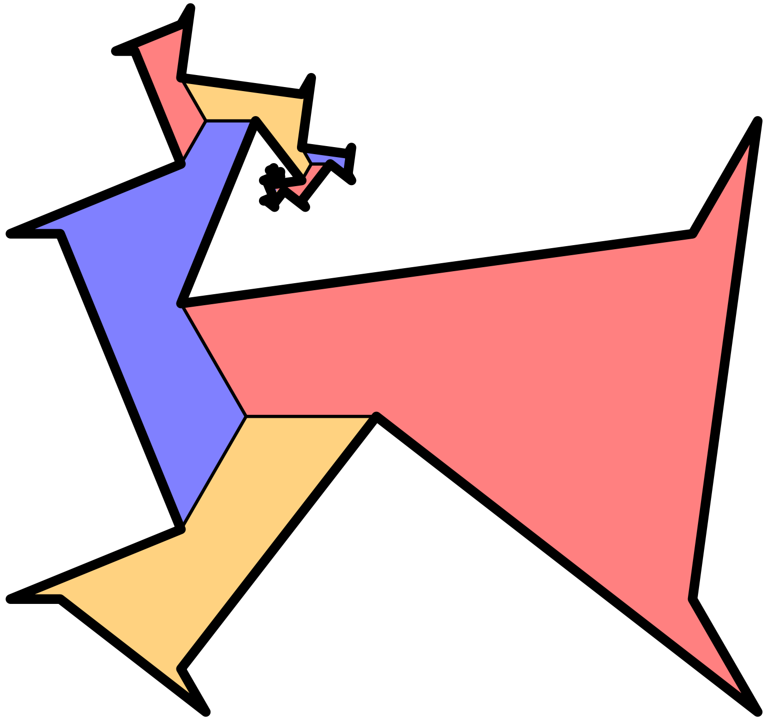
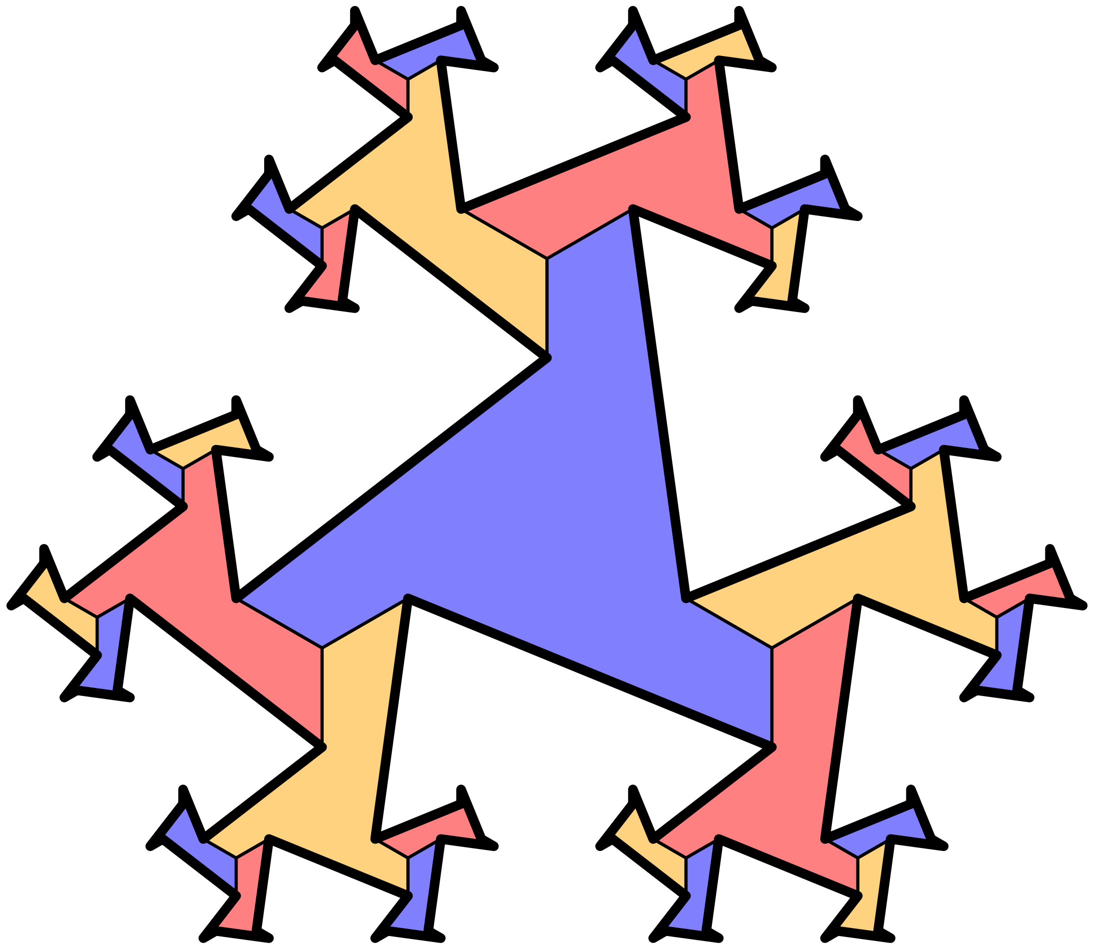
Minimum Star Partitions of Simple Polygons in Polynomial Time

Mikkel Abrahamsen[1]
Joakim Blikstad[2]
André Nusser[1,3]
Hanwen Zhang[1]

[1] BARC
[2] KTH & MPI-INF
[3] CNRS

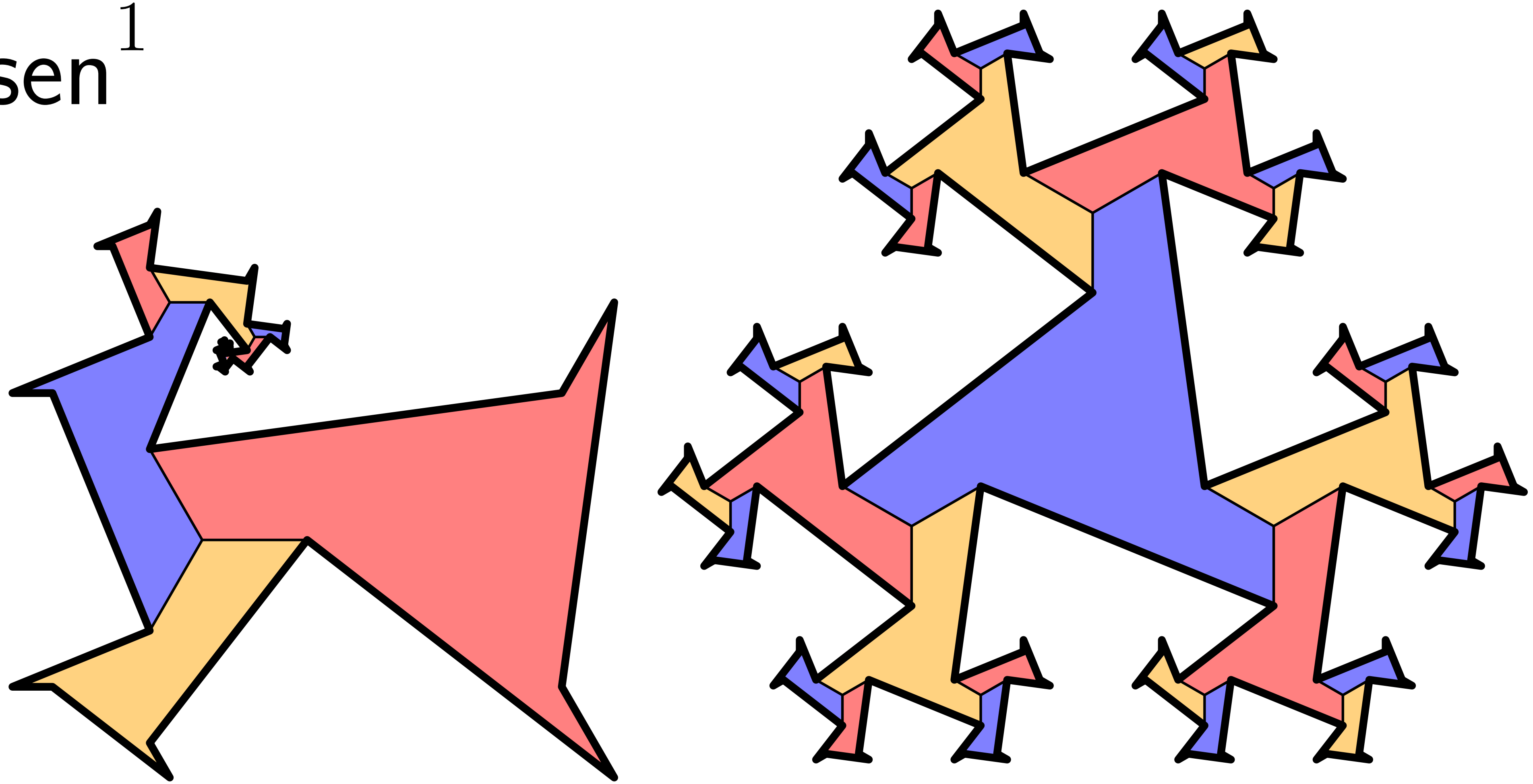




Minimum Star Partitions of Simple Polygons in $O(n^{107})$ Time

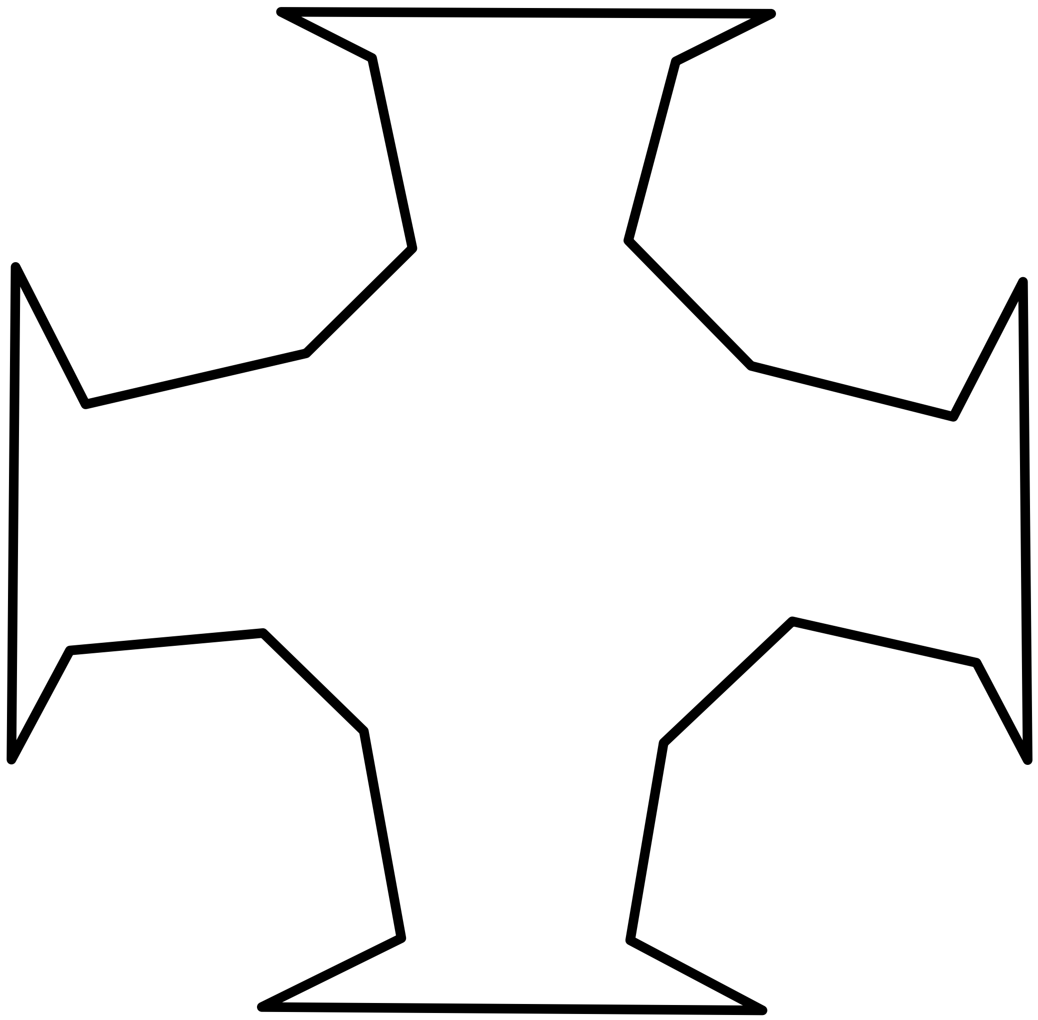
Mikkel Abrahamsen[1]
Joakim Blikstad[2]
André Nusser[1,3]
Hanwen Zhang[1]

[1] BARC
[2] KTH & MPI-INF
[3] CNRS



Suprising Open Problem — Triangle Partitions

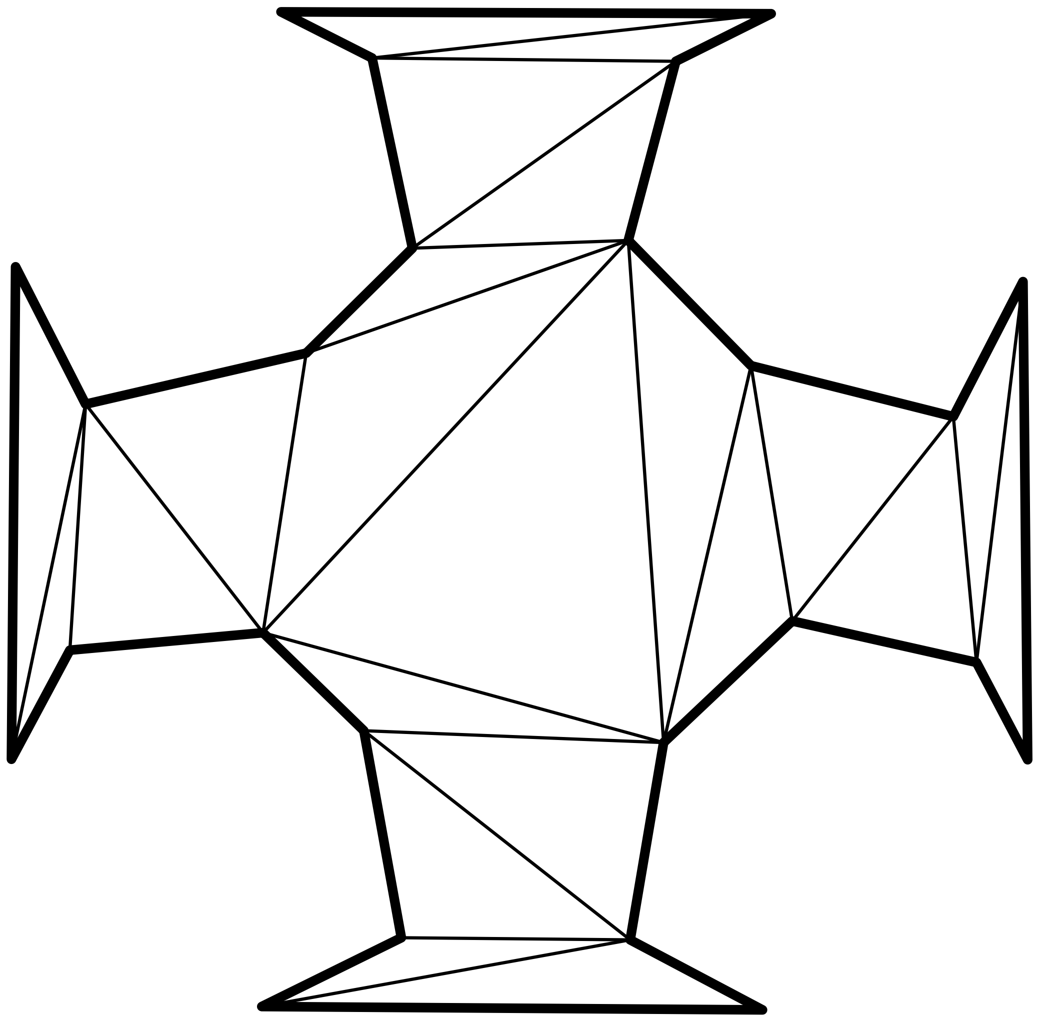
Given polygon (n = #corners),
Partition into few triangle pieces



Suprising Open Problem — Triangle Partitions

Given polygon (n = #corners),
Partition into few triangle pieces

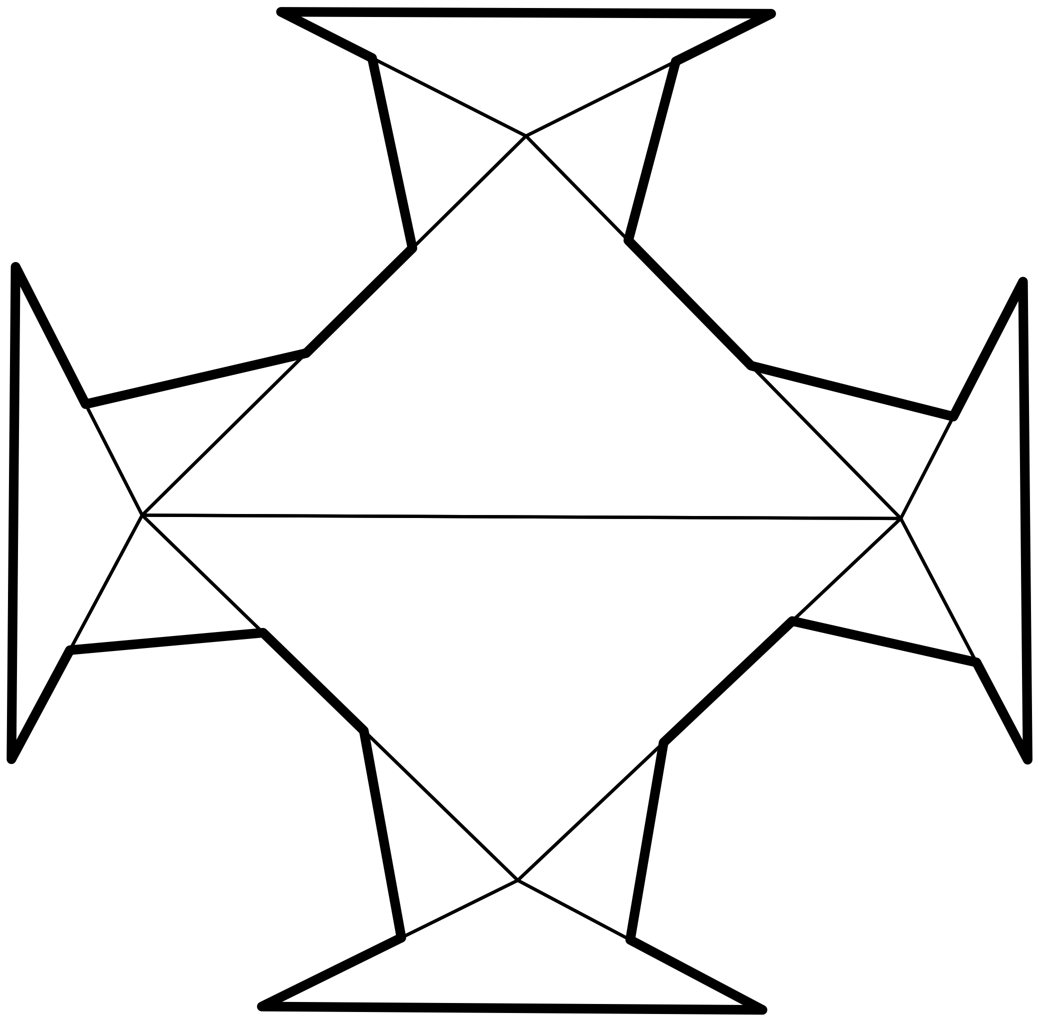
Triangulation: $n - 2$ triangles **22**



Given polygon (n = #corners),
Partition into few triangle pieces

Triangulation: $n - 2$ triangles **22**

Not always optimal! **14**

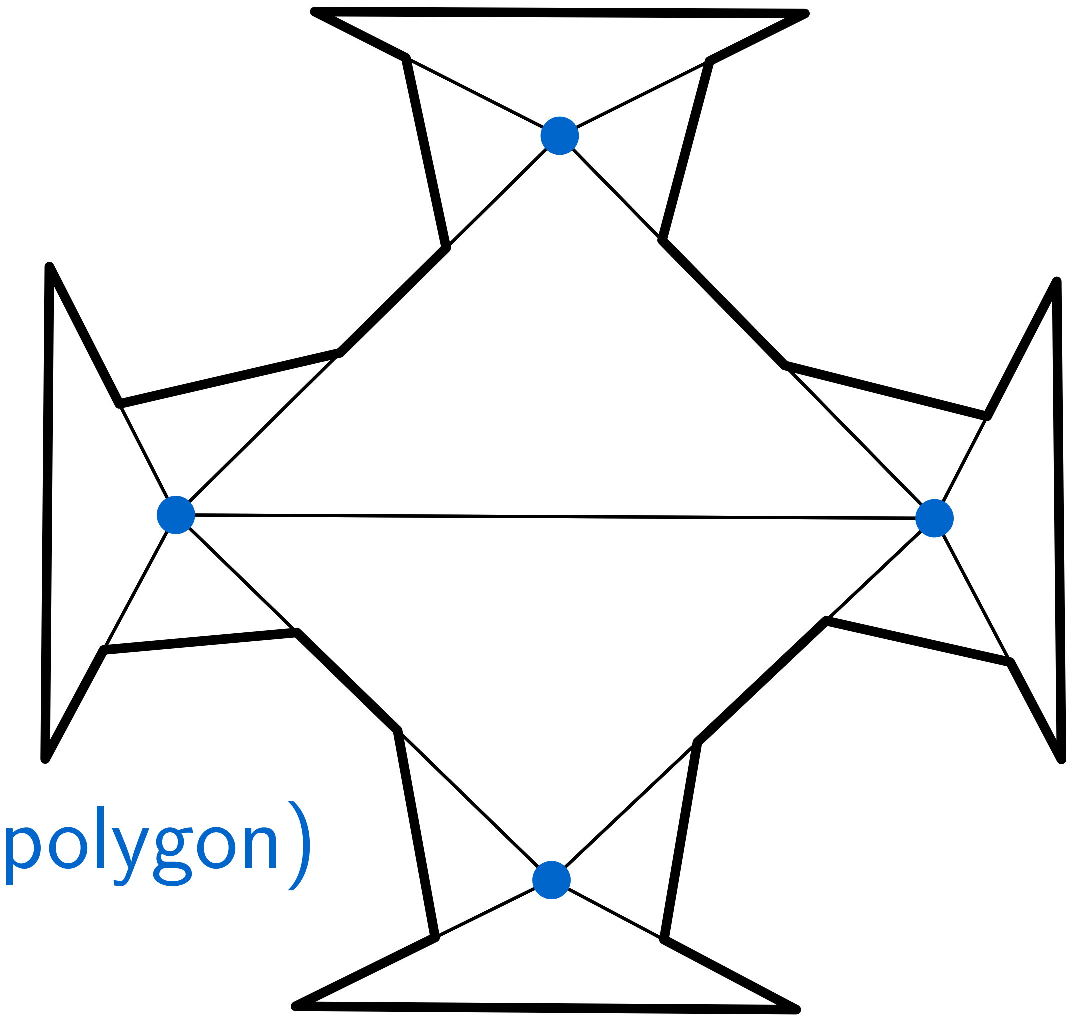


Suprising Open Problem — Triangle Partitions

Given polygon (n = #corners),
Partition into few triangle pieces

Triangulation: $n - 2$ triangles **22**

Not always optimal! **14**



Difficulty: Steiner Points

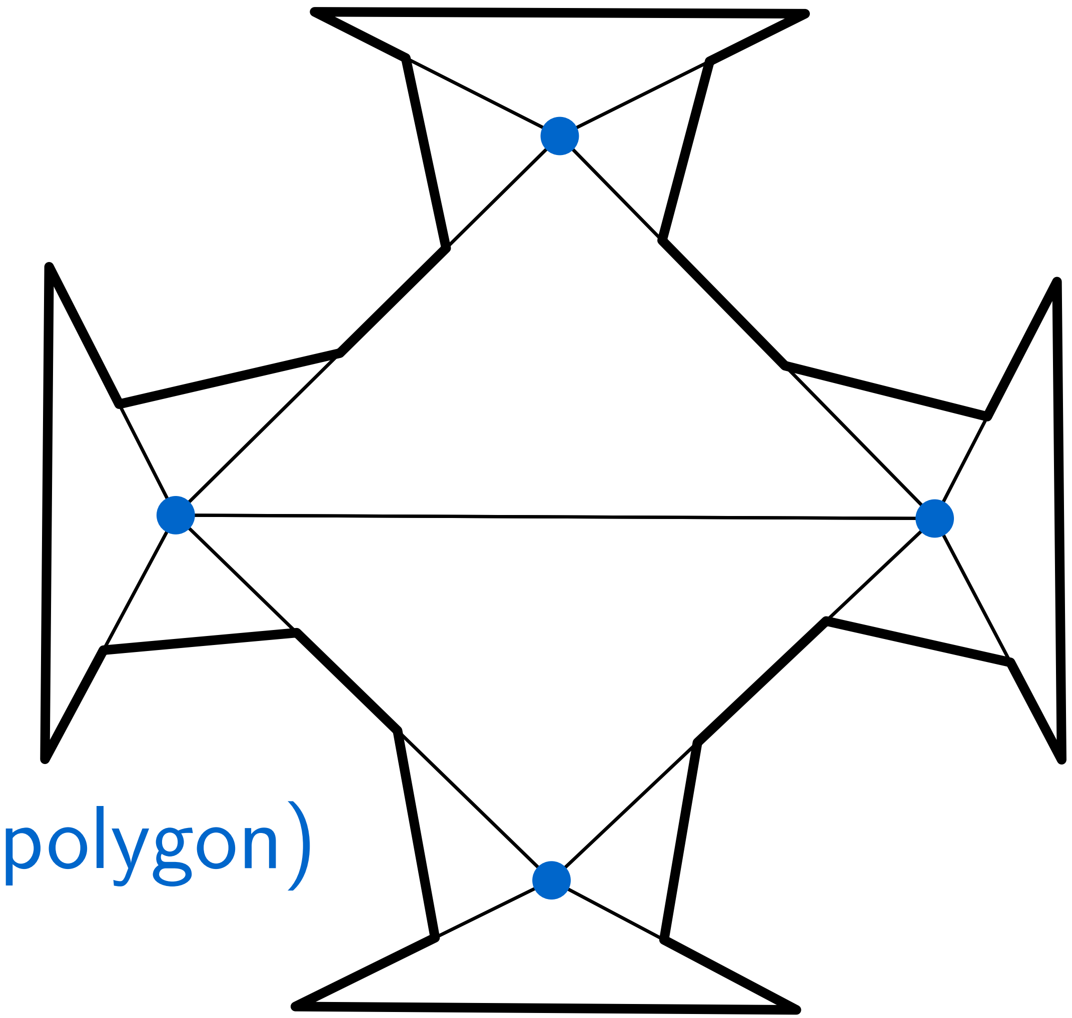
(corner of the solution but not of input polygon)

Suprising Open Problem — Triangle Partitions

Given polygon (n = #corners),
Partition into few triangle pieces

Triangulation: $n - 2$ triangles **22**

Not always optimal! **14**



Difficulty: Steiner Points
(corner of the solution but not of input polygon)

Open Problem: Polynomial time?

Star Partitions

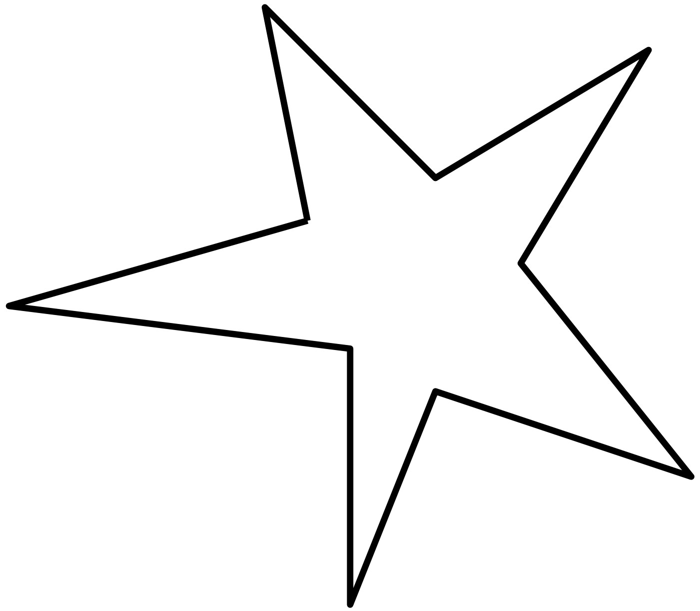
This Talk: Triangle Partitions

Star Partitions

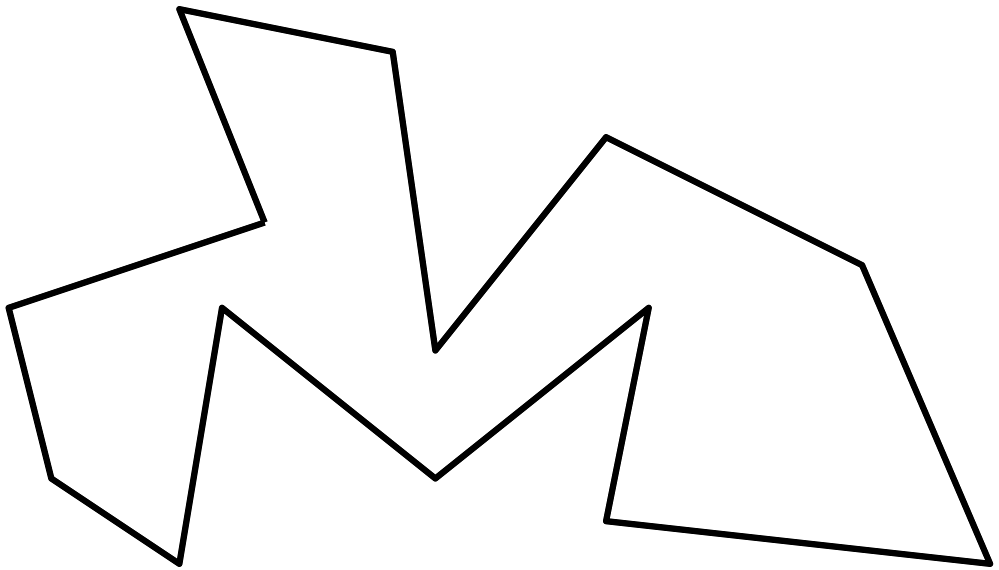
This Talk: **Star** Partitions

Star Partitions

This Talk: **Star** Partitions



Star

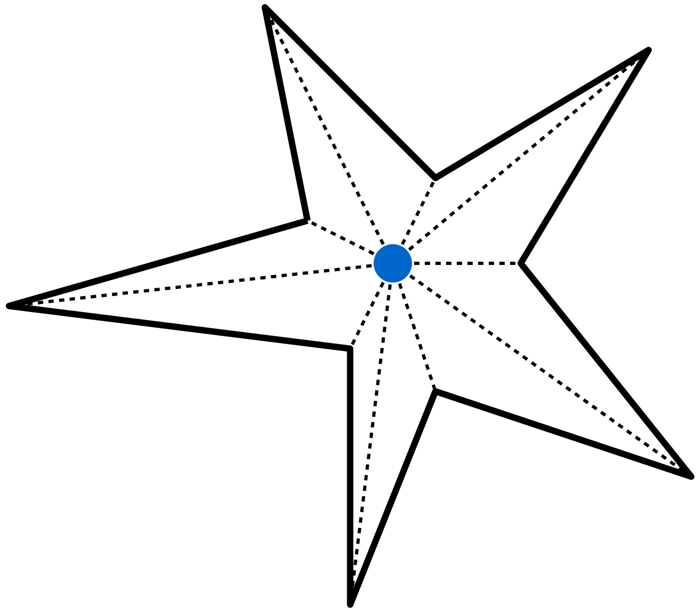


Not a Star

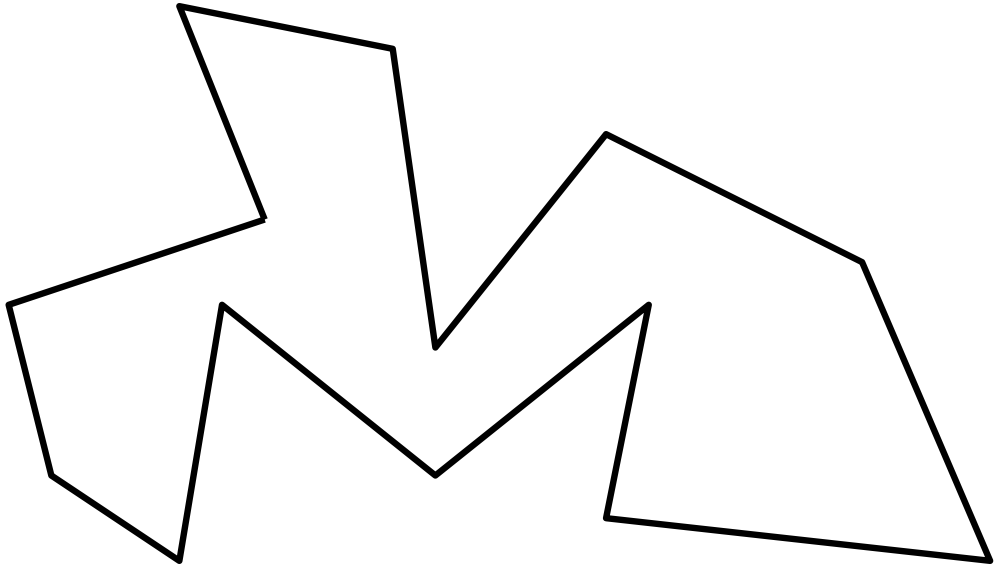
Star Partitions

This Talk: Star Partitions

Definition: Star iff exists star-center point which can see all of the polygon



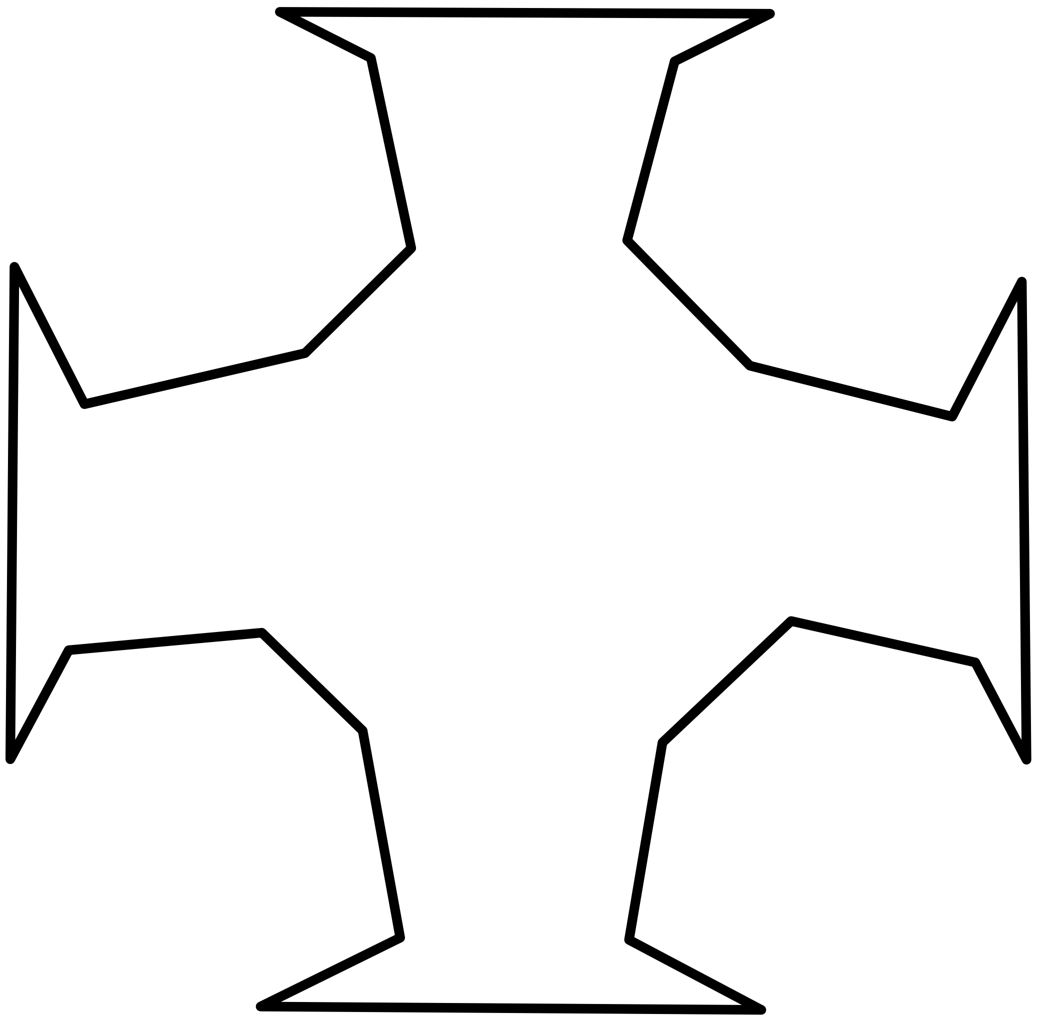
Star



Not a Star

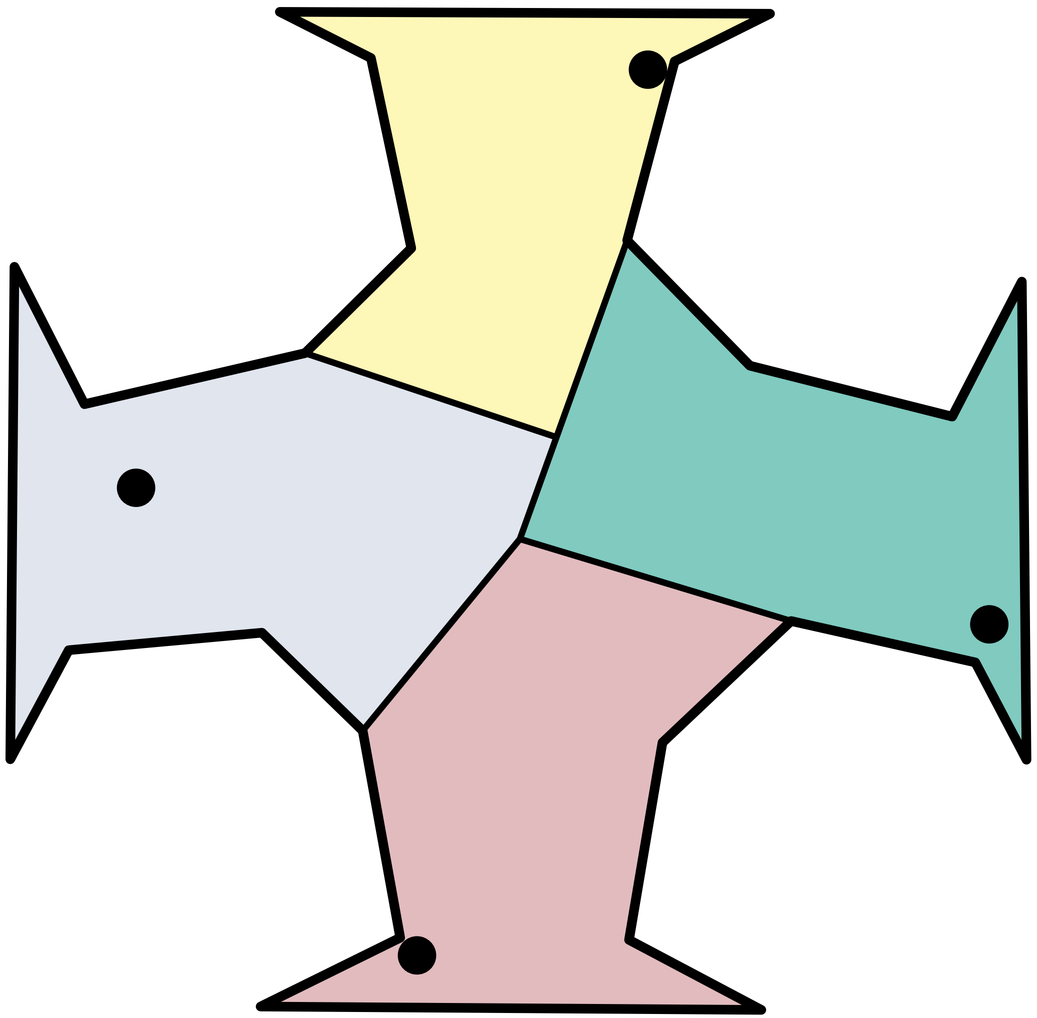
Star Partitions of Polygons

Given polygon (n = #corners),
Partition into few star pieces



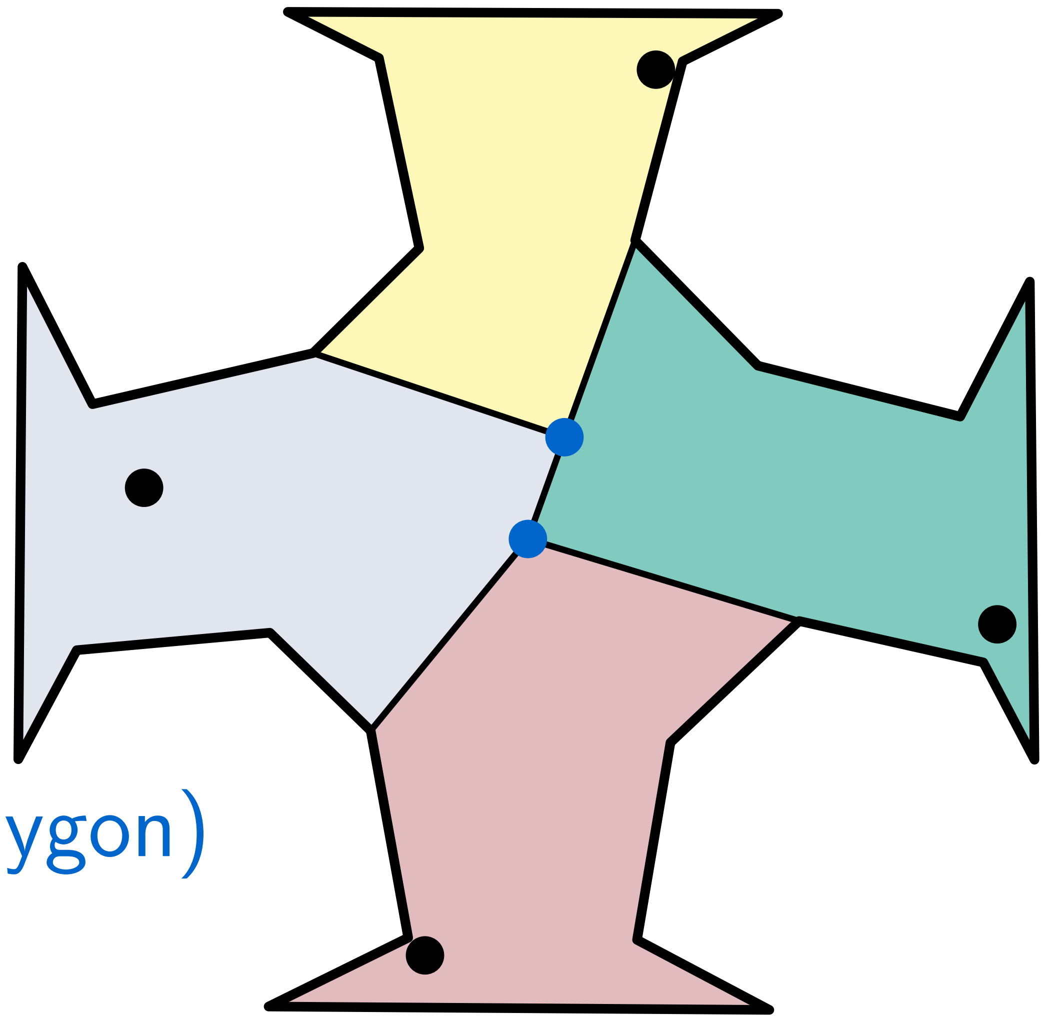
Star Partitions of Polygons

Given polygon (n = #corners),
Partition into few star pieces



Star Partitions of Polygons

Given polygon (n = #corners),
Partition into few star pieces

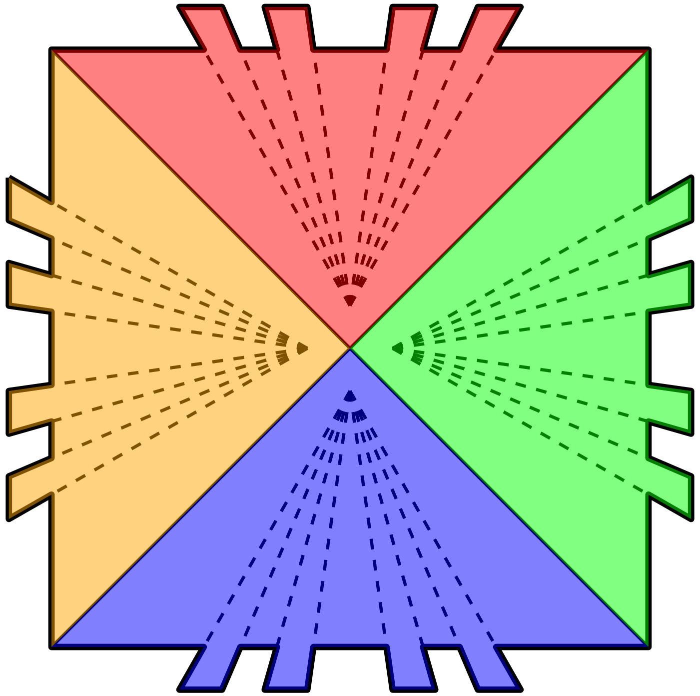


Difficulty: Steiner Points

(corner of the solution but not of input polygon)

Star Partitions of Polygons

Given polygon (n = #corners),
Partition into few star pieces



Difficulty: Steiner Points

(corner of the solution but not of input polygon)

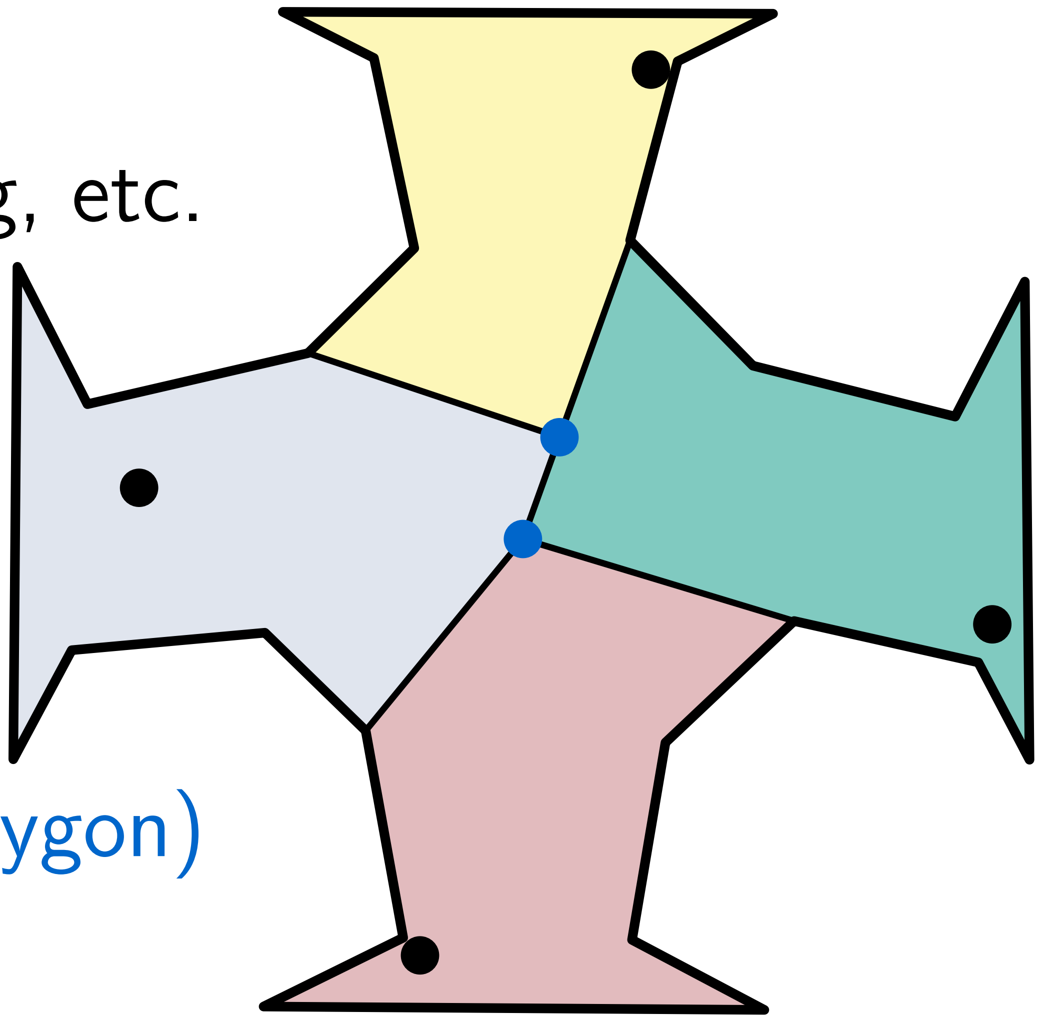
Without Steiner Points: $\tilde{O}(n^7)$ [Kei'85]

With Steiner points: $\text{ans} = 4$
Without Steiner points: $\text{ans} = \Omega(n)$

Star Partitions of Polygons

Given polygon (n = #corners),
Partition into few star pieces

Applications in CNC milling, route planning, etc.
Open for > 40 years if in P (or even NP)

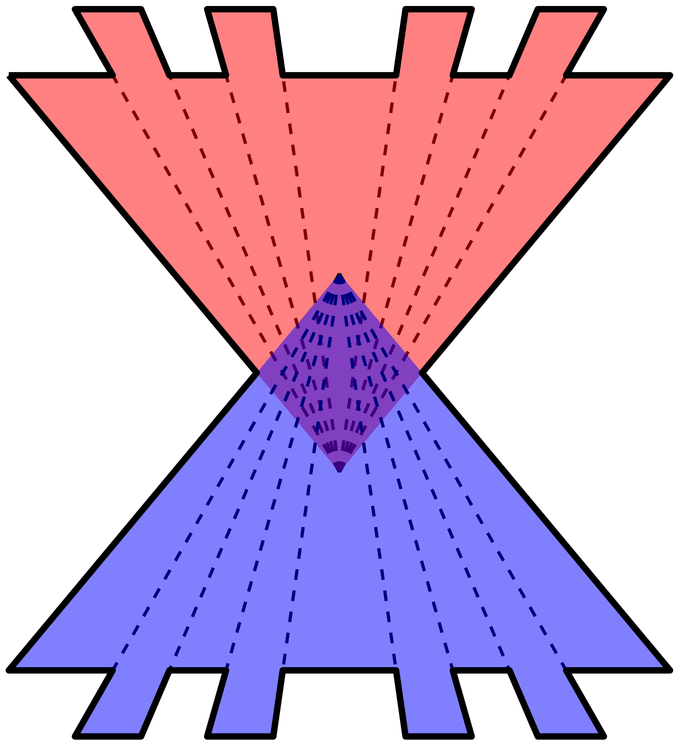


Difficulty: Steiner Points

(corner of the solution but not of input polygon)

Art Gallery Problem

Related Problem: **Cover** vs **Partition**

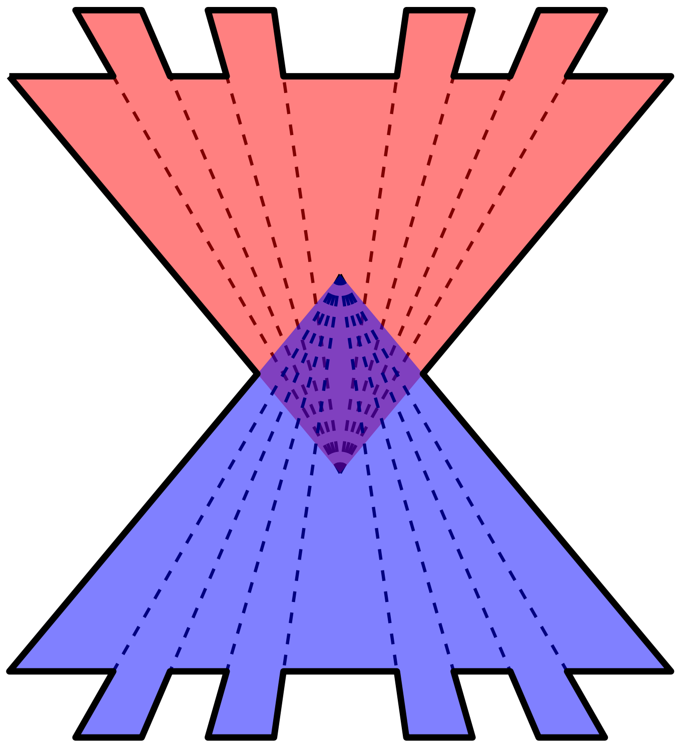


Art Gallery Problem

Related Problem: **Cover** vs **Partition**

Cover: Pieces can overlap ans=**2**

Partition: Pieces cannot overlap ans=$\Omega(n)$



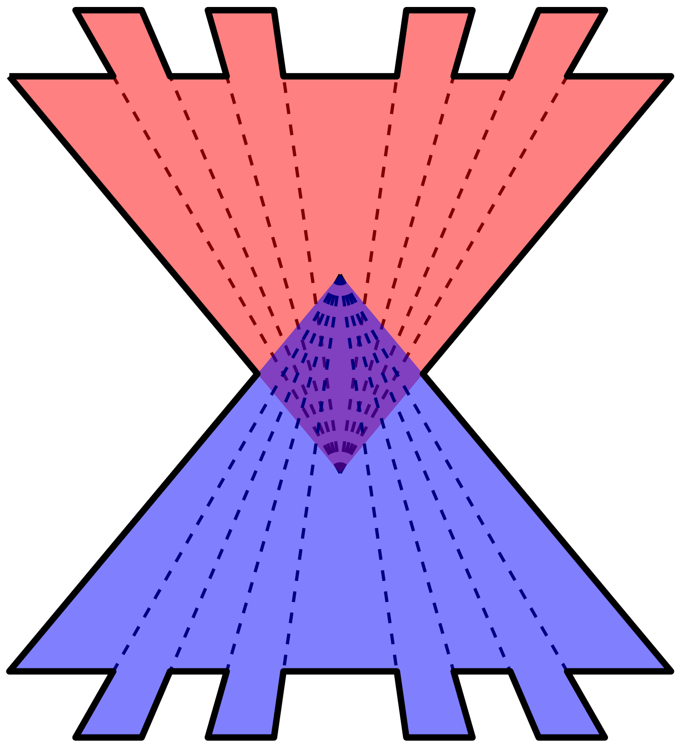
Art Gallery Problem

Related Problem: **Cover** vs **Partition**

Cover: Pieces can overlap ans=**2**

Partition: Pieces cannot overlap ans=$\Omega(n)$

Theorem: (Cover)
Art Gallery Problem is $\exists\mathbb{R}$-complete [AAM'22]
(i.e., probably not even in NP)

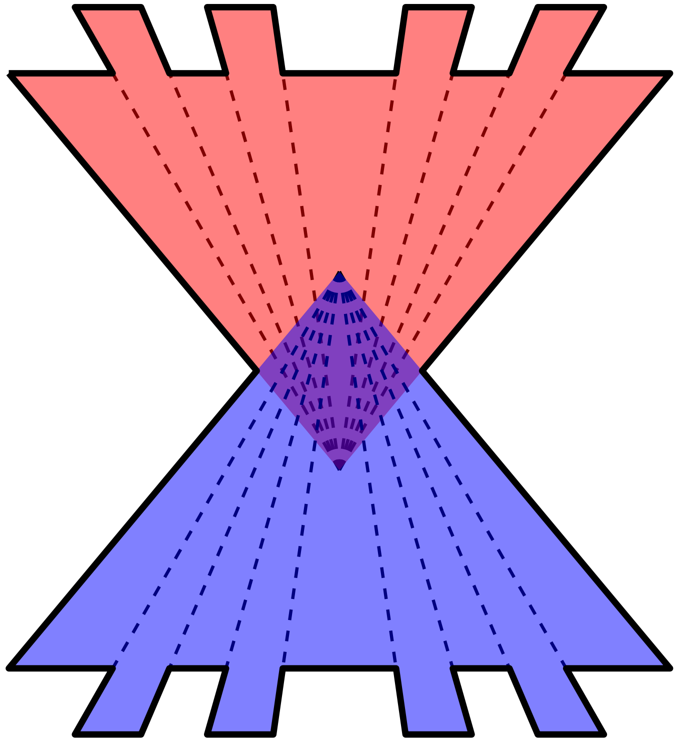


Art Gallery Problem

Related Problem: **Cover** vs **Partition**

Cover: Pieces can overlap ans$=$**2**

Partition: Pieces cannot overlap ans$=\Omega(n)$



Theorem: (Cover)
Art Gallery Problem is $\exists\mathbb{R}$-complete [AAM'22]
(i.e., probably not even in NP)

Main Result

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Main Result

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Polynomial!

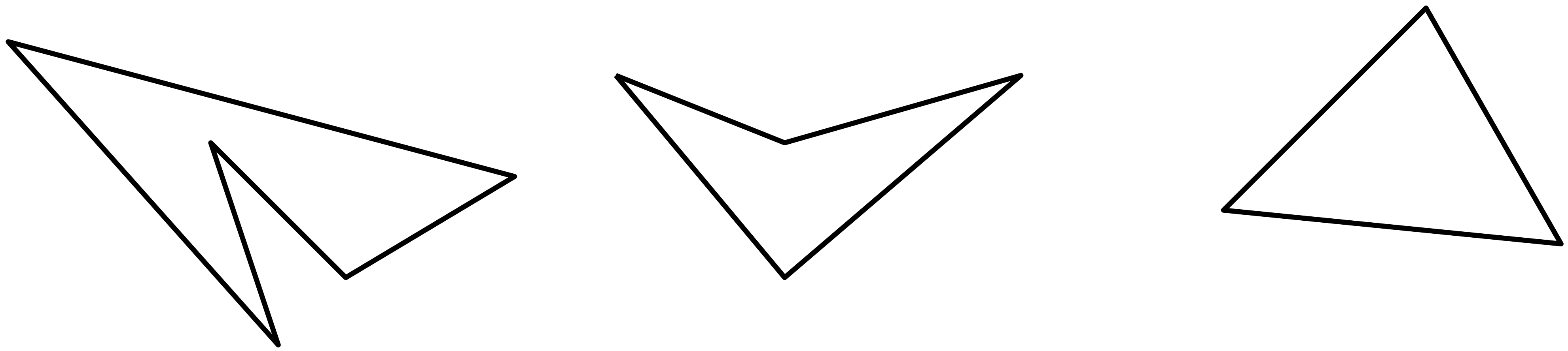
Main Result

Our Main Result:

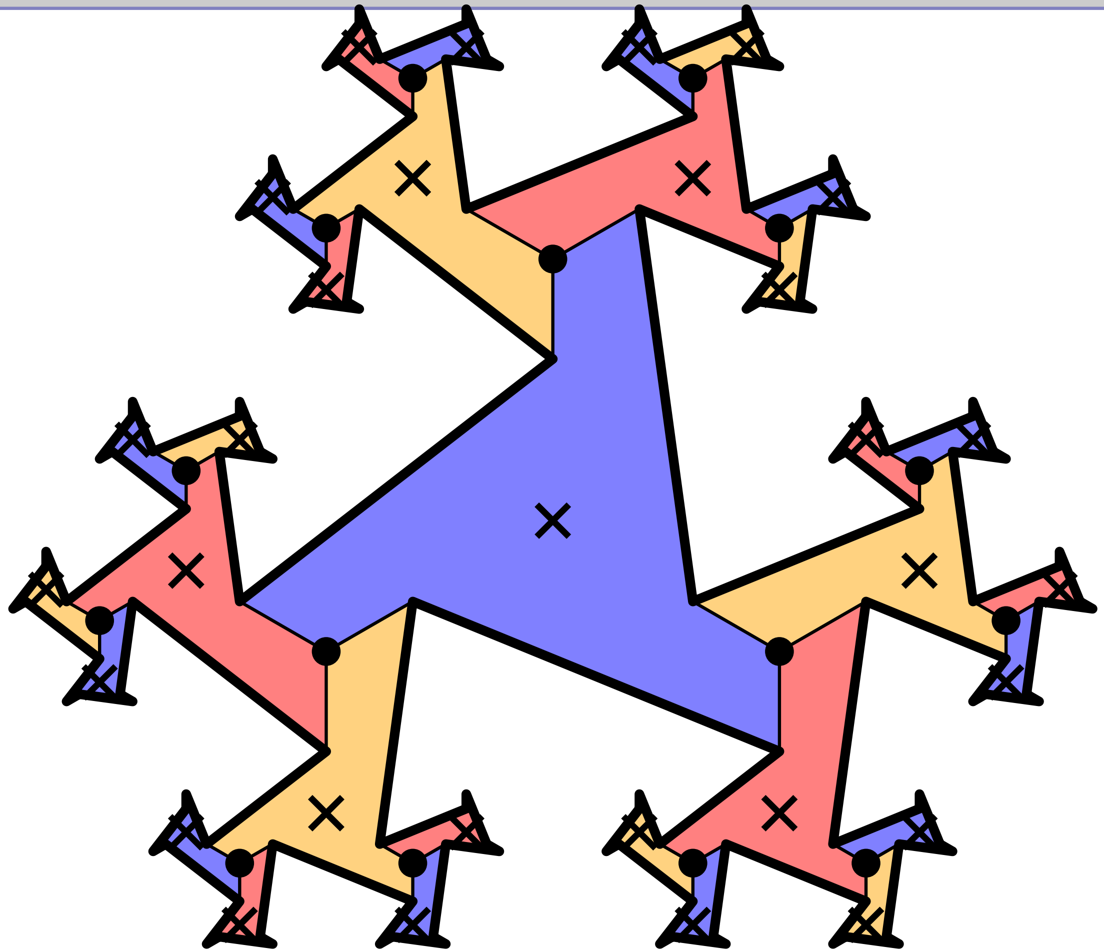
Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

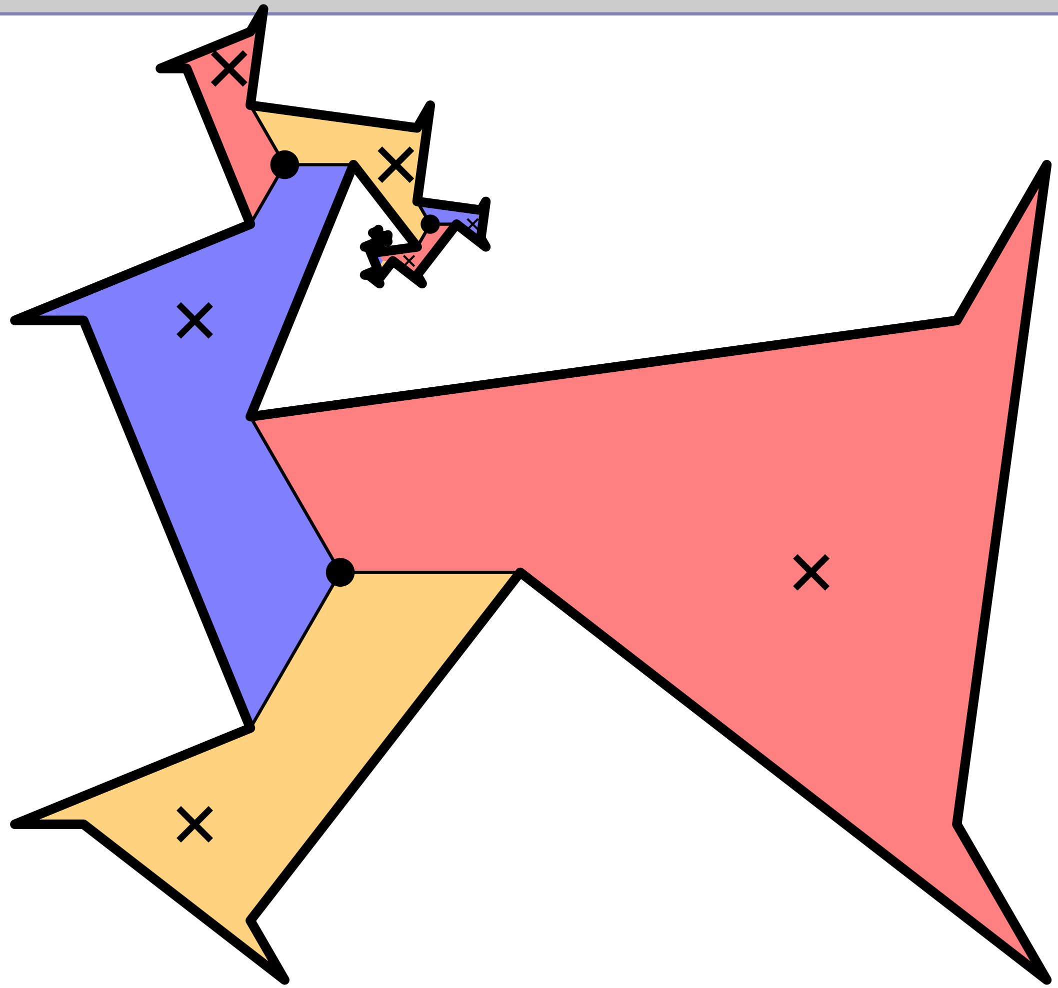
Polynomial!

Works in $\ll 1$ sec for $n \leq 5$, but $n = 6$ it is a bit slow...

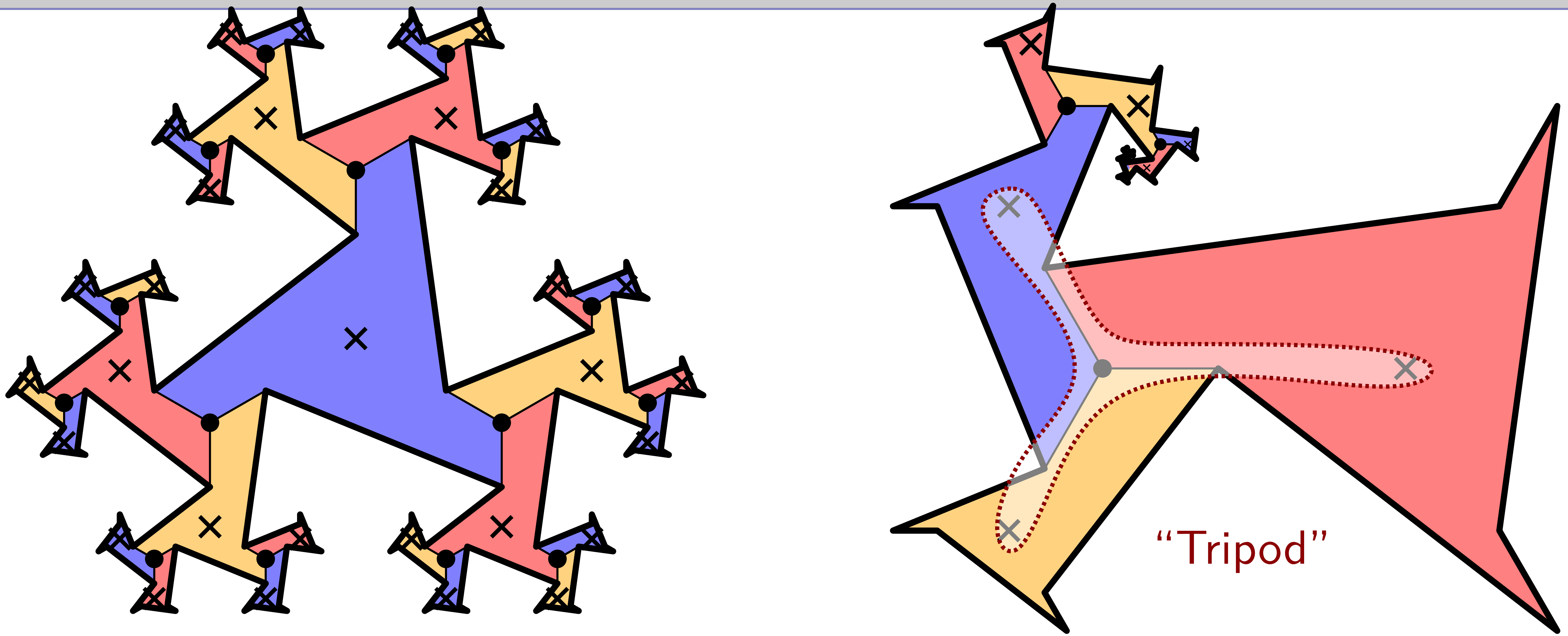


Tricky Instances



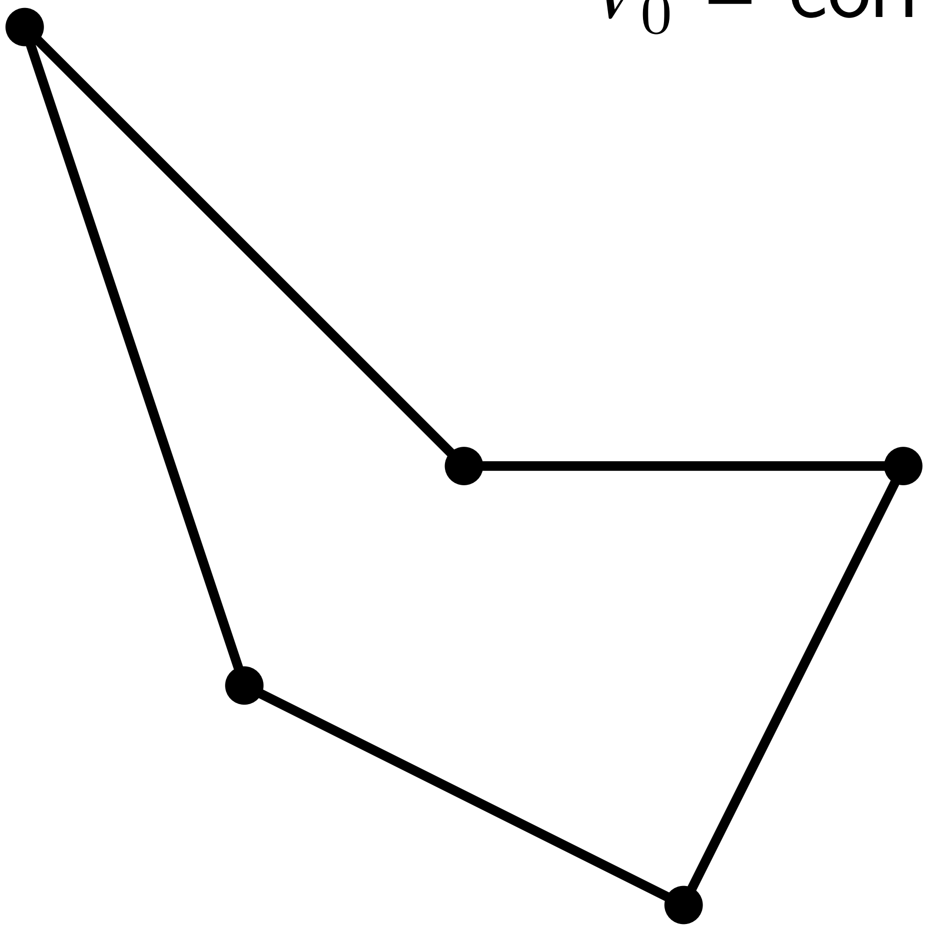


Tricky Instances



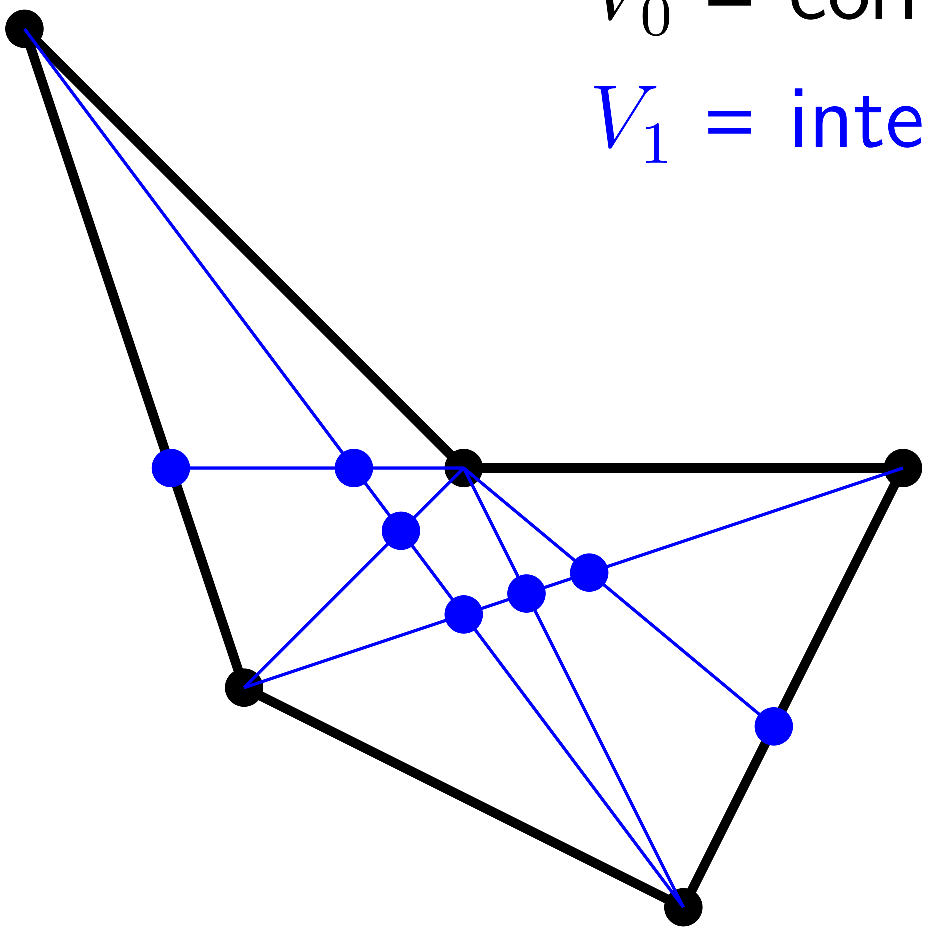
Tricky Instances

V_0 = corners of P



$|V_0| = n$

Tricky Instances

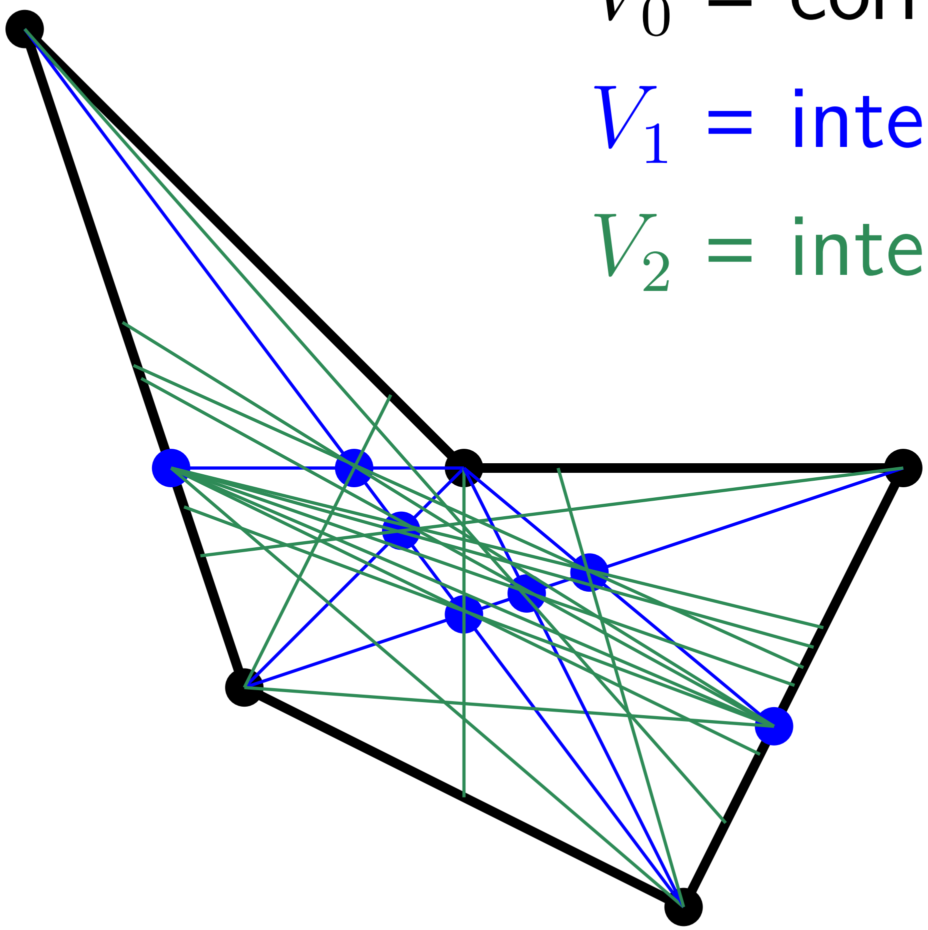


V_0 = corners of P

V_1 = intersections of lines between points in V_0

$|V_0| = n$ $|V_1| = O(n^4)$

Tricky Instances



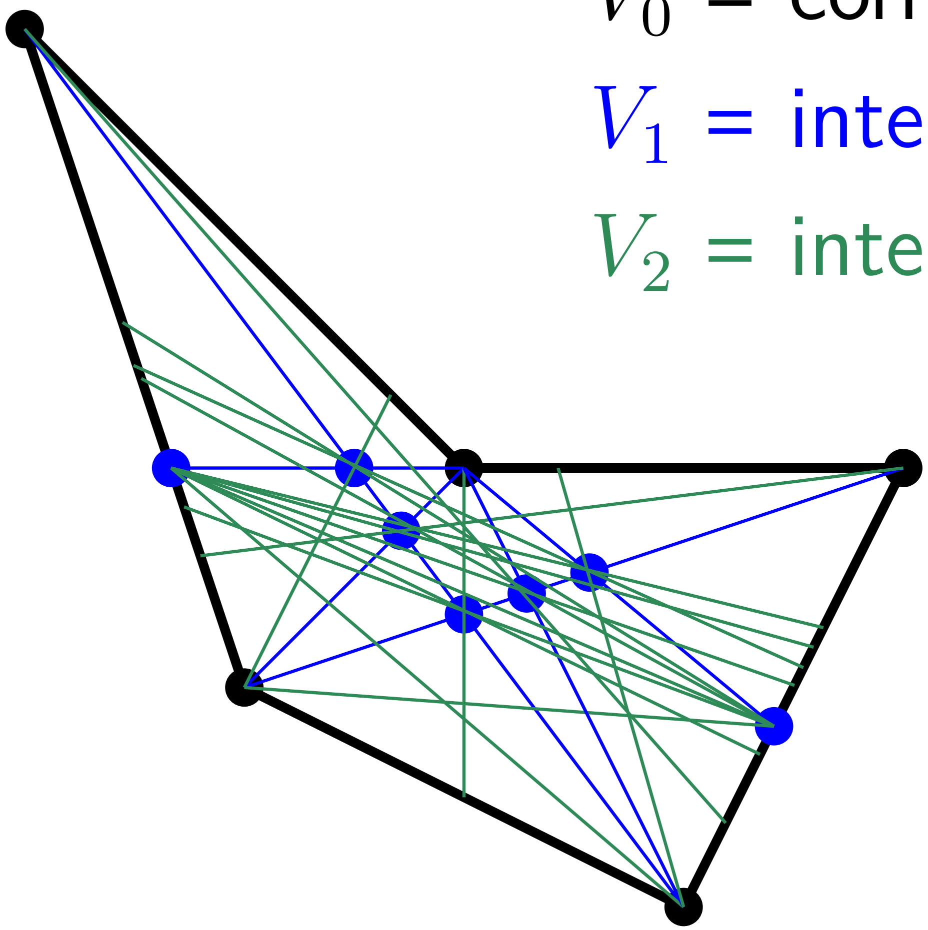
V_0 = corners of P

V_1 = intersections of lines between points in V_0

V_2 = intersections of lines between points in V_1

$|V_0| = n$ $\quad |V_1| = O(n^4)$ $\quad |V_2| = O(n^{16})$

Tricky Instances



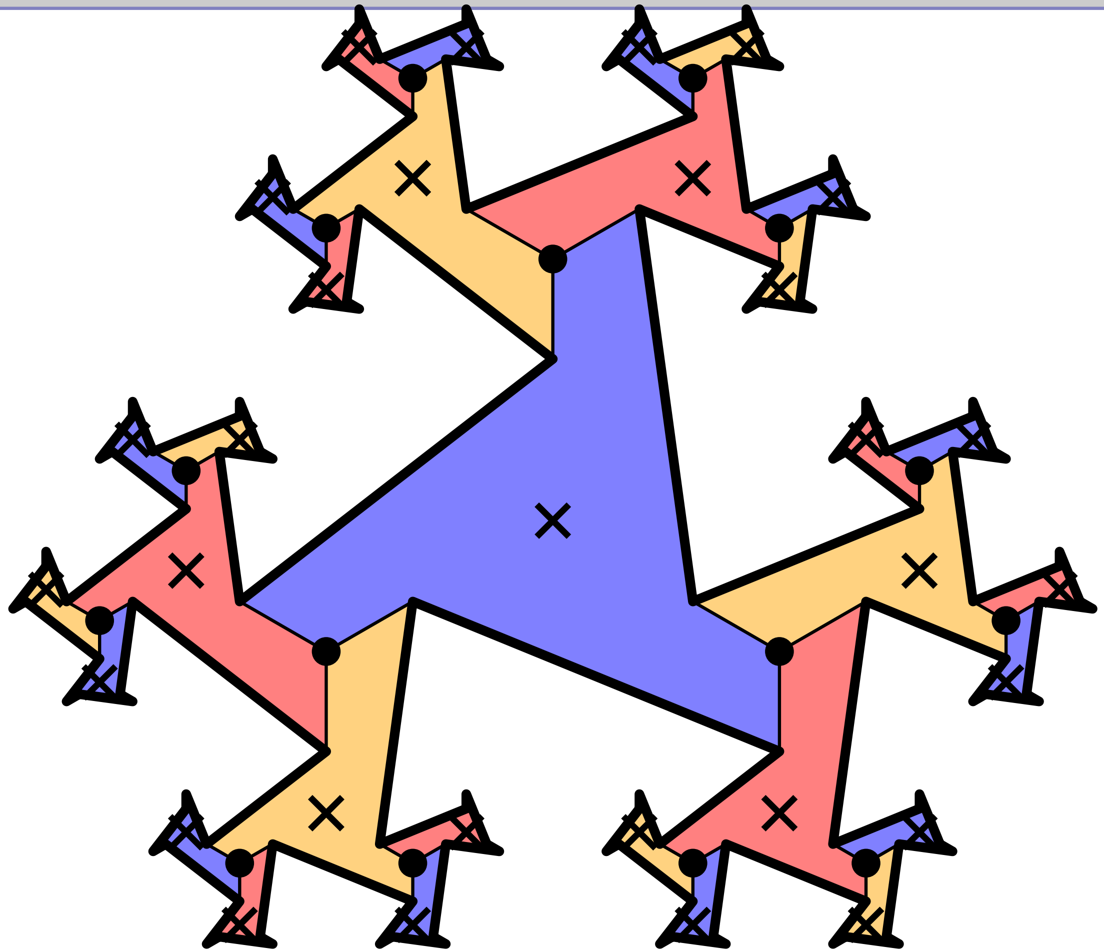
V_0 = corners of P

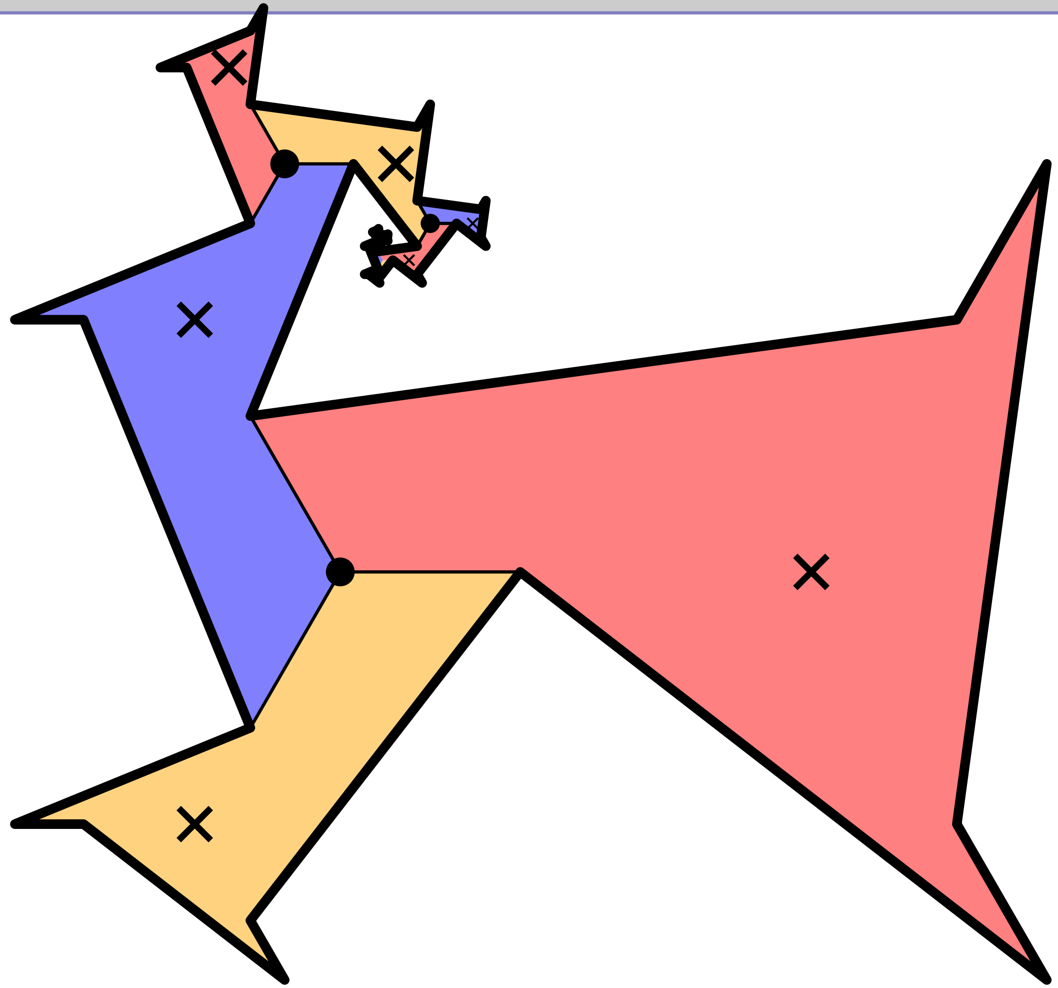
V_1 = intersections of lines between points in V_0

V_2 = intersections of lines between points in V_1

$|V_0| = n$ $|V_1| = O(n^4)$ $|V_2| = O(n^{16})$ $|V_k| = O(n^{(4^k)})$

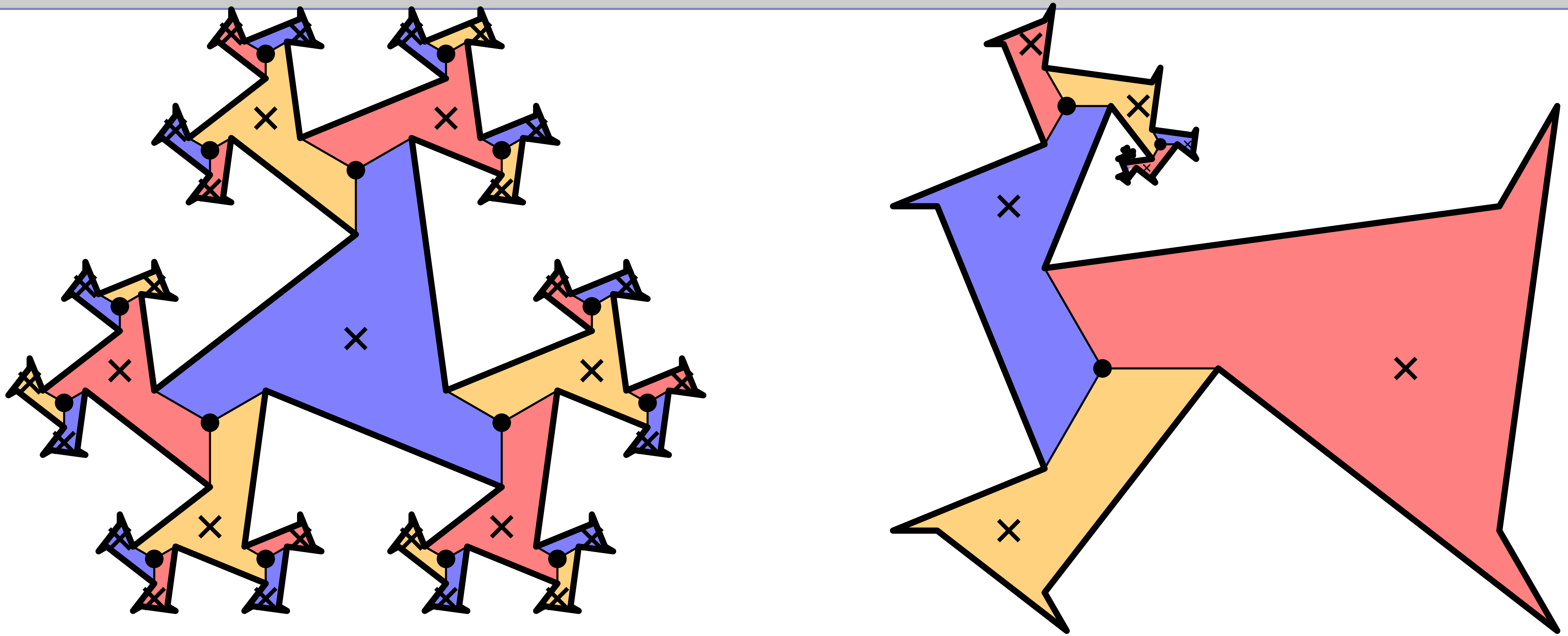
Tricky Instances





Good News: Exists solution with all (Steiner) points in V_n (unlike covering!)

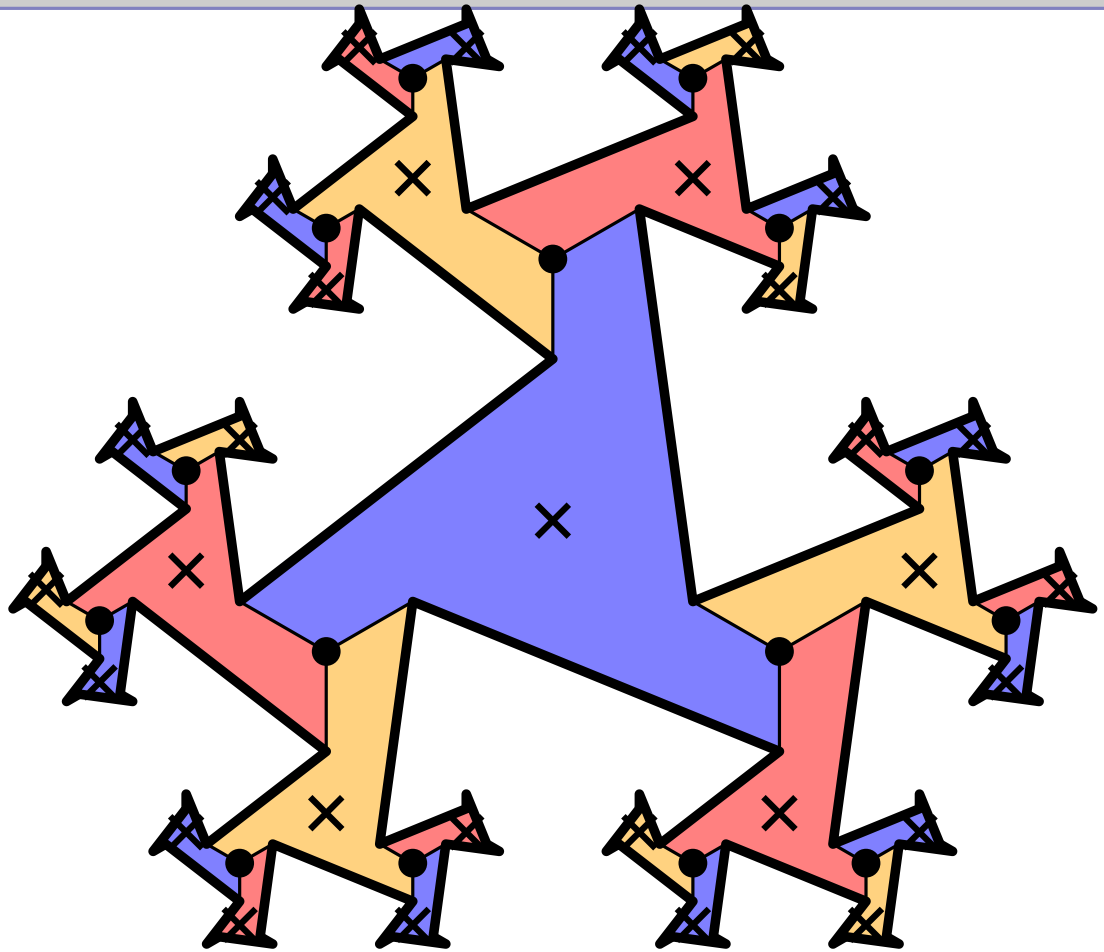
Tricky Instances

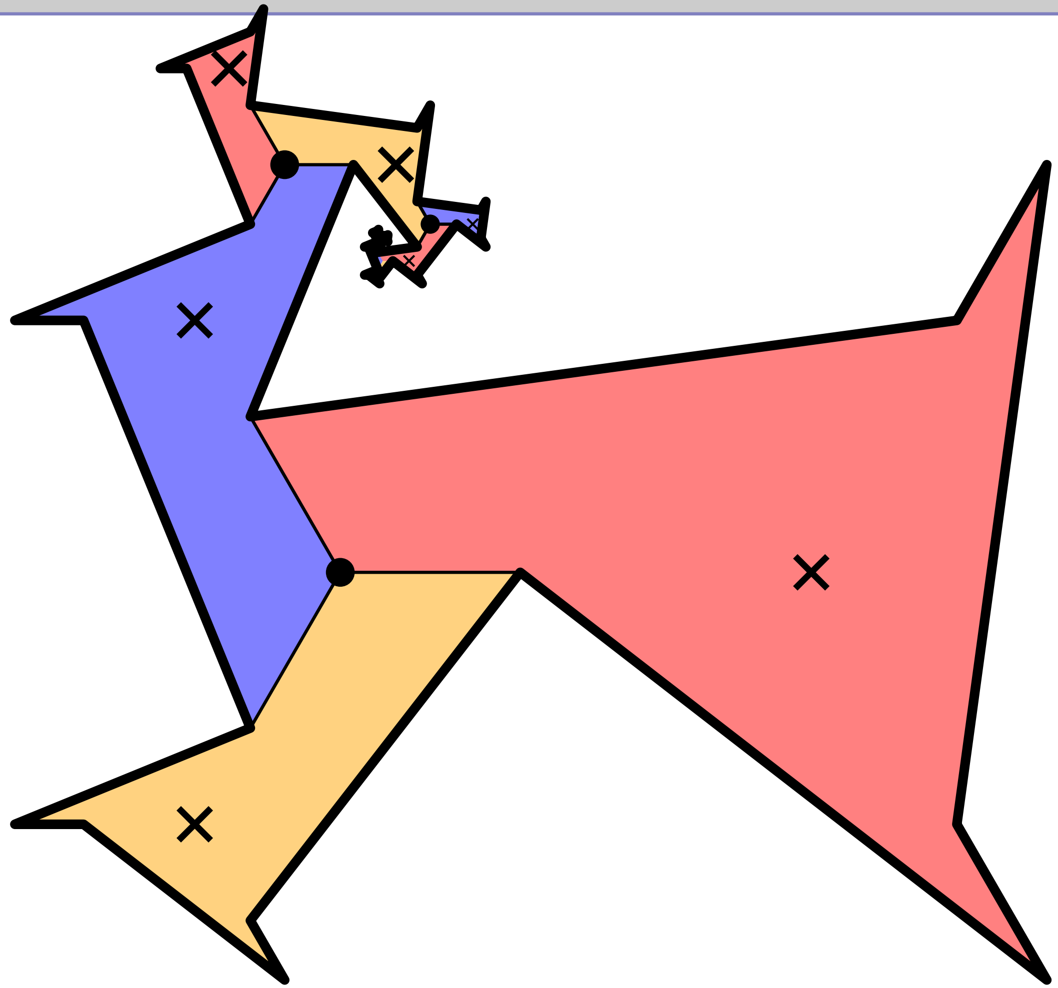


Good News: Exists solution with all (Steiner) points in V_n (unlike covering!)

Bad News: Some examples require points in $V_{\Omega(n)}$; size $n^{2^{\Omega(n)}}$

Tricky Instances



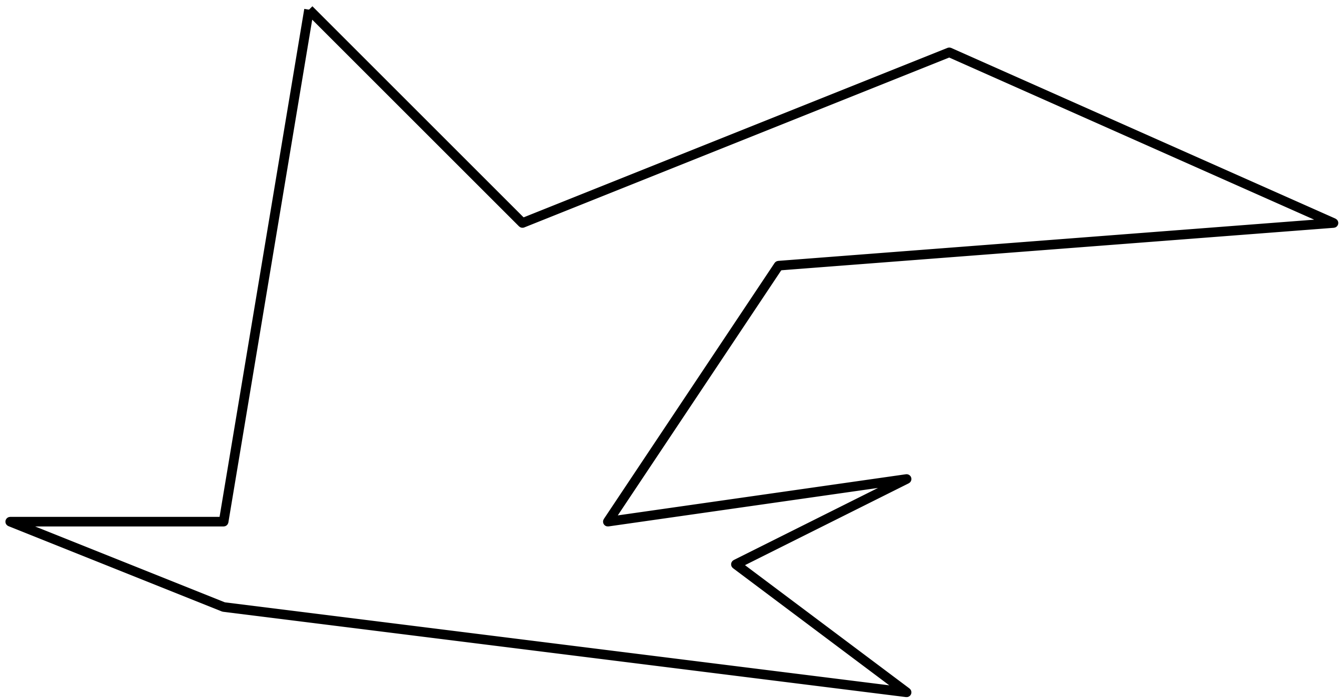


Good News: Exists solution with all (Steiner) points in V_n (unlike covering!)

Bad News: Some examples require points in $V_{\Omega(n)}$; size $n^{2^{\Omega(n)}}$

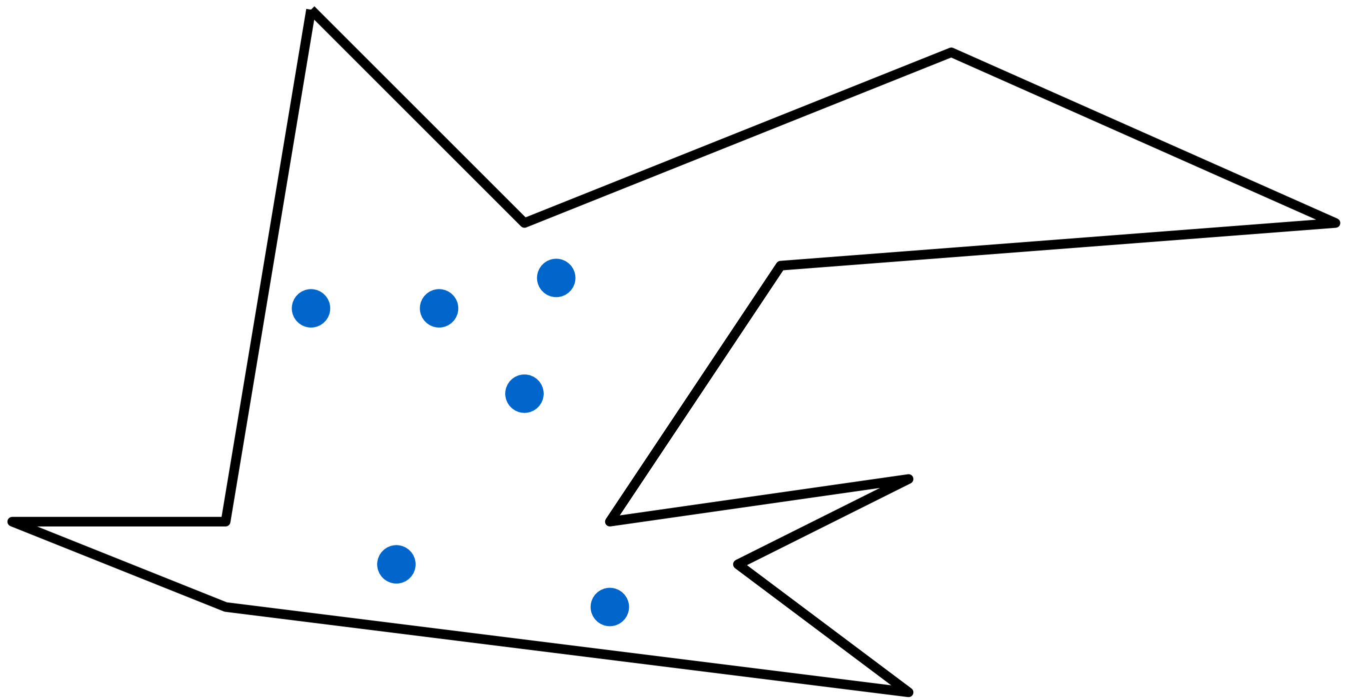
Good News: "Tripod" structure is only "tricky" case

Algorithm Outline



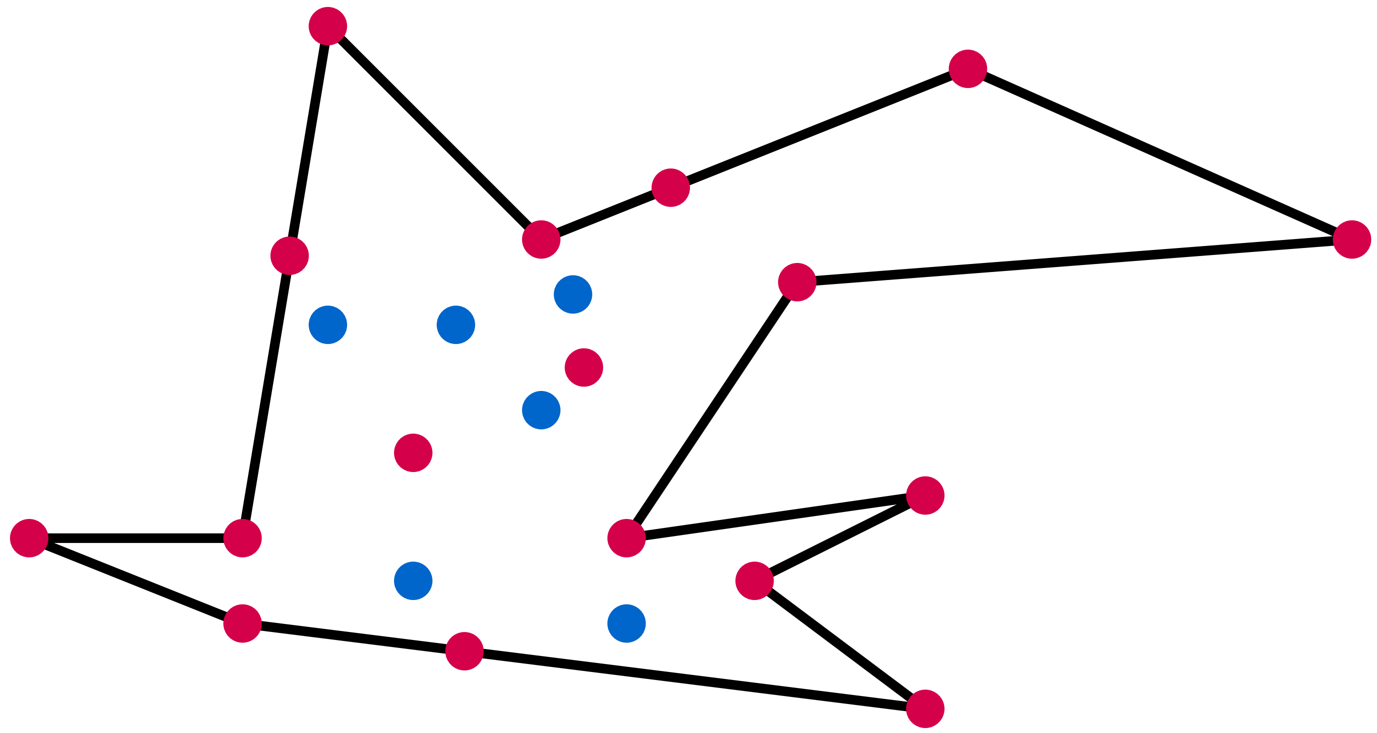
Algorithm Outline

1. Find small set of potential star-centers: S^{centers}



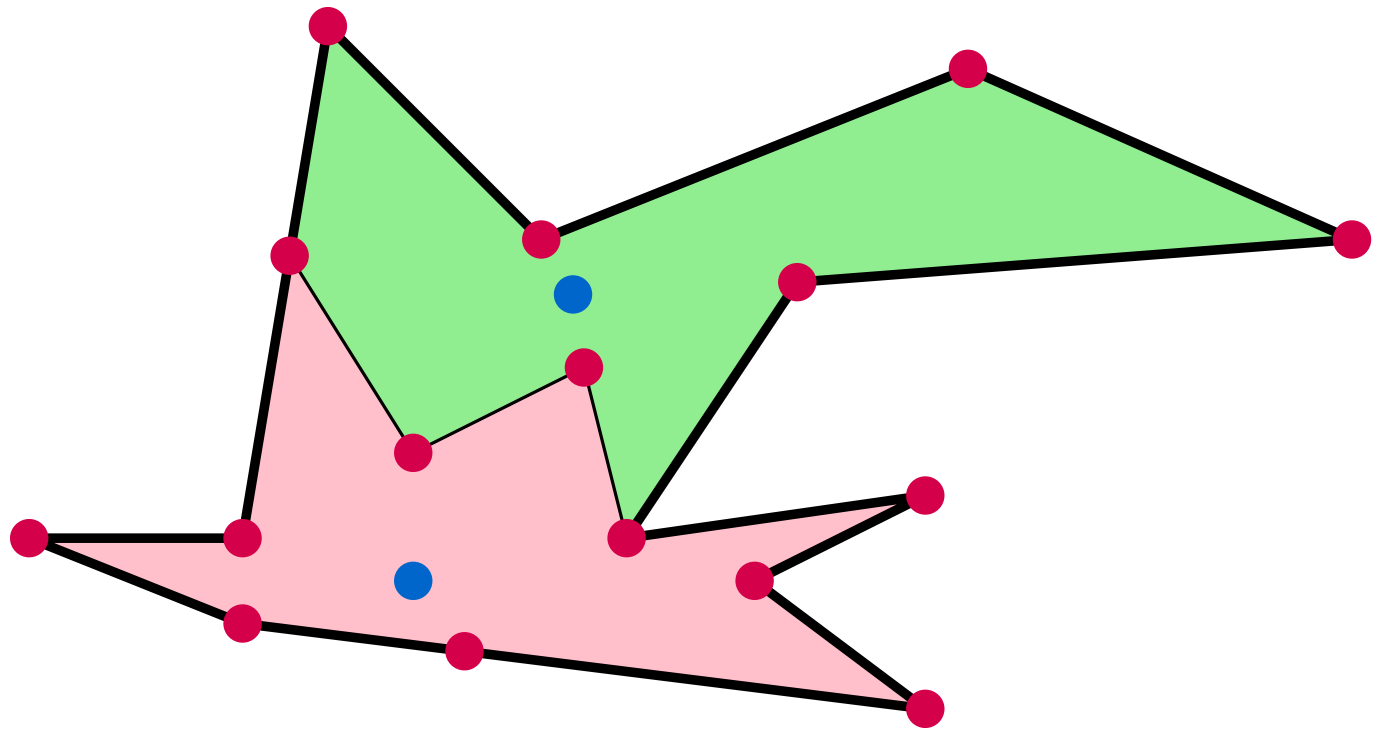
Algorithm Outline

1. Find small set of potential star-centers: S^{centers}
2. Find small set of potential "corner" points: S^{corners}



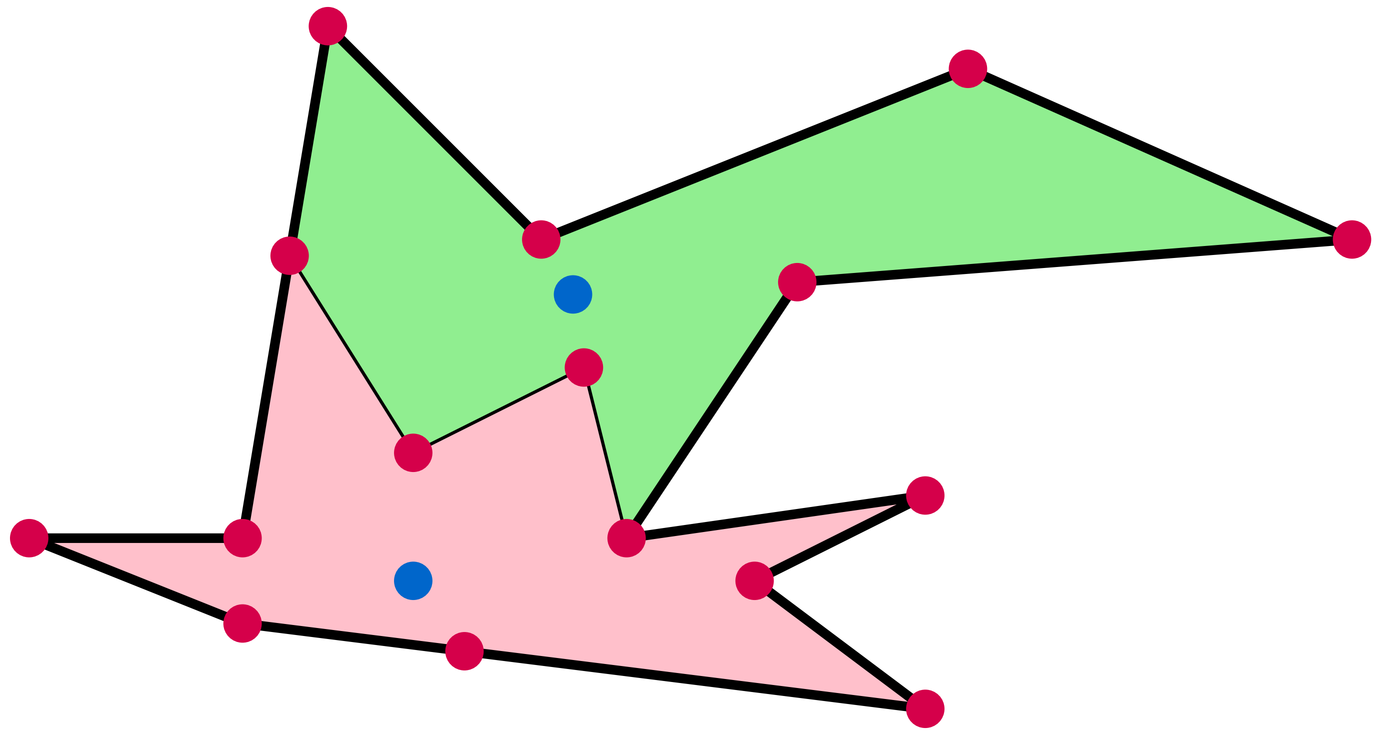
Algorithm Outline

1. Find small set of potential star-centers: S^{centers}
2. Find small set of potential "corner" points: S^{corners}
3. Use dynamic programming to find the minimum star partition



Algorithm Outline

1. Find small set of potential star-centers: S^{centers} $O(n^6)$
2. Find small set of potential “corner” points: S^{corners} $O(n^{32})$
3. Use dynamic programming to find the minimum star partition

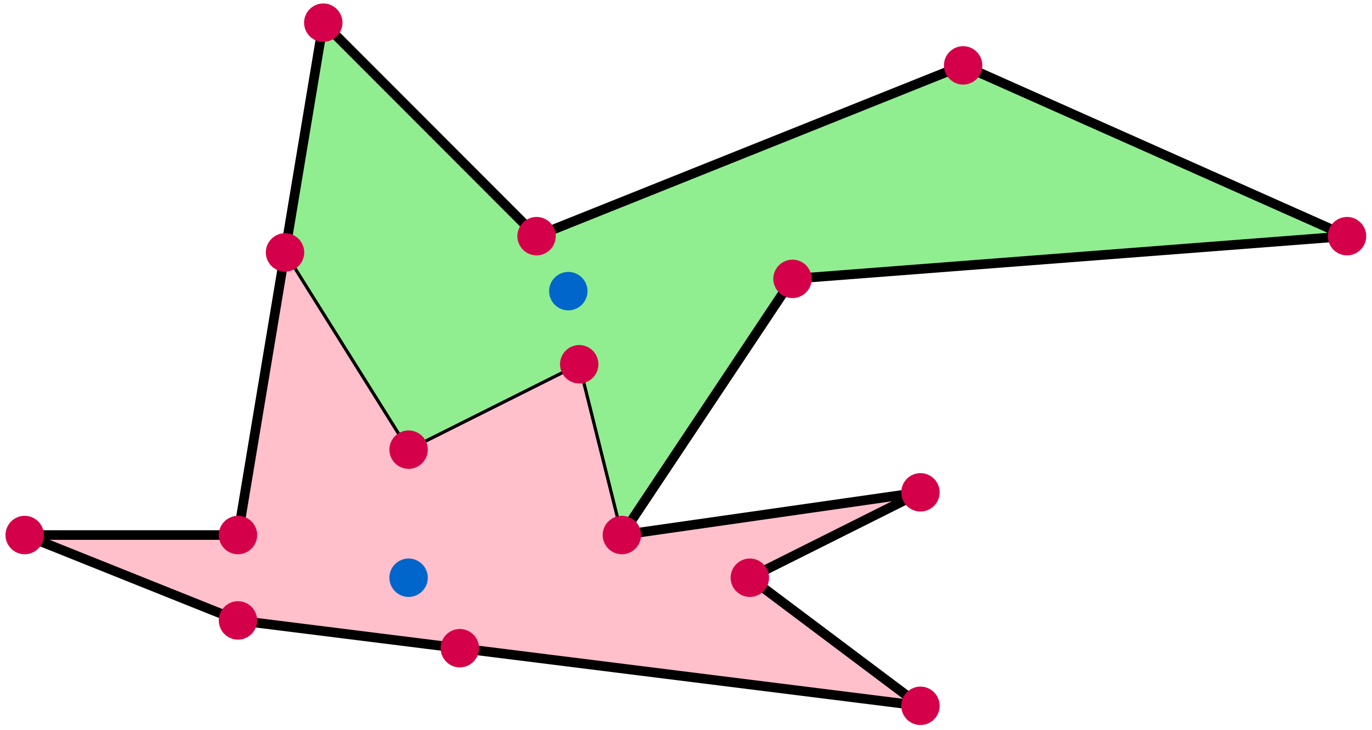


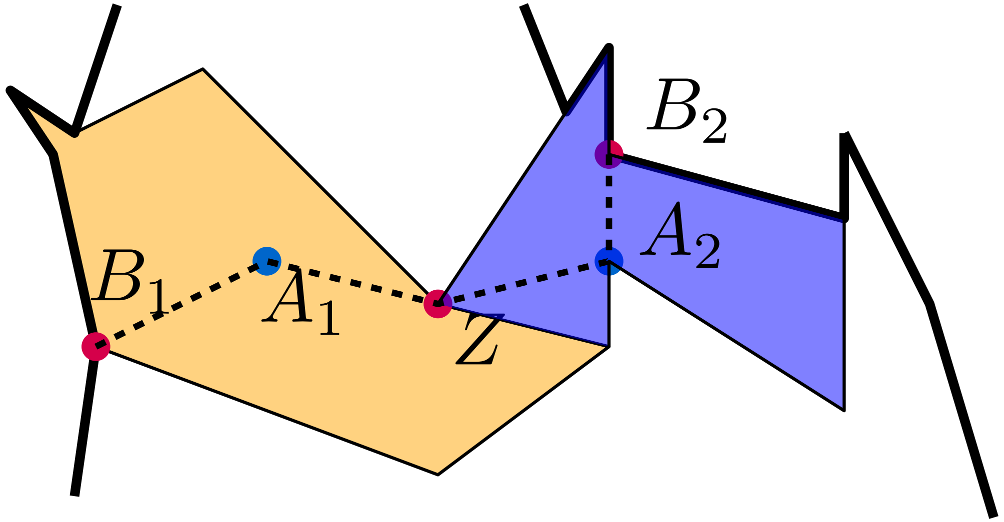
Algorithm Outline

1. Find small set of potential star-centers: S^{centers} $O(n^6)$
2. Find small set of potential "corner" points: S^{corners} $O(n^{32})$
3. Use dynamic programming to find the minimum star partition

Kind of easy

Every piece touches the boundary



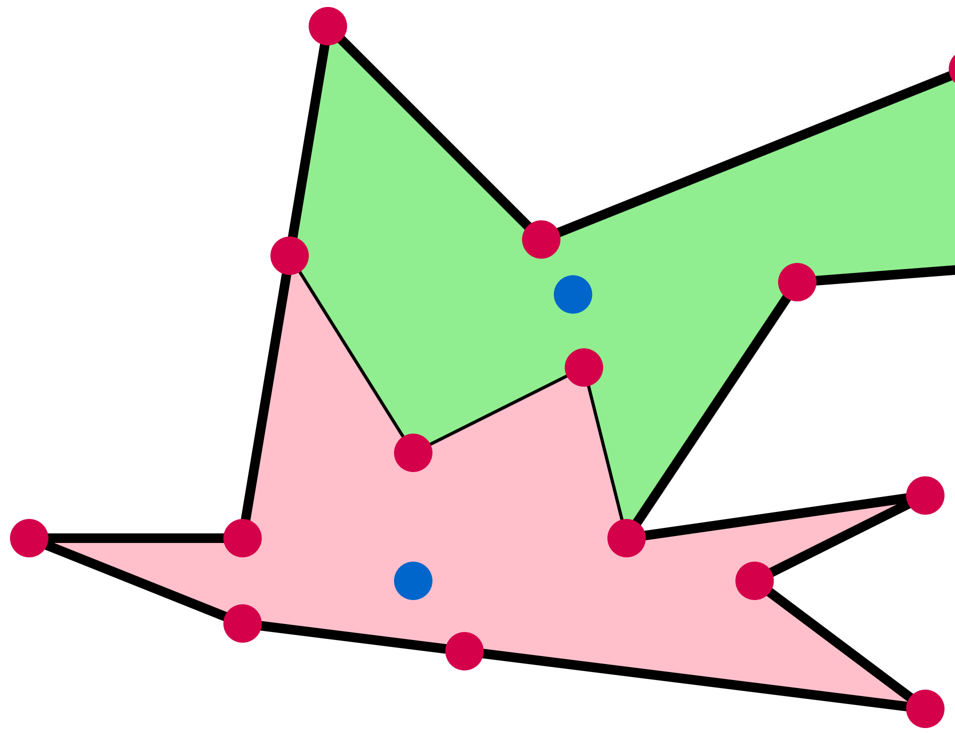


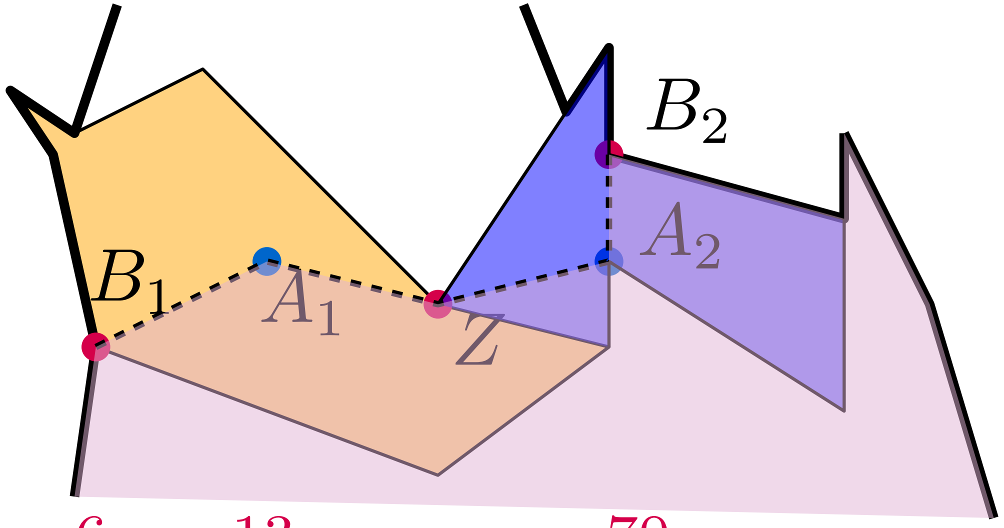
Algorithm Outline

1. Find small set of potential star-centers: S^{centers} $O(n^6)$
2. Find small set of potential "corner" points: S^{corners} $O(n^{32})$
3. Use dynamic programming to find the minimum star partition

Kind of easy

Every piece touches the boundary





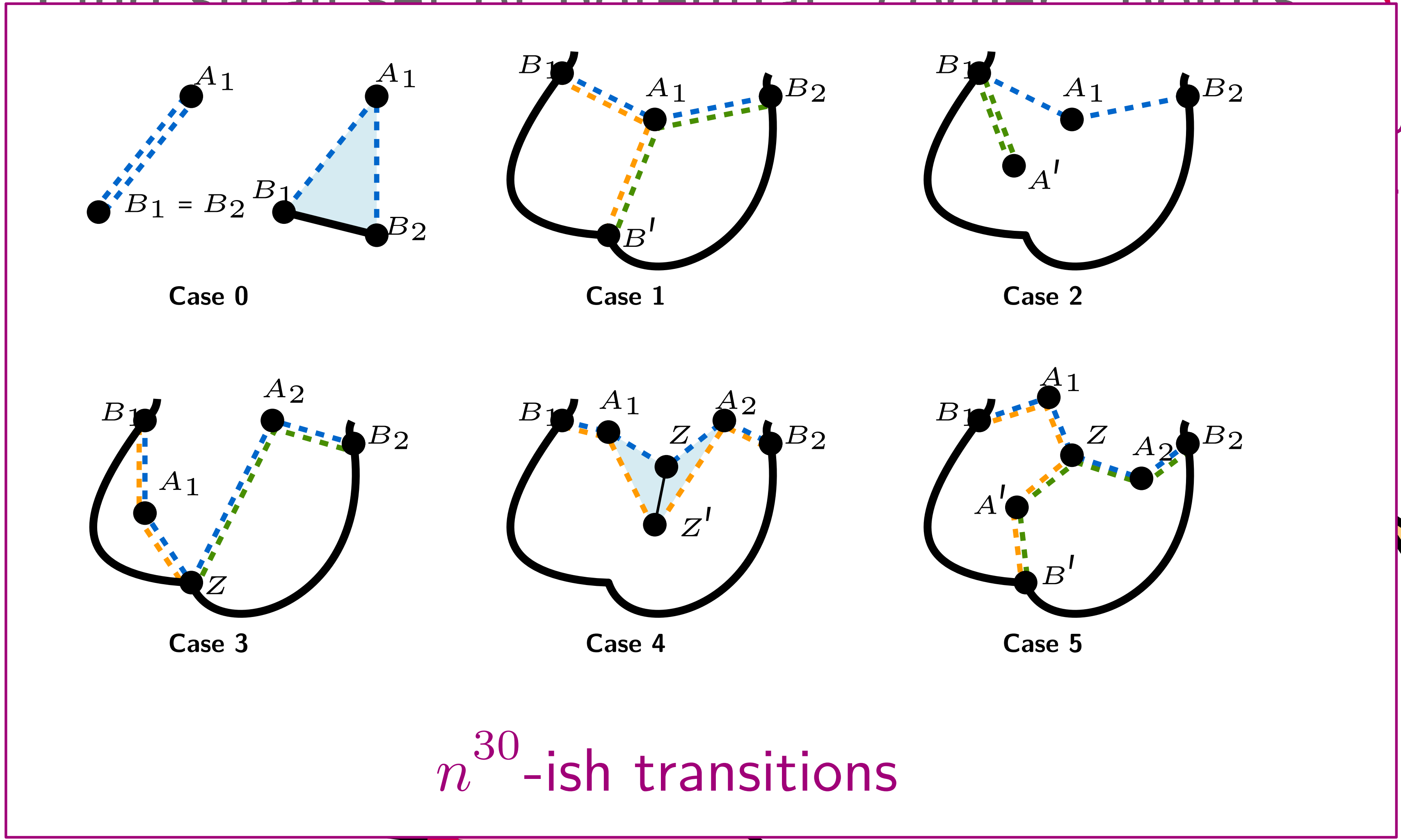
$O(n^{13} \cdot n^6 \cdot n^{32} \cdot n^6 \cdot n^{13}) = O(n^{70})$ states

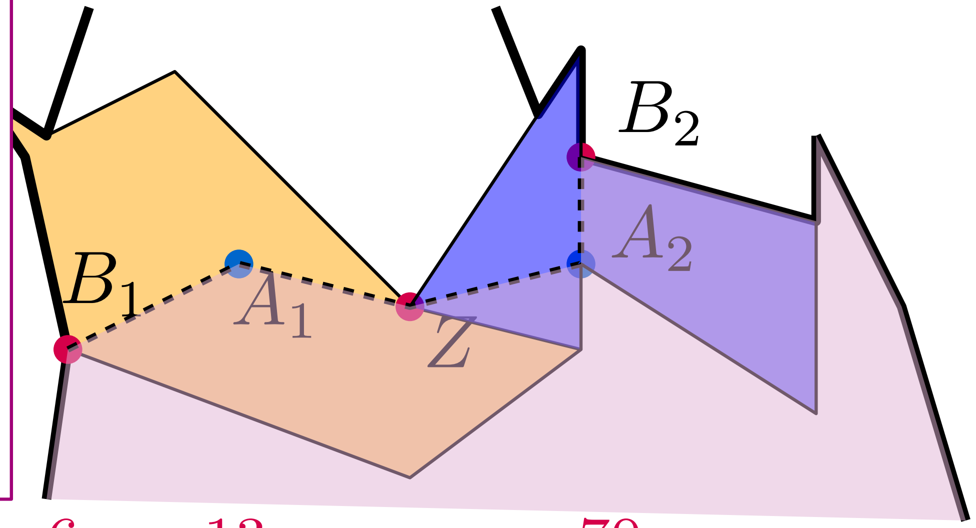
Algorithm Outline

1. Find small set of potential star-centers: S^{centers} $O(n^6)$
2. Find small set of potential "corner" points: S^{corners} $O(n^{32})$
3. ...um star partition

...easy

...iece touches the boundary





$O(n^{13} \cdot n^6 \cdot n^{32} \cdot n^6 \cdot n^{13}) = O(n^{70})$ states

Algorithm Outline

1. Find small set of potential star-centers: S^{centers}
2. Find small set of potential "corner" points: S^{corners}
3. Use dynamic programming to find the minimum star partition

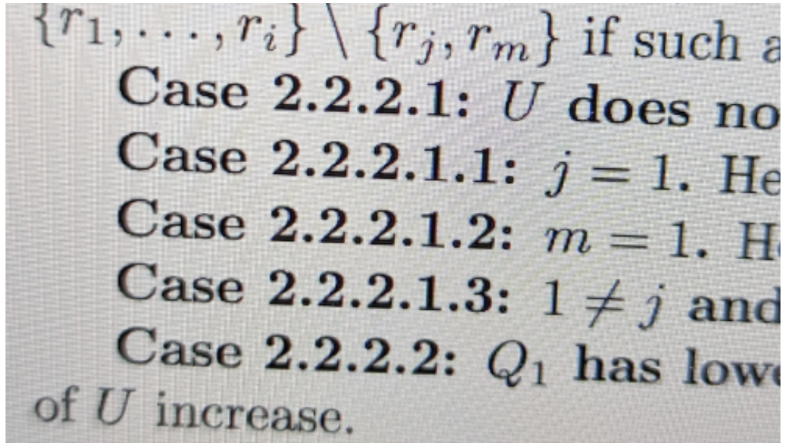
Kind of straightforward...

Algorithm Outline

1. Find small set of potential star-centers: S^{centers}
2. Find small set of potential "corner" points: S^{corners}
3. Use dynamic programming to find the minimum star partition

Kind of straightforward...

By case analysis...



$\{r_1, \ldots, r_i\} \setminus \{r_j, r_m\}$ if such a

Case 2.2.2.1: U does no

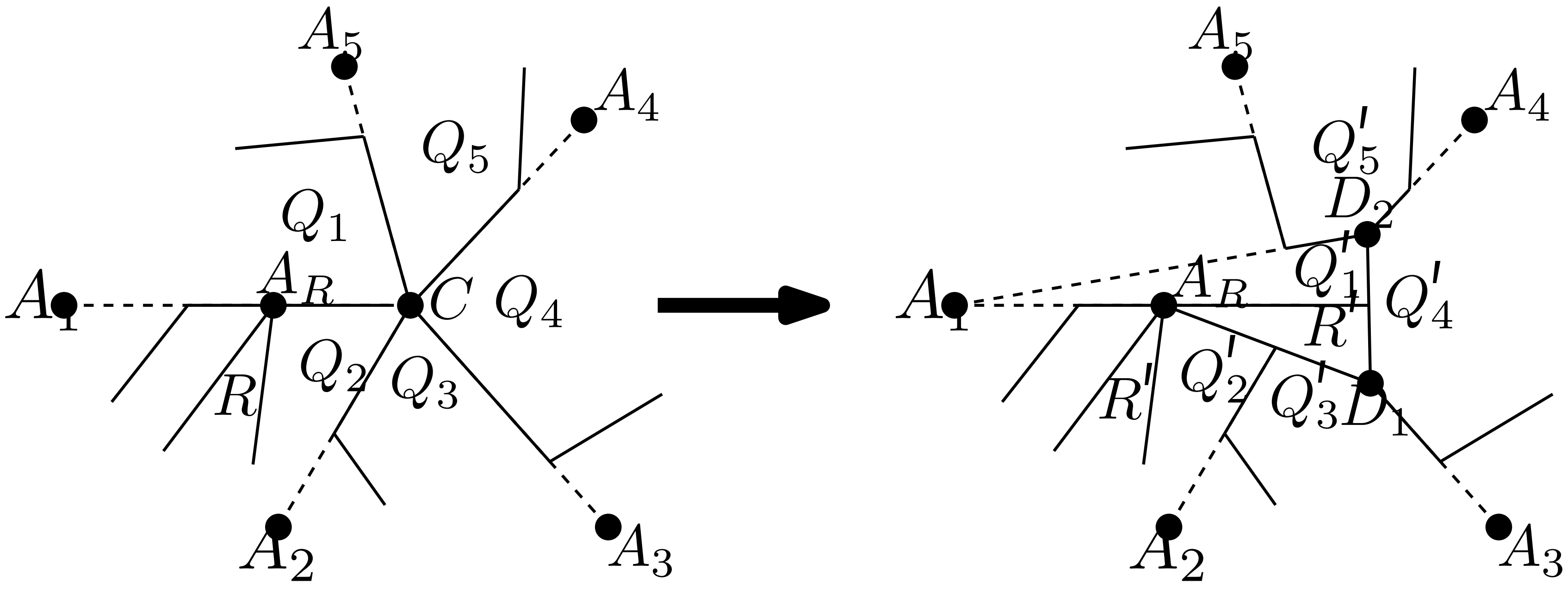
Case 2.2.2.1.1: $j = 1$. He

Case 2.2.2.1.2: $m = 1$. H

Case 2.2.2.1.3: $1 \neq j$ and

Case 2.2.2.2: Q_1 has low

of U increase.

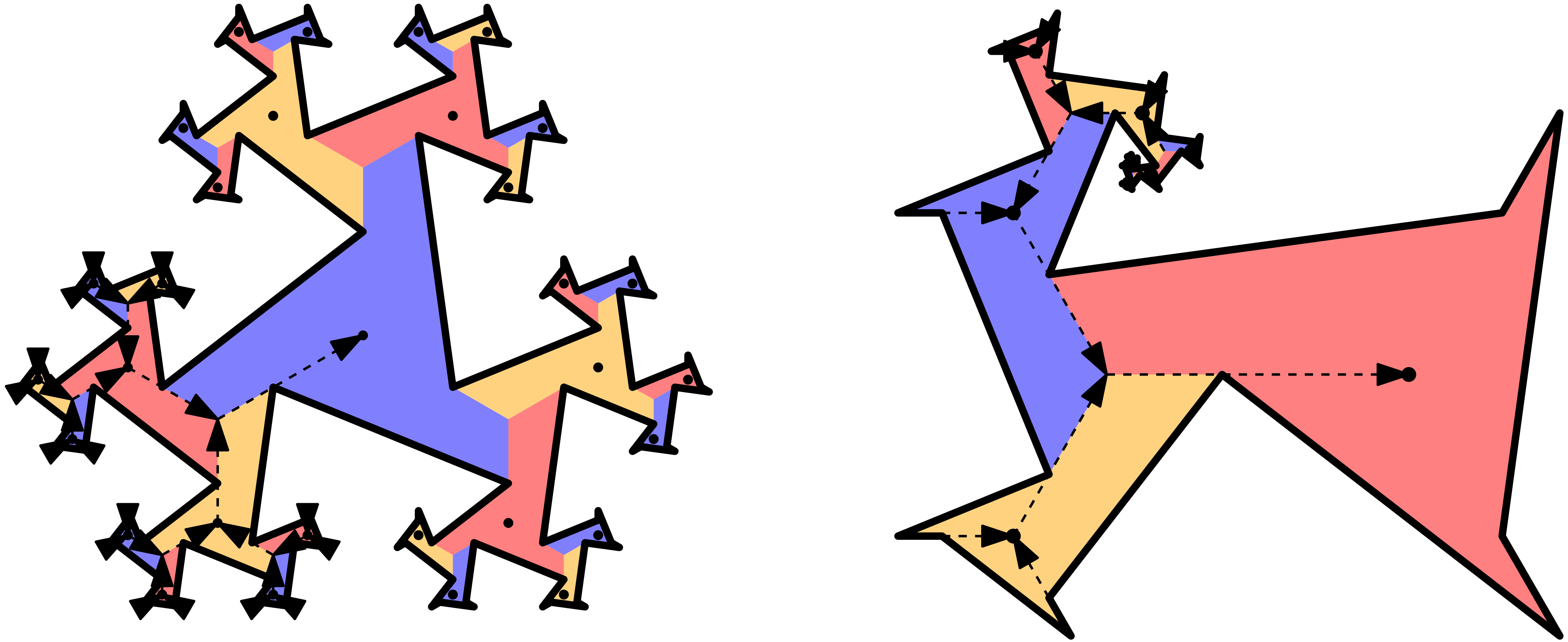


Hard part

1. Find small set of potential star-centers: S^{centers}

Hard part

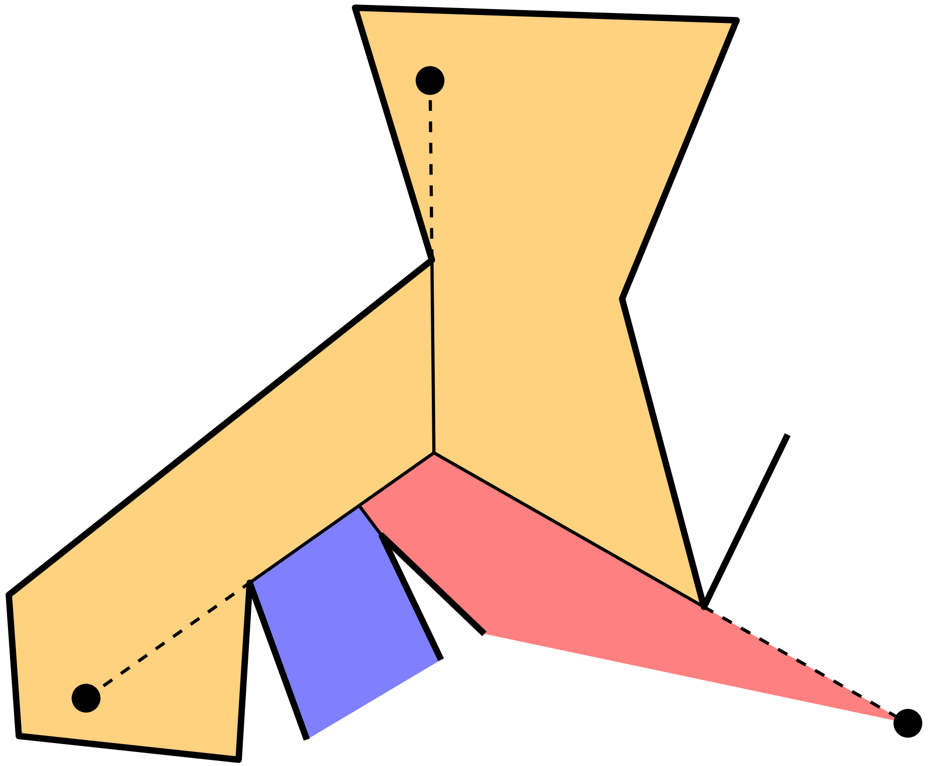
1. Find small set of potential star-centers: S^{centers}



Ingredient 1: “Tripods” form rooted trees

Hard part

1. Find small set of potential star-centers: S^{centers}

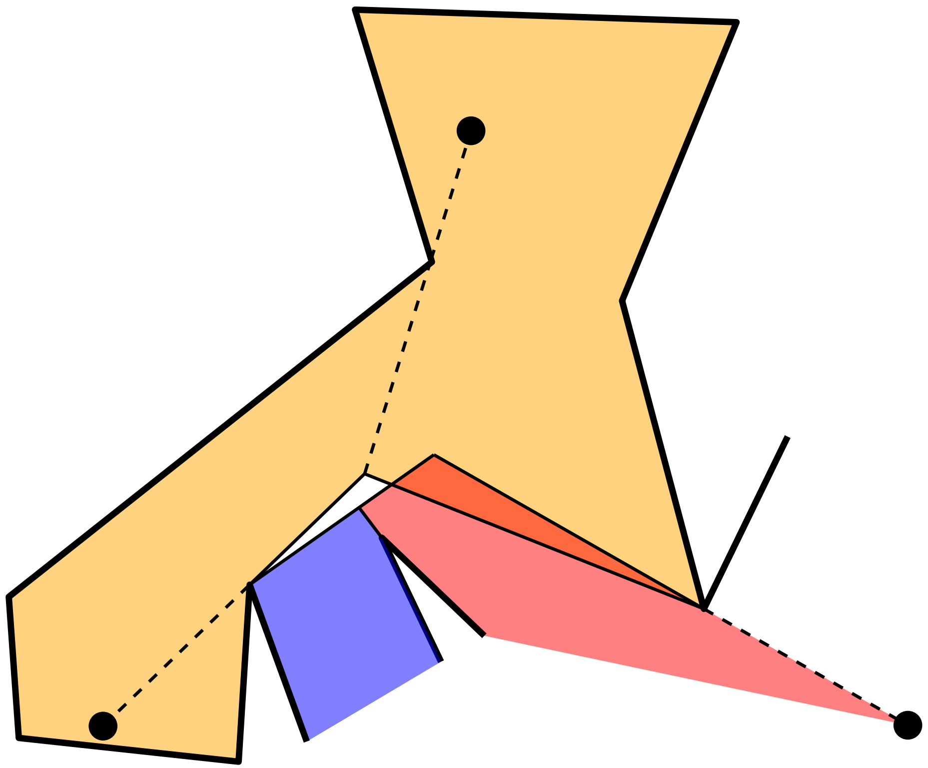


Ingredient 1: “Tripods” form rooted trees

Ingredient 2: “Greedy Choice”

Hard part

1. Find small set of potential star-centers: S^{centers}

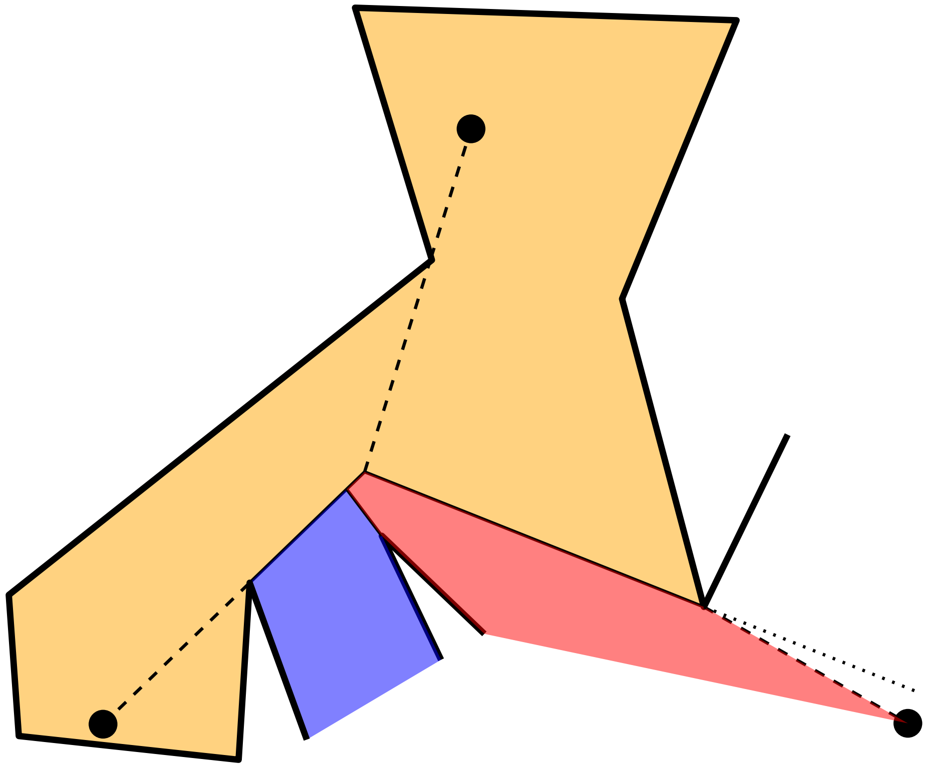


Ingredient 1: “Tripods” form rooted trees

Ingredient 2: “Greedy Choice”

Hard part

1. Find small set of potential star-centers: S^{centers}

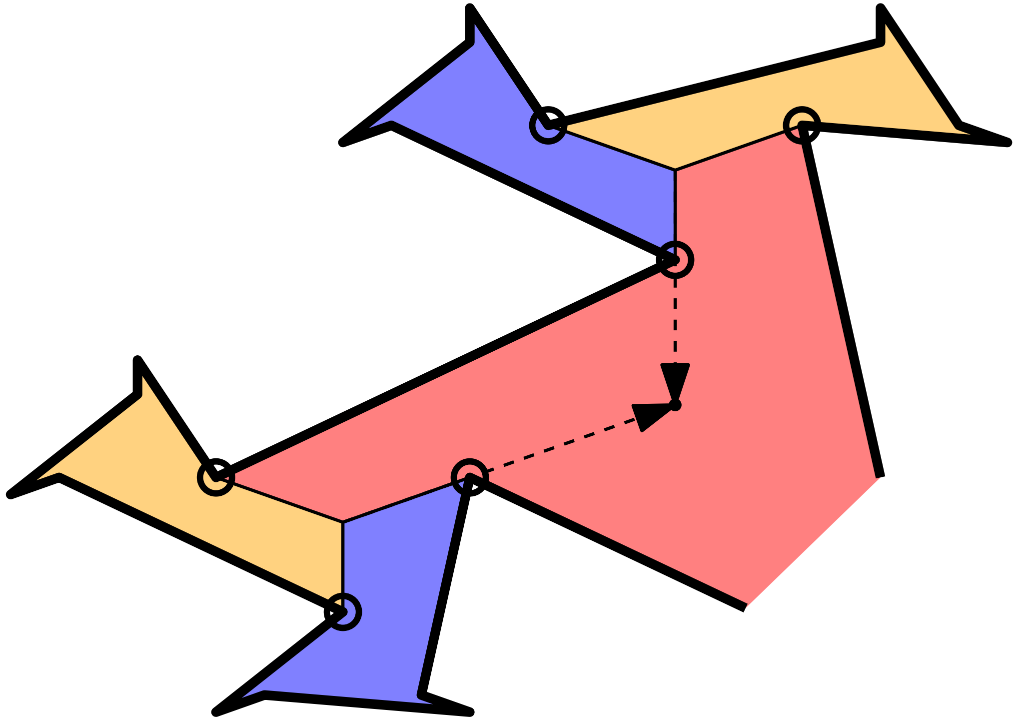


Ingredient 1: “Tripods” form rooted trees

Ingredient 2: “Greedy Choice”

Hard part

1. Find small set of potential star-centers: S^{centers}



$O(n^6)$ candidate star-centers

Ingredient 1: “Tripods” form rooted trees

Ingredient 2: “Greedy Choice”

Ingredient 3: Bootstrap whole algorithm on smaller polygons

Summary

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Summary

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Techniques:

Structural Properties of Optimial Solutions $+$ DP

Summary

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Techniques:

Structural Properties of Optimial Solutions + DP

Open Problems:

Triangle Partition?

Spiral Partition?

Fast (linear/quadratic) Approximation Algorithm?

3D?

Summary

Our Main Result:

Minimum Star Partition of Simple Polygons in $O(n^{107})$ time

Techniques:

Structural Properties of Optimal Solutions + DP

Open Problems:

Triangle Partition?

Spiral Partition?

Fast (linear/quadratic) Approximation Algorithm?

3D?

Thanks!