# *Approximation Algorithms for Some Geometric Packing and Covering Problems*

Joe Mitchell

Stony Brook University

In memoriam: Godfried Toussaint 7/31/44-7/14/19

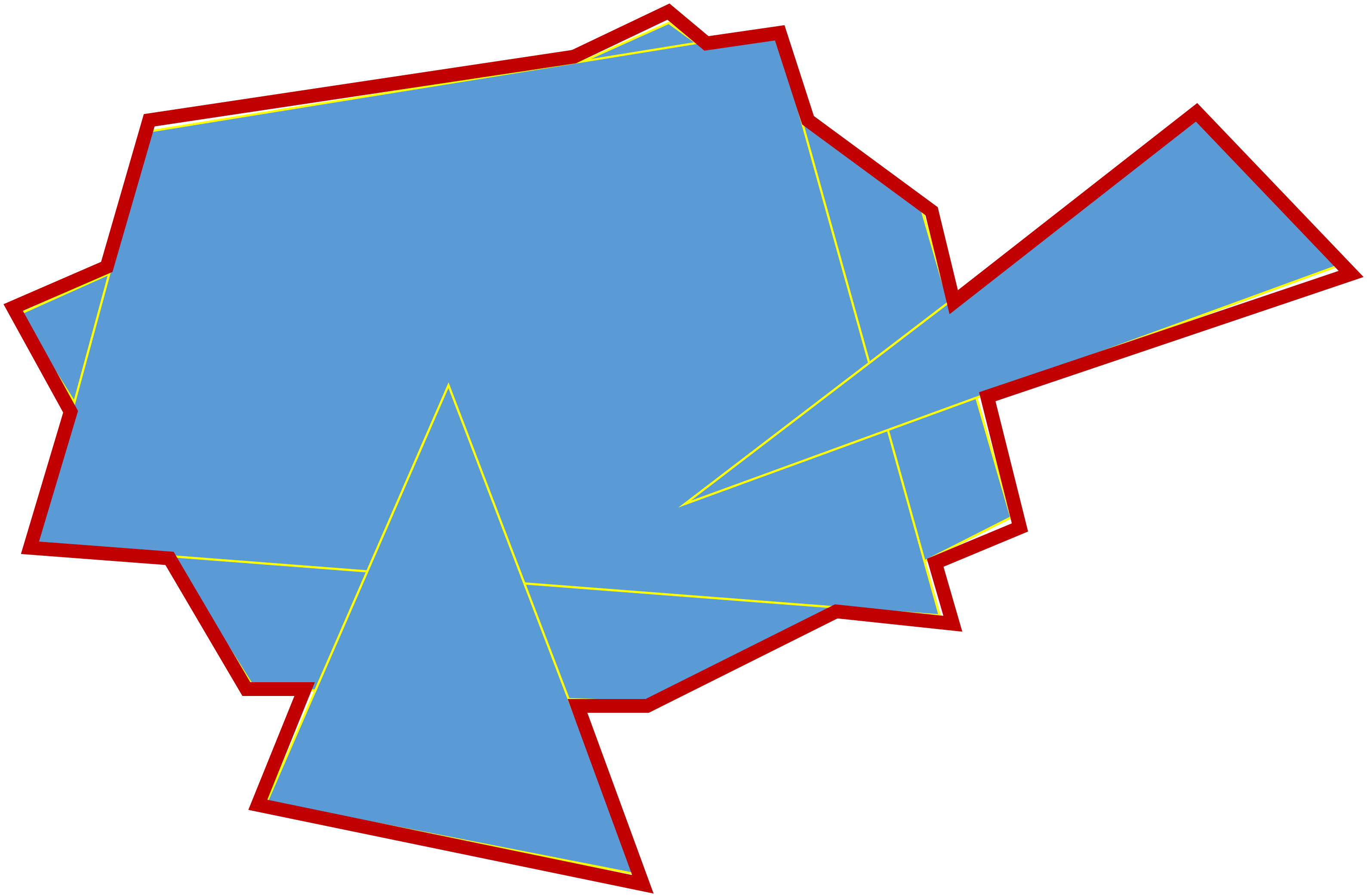
WADS, July 31, 2023

Support from NSF, DARPA, Sandia National Labs, US-Israel Binational Science Foundation

# Some Classic Packing and Covering Problems in Geometry

- Convex Cover: Cover a polygon P with fewest convex subpolygons
- Hidden Set: Max # points packed in P (no 2 see each other)
- Art Gallery/Guarding Problem: Cover P with fewest star-shaped subpolygons (fewest guards)
- Maximum Independent Set: geometric objects
- Mobile coverage: watchman routes
  - Min-length full coverage routes
  - Max coverage routes of bounded length
  - Coordinated routes: segment sweeping

# Convex Cover of Simple Polygon P

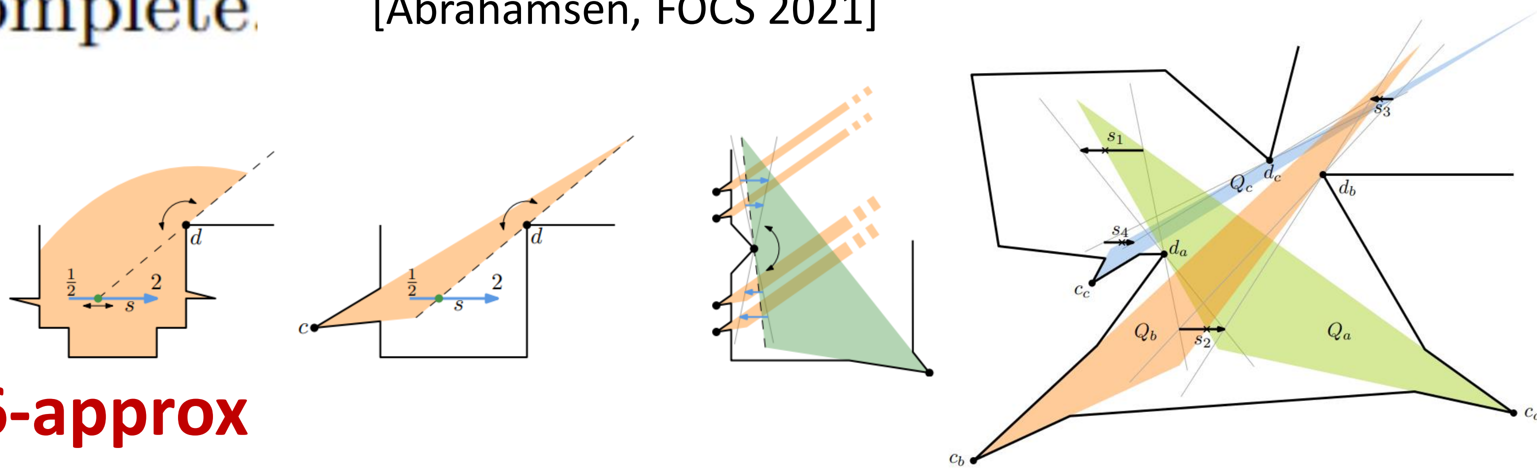


# Convex Cover of a Simple Polygon

- CC: Given a simple polygon P with n vertices, cover P with min # convex polygons within P
- NP-hard, APX-hard
- O(log n)-approx

[Eidenbenz, Widmeyer, 2003]

∃ℝ-complete [Abrahamsen, FOCS 2021]

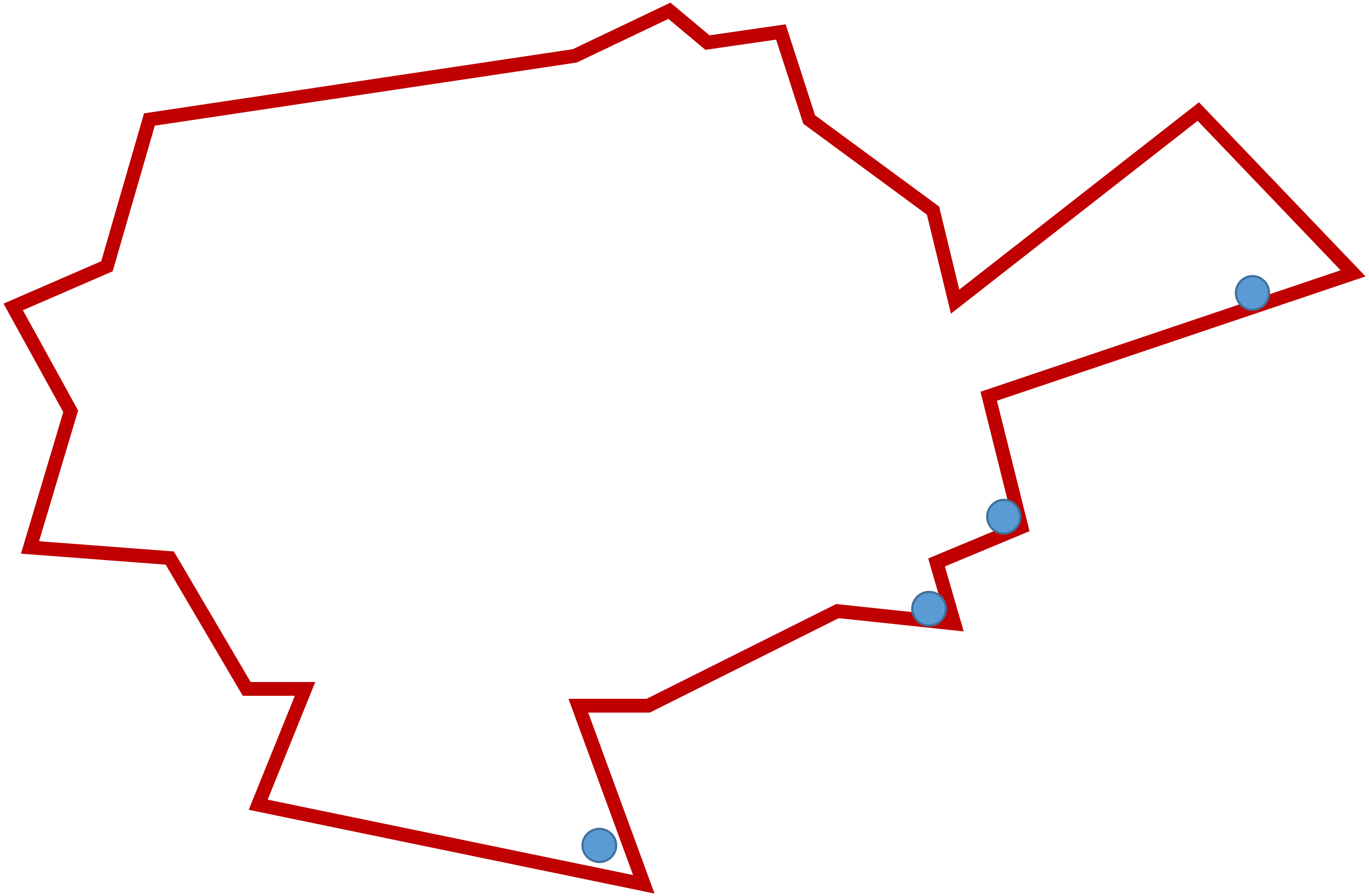



- **New: 6-approx**

[Reilly Browne, Prahlad Narasimham Kasthurirangan, JM, Valentin Polishchuk, to appear, FOCS 2023]

# Hidden Set of Points in P

HS: Given a simple polygon P, pack as many points in P so that no two see each other



# Hidden Set

- HS: Given a simple polygon P, pack as many points in P so that no two see each other

- No 2 hidden points can be in the same convex subset of P:

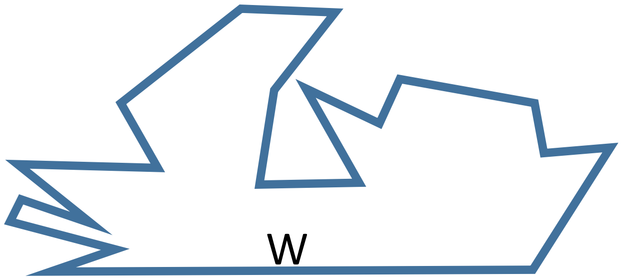
  $hs(P) \leq cc(P)$

- APX-hard [Eidenbenz, 2002]
- No prior approx to compute hs(P)

  For hidden ***vertex*** a ¼-approx is known [Alegria, Bhattacharya and Ghosh, EuroCG'19]
- **New: 1/8-approx for hs(P)**

[Reilly Browne, Prahlad Narasimham Kasthurirangan, JM, Valentin Polishchuk, to appear, FOCS 2023]

# Overview




- Give a 2-approx for cc(P) if P is weakly visible from W
  - Cover edges (except W) of P: formulate as a path cover
    - Obtain k convex polygons (k paths in min path cover in DAG)
    - Dilworth: “antichain”: k edges no two of which are strongly visible
      - Lemma: k hidden points, one on each edge of antichain
      - Thus, OPT cc(P) ≥ k
  - Cover all of P by adding k additional triangles, one associated with each path
- General P: In a window partition of P, no convex body intersects more than 3 faces, each of which is a weakly visible polygon

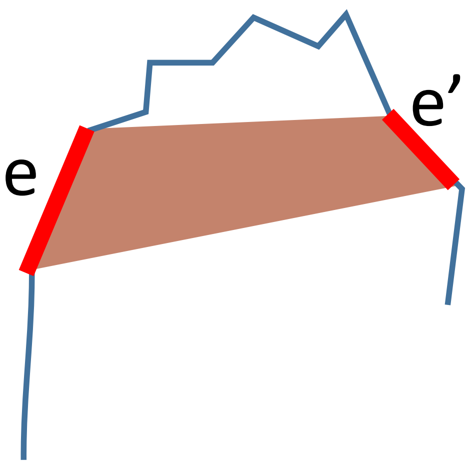
DAG: G=(V,E) where V={edges of polygon}, E={ (e,e’): CH(e,e’) $\subseteq$ P }

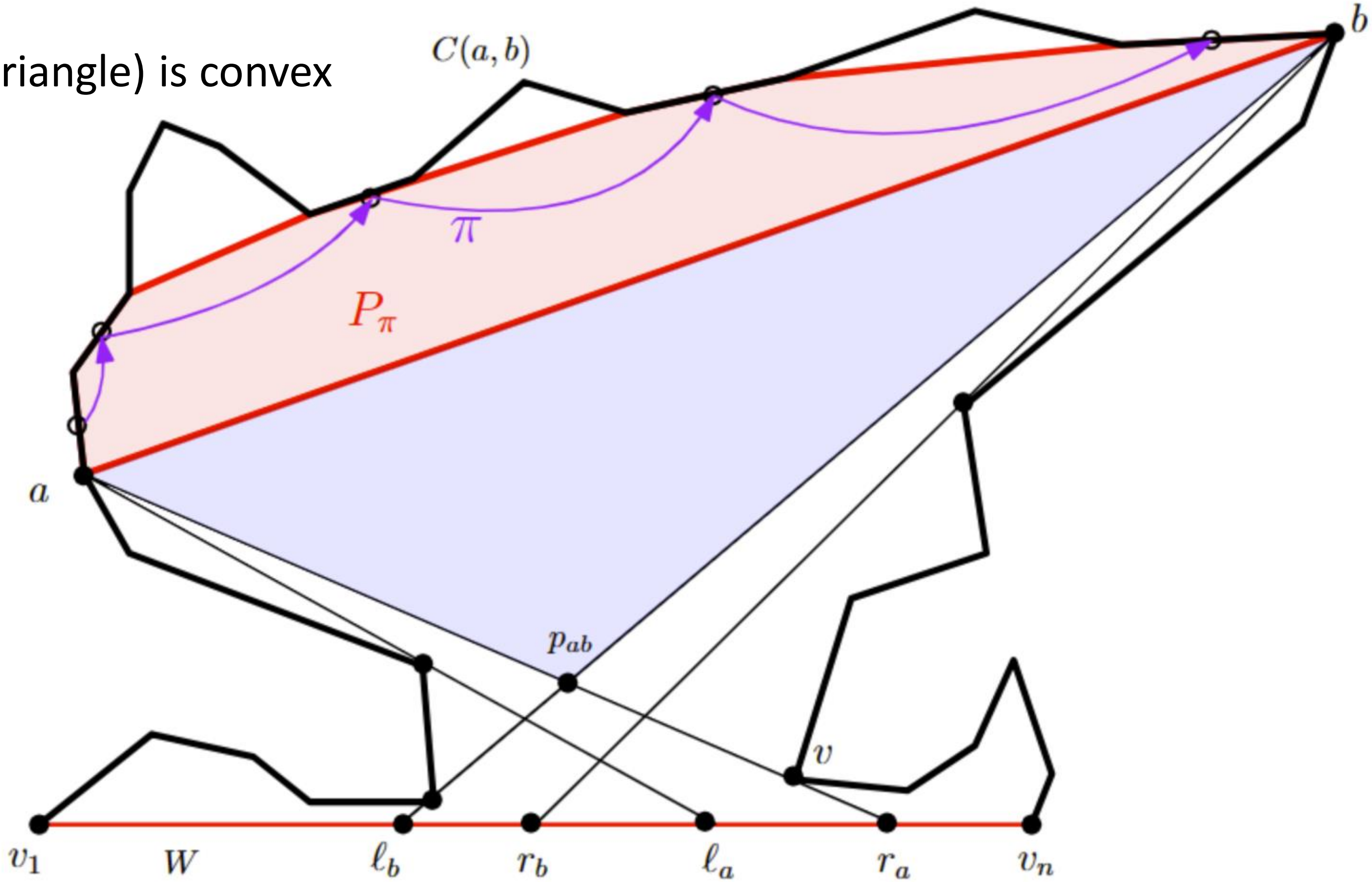
Path cover problem: Cover all nodes V with fewest directed paths in G.

[solve: using flows; [CLRS] ]

$P_\pi$ (red) is convex

$P'_\pi$ (red union blue triangle) is convex

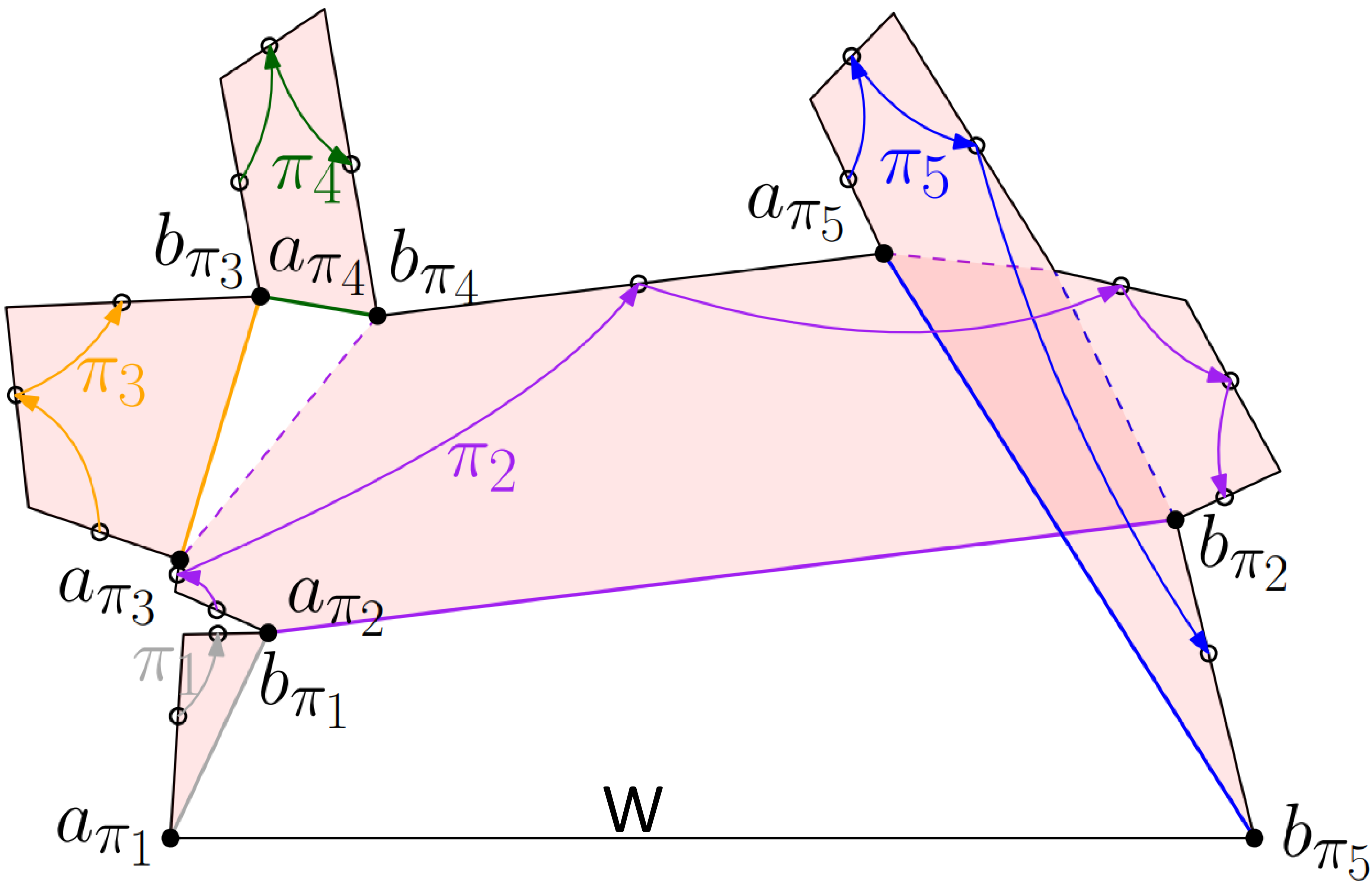


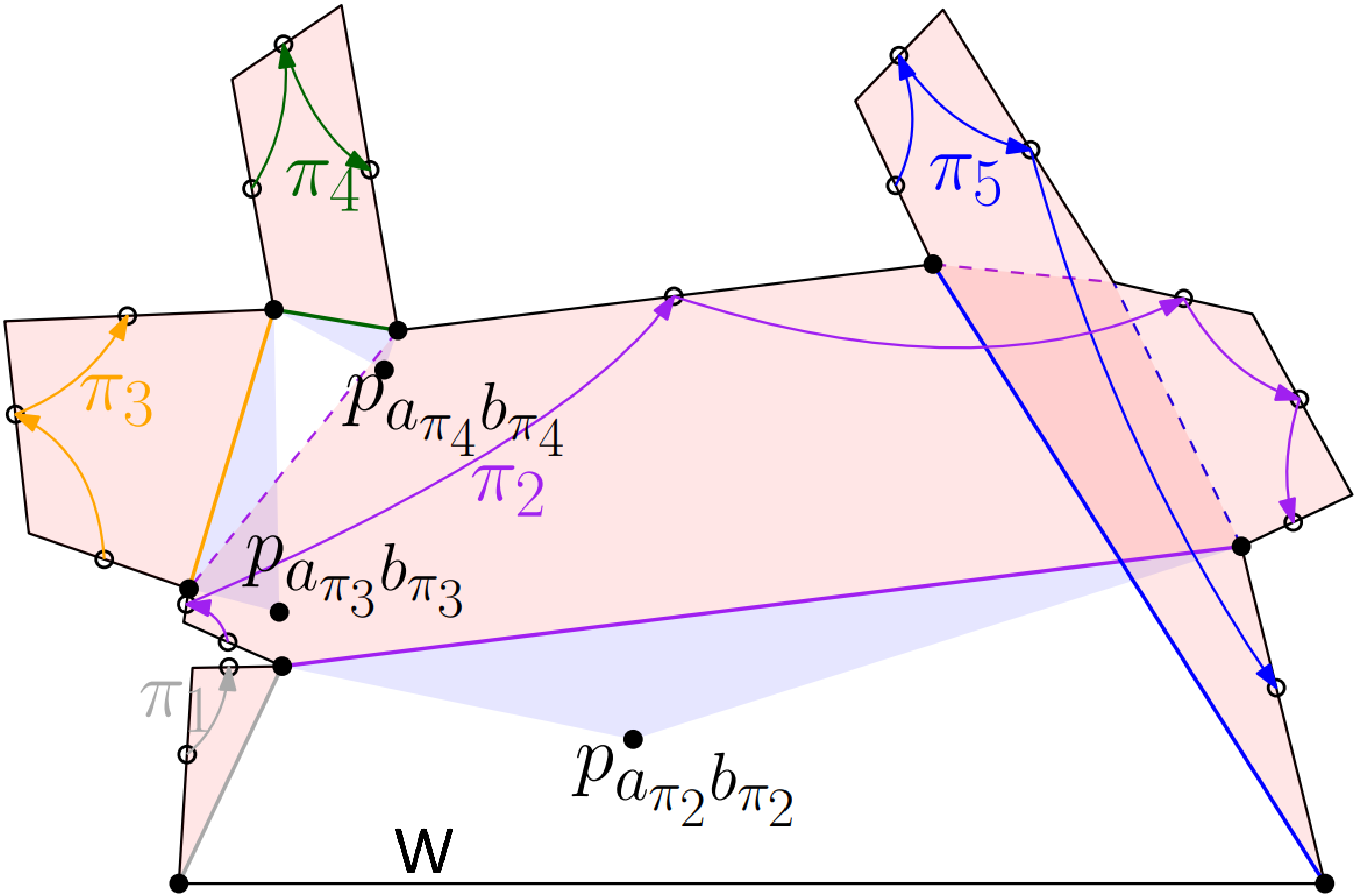

Property of weakly visible P:

**Fact 3.1.** [Chord property] *If $ab \cap C(a,b) = \{a,b\}$, then $a$ and $b$ see each other.*

# Example: 5 paths covering all edges of P



# Augment with blue triangles: $P'_{\pi}$



DAG: G=(V,E) where V={edges of polygon}, E={ (e,e'): CH(e,e') $\subseteq$ P }

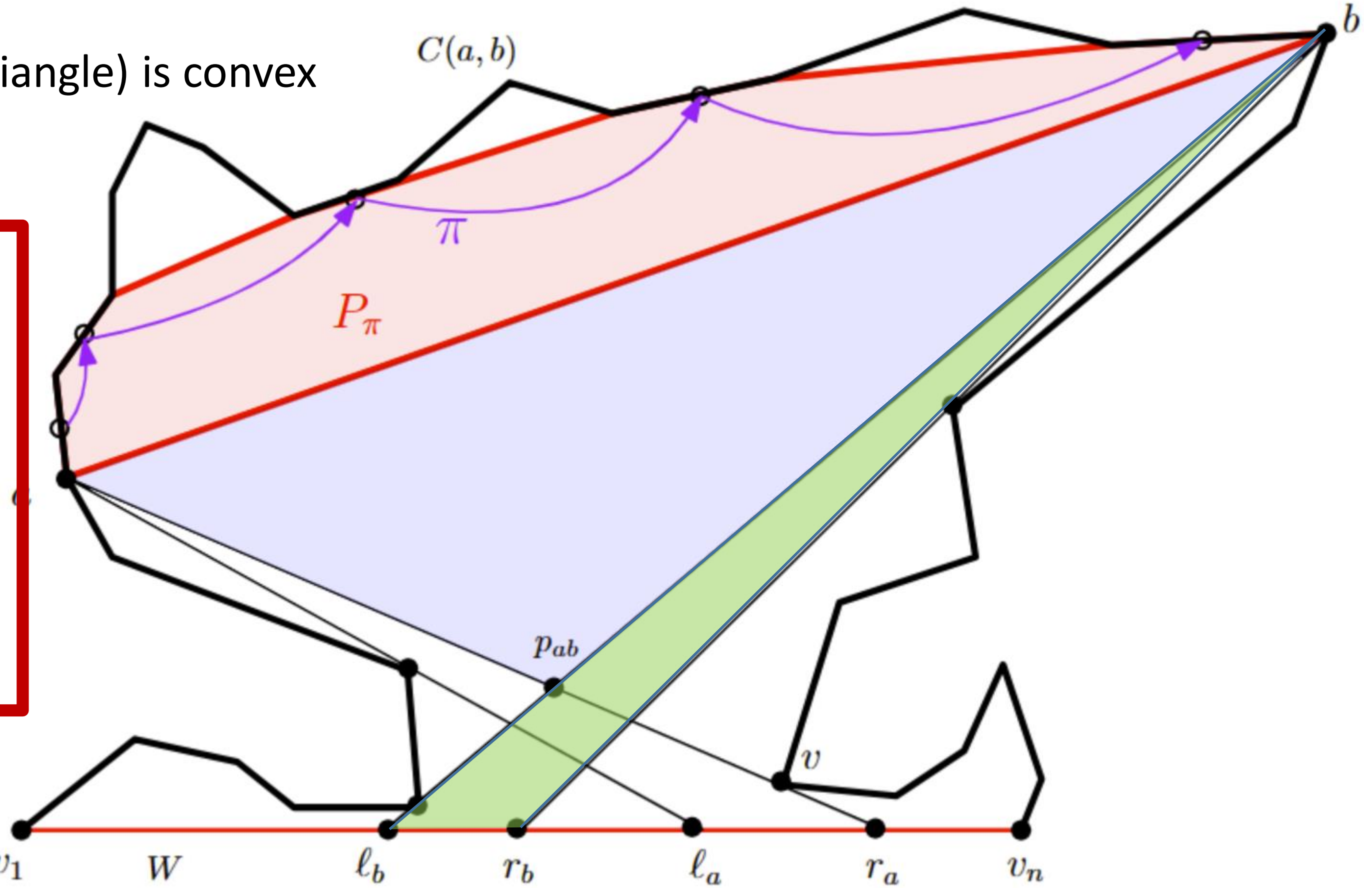
Path cover problem: Cover all nodes V with fewest directed paths in G.

[solve: max bipartite matching; [CLRS] ]

$P_\pi$ (red) is convex

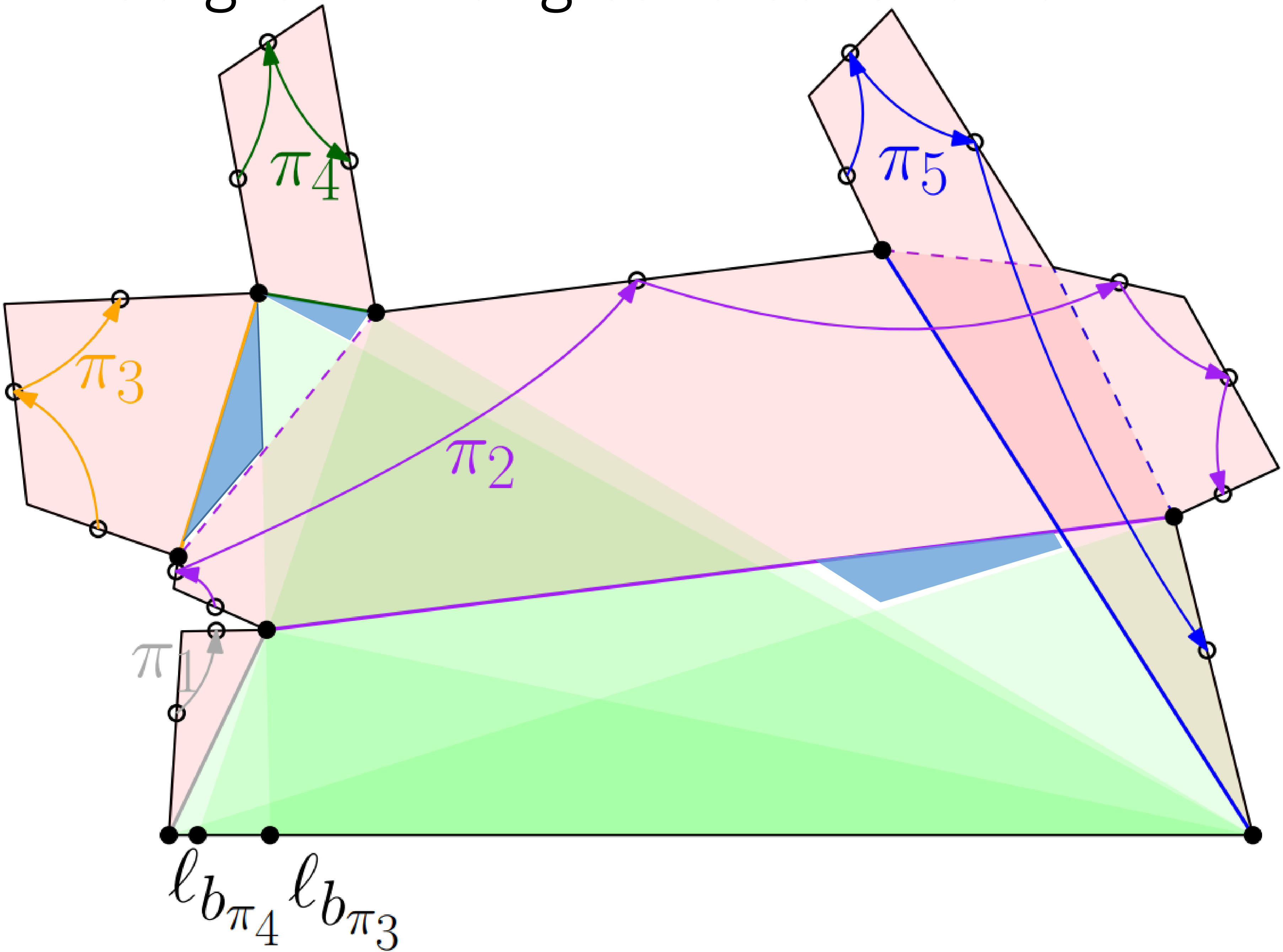
$P'_\pi$ (red union blue triangle) is convex

Add **green triangle**, one per path/ $P'_\pi$

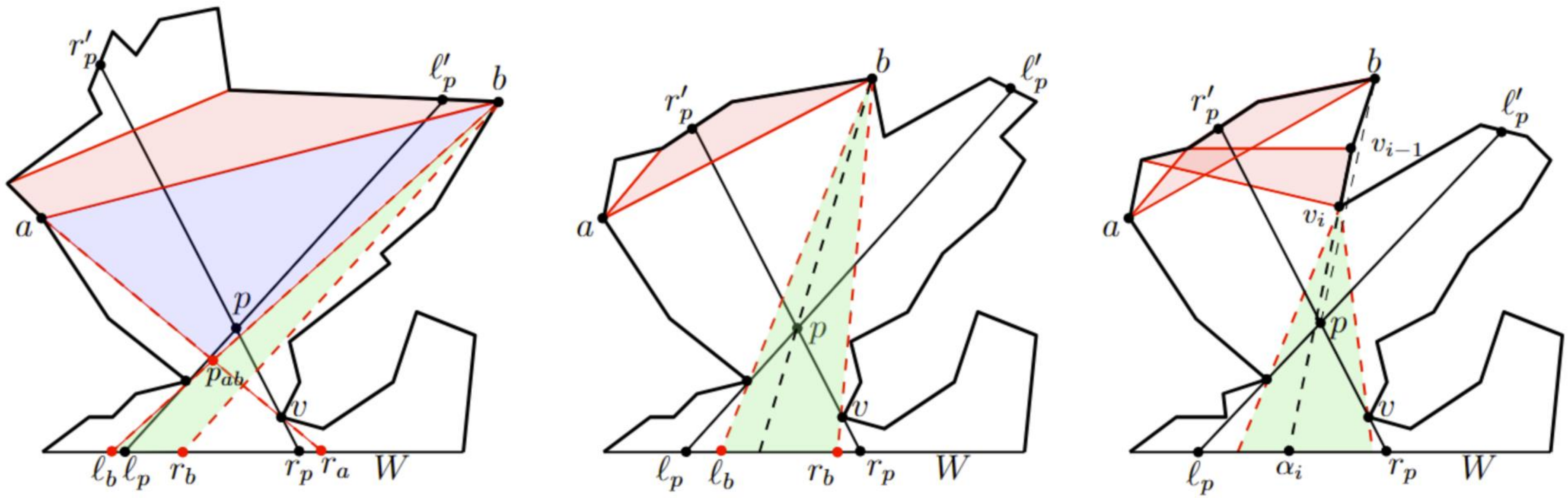



**Fact 3.1.** [Chord property] *If* $ab \cap C(a,b) = \{a,b\}$*, then a and b see each other.*

# Add green triangles to cover all of P



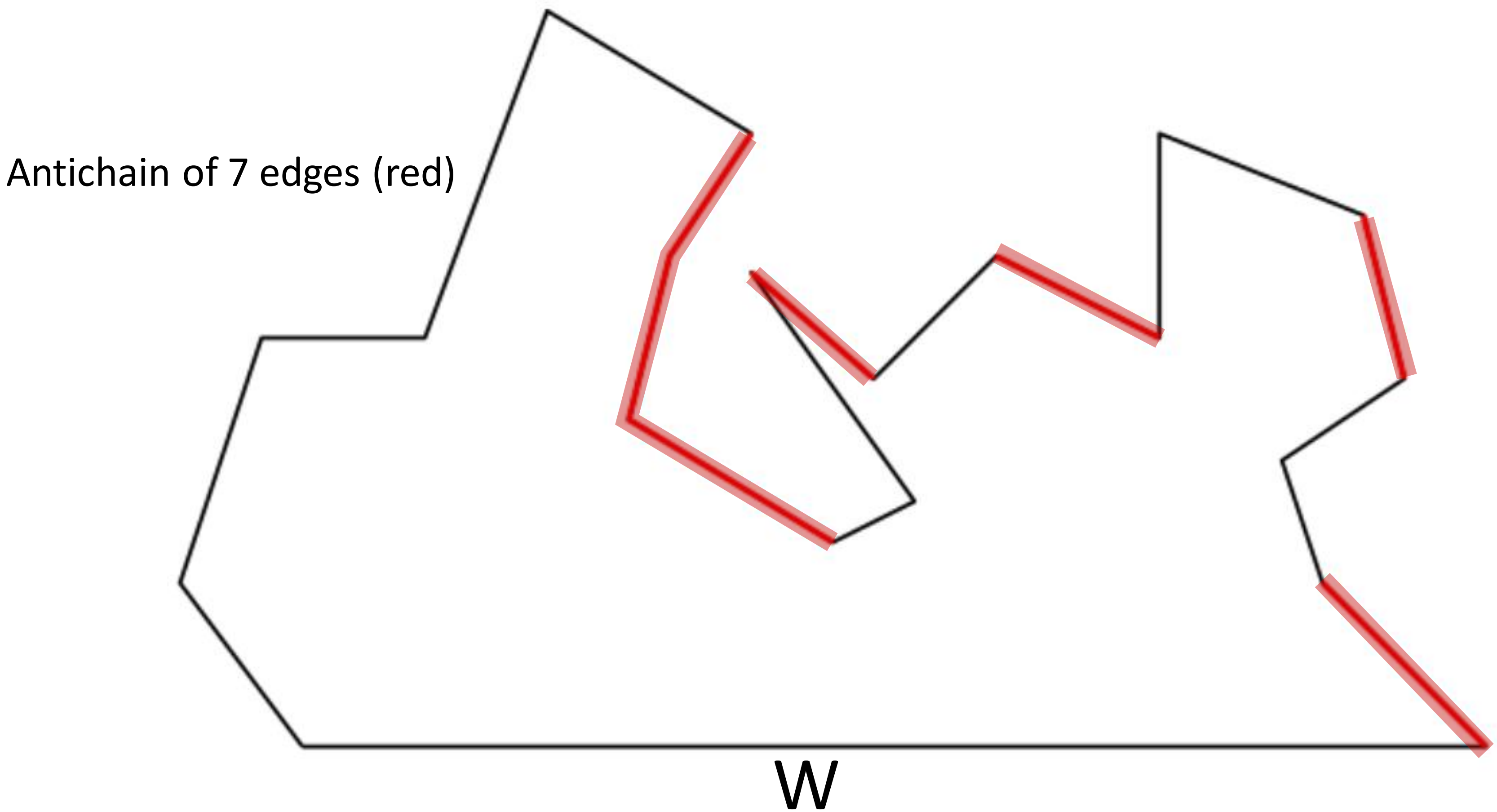
# Lemma: All of P is covered by augmented path polygons, green triangles




**Theorem 3.5.** *For a weakly visible polygon P, there is a polynomial-time algorithm to compute a set of at most 2k convex polygons within P that cover P, where k is the size of an optimal path cover of G.*

# Hidden Set from Antichain of Edges

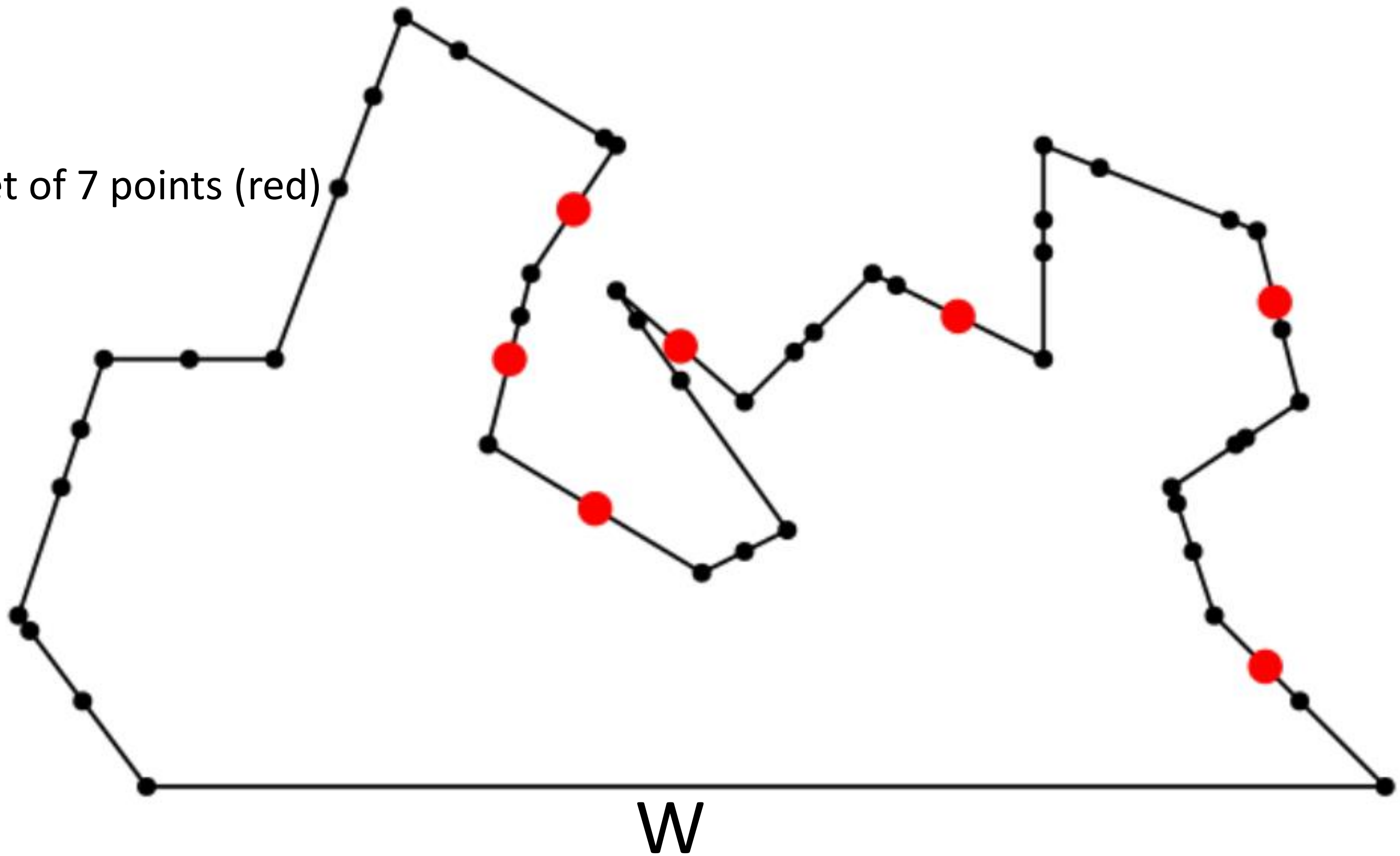
**Lemma 3.6.** *Given an antichain $I$ in $G$, we can compute in polynomial time a set $H$ of $|I|$ points, each interior to one of the edges of $I$, such that $H$ is a hidden set (no two points of $H$ are visible to each other).*

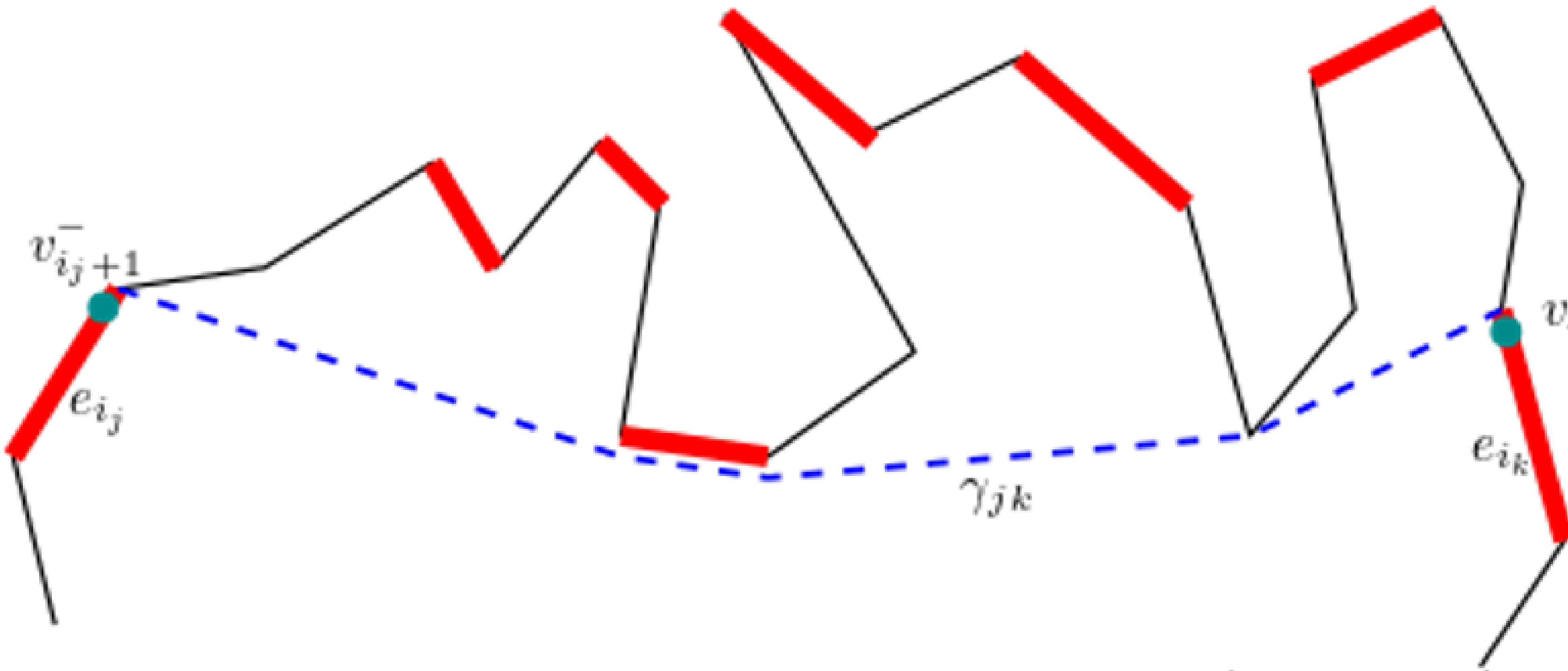


# Hidden Set from Antichain of Edges

**Lemma 3.6.** *Given an antichain $I$ in $G$, we can compute in polynomial time a set $H$ of $|I|$ points, each interior to one of the edges of $I$, such that $H$ is a hidden set (no two points of $H$ are visible to each other).*

Hidden set of 7 points (red)






4 pockets induced by $\gamma_{jk}$, with one degenerate pocket (single edge)

# Weakly Visible Simple Polygon P

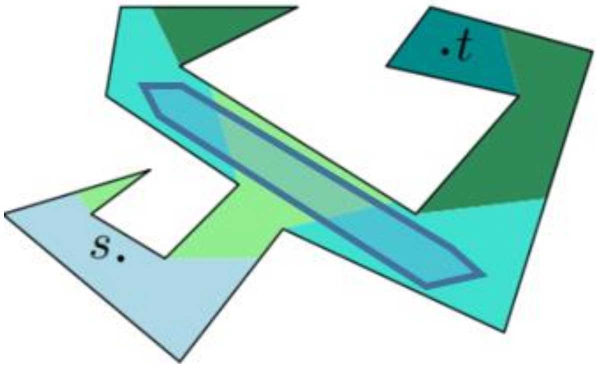
$$k = |H| \leq hs(P) \leq cc(P) \leq |B| = 2k$$

- Theorem: For weakly visible P, we can compute a set B of at most 2k convex polygons covering P, where k= #paths=|largest antichain|

2-approx for CC(P)

½-approx for HS(P)

# General Simple Polygon P



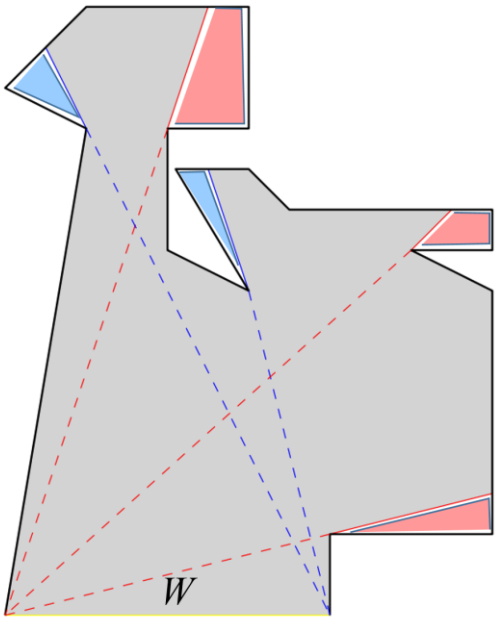

- Use a window partition of P; faces are weakly visible

- **Lemma**:

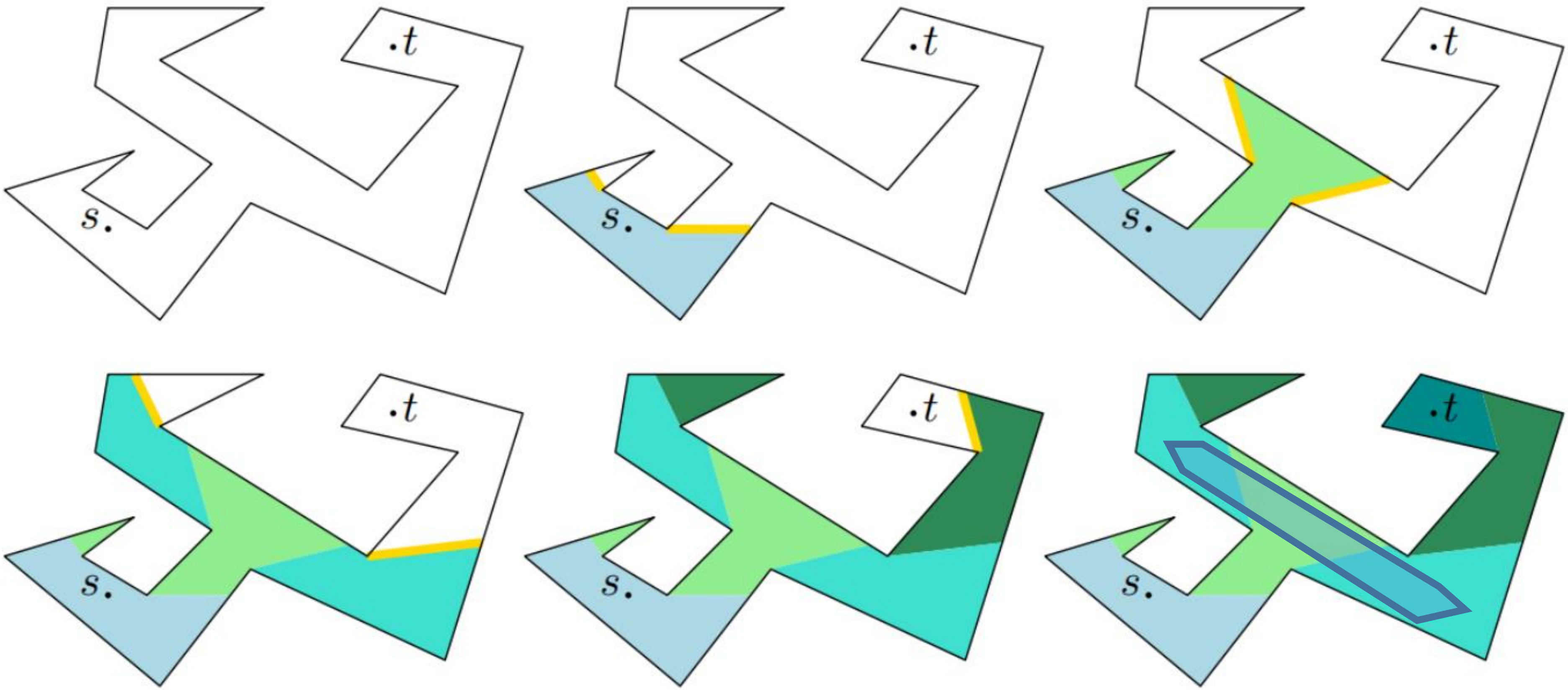
Any convex K in P intersects ≤3 faces

Faces are of 4 types: even/odd link distance, from source s; left/right window pocket.

No point p in a face of type i∈{1,2,3,4} can see any point in another face of type i



# Window Partition: Link Distance Map LDM(s)




Staged illumination: Windows are yellow; colors indicate regions of the same link distance from s

# Summary

Polytime 6-approx algorithm for convex cover (CC) of a simple n-gon; 1/8-approx algorithm for hidden set (HS)

$O(n^{2+o(1)})$ time

Prior: O(log n)-approx for CC, in time $O(n^{29} \log n)$ [Eidenbenz, Widmayer, SICOMP 2003]

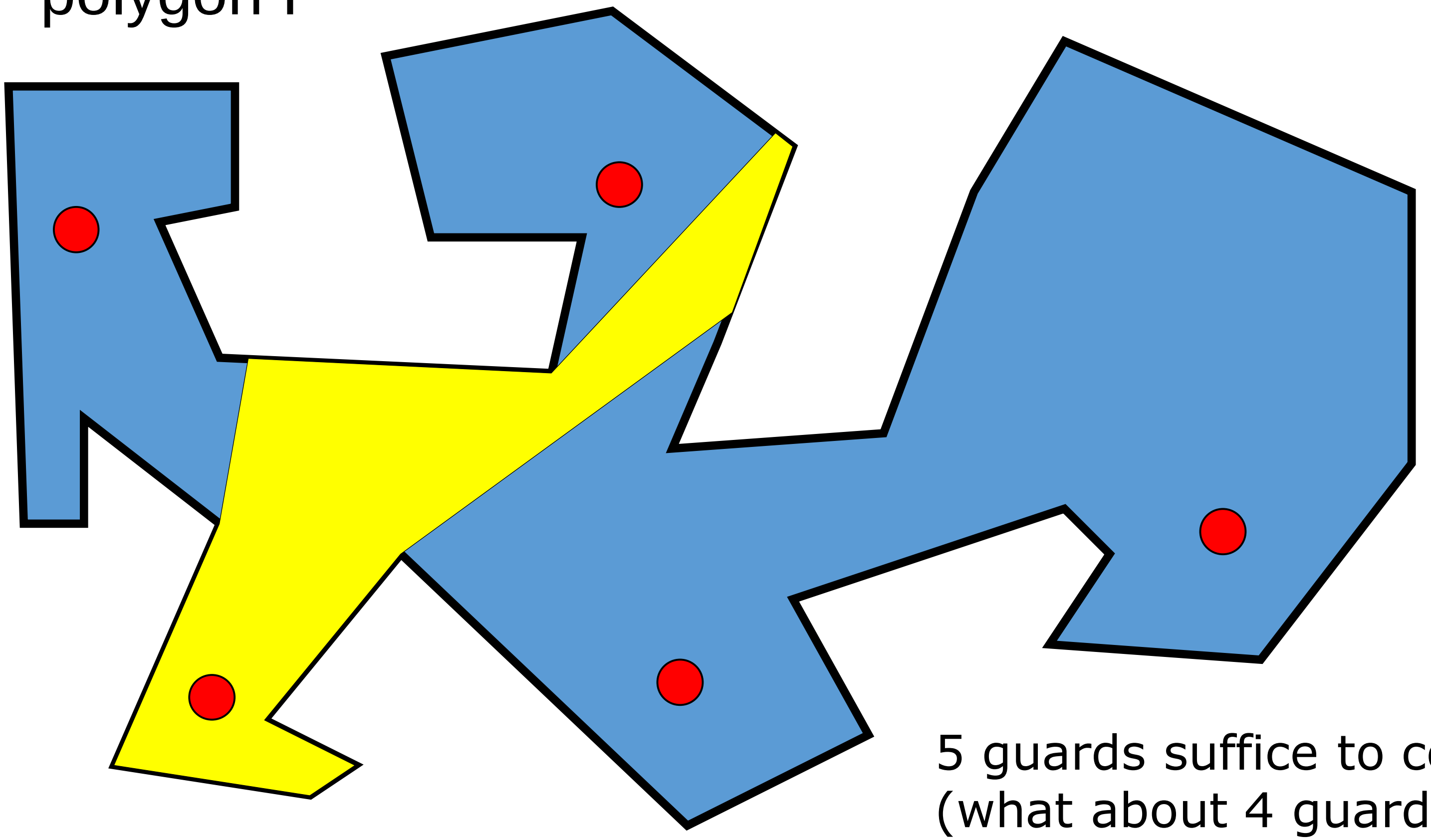
Factors are better for weakly visible polygons:
2-approx for CC, ½-approx for HS

Combinatorial bounds shown:
- cc(P) ≤ 8*hs(P), confirming a conjecture from [Browne & Chiu, YRF'22]
- cc(P) ≤ 2*hs(P), for weakly visible P

Polygons with holes?
- HS cannot be $n^{\varepsilon}$ -approximated, for some ε>0 (unless P=NP) [Eidenbenz]
- CC is APX-hard, has O(log n)-approx [EW'03]: Can this be improved?

# Min-Guard Coverage Problem

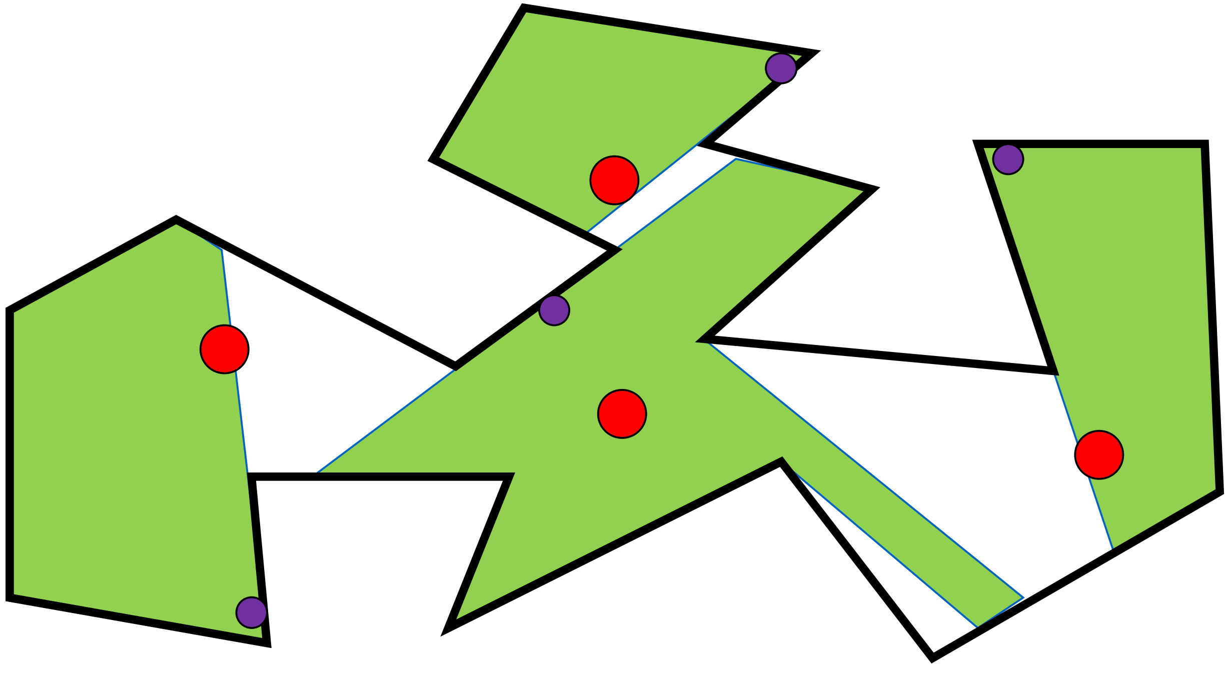
- Determine a small set of guards to see all of a given polygon P



5 guards suffice to cover P
(what about 4 guards? 3?)

# Lower Bound on g(P)

- Fact: If we can “pack” w visibility independent witness points, then g(P) ≥ w.



$g(P) \geq 4$

$g(P) \leq 4$; thus, $g(P)=4$

# Witness Number

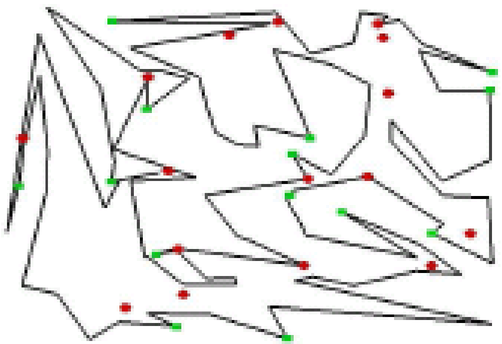
- Let w(P) = max # of independent witness points possible in a set of visibility independent witness points for P
- Then, g(P) ≥ w(P)
- Compute g(P): NP-hard, APX-hard, $\exists\mathbb{R}$-complete
- DP allows one to compute w(P) [at least if candidates given]

Some polygons have g(P)=w(P); I call these *perfect polygons* – they are very special; many (most?) polygons P have a "gap": g(P)>w(P) Characterize perfect polygons?

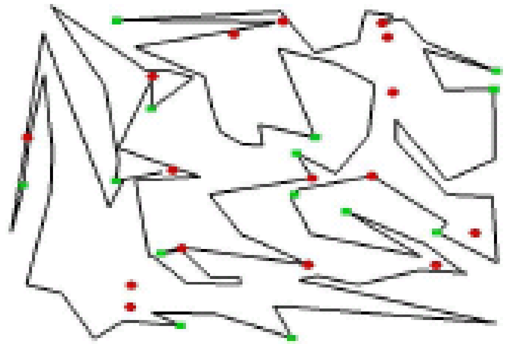
# Guarding: Experimental Investigations

- Early work:
  - [AMP] Proposed/implemented several heuristics for computing guards [Amit,M,Packer]
  - Experimental analysis and comparison
  - Compute both upper bounds and lower bounds on OPT, so we can bound how close to OPT we get
  - Conclude: heuristics work well in practice:
    - Either find OPT solution or close to optimal
    - Almost always 2-approx (always for "random" polygons)
- More recent: Sophisticated methods based on LP/IP, and understanding of combinatorial structure
  Extensive experiments, achieving optimal solutions
  [Sandor Fekete et al; Cid de Souza et al]
  www.ic.unicamp.br/~cid/Problem-instances/Art-Gallery/AGPPG

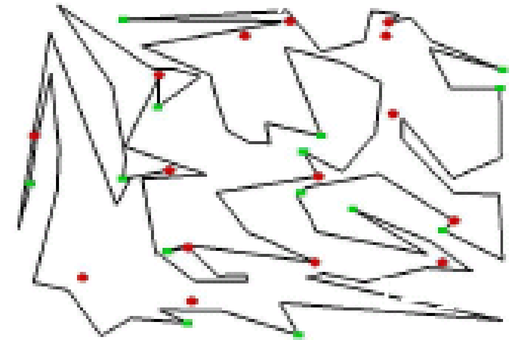
  [Hengeveld,Miltzow SoCG'21]: practical methods, vision stability $\delta$



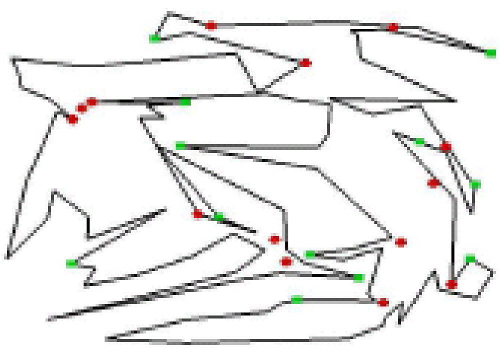
(g) 16 guards



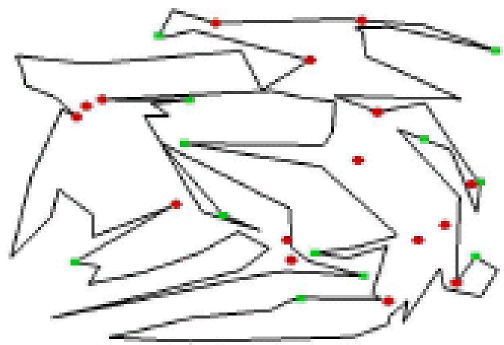
(h) 16 guards



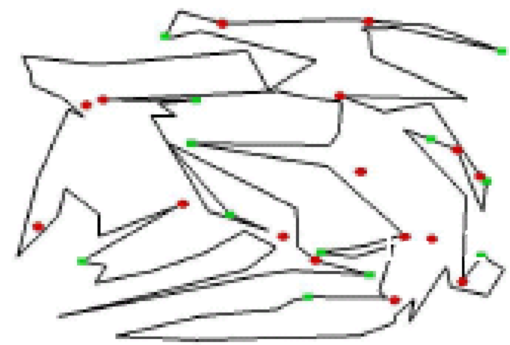
(i) 15 guards



(j) 14 guards

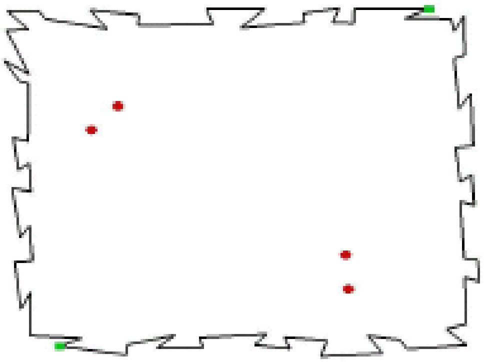


(k) 16 guards

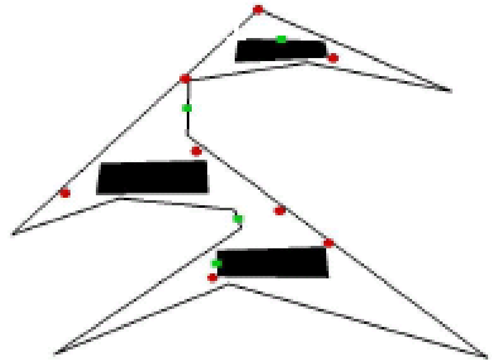


(l) 16 guards

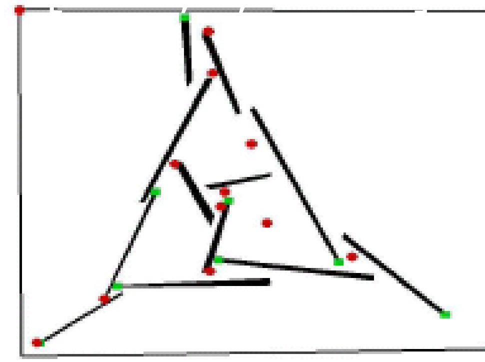
[AMP]



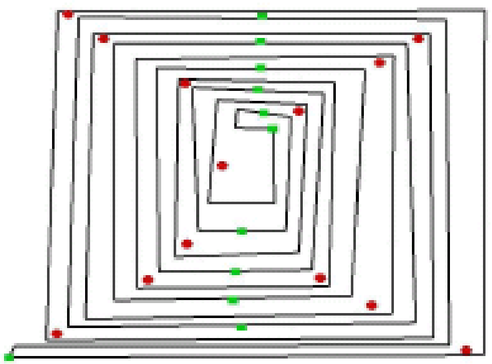
(a)



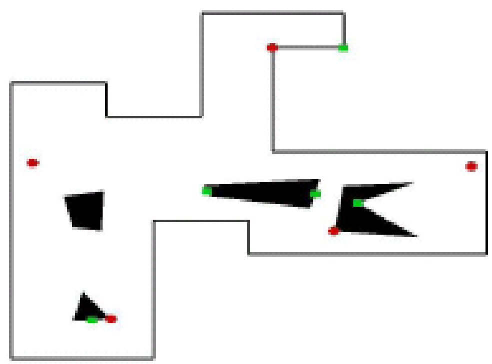
(b)



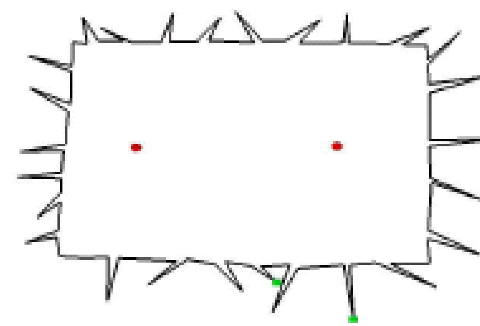
(c)



(d)

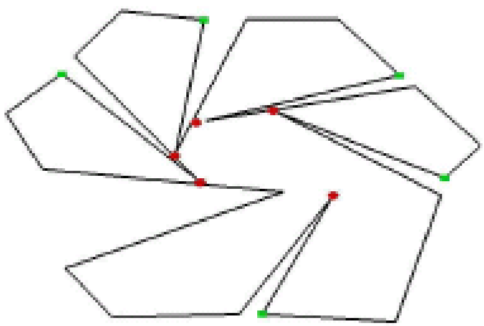


(e)

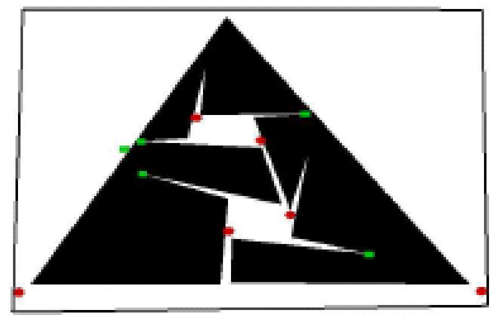


(f)

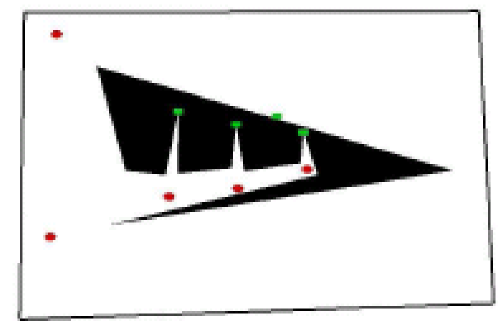
[AMP]



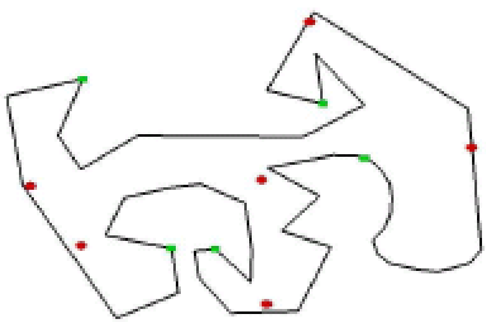
(g)



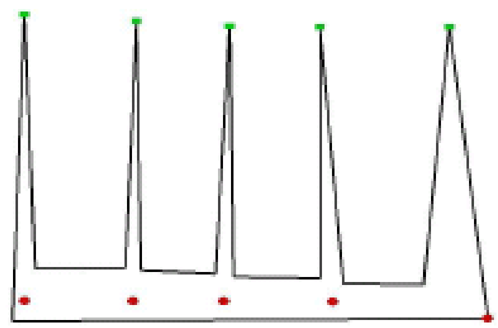
(h)



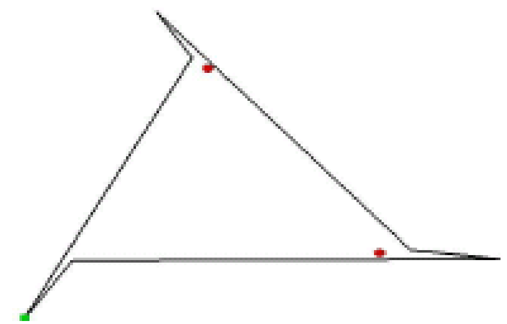
(i)



(j)



(k)



(l)

[AMP]

# Complexity of Computing Guards

- NP-hard, even in simple polygons, terrains
- APX-hard in simple polygons
- Need for irrational guards [Miltzow,Adamaszek,Abrahamsen,SoCG'17]
- $\exists\mathbb{R}$-complete (unlikely in NP)

[Miltzow,Adamaszek,Abrahamsen,JACM'21]

# Approximation Algorithms

Approximation algorithms for discrete candidate sets (vertex guards, grid-point guards, etc):

- $O(\log n)$-approx: set cover (greedy) [G87]
- $O(\log g^*)$-approx: reweighting ([Cl,BG]) [EH03,GL01]
- $O(\log\log g^*)$: [KK11]
- $O(1)$-approx in special cases:
  - 1.5D terrains (O(1), PTAS) [BKM05,K06,EKMMS08,GKKV14]
  - Monotone polygons [Ni05]
  - Triangle-free arrangements (3-approx) [JN14]
- PTAS:
  - Bounded depth, bounded vision disks [AKMY12]
  - Robust (one model) guarding [M]

Pseudo-poly $O(\log g^*)$-approx (poly in spread, n) [DKDS07]

Point guards (any, but integer coords,nondeg P): $O(\log g^*)$-approx

[Bonnet,Miltzow,SoCG'17] (correcting [DKDS07])

Exact poly-time solutions:

- Rectangle visibility in rectilinear polygons [WK06]
- *Partitioning* P into min # star-shaped pieces [Ke85] (diagonals)
- Min-length watchman tour (mobile guard) [CN86,...]

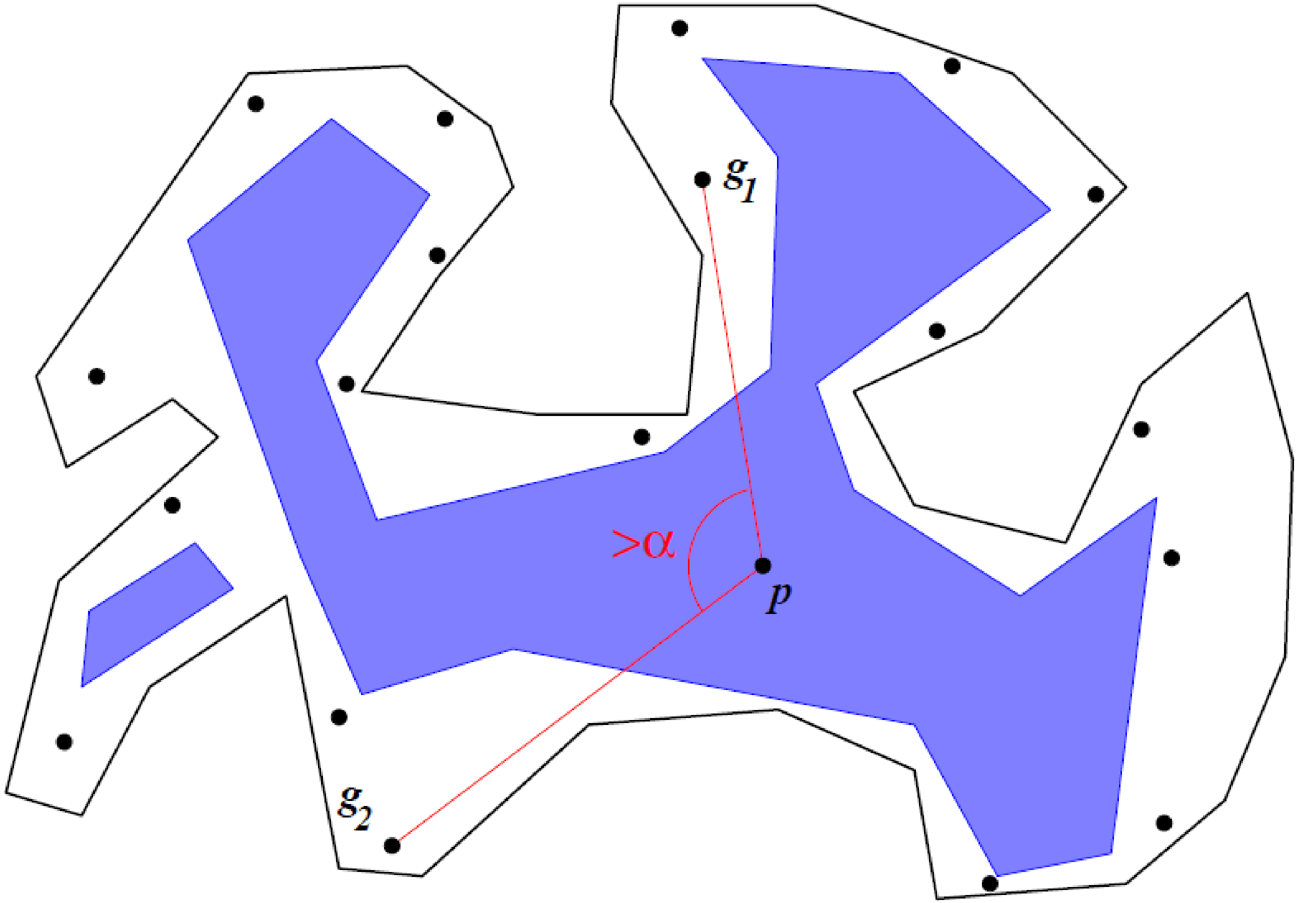
# Notions of Robust Guarding

- k-guarding
  - [Busto,Evans,Kirkpatrick'13] O(k loglog $g^*$)-approx in simple polygons (use ($\varepsilon$, k)-nets)
- Angle-constrained 2-guarding
- Triangle guards
- ($\varepsilon$,R)-guards
- Universal guards
- Polygons with vision stability $\delta$ [Hengeveld,Miltzow, SoCG'21]
- $\alpha$-robust guards [Das,Filtser,Katz,M, 2023]

# Angle-Constrained 2-Guarding

[Efrat, Har-Peled, M]

Goal: See all of a region $Q$ very "well"




$p \in Q$ is *2-guarded at angle* $\alpha$ by $G$:

# Main Idea

[Efrat, Har-Peled, M]

Follow a Clarkson/Brönnimann-Goodrich approach:

- Distribute weights on candidate guard locations (grid $\Gamma$, implicitly maintained)
- Each main iteration: Select a subset of candidates using the weight distribution

  (larger weight implies more likely to select)
- If we ever satisfy the covering criterion, DONE
- Else, pick a $q \in Q$ not yet "covered", increase weights of candidates that see $q$, REPEAT

# Angle-Constrained 2-Guard Cover

[Efrat, Har-Peled, M]

Two phases:

- Find $G_1$ – approx min 1-guard cover of $Q$ [EH]
- Find $G_2$ such that $G_1 \cup G_2$ 2-guards $Q$ at angle $\alpha/2$

**LEM:** Let $G^*$ be a set of $k^*$ sensors that 2-guard $Q$ at angle $\alpha$. Let $G_1$ 1-guard $Q$. Then, for any point $p \in Q$ there exist sensors $g_1 \in G_1$ and $g_2 \in G^*$ that 2-guard $p$ at angle $\alpha/2$.

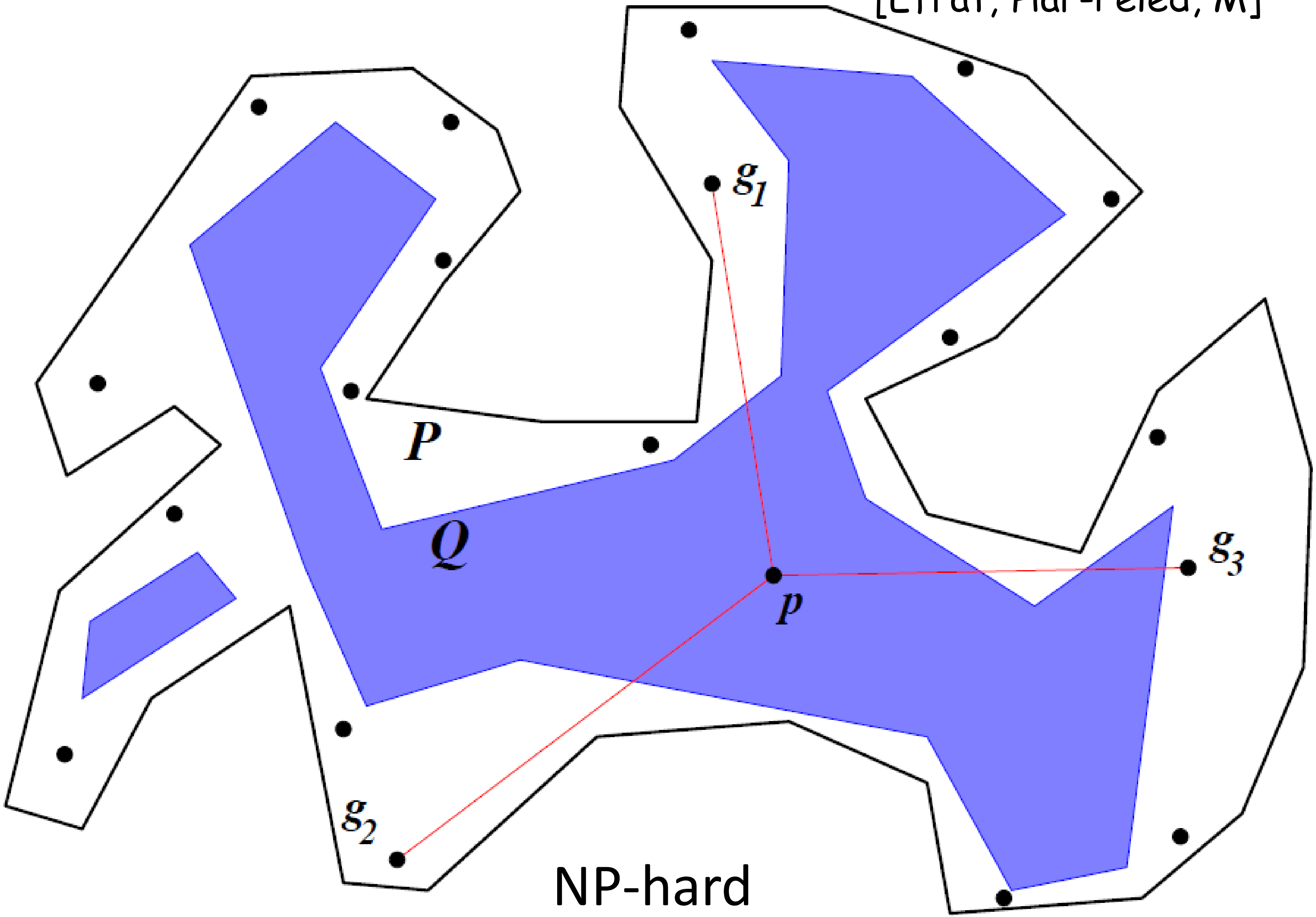
Apply Clarkson/Brönnimann-Goodrich approach:

**THM:** Given $P$, $Q$, grid $\Gamma$, we can find sensors $G \subset P$ that 2-guard $Q$ at angle $\alpha/2$, and $|G| = O(k^* \log k^*)$, where $k^*$ is the cardinality of smallest set of vertices of $\Gamma$ that 2-guard $Q$ at angle $\alpha$.
The running time is $O(nk^{*4} \log^2 n \log m)$, where $m = \#$ vertices of $\Gamma \cap P$.

(dual approx)

# Triangle Guarding

[Efrat, Har-Peled, M]




NP-hard

# Triangle Guarding

- Method: Find a min-link cycle surrounding Q, and place guards at these vertices

- Analysis: OPT can be converted to a set of 3|OPT| points outside of Q, within P, such that the $VG_{P-Q}$ of these points is connected, and Q lies within a face of the arrangement

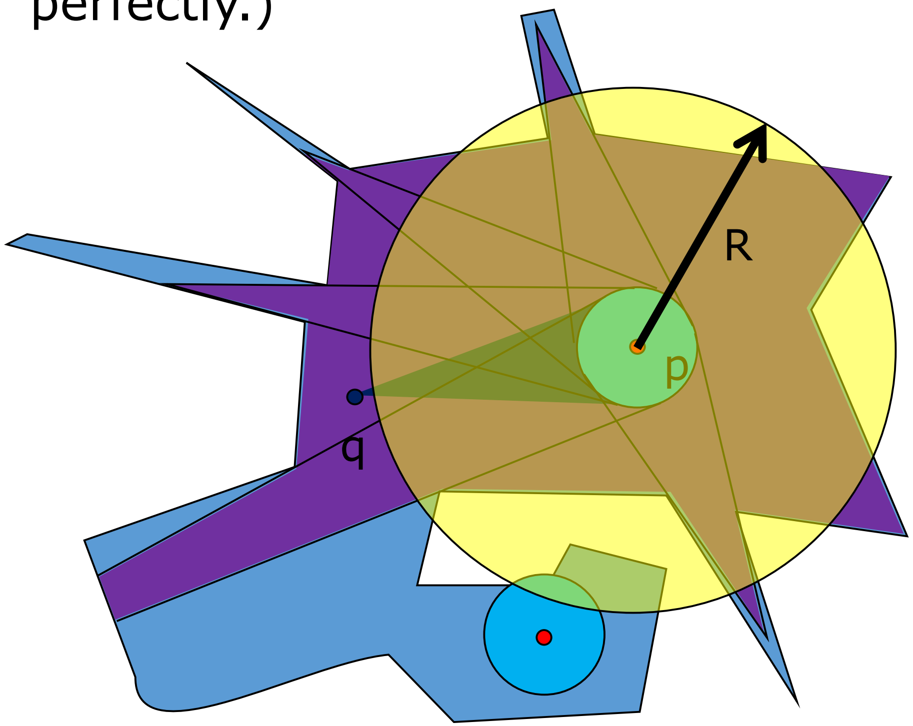
Thm: Complexity of face is O(OPT log OPT), since O(OPT) vertices in this arrangement. [AHKMN, *DCG*]

- Result: O(log OPT)-approx

# (ε,R)-Robust Guards

Issue: Even if we computed exactly a minimum cardinality set of guards, could we know with confidence the domain is really guarded?

Guards may not be placed exactly. (Human guards don't usually stand exactly still, and cameras/sensors cannot be placed perfectly.)




**Model**: When a guard is placed at p, it will actually reside at some point within a disk, $B_\varepsilon(p)$, of radius $\varepsilon$

In order for q to be "seen" by guard p, it must be able to see the guard no matter where it is within the disk $B_\varepsilon(p)$

Bounded radius, R, of vision

Useful model for guarding point-cloud models of domains

# Robust Guards: Approximation

**Theorem**: There is a PTAS for computing a min # of robust, radius-bounded guards in a polygonal domain (with holes), assuming $R/\varepsilon$ is bounded, and a poly-size set G of candidate guard locations is given.

One option for G: use a set L of $O(\lambda \log^2 \lambda)$ *landmarks*, as in [AEG08], and then guarantee at least $(1-\varepsilon_1)$-fraction of the area is seen.

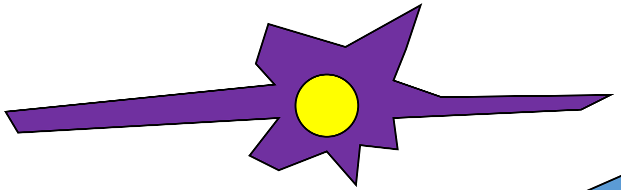
$\lambda = (g_{opt} / \varepsilon_1) \log h$ (h = # holes)

[AEG08] also give randomized greedy algorithm that, whp, computes $O(g_L \log \lambda)$ guards to cover L, where $g_L \leq g_{opt}$ is opt # of guards to cover L

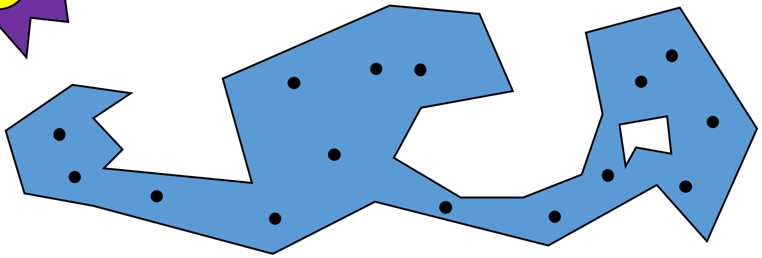
**Method**: m-guillotine optimization: Convert any OPT to an m-guillotine version; apply DP to optimize

# What is Needed for PTAS to Apply

**Suffices**: Visible regions, VP(g), from candidate guard locations g∈G have area(VP(g)) ≥ c diam²(VP(g)), for some c. (e.g., each VP(g) contains a disk of radius Ω(diam(VP(g)) )



Special Case: Bounded radius visibility in polyominoes



**Another Sufficient Model**:

Sample points S in P.
Guards placed at subset of S.
Guards must see all of S: Problem is **Dominating Set** in VG(S)

If samples S are δ-*well dispersed* (e.g., no disk of radius δ has more than O(1) samples of S), and guards have visibility radius R, with R/δ bounded, then PTAS also applies

**Minimum Dominating Set**:
- best approx in general is log-approx
- PTAS for planar graphs, UDG
- APX-complete for degree-B, B≥3

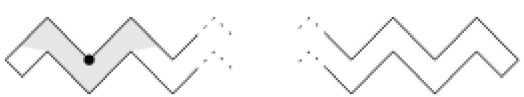
Here, the graph VG(S) is not planar, not UDG, but has bounded degree, depending on R/δ

# Guarding "Fat Vision" Polygons

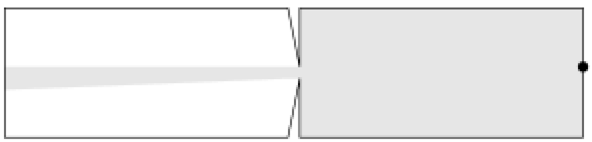
[Das,Filtser,Katz,M, 2023]

- If P has the property that for every point p in P the polygon VP(p) is $\alpha$-fat, we say P is "fat vision"
- Theorem: For fat vision P (even with holes), we can compute a set Q of $O(n^2)$ points such that Q contains a guard set of size O(OPT).

  Dependence on $\alpha$: $|Q|=O(\alpha^{-1} n^2)$, approx factor $O(\alpha^{-1})$
- Theorem: For fat vision P (even with holes), there is an $O(\alpha^{-3})$-approximation algorithm, poly(n).



P is fat vision
P is not fat

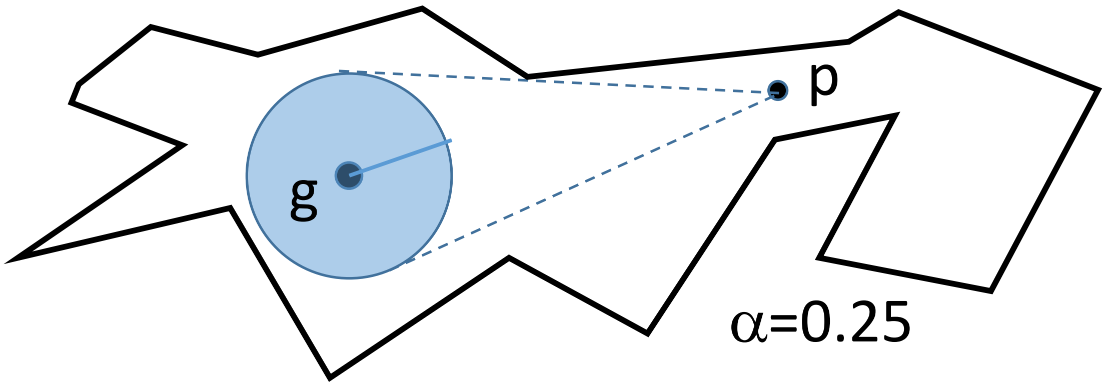


P is fat
P is not fat vision

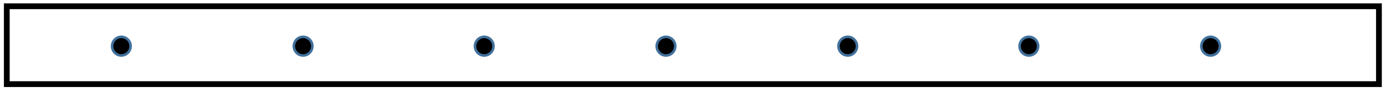
# Robust Vision Guarding

[Rathish Das, Omrit Filtser, Matya Katz, JM, 2023]

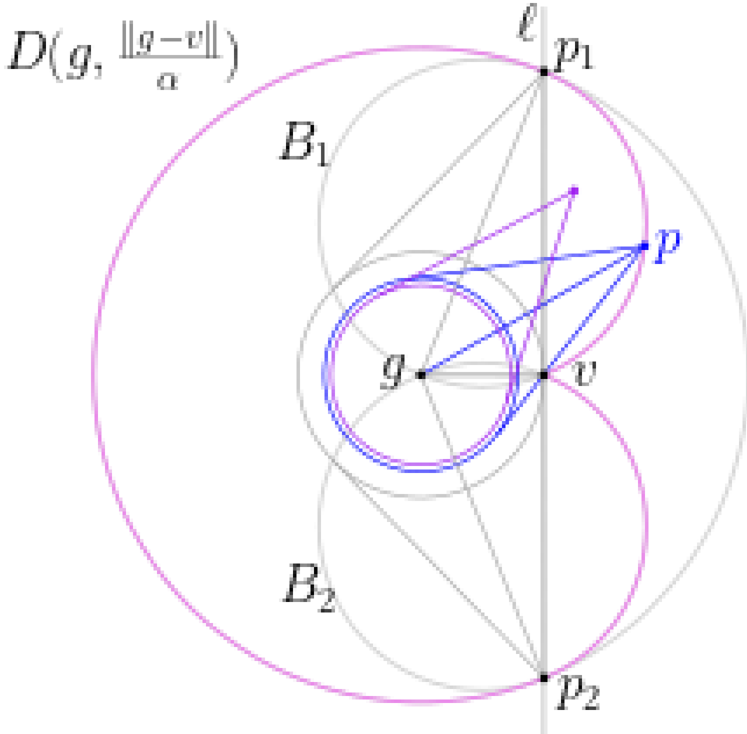
Point g is said to *$\alpha$-robustly see* point p iff p is seen by a guard that is anywhere inside the disk D(g, $\alpha$|gp|)

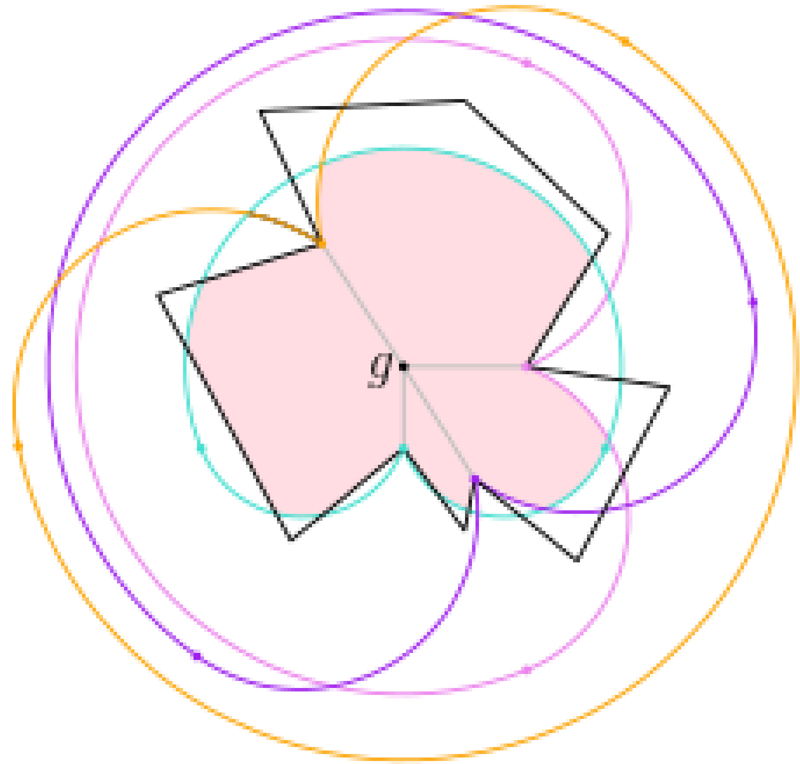



Note that many guards may be needed to $\alpha$-robustly guard a skinny polygon:



# What does g see $\alpha$-robustly?




$\text{Vis}_\alpha(g)$ is $O(\alpha)$-fat, and can be computed efficiently

# Method/Results

- Compute a carefully crafted discrete set Q of candidate guards

**Theorem 3.** *The set* $Q = M \cup \bigcup_{v \in M} Q_v$ *contains a set of* $O(\alpha^{-4})|OPT_\alpha|$ *points that* $\alpha/4$*-robustly guard* $P$*.*

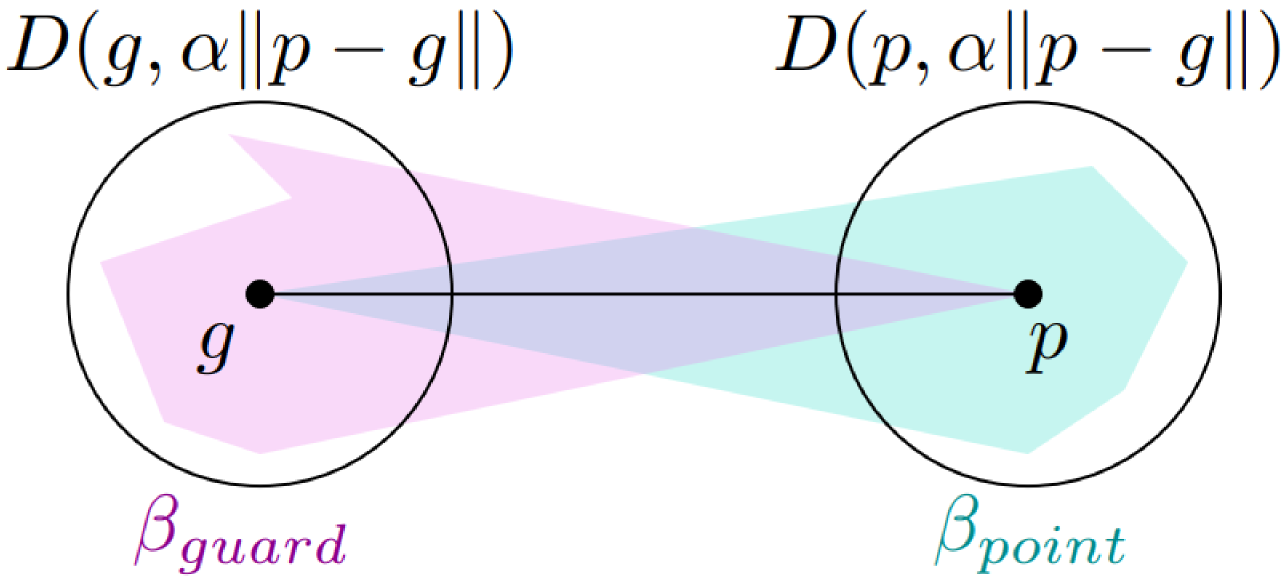
In addition, we claim that the size of $Q$ is linear in $n = |P|$ and $|OPT_\alpha|$.

- Apply a greedy algorithm and prove:

**Theorem 4.** *Given a polygon* $P$ *with* $n$ *vertices, one can compute in* $\text{poly}(n, |OPT_\alpha|)$ *time a set of* $O(\alpha^{-6})|OPT_\alpha|$ *points that* $\alpha/8$*-robustly guard* $P$*, where* $OPT_\alpha$ *is a minimum-cardinality set of guards that* $\alpha$*-robustly guard* $P$*.*

Time is polynomial in (input,output)

# More General Definition

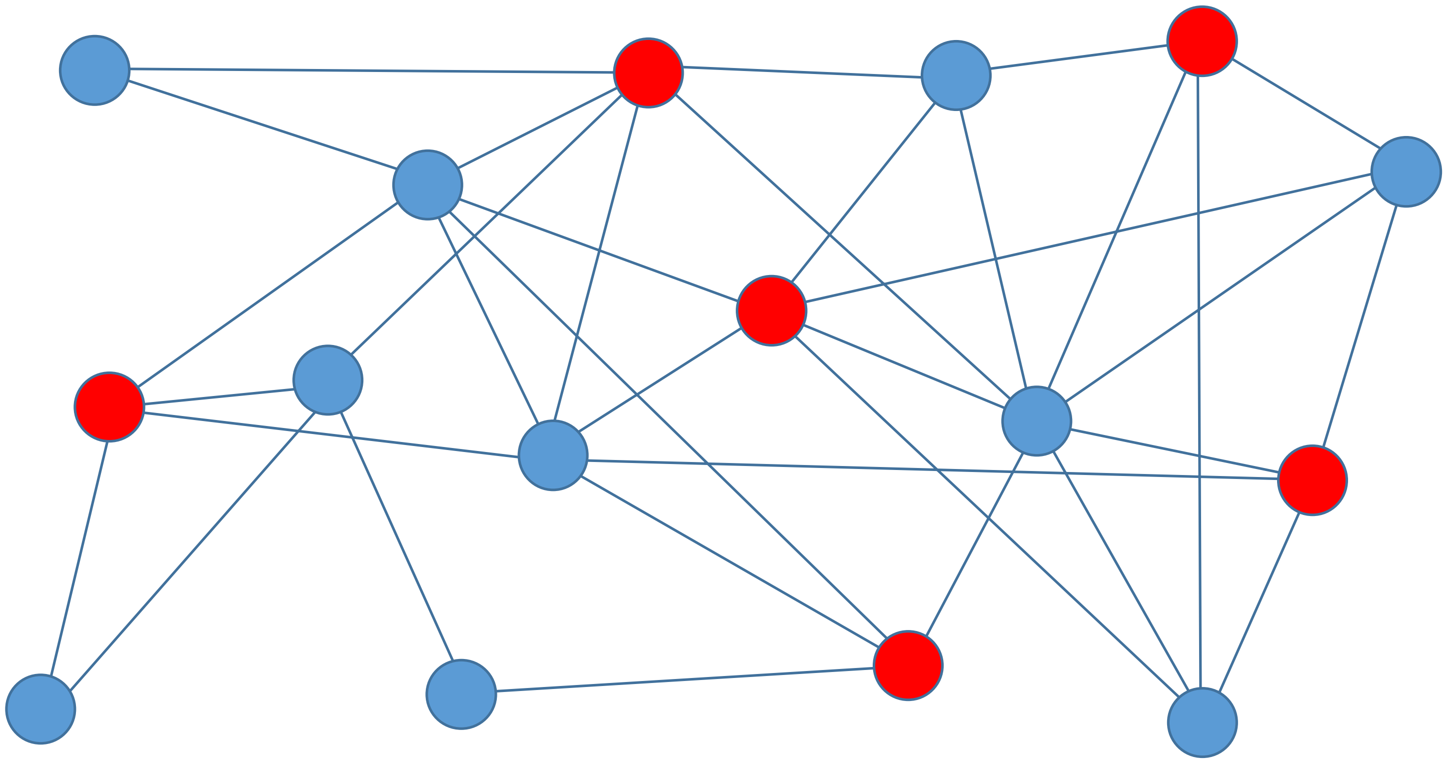



**Definition 6.1.** *Given a polygon $P$ and parameters $0 < \beta_{guard}, \beta_{point}, \alpha \leq 1$, we say that a point $g \in P$* ($\beta_{guard}, \beta_{point}, \alpha$)**-robustly guard** *another point $p \in P$ if $\overline{gp} \in P$, and*

1. *the area of $Vis(p) \cap D(g, \alpha \cdot \|p - g\|)$ is at least $\beta_{guard} \cdot \pi(\alpha \cdot \|p - q\|)^2$, and*
2. *the area of $Vis(g) \cap D(p, \alpha \cdot \|p - g\|)$ is at least $\beta_{point} \cdot \pi(\alpha \cdot \|p - g\|)^2$*

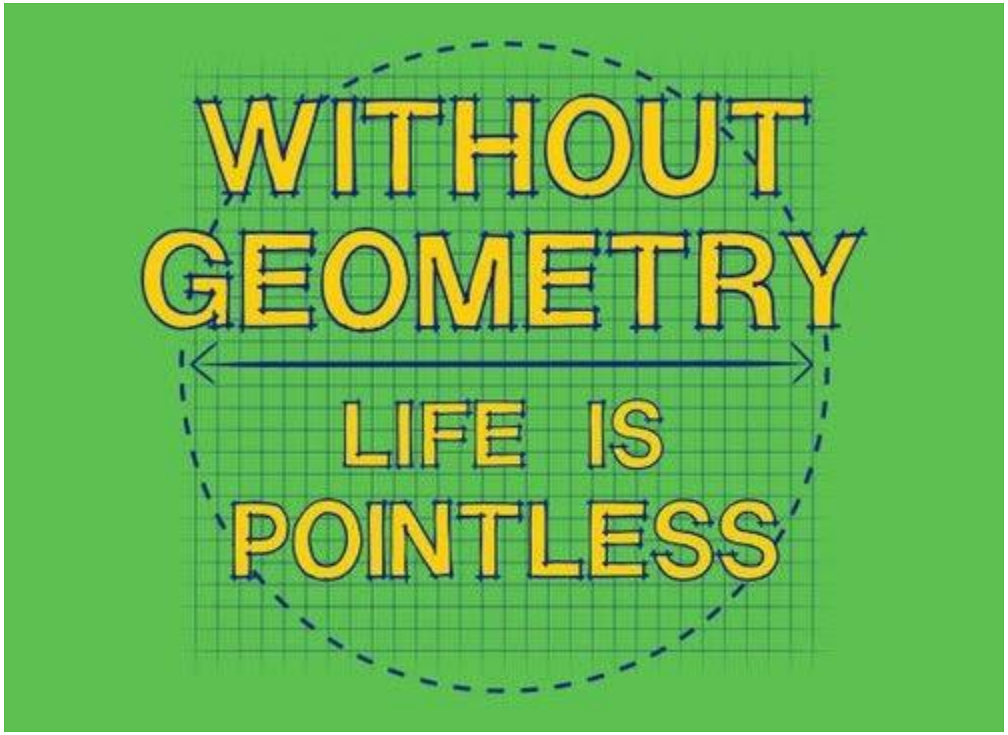
**Theorem 7.** *If $Vis^{\alpha}_{(1/6,0)}(g)$ can be computed in polynomial time, then a set of $O(\alpha^{-3})|OPT_{\alpha}|$ points that $(1/6, 0, \alpha/2)$-robustly guard $P$ can be computed in polynomial time.*

# Maximum Independent Set (MIS)



Best known polytime approx factor: $O(n/\log^2 n)$ [Boppana-Halldórsson]
No polytime algorithm with approx $n^{1-\delta}$ for $\delta>0$, unless P=NP [Zuckerman]
PTAS in planar graphs

# Can Geometry Help?

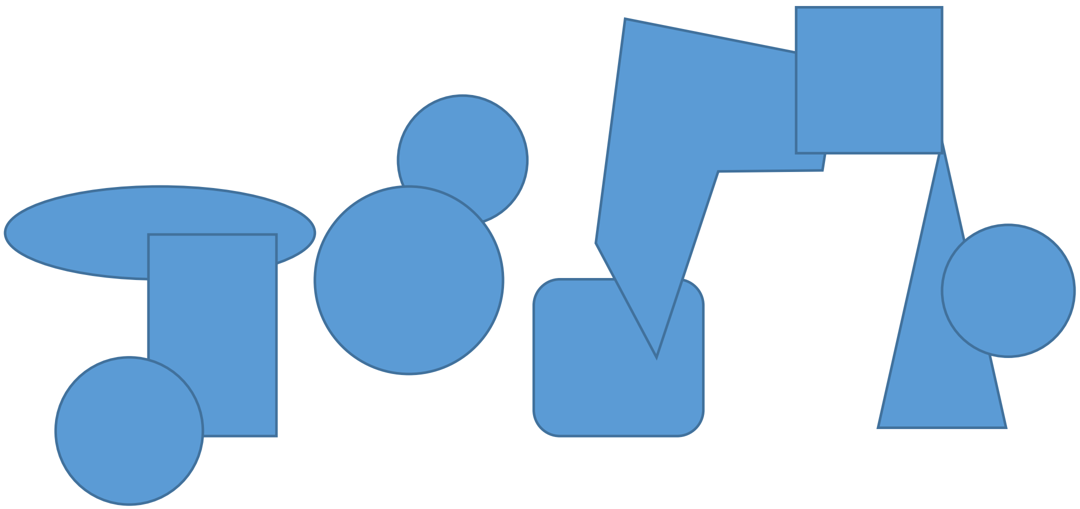


# A Basic Geometry Problem

Maximum Independent Set (MIS):

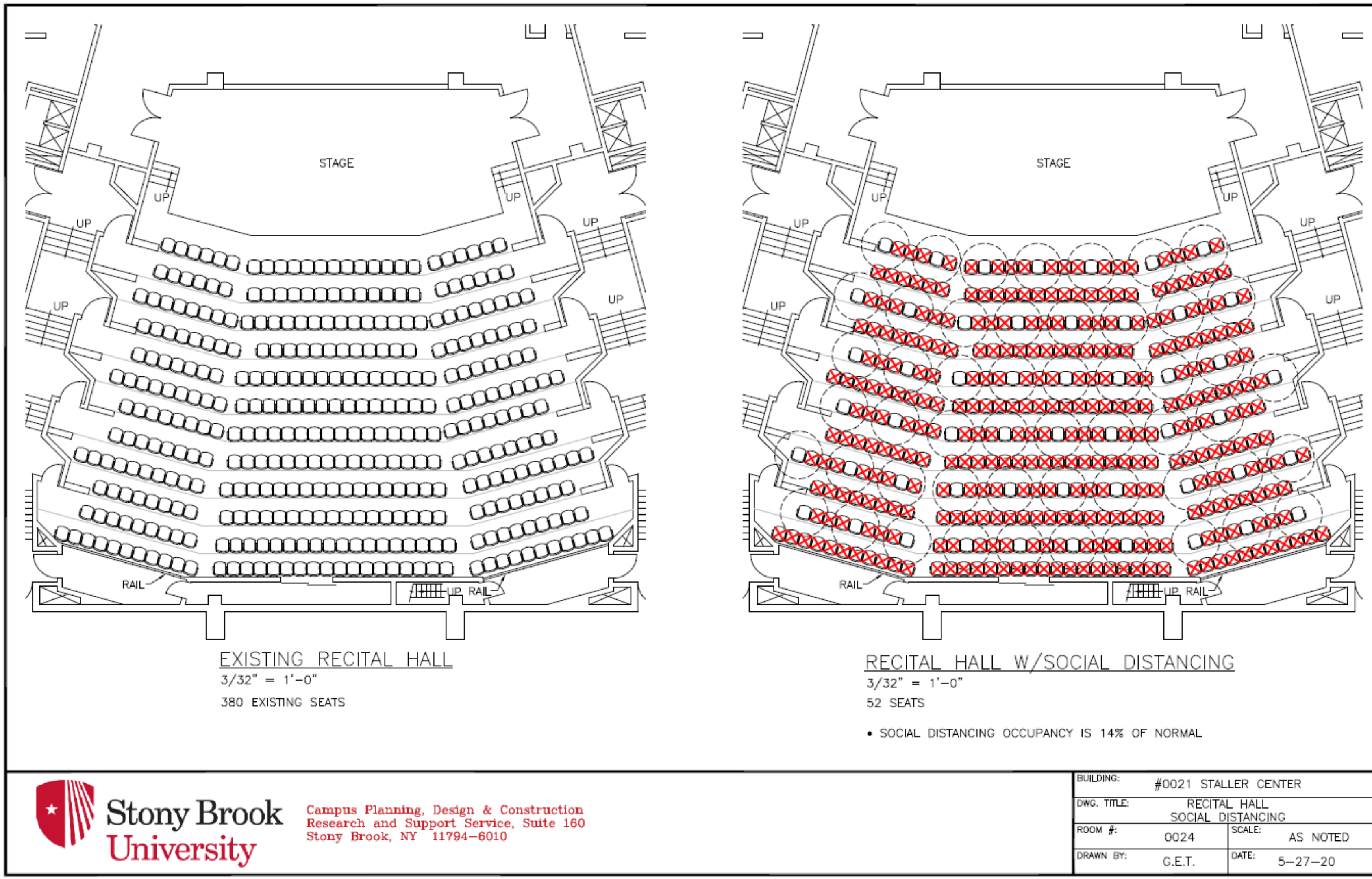
Given a set S of bodies in the plane.

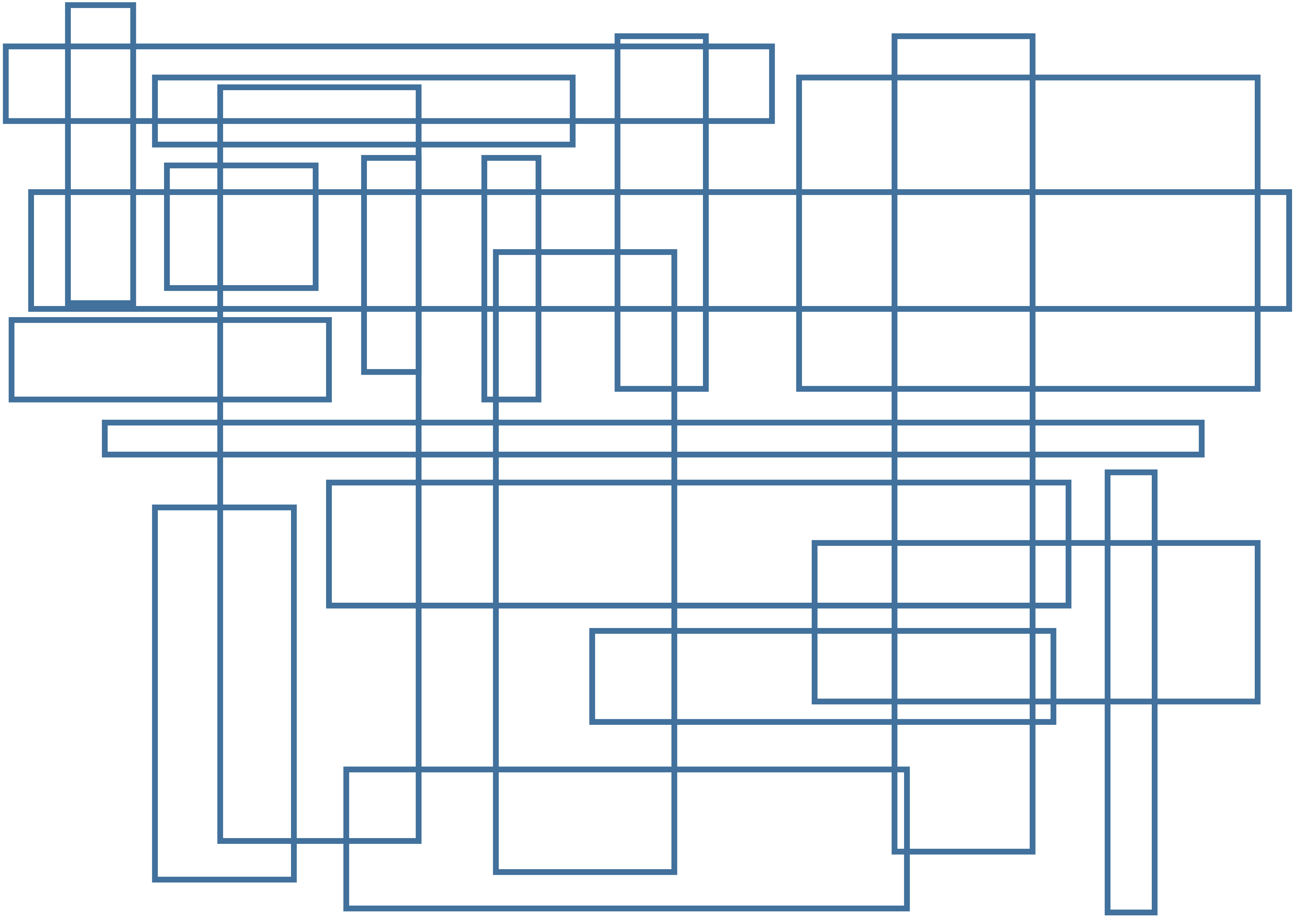
Find a max-cardinality subset, $S^*$, that is pairwise-disjoint.

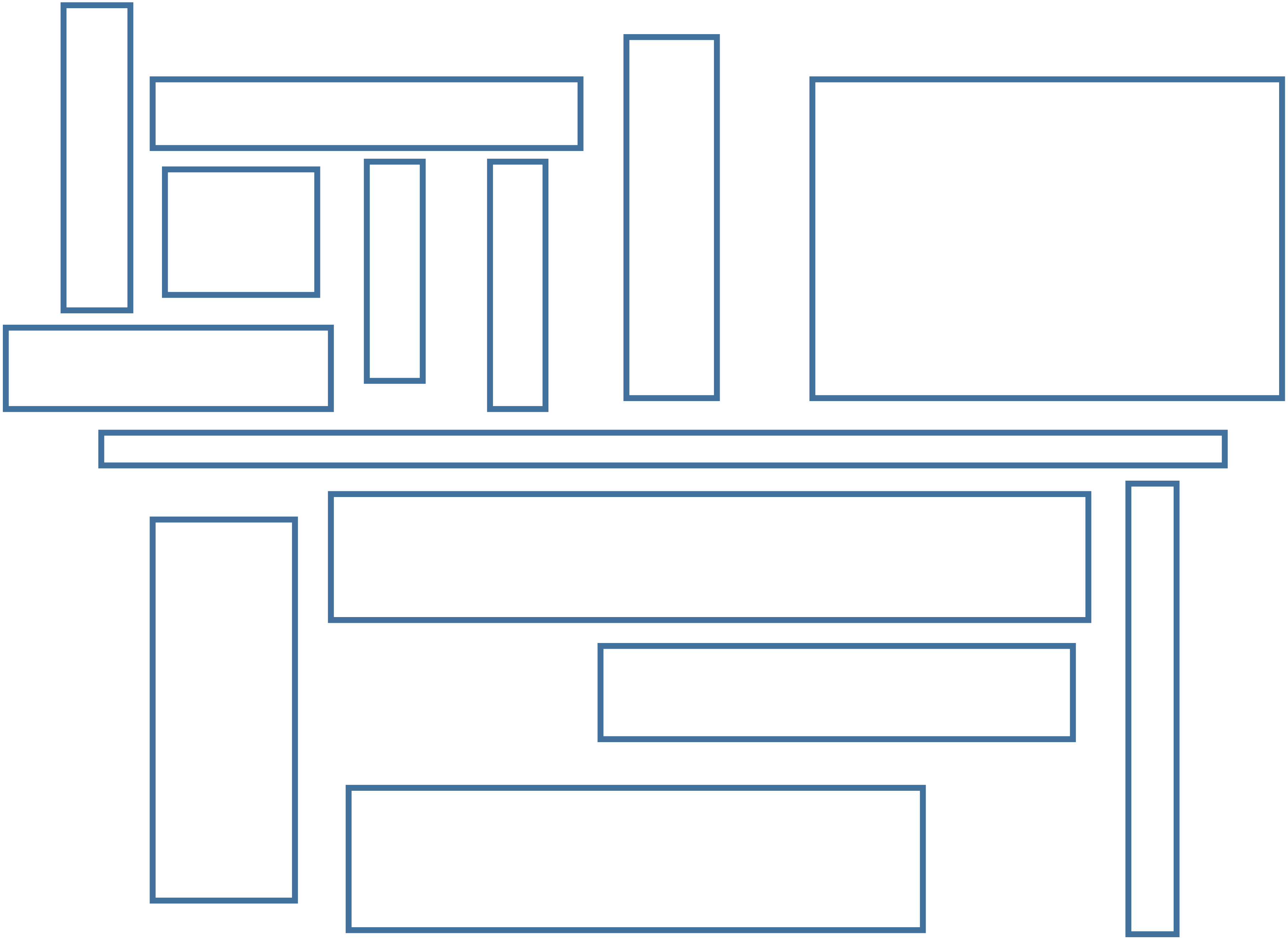


# MIS=Most Efficient Social Distancing

**Figure 5 – Lecture Hall Social Distancing Mock-Up**







# Approximations

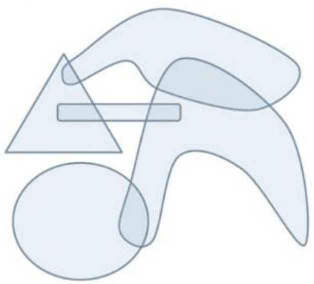
- Disks, fat regions: PTAS (1-ε)-approx, for any ε>0, in polytime

  (1- ε)-approx in $n^{O(1/\varepsilon^{d-1})}$ [Chan]

  Also: PTAS for pseudodisks [Chan, Har-Peled]

- Rectangles: MISR

  Rectangles are neither fat nor pseudodisks!

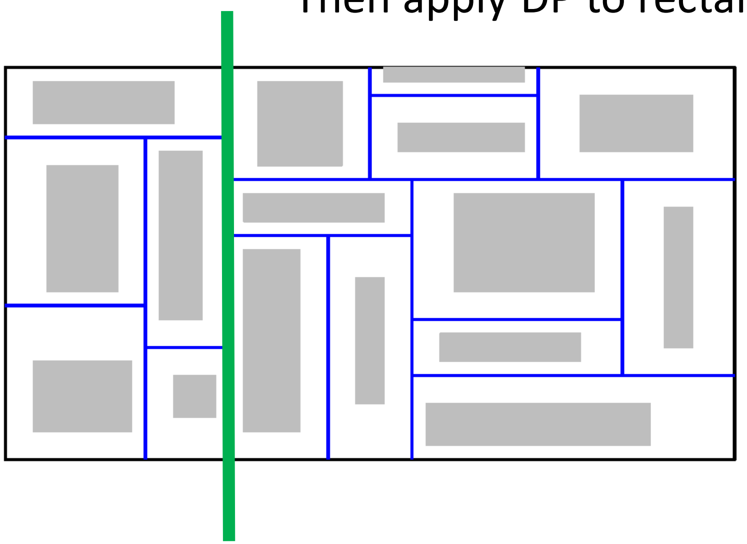
  – QPTAS
    - $n^{poly((\log n)/\varepsilon)}$ [Adamaszek, Har-Peled, and Wiese]
    - $n^{O(((\log\log n)/\varepsilon)^4)}$ [Chuzhoy and Ene]
  – PTAS for "long" rectangles [Adamaszek, Har-Peled, and Wiese]
  – Polytime: O(loglog n)-approx [Chalermsook, Chuzhoy]
  – Parameterized Approximation Scheme: [Grandoni,Kratsch,Wiese,2019]

    For any k, ε, in time $f(k, \varepsilon)n^{g(\varepsilon)}$ either gives indep subset of ≥k/(1+ ε), or declares OPT<k
  – **Here: O(1)-Approx in polytime**

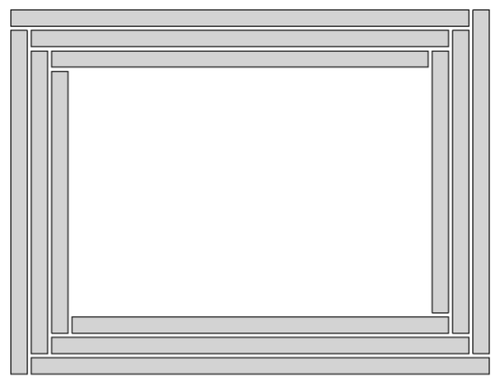
# MISR: One Approach

- Show that any set of disjoint rectangles (e.g., the rectangles of OPT) has a constant fraction subset that has a perfect BSP (or "guillotine separable")

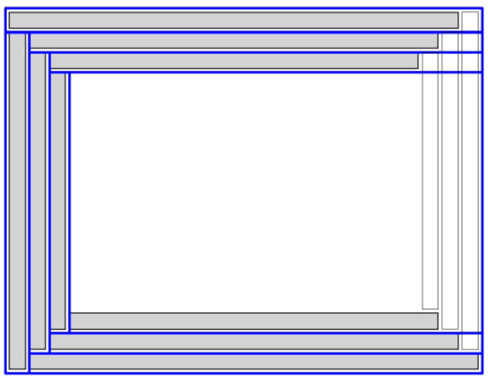
Then apply DP to rectangular "subproblems"



cut



No "free" guillotine cut



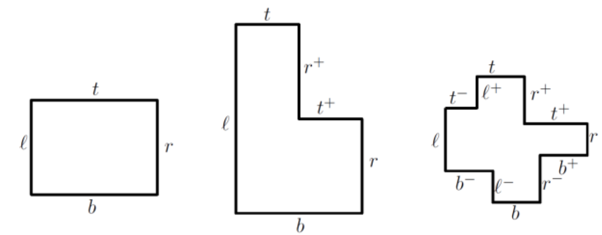
Subset (3/4) has perfect BSP

**Conjecture 1.** *For any set of $n$ interior-disjoint axis-aligned rectangles in the plane, there exists a subset of size $\Omega(n)$ that has a perfect orthogonal BSP.*

Pach-Tardos Conjecture

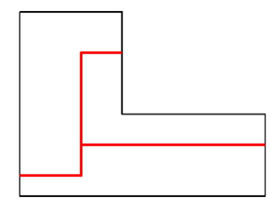
# Main Ideas

- Use more general cuts to get O(1) complexity pieces – one class “CCRs”




- Use K-ary cutting instead of just binary

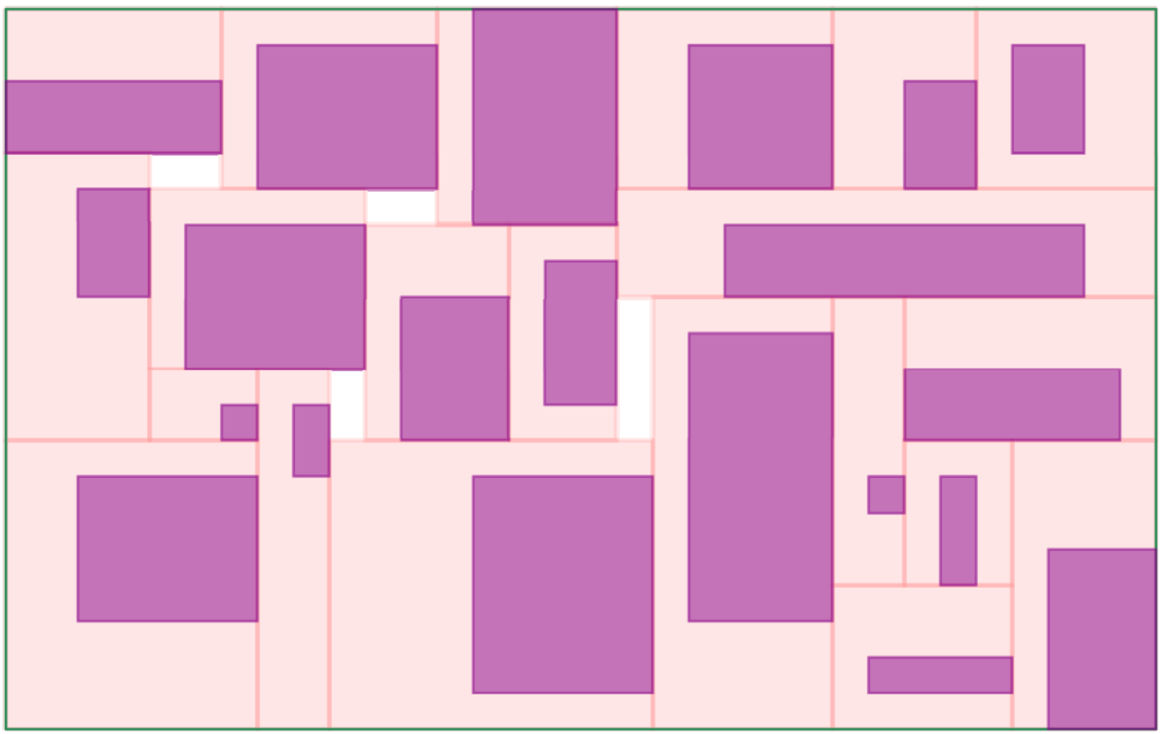
  K≤3



- Charging scheme to prove a structural theorem: Can afford to discard a constant fraction of input rectangles, to enable a “nearly perfect CCR-partition”
- DP to optimize

# Maximal Rectangles

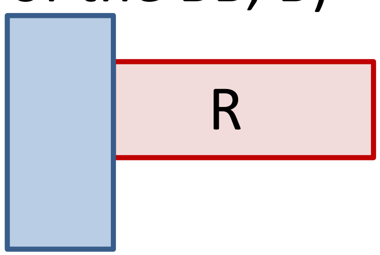
- Transform any set I of k disjoint rectangles into a set I’ of *maximal* disjoint rectangles



Will show that I’ has a constant-fraction subset for which there is a “nearly perfect CCR-partition” wrt the subset

# Nesting Among Maximal Rectangles

**Def**: A rectangle R is *nesting* to its left/right/top/bottom if its corresponding side is contained in the interior of an abutting rectangle's side (or the side of the BB, B)

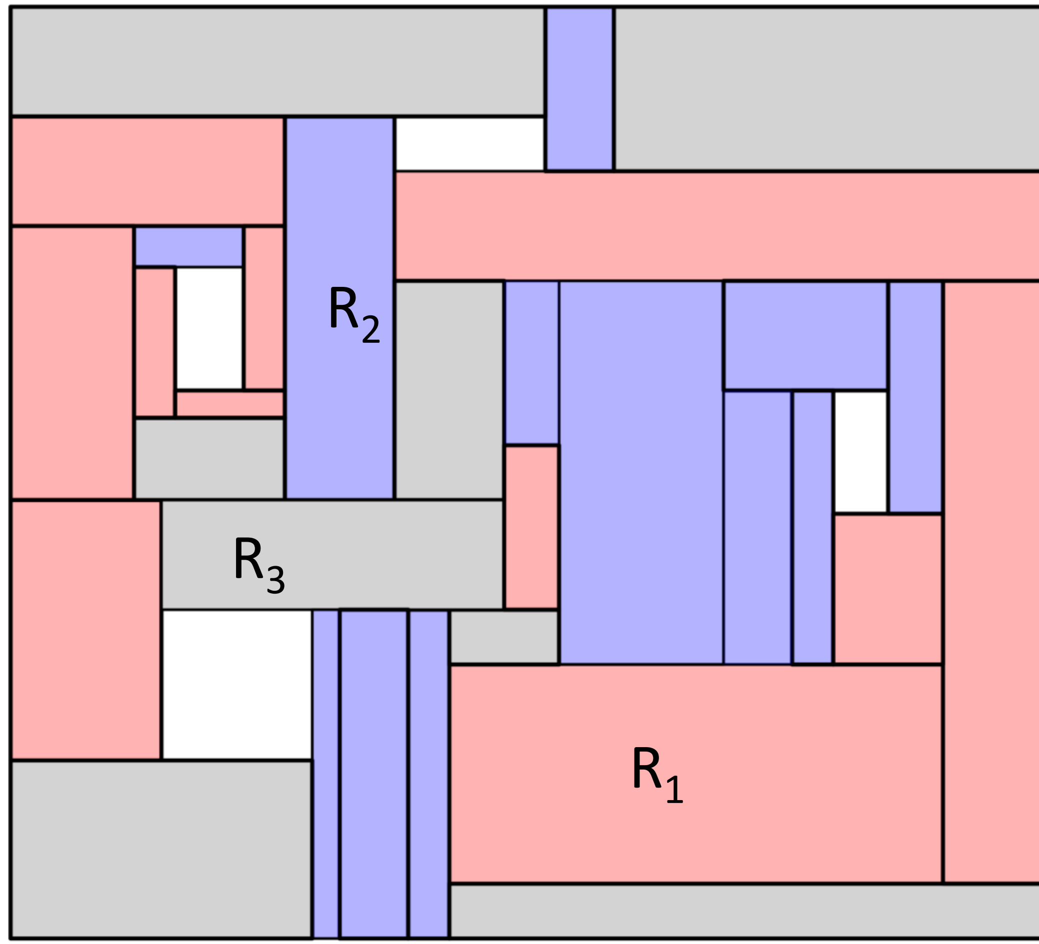



**Example**:

$R_1$ is horiz nested (red)

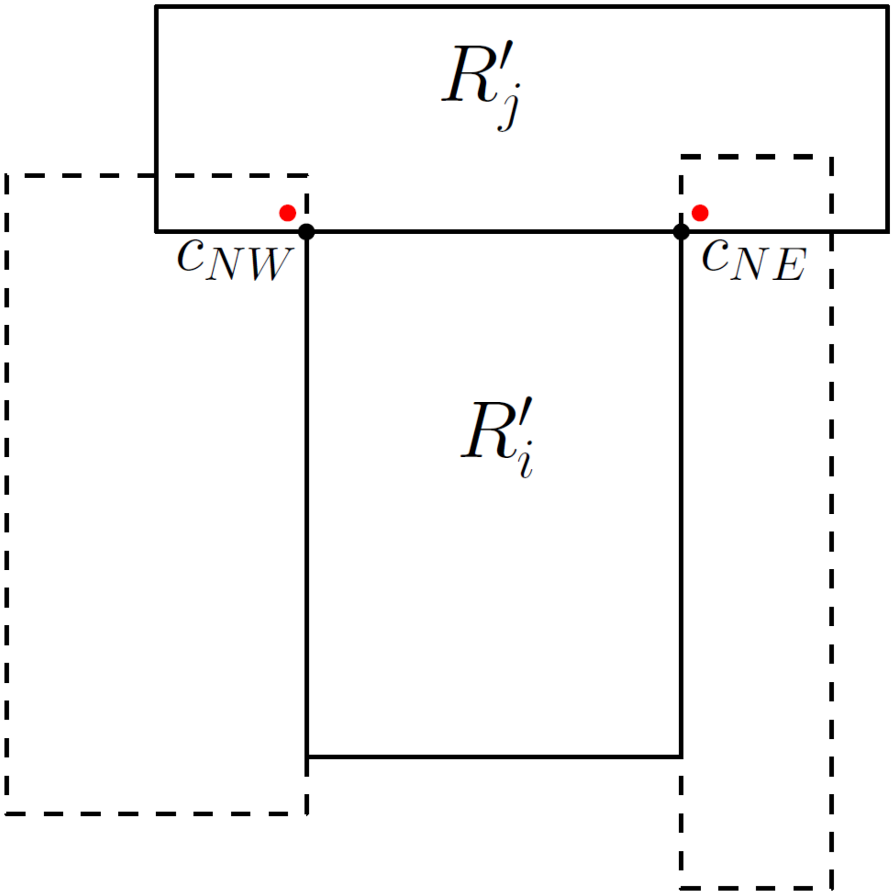
$R_2$ is vert nested (blue)

$R_3$ is not nested in any direction

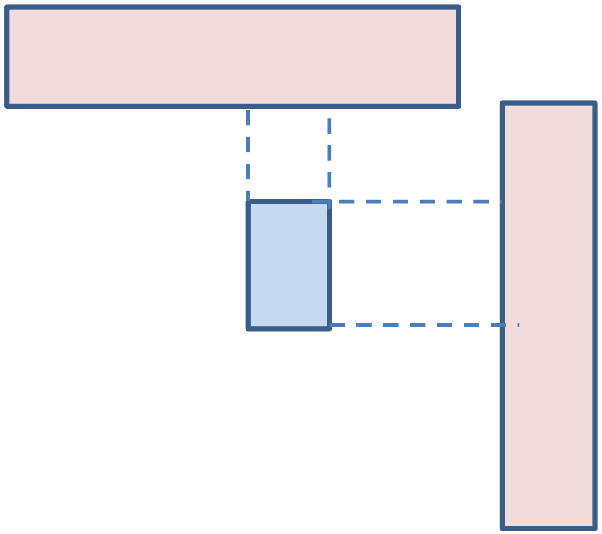


# Why Maximality Is Useful

**Observation 1.** *For a set $I'$ of independent rectangles that are maximal within $BB(\mathcal{R})$, a rectangle $R'_i \in I'$ cannot be nested both vertically and horizontally.*

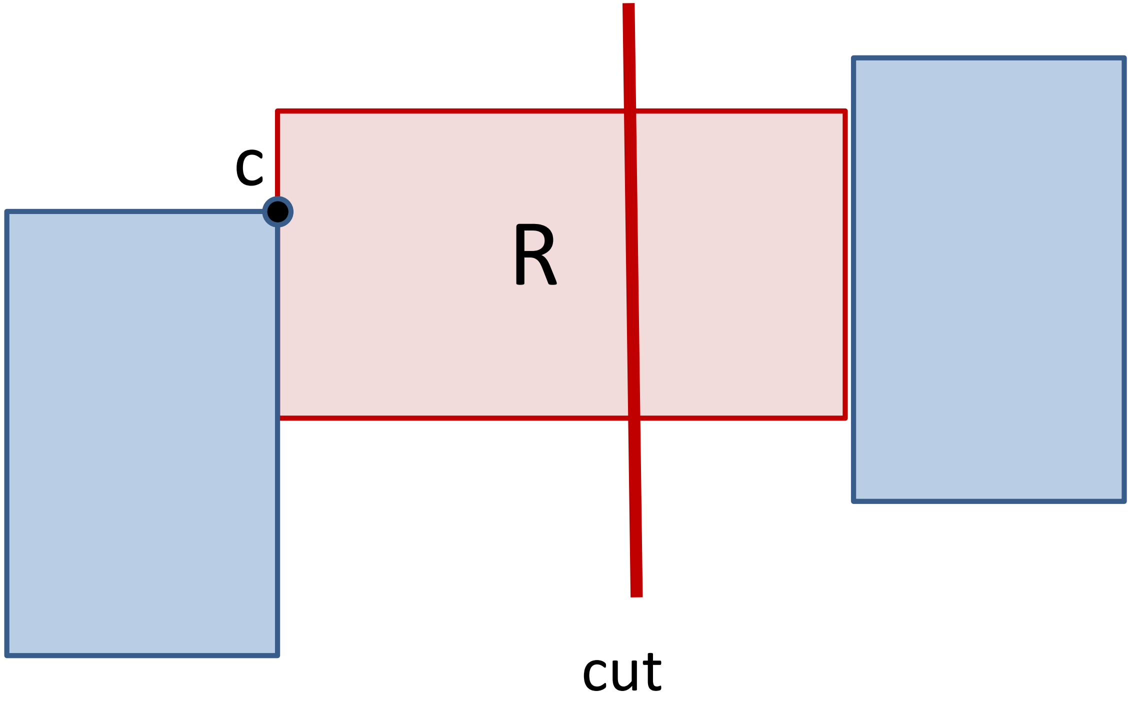



Note that the claim is not true without maximality:



# Why Nesting Concept Is Useful

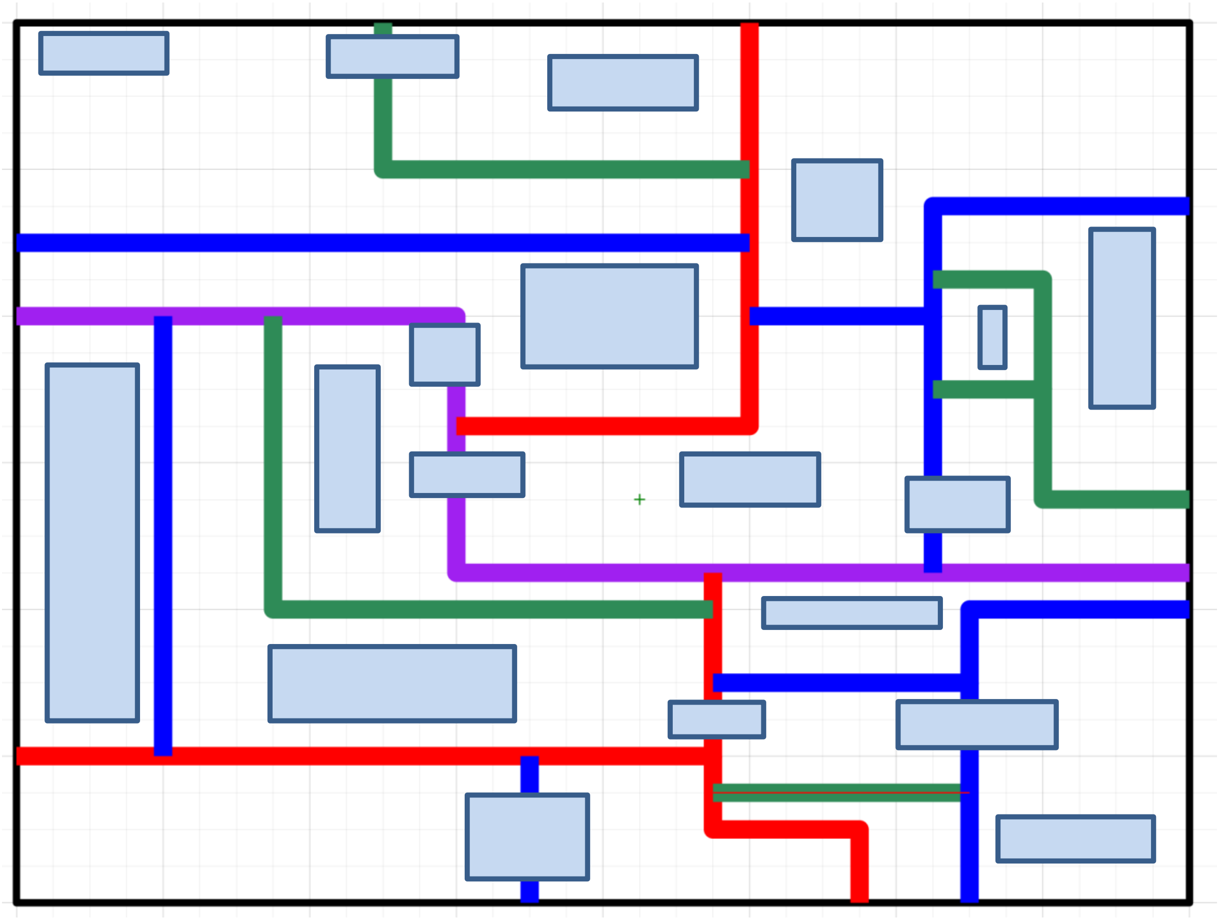
If R is *not* nested on at least one side, there is hope to be able to “charge” R to a corner, c, when a cut segment crosses R



# CCR-Partitions

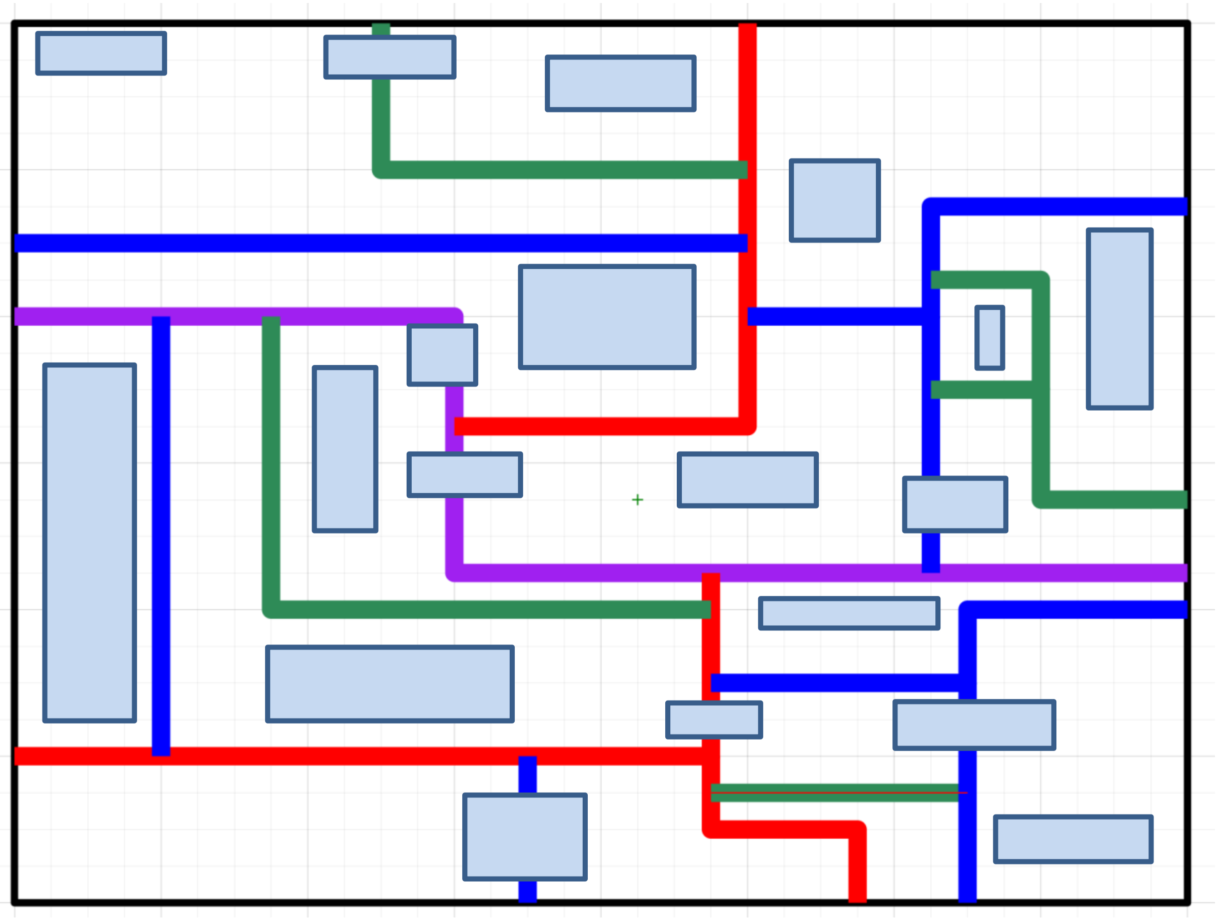
- Recursive partitioning of the BB, B, of input
- Each face Q is a CCR
- A *cut*, consisting of O(1) hor/vert segments partitions Q into at most 3 subfaces (CCRs)
- A CCR-partition is *perfect* wrt input rectangles if no rectangle is penetrated by a cut segment, each leaf face has exactly 1 input rectangle
- *Nearly perfect* CCR-partition: each cut segment penetrates at most 2 input rectangles, each leaf face has ≤1 input rectangle

# Nearly Perfect CCR Partition



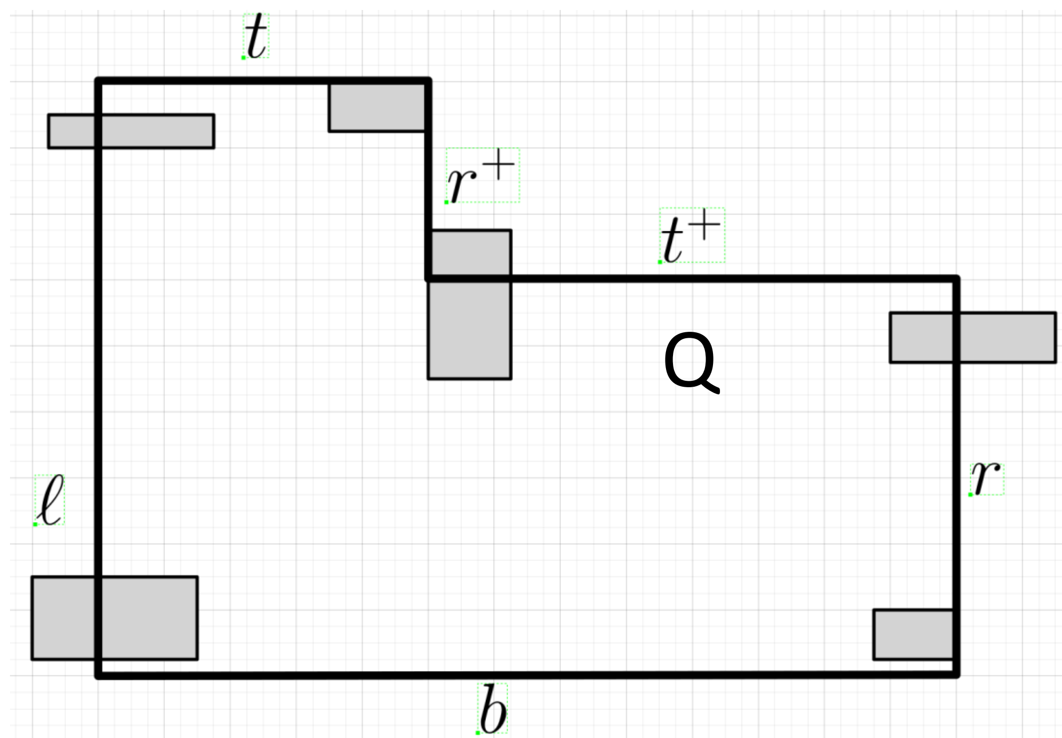
# The Structure Theorem

**Theorem 3.1.** *For any set $I = \{R_1, \ldots, R_k\}$ of $k$ interior disjoint (axis-aligned) rectangles in the plane within a bounding box $B$, there exists a $K$-ary CCR-partition of the bounding box $B$, with $K \leq 3$, recursively cutting $B$ into rectangles and (L-shaped) corner-clipped rectangles (CCRs), such that the CCR-partition is nearly perfect with respect to a subset of $I$ of size $\Omega(k)$.*



# The Algorithm: DP Subproblem

Subproblem $S=(Q,I_S)$, where $I_S$ is a set of "special" (specified) rectangles, at most 2 per vertical side of the CCR face Q.



# Dynamic Program

- Optimize over K-ary cuts (K≤3) for a CCR subproblem, S, to compute f(S), the max cardinality of an indep subset of input rectangles for which there is a nearly perfect CCR-partition

$$f(\mathcal{S}) = \begin{cases} 0 & \text{if } \mathcal{R}(\mathcal{S}) = \emptyset, \\ \max_{\chi \in \gamma(\mathcal{S}), I_\chi} (f(\mathcal{S}_1) + \cdots + f(\mathcal{S}_K) + |I_\chi|) & \text{otherwise,} \end{cases}$$

Here, $I_\chi$ is the set of rectangles (at most 2 per vertical segment of χ) that are penetrated by vertical cut segments and become special rectangles specified for the new subproblems, and $\gamma(\mathcal{S})$ is the set of all eligible K-ary CCR-cuts

**Theorem 4.1.** *There is a polynomial-time $O(1)$-approximation algorithm for maximum independent set for a set of axis-aligned rectangles in the plane.*

Crudely counted: time is $O(n^{21})$

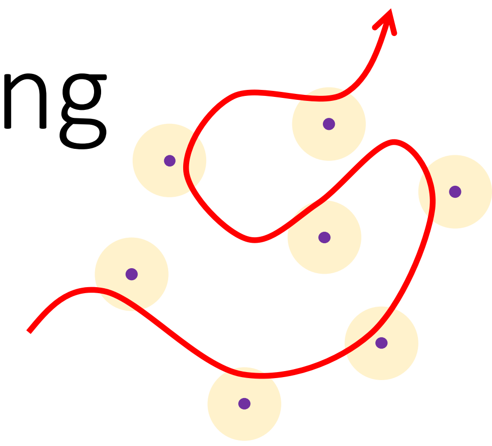
# Better Factors

- Original factor (Jan, 2021): 10
- Here: 4 [FOCS'21]
- Small variant: Offload charge on $R_r$ if both left corners charged (cases (5),(6)), by examining its top-left neighbor: Get factor 10/3

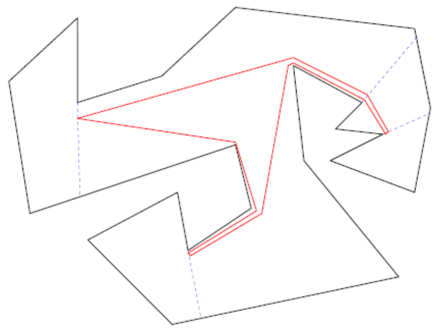
  Now: fence may penetrate 2 rectangles instead of 1
  Still get O(1) complexity subproblems

- Continuing: 22/7,….., (3+ ε)
- Further improvements:
  – Factor 3 [SODA'22], (2+ ε) [Galvez,Khan,Mari,Momke,Reddy,Wiese]

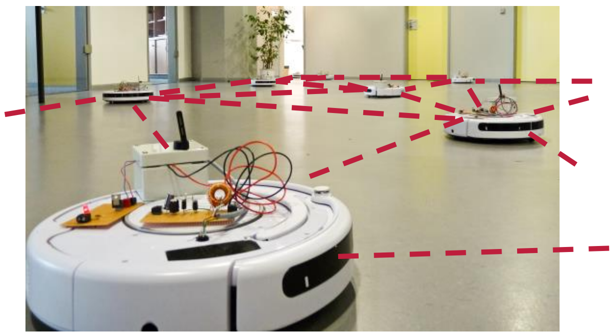
# Combining Coverage, Routing



- Optimal routing problems:
  - Optimal routes/networks to visit regions
  - Optimization of routes for vision/coverage
- Aspects of particular interest:
  - Uncertainty, robustness of solutions
  - Handling time constraints



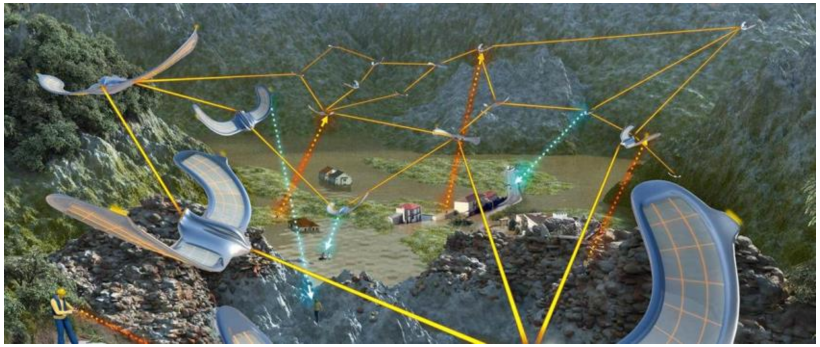
- Motivating applications:
  - Robotics
  - Sensor networks
  - Vehicle routing, logistics



# Cooperative Heterogeneous Vehicle Mission Planning

Motivating applications: search and rescue; casualty/disaster response; surveillance; mosaic battlefield

- Vehicles:  various classes (ground, air, sea), speeds, capacities, capabilities
- Targets: points, regions; mission task times; precedence constraints
- Constraints: domains of operation; tethers (distance); rendezvous requirements, formations
- Tactical vs strategic; online vs offline

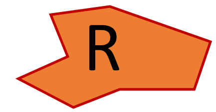
# Missions for Agents, UAVs

Types of mission tasks:

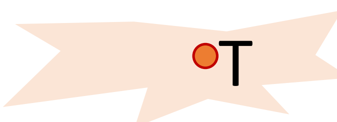
- Visit target site (point) p


- Visit (any point) of target region R
  - Possible constraint: Mission time (minimum) within R

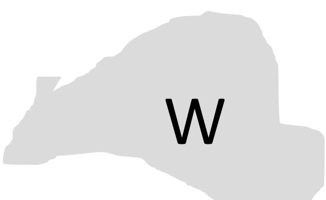
[Jia, Mitchell, 2019: TSPN with time lower bounds. PTAS, dual approximation algorithms]

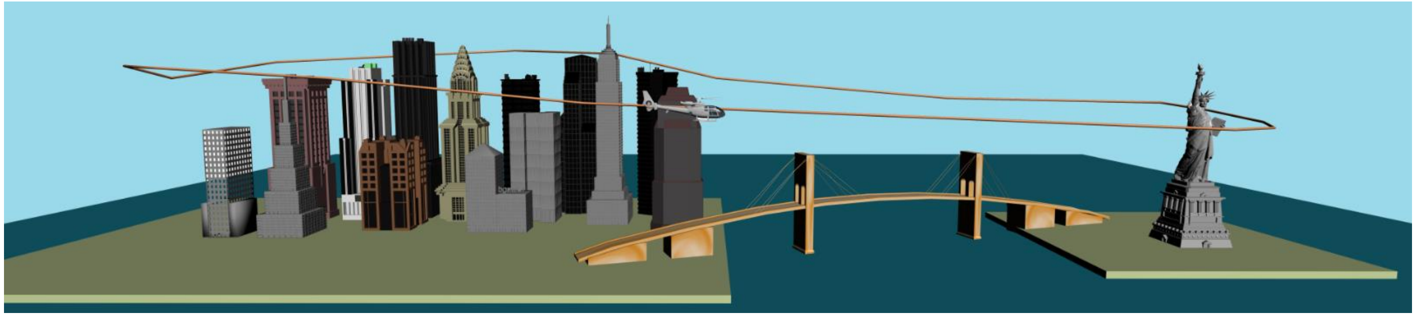



- View a target (point/region) T: visit any point that is visible to T “watchman route problem”



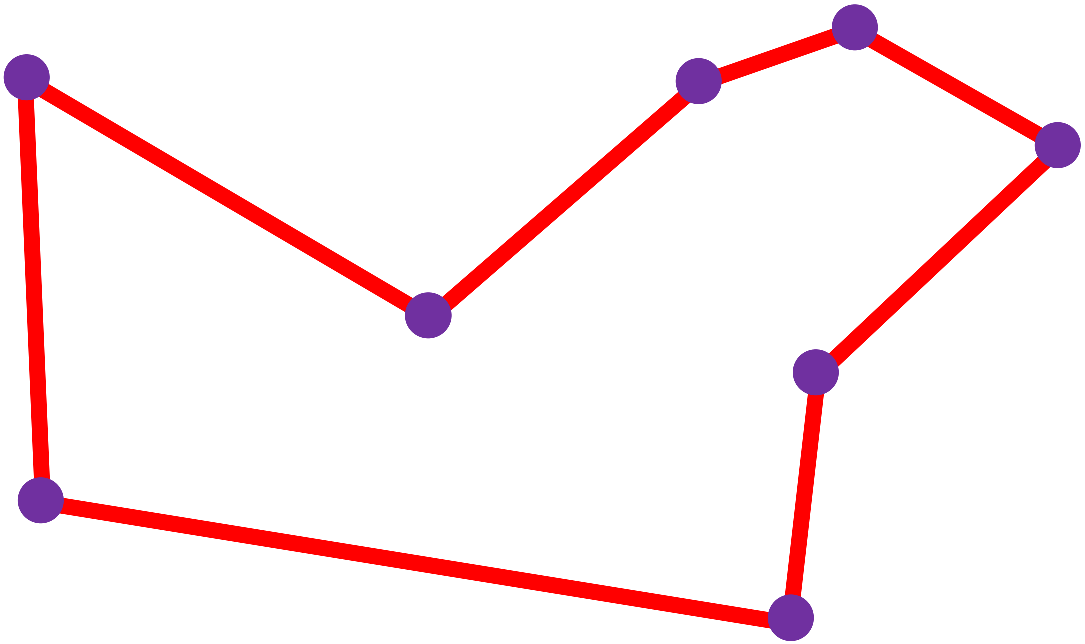

- Sweep a target region (recon, search), W



# Covering Tours

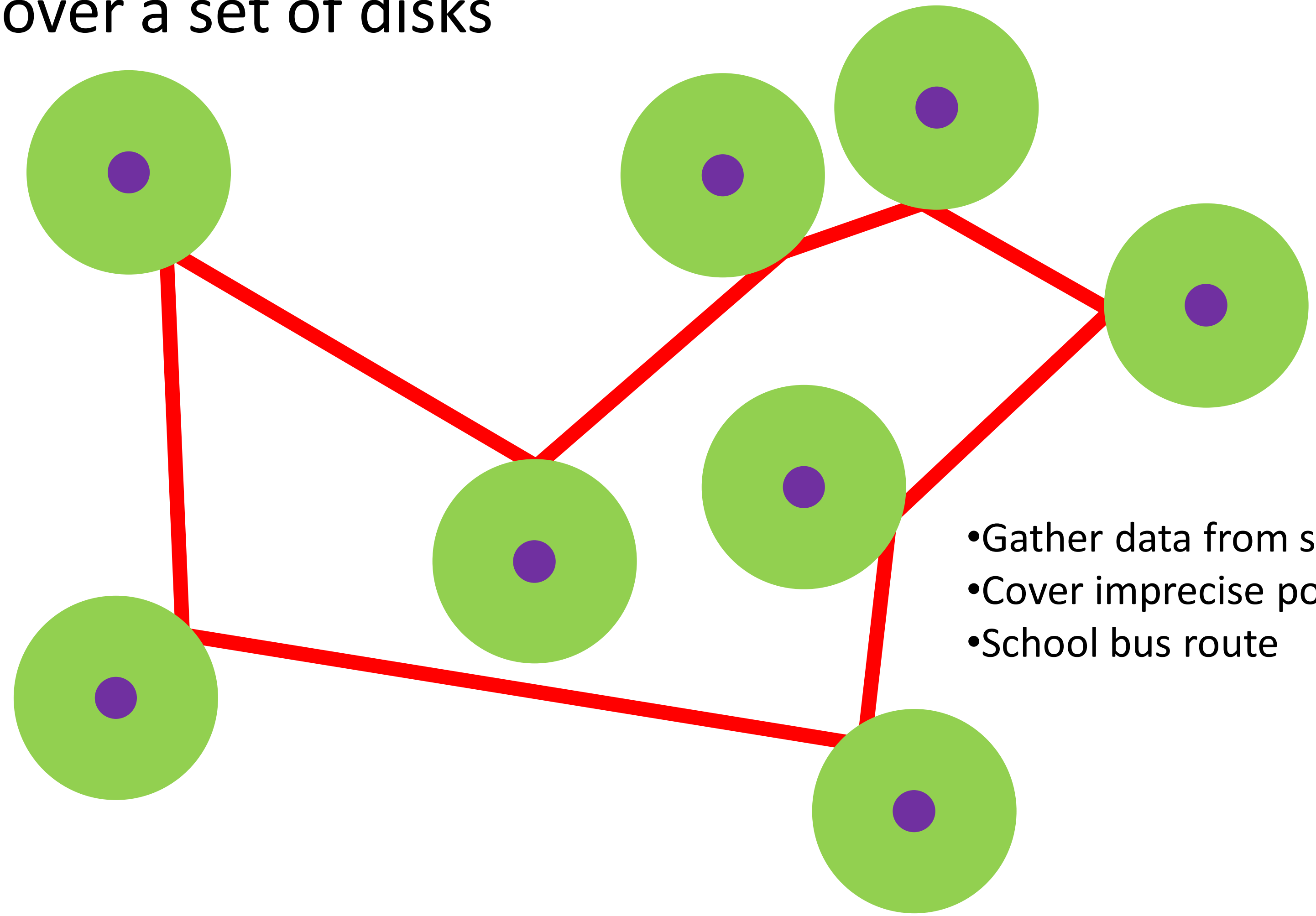
- Cover a point set S



Just geometric TSP

# Covering Tours

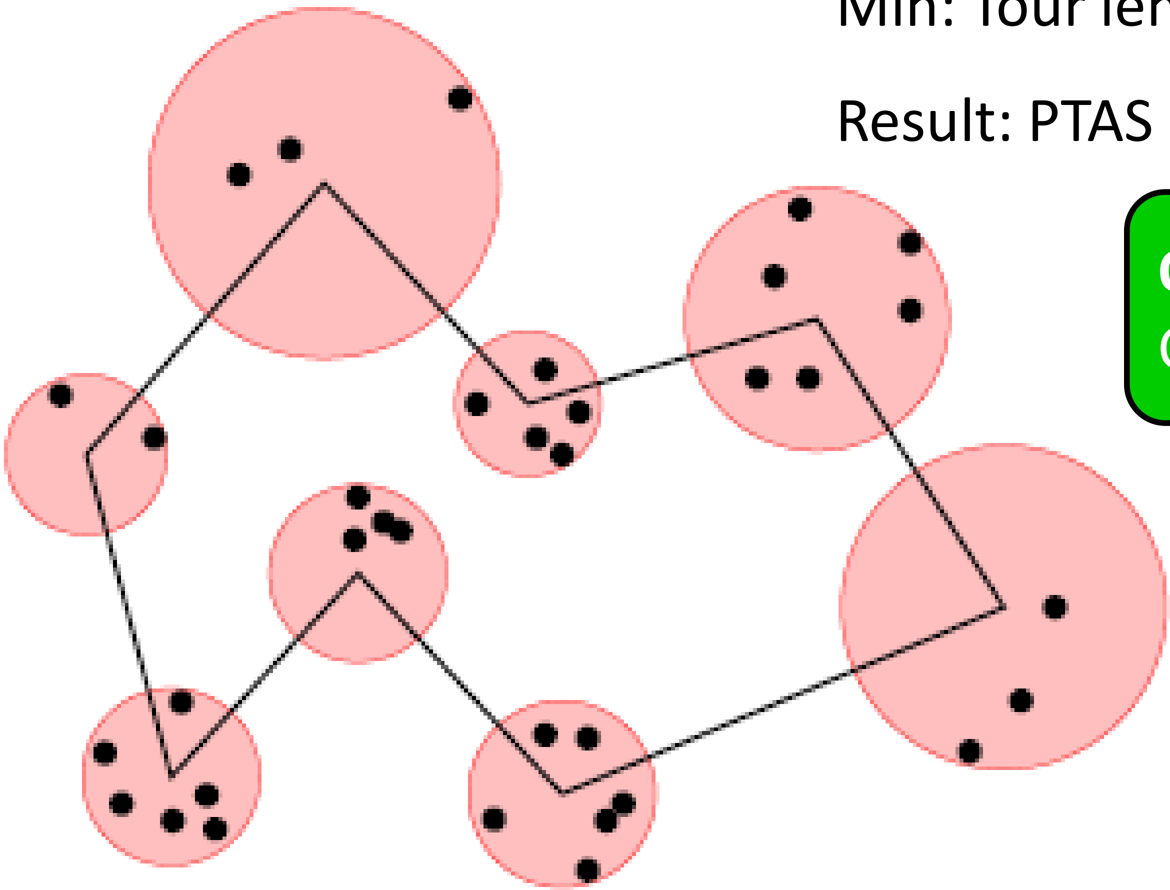
- Cover a set of disks



- Gather data from sensors
- Cover imprecise points
- School bus route

TSP with (circular) neighborhoods

# Sensor Network Application: Cover Tour Problem



Min: Tour length + C * (sum of radii)

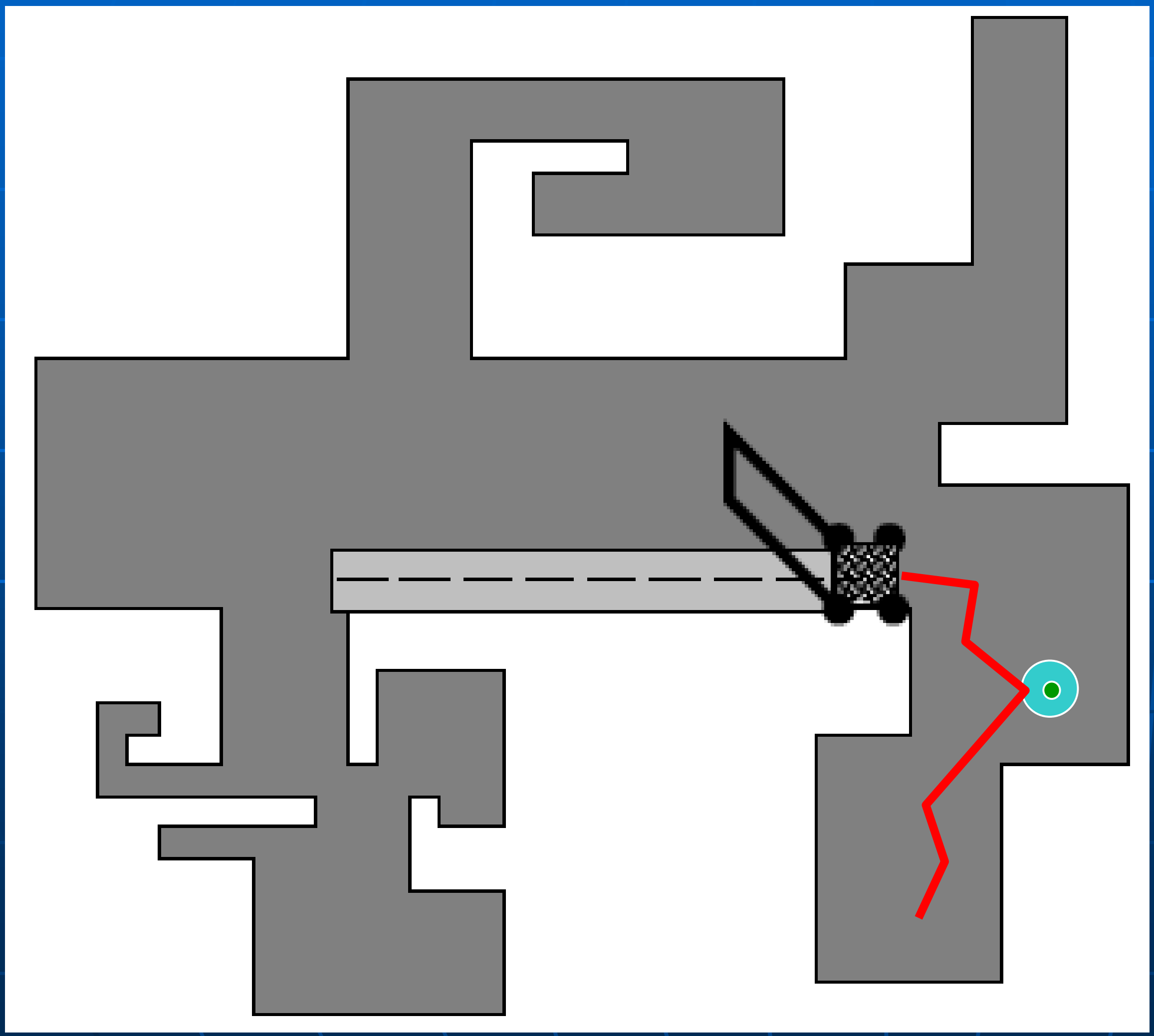
Result: PTAS

**Q:** Min Tour length + C * (sum of radii$^2$)?

C > 4 ; else OPT is a single disk

Alt, Arkin, Bronnimann, Erickson, Fekete, Knauer, Lenchner, M, Whittlesey, SoCG'06

# Lawnmower/Milling Problem



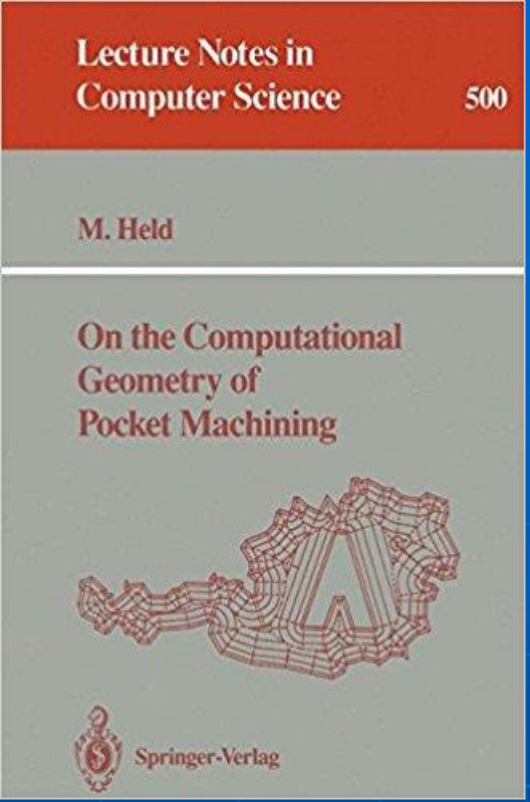
[AFM]

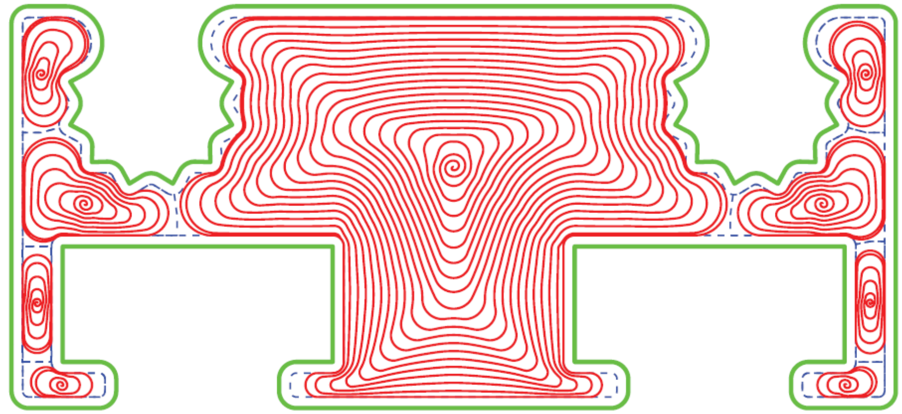
Best method of mowing the lawn?

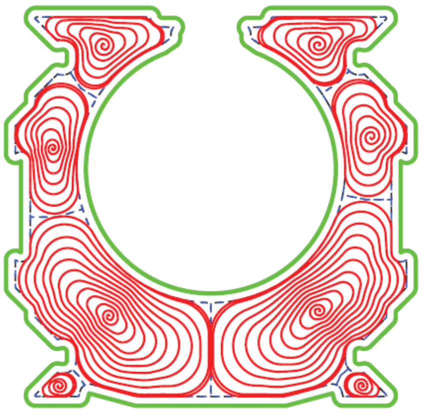
TSPN: Visit the disk centered at each blade of grass

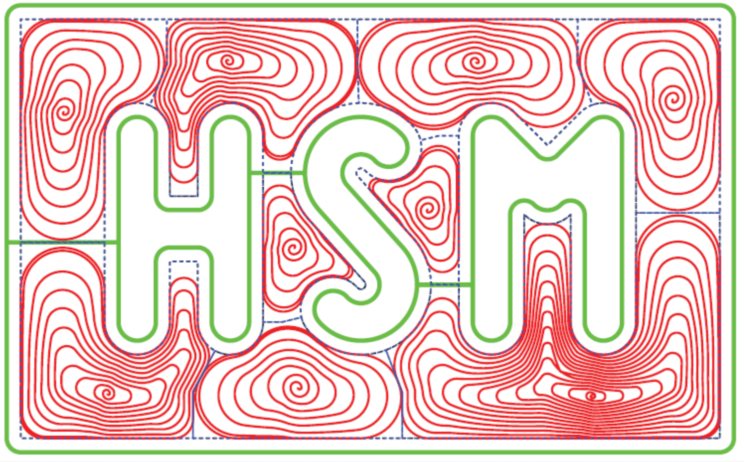
# Pocket Machining

[Martin Held]








# Watchman Route Problem

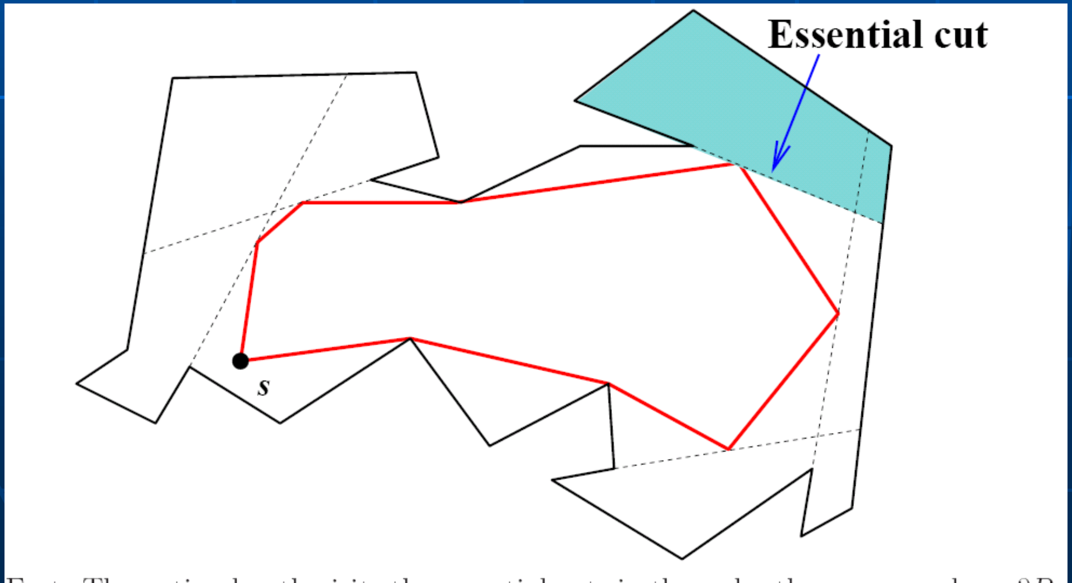
- Cover set of all visibility polygons

Subject to: stay inside polygonal domain P

Watchman Route Problem (WRP)

# Watchman Route Problem

- SoCG 1986: Chin and Ntafos
  - NP-hardness in 2D,3D; Revisited:[Dumitrescu, Toth 2012]
  - O(n) in rectilinear, simple polygons
- WRP in simple polygons: polytime
  - Long history…Current fastest: $O(n^3 \log n)$ for *anchored*, $O(n^4 \log n)$ for *floating* [STOC 2003: Dror, Efrat, Lubiw, M]




Fact: The optimal path visits the essential cuts in the order they appear along $\partial P$.

# WRP Approximation

- Simple polygons:
  - Sqrt(2)-approx, O(n), for anchored [Tan, DAM 2004]
  - $14(\pi+4)=99.98$-approx, O(n log n), for floating [Carlsson, Jonsson, Nilsson, TR 1997]
  - 2-approx, O(n), for floating [Tan, TCS 2007]
  - 4-approx, $O(n^2)$, for min-link [Alsuwaiyel, Lee, IPL 1995]
- Polygons with holes?
  - O(log n)-approx, rectilinear, rectangle-visibility
- WRP in 3D: No constant-factor, unless P=NP

[Safra, Schwartz 2003]

SODA'13: $O(\log^2 n)$, $\Omega(\log n)$

$\Omega(\log n)$, even for terrains

# General Case: WRP in Polygonal Domain (2D)

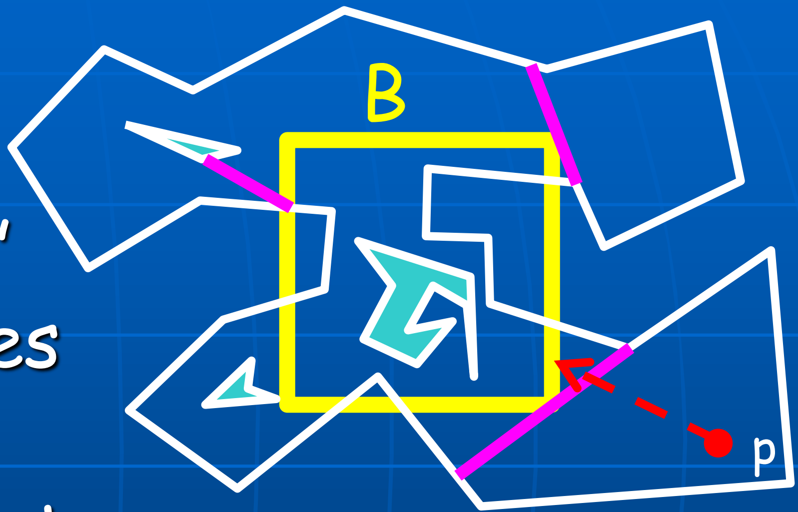
[M, SODA'13]

- **Theorem**: The WRP has an $O(\log^2 n)$-approximation algorithm.
- *Also*: WRP has an $O(\log n)$-approx in domain P satisfying the bounded perimeter assumption (BPA): perim(VP(p)) = O( diam(VP(p)), for p in P

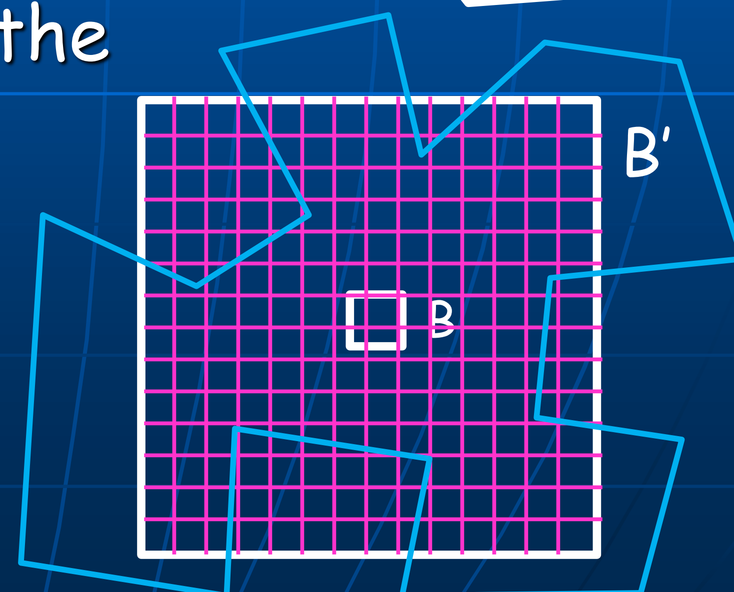
e.g., bounded degree corridor domains

# Main Ideas

- Localization: Consider a polynomial # of "minimal outer-illuminating squares" (MOIS), B, that OPT passes near/through




- Discretization: Show that the continuous problem can be discretized, using an appropriate grid



# Main Ideas

- Solve 2 separate problems:
  - OWRP: Outer WRP: Find a short tour γ within P that sees all of P outside the tour.
    - Discrete-OWRP: exact DP algorithm
    - OWRP: PTAS


  - IWRP: Inner WRP: For a *given* simple closed curve, γ, within P, augment γ (if needed) into a short network that sees all of P that is inside γ.
    - $O(\log^2 n)$-approx


- Combine

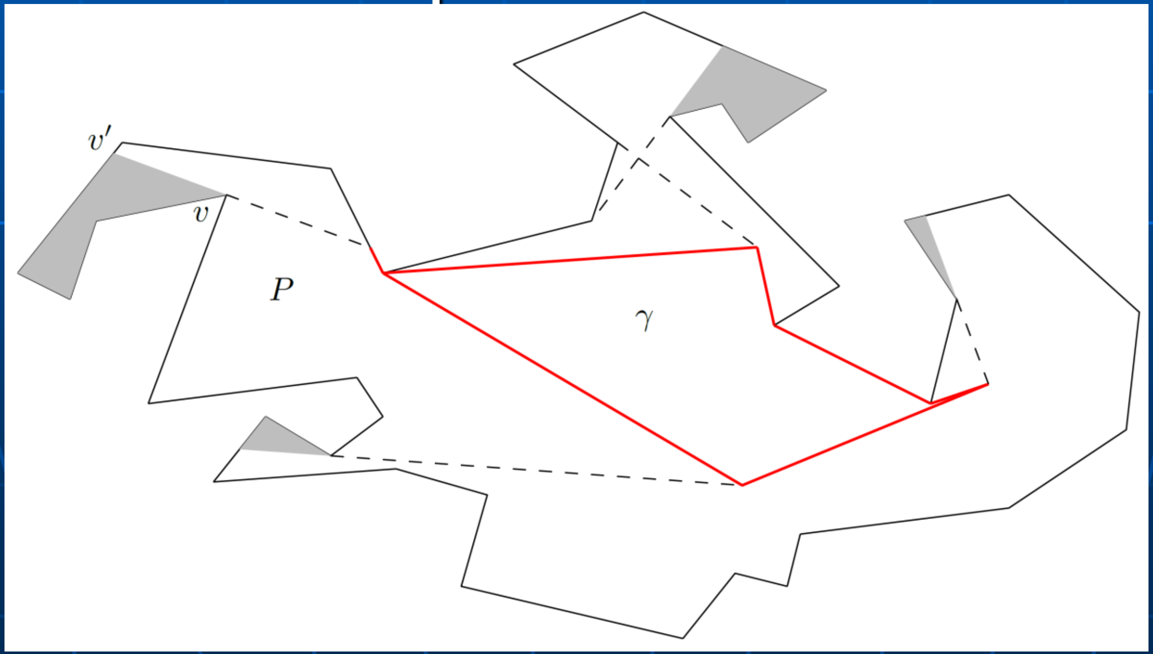
# Budgeted Watchman Route Problem

Orienteering Watchman

- BWRP: See as much as possible (e.g., area) on a route of length at most L

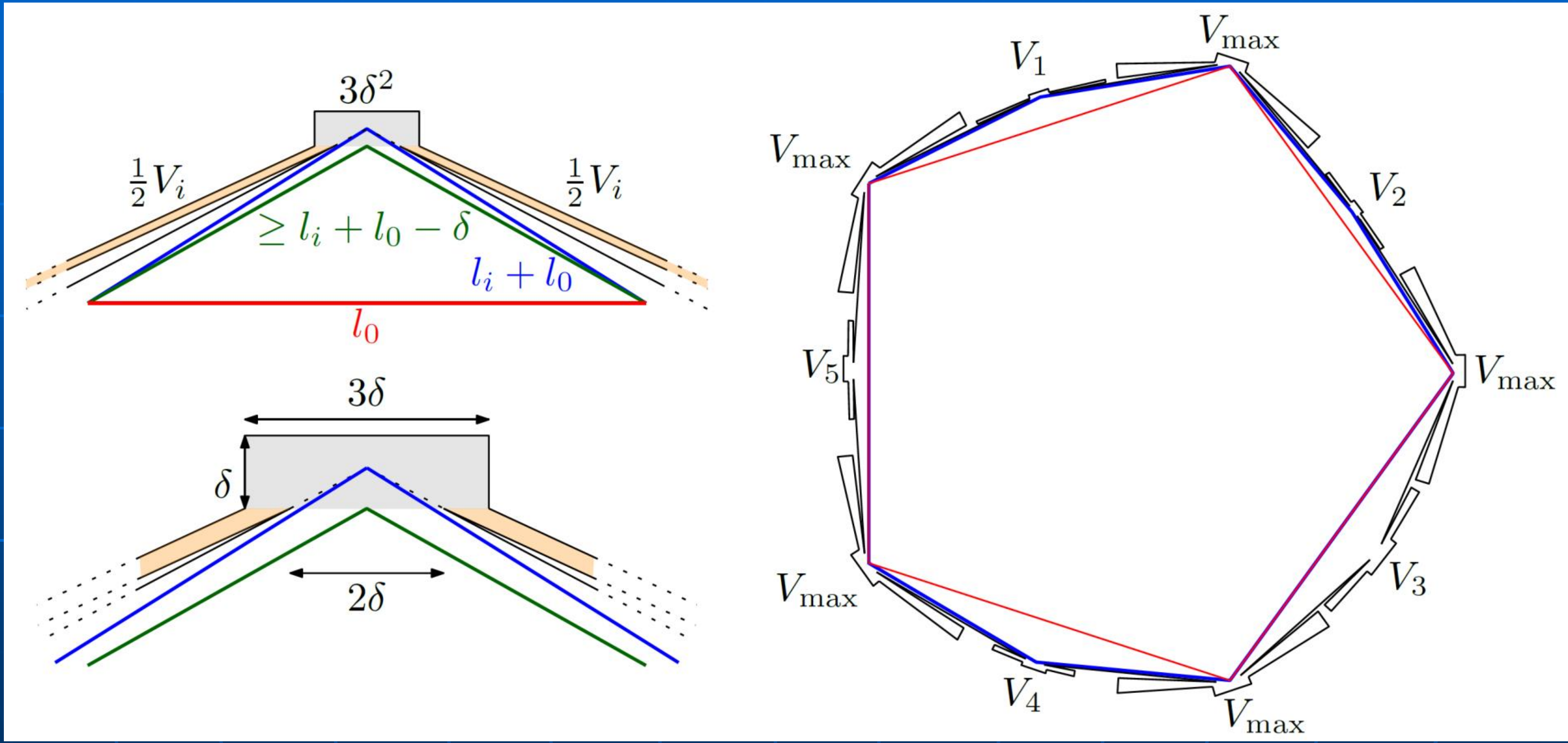
Special case: L=0: Find a point guard to see as much as possible [CEH, DCG'07]

- QWRP: Quota WRP: See area at least A using shortest route possible




[ongoing work with Kien Huynh, Linh Nguyen, Valentin Polishchuk]

# Hardness of BWRP




From KNAPSACK

Hardness also of QWRP, from INVERSE-KNAPSACK

# Approximation Algorithms

- Method for simple polygon P:
  - Localization of OPT (or possible depot, s)
  - Discretization (round to appropriate grid)
  - Dynamic programming
- BWRP: An FPTAS, poly(n,1/ε), to compute a tour seeing area ≥(1-ε)*$OPT_L$, using length ≤(1+ ε)L

  "floating", convex, no s: $n^{O(\frac{1}{\sqrt{\varepsilon}})}$
- QWRP: An FPTAS, poly(n,1/ε), to compute a tour seeing area ≥(1-ε)*A, using length ≤(1+ ε)*OPT

# Polygons with Holes

**Theorem 5.1.** *The BWRP in a polygon with holes cannot be approximated, in polynomial time, to a factor of* $(1-\varepsilon)$ *for arbitrary* $\varepsilon > 0$, *unless* $P = NP$.

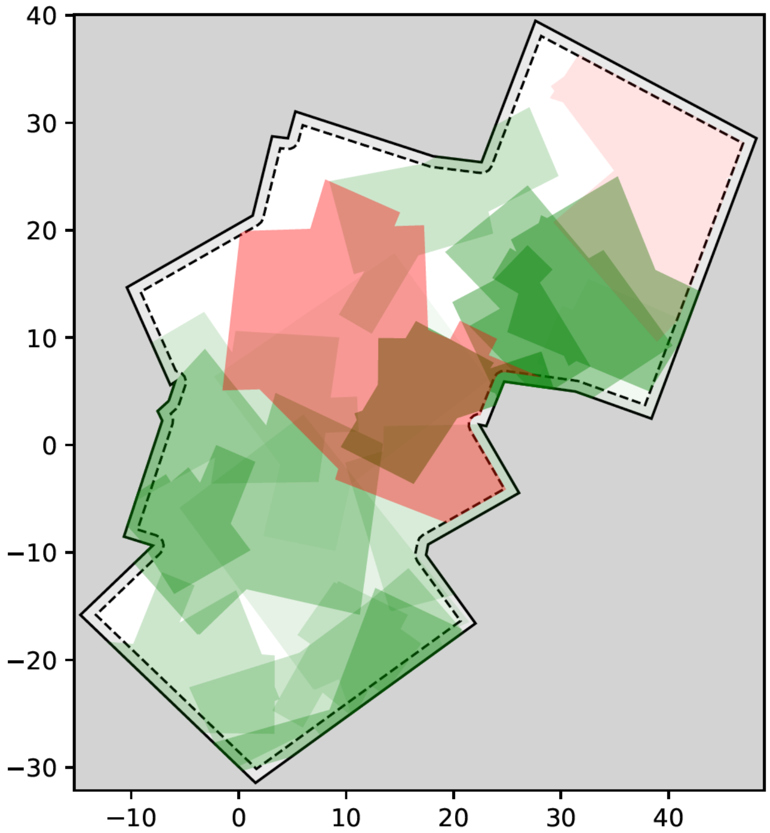
From Max-k-Vertex-Cover in cubic graphs

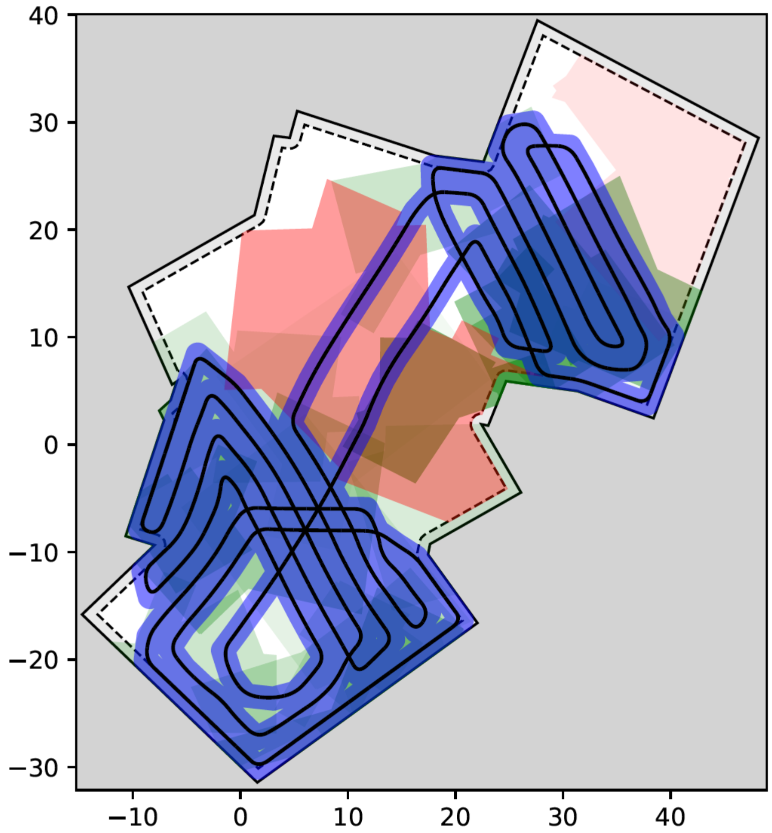
**Theorem 5.3.** *Given a polygon* $P$ *with holes, the BWRP has a dual approximation algorithm that computes a tour of length at most* $(1+\varepsilon)L$ *that sees at least* $\Omega\left(\frac{OPT \log \beta}{\log OPT}\right)$ *with running time* $\left(\frac{n}{\varepsilon}\log L\right)^{O\left(\beta \log \frac{n}{\varepsilon} / \log \beta\right)}$.

For any $\beta \leq 2$

# Practical Methods

Sweeping with a bounded radius disk



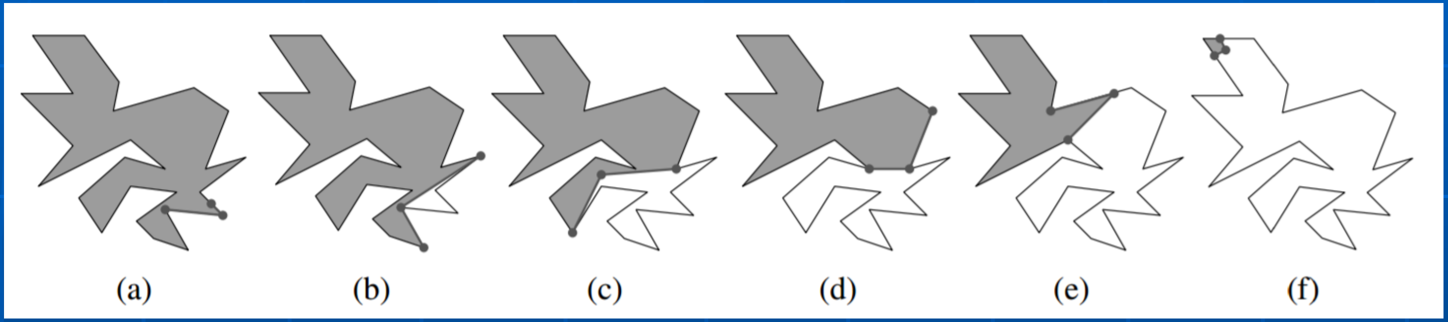




[thesis, Dominik Krupke, 2022; Fekete+Krupke, ALENEX'19]

# More Sweeping

- Sweeping with chains of visible agents, to "clean" a polygon with mobile evader

[Efrat,Guibas,Har-Peled,Mitchell,Murali DCG]




- Sweeping with a pair of agents/segment

[Kien Huynh, JM, Val Polishchuk]

# Sweeping with 2 Covisible Guards

[Kien Huynh, JM, Valentin Polishchuk, 2023]

- NP-hard, even in a simple, orthogonal polygon
- O(1)-approx
  - Simple polygons
  - Polygons with holes