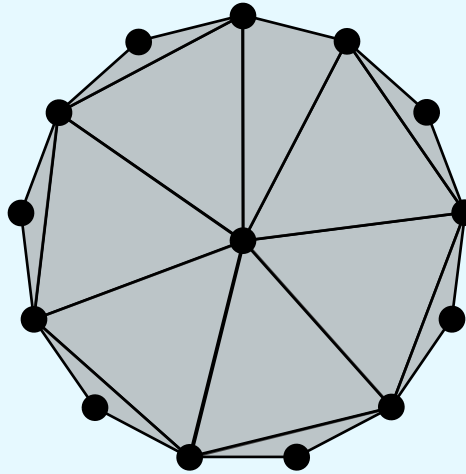


Fractional MWT
Each triangle has weight 1/2

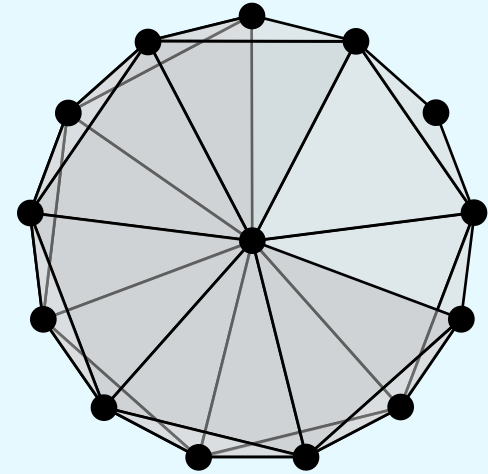


Integer vs. fractional MWT

Integer MWT

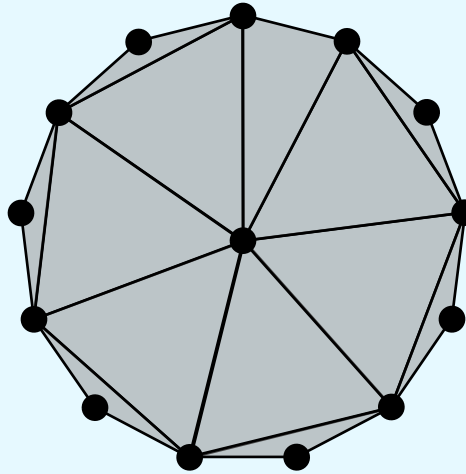


Fractional MWT
Each triangle has weight 1/2

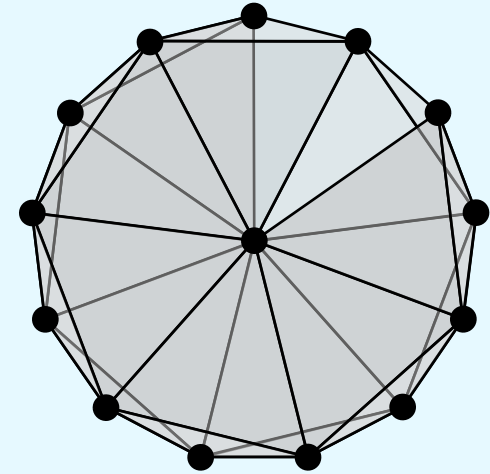


Integer vs. fractional MWT

Integer MWT

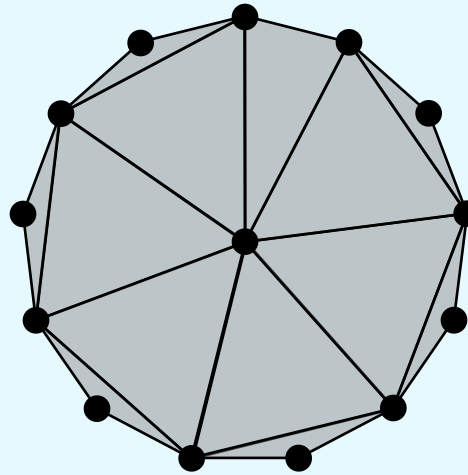


Fractional MWT
Each triangle has weight 1/2

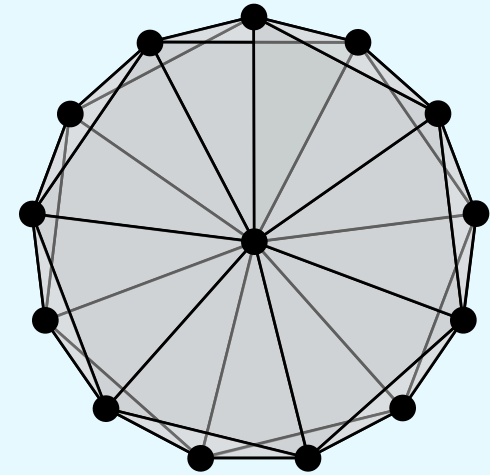


Integer vs. fractional MWT

Integer MWT



Fractional MWT
Each triangle has weight $1/2$



Ratio of costs is about 1.001

known results

- The integrality gap is at least 1.001 [2004 Kirsanov]
- For **simple-polygon** instances, the LP finds the MWT.
[1985 Dantzig et al; 1996 Loera et al; 2004 Kirsanov; etc]

first new result

THM 1: *The integrality gap of the LP is constant.*

first new result

THM 1: *The integrality gap of the LP is constant.*

proof idea:

As Levcopoulos and Krznaric [1996] show, their algorithm produces triangulation T of cost at most

$O(1)$ times the MWT (optimal integer solution).

We show that their triangulation T has cost at most

$O(1)$ times the optimal ***fractional*** LP solution.

second new result

THM 2: *If the heuristics find the MWT for a given instance, then so does the LP.*

second new result

THM 2: *If the heuristics find the MWT for a given instance, then so does the LP.*

proof idea:

If a heuristic shows that an edge is not in any MWT, we show that the optimal *fractional* triangulation cannot use the edge either.

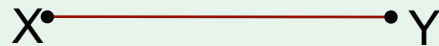
If a heuristic shows that an edge is in every MWT, we show that the optimal *fractional* triangulation must use the edge fully as well.

Requires painstakingly adapting each analysis.

example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

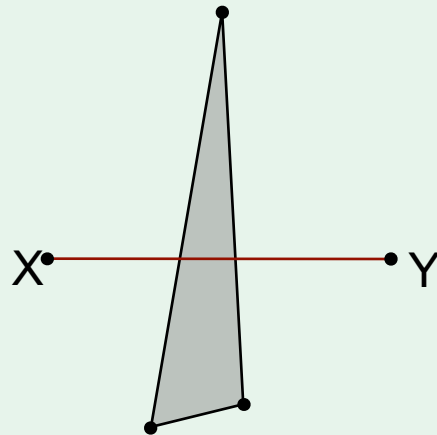


example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

Then some triangle in the triangulation must cross (x,y) :



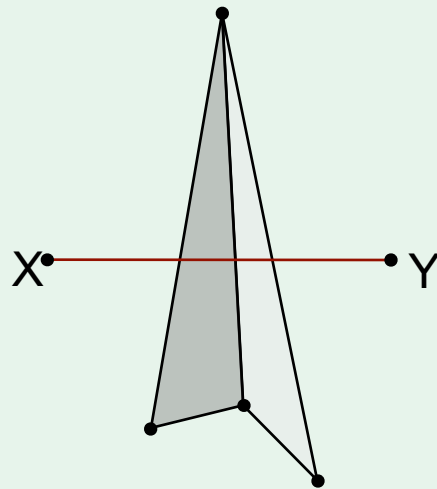
example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

Then some triangle in the triangulation must cross (x,y) .

The triangulation must extend this triangle on each side:



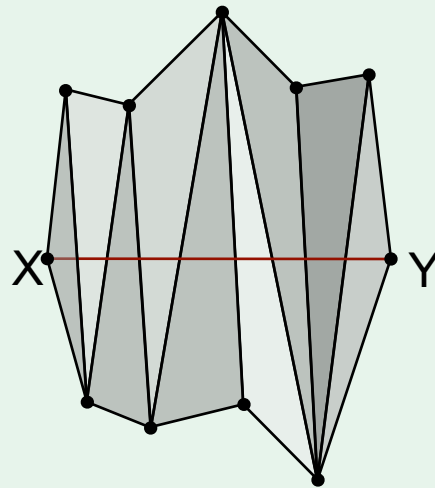
example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

Then some triangle in the triangulation must cross (x,y) .

Continuing, the triangulation covers (x,y) **locally** something like this:

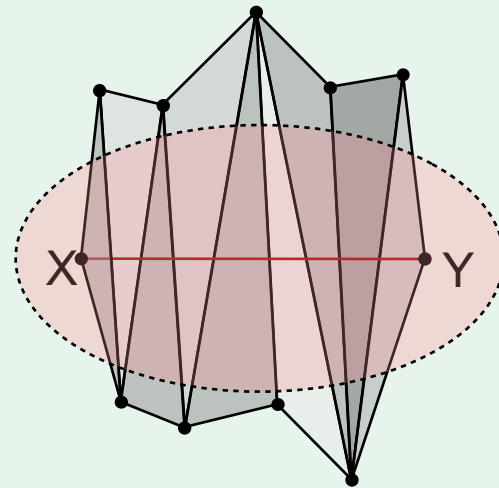


example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

One shows that, given the heuristic condition, this triangulation can be improved, contradicting MWT.

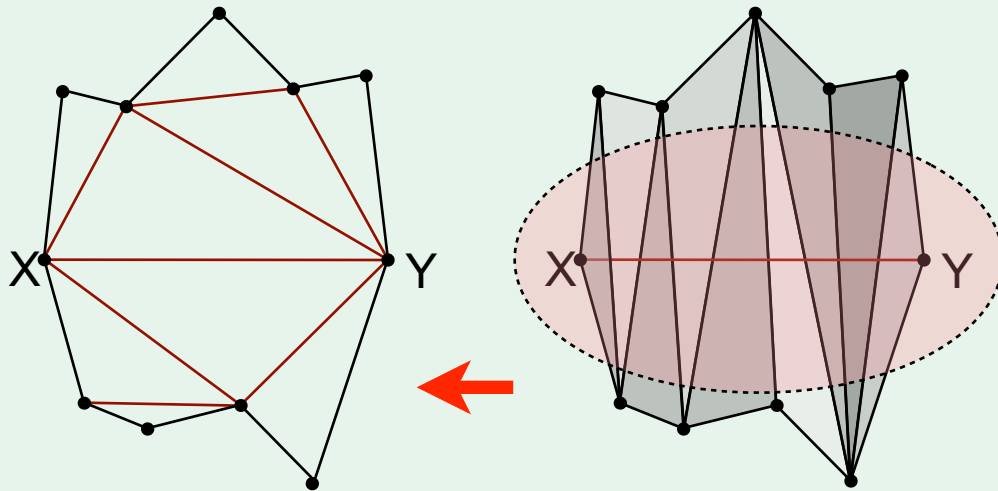


example

- Most heuristics based on local-improvement arguments.

For example, a heuristic might show (x,y) is in every MWT by contradiction. Suppose (x,y) is not in a given triangulation.

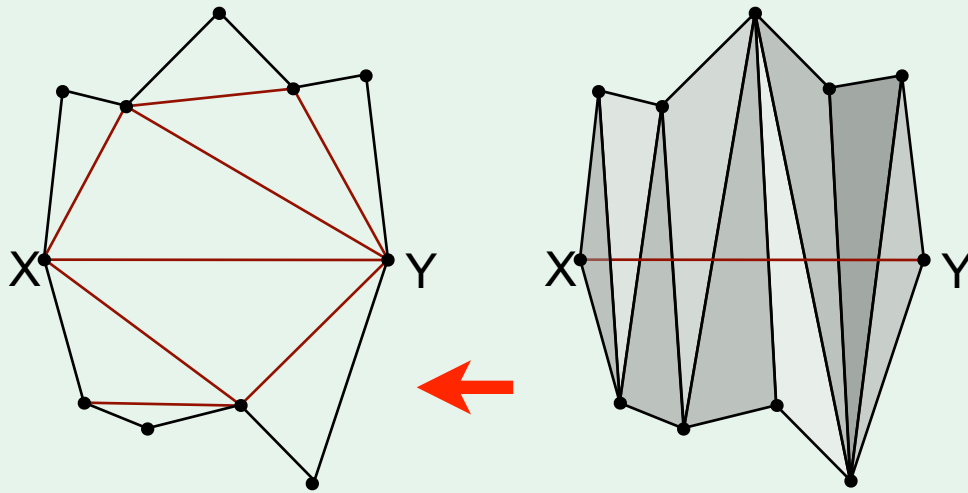
One shows that, given the heuristic condition, this subtriangulation can be improved, contradicting MWT.



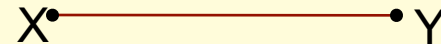
example - extending to fractional MWT

Assume for contradiction that (x,y) edge is not used fully (with total weight 1) in the fractional MWT.

Some triangle that crosses (x,y) must have positive weight.



extending to fractional triangulation

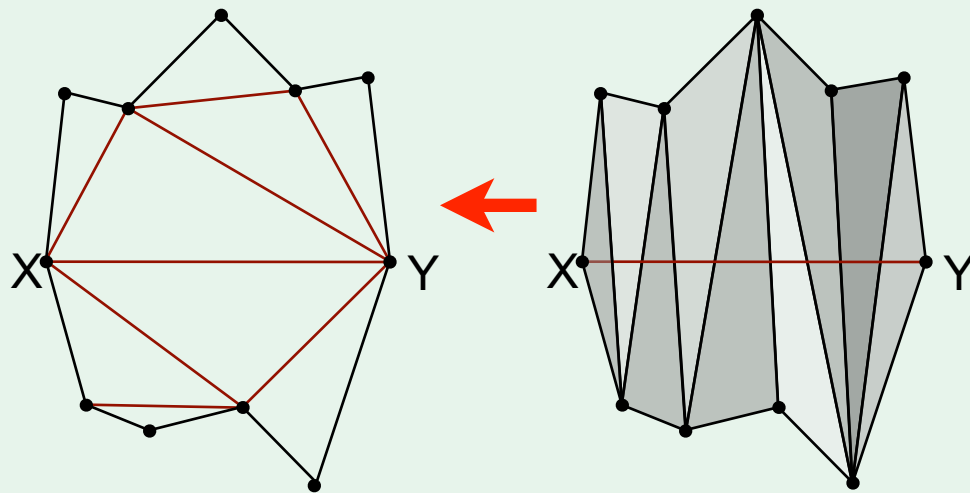


example - extending to fractional MWT

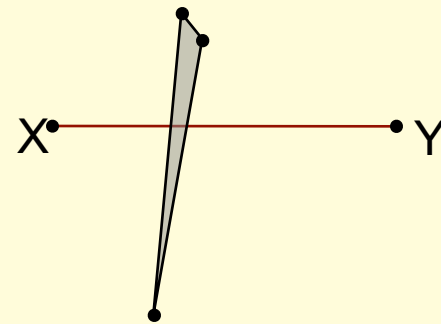
Assume for contradiction that (x,y) edge is not used fully (with total weight 1) in the fractional MWT.

Some triangle that crosses (x,y) must have positive weight.

Can again find a sub-triangulation over (x,y) with positive wt.



extending to fractional triangulation



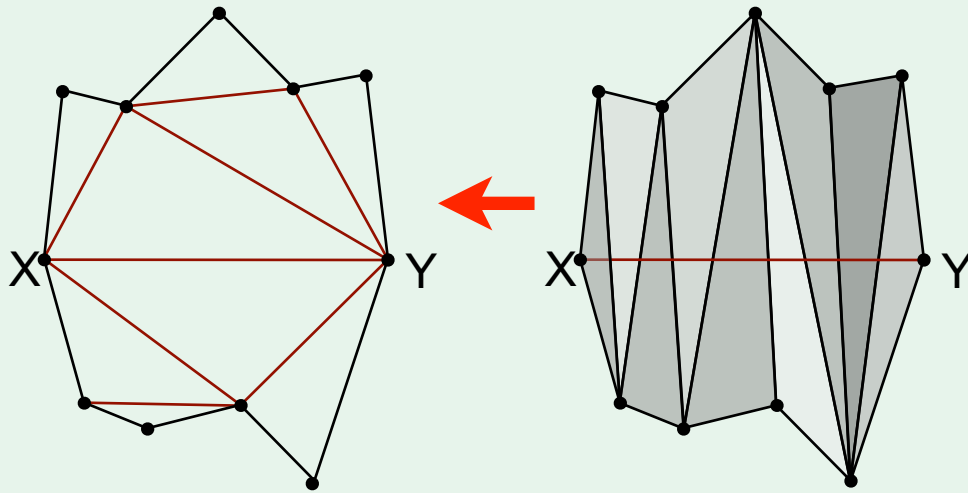
example - extending to fractional MWT

Assume for contradiction that (x,y) edge is not used fully (with total weight 1) in the fractional MWT.

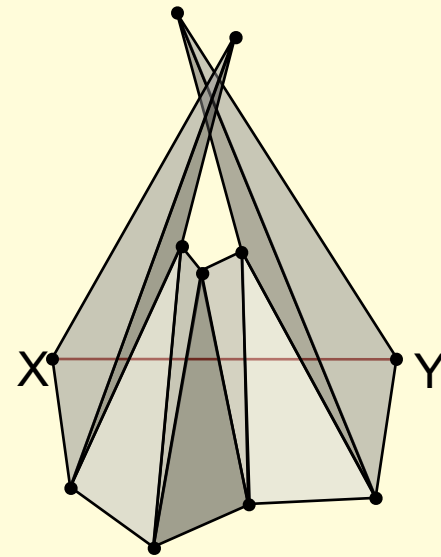
Some triangle that crosses (x,y) must have positive weight.

Can again find a sub-triangulation over (x,y) with positive wt.
But the triangles covering (x,y) may overlap!

Complicates argument, but is not fatal.



extending to fractional triangulation



open problems

- What is the integrality gap of the LP? All we know:

$$1.001 \leq \text{integrality gap} \leq 54(\lambda + 1)$$

(λ is a very large constant.)

- Find an algorithm with **small** constant approximation ratio.
- Primal dual? Randomized rounding?
- Is there a PTAS?

Do constantly many rounds of lift-and-project bring the integrality gap of the LP to $1+\epsilon$?