Approximation Algorithms and Inapproximability Results for Art Gallery Problems

Pritam Bhattacharya

Department of Computer Science & Technology,
Indian Institute of Technology Kharagpur,
Kharagpur, West Bengal - 721302.

pritam.bhattacharya@cse.iitkgp.ernet.in

March 27, 2017

Polygons and Visibility

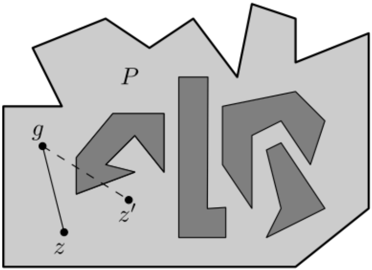


Figure: Polygon with holes

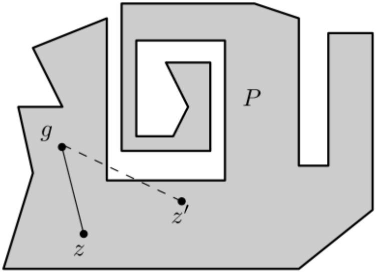


Figure: Polygon without holes

Definition (Visibility of a Point)

A point $z \in P$ is said to be *visible* from another point $g \in P$ if the line segment zg does not intersect the exterior of P.

Art Gallery Problem

The art gallery problem (AGP) enquires about the least number of guards that are sufficient to ensure that an art gallery is fully guarded, assuming that a guard's field of view covers 360° as well as an unbounded distance.

Art Gallery Problem

The art gallery problem (AGP) enquires about the least number of guards that are sufficient to ensure that an art gallery is fully guarded, assuming that a guard's field of view covers 360° as well as an unbounded distance.

An art gallery can be viewed as an n-sided polygon P (with or without holes) and guards as points in P.

Art Gallery Problem

The art gallery problem (AGP) enquires about the least number of guards that are sufficient to ensure that an art gallery is fully guarded, assuming that a guard's field of view covers 360° as well as an unbounded distance.

An art gallery can be viewed as an n-sided polygon P (with or without holes) and guards as points in P.

Guards may be allowed to be placed anywhere within P (*point guards*), or they may be allowed to be placed only on the vertices of P (*vertex guards*).

Art Gallery Problem

The art gallery problem (AGP) enquires about the least number of guards that are sufficient to ensure that an art gallery is fully guarded, assuming that a guard's field of view covers 360° as well as an unbounded distance.

An art gallery can be viewed as an n-sided polygon P (with or without holes) and guards as points in P.

Guards may be allowed to be placed anywhere within P (*point guards*), or they may be allowed to be placed only on the vertices of P (*vertex guards*).

Victor Klee (1973) $\rightarrow$ How many point guards or vertex guards are always sufficient to guard a simple polygon having n vertices?

Sufficient Number of Guards

Theorem (Chvatal (1975), Fisk (1978))

For guarding a simple polygon with n vertices, $\lfloor \frac{n}{3} \rfloor$ point guards or vertex guards are sufficient and sometimes necessary.

Sufficient Number of Guards

Theorem (Chvatal (1975), Fisk (1978))

For guarding a simple polygon with n vertices, $\lfloor \frac{n}{3} \rfloor$ point guards or vertex guards are sufficient and sometimes necessary.

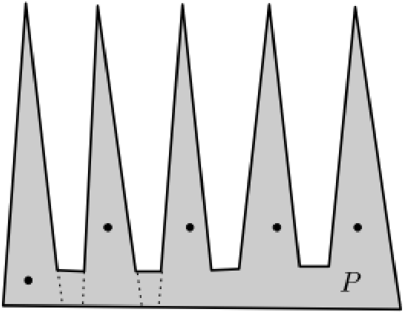


Figure: A polygon where $\lfloor \frac{n}{3} \rfloor$ point guards or vertex guards are necessary.

Sufficient Number of Guards

Theorem (O'Rourke(1983))

For guarding a simple orthogonal polygon with n vertices, $\lfloor \frac{n}{4} \rfloor$ point guards or vertex guards are sufficient and sometimes necessary.

Sufficient Number of Guards

Theorem (O'Rourke(1983))

For guarding a simple orthogonal polygon with n vertices, $\lfloor \frac{n}{4} \rfloor$ *point guards or vertex guards are sufficient and sometimes necessary.*

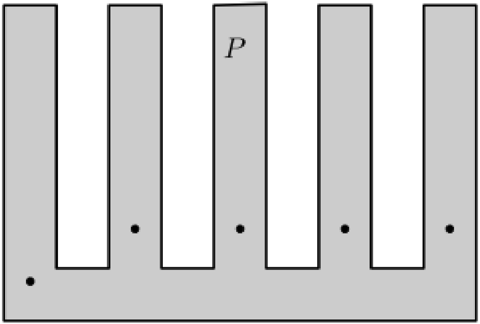


Figure: A polygon where $\lfloor \frac{n}{4} \rfloor$ stationary guards are necessary.

Literature Survey - Hardness Results

Definition (Decision Version of the Art Gallery Problem)

Given a polygon P and a number k as input, can the polygon P be guarded with k or fewer guards?

Literature Survey - Hardness Results

Definition (Decision Version of the Art Gallery Problem)

Given a polygon P and a number k as input, can the polygon P be guarded with k or fewer guards?

Hardness results known for the decision version of AGP:

- Proved to be NP-complete for vertex guards (Lee and Lin).

Literature Survey - Hardness Results

Definition (Decision Version of the Art Gallery Problem)

Given a polygon P and a number k as input, can the polygon P be guarded with k or fewer guards?

Hardness results known for the decision version of AGP:

- Proved to be NP-complete for vertex guards (Lee and Lin).
- Proved to be NP-complete for point guards (Aggarwal).

Literature Survey - Hardness Results

Definition (Decision Version of the Art Gallery Problem)

Given a polygon P and a number k as input, can the polygon P be guarded with k or fewer guards?

Hardness results known for the decision version of AGP:

- Proved to be NP-complete for vertex guards (Lee and Lin).
- Proved to be NP-complete for point guards (Aggarwal).
- Proved to be APX-complete (Eidenbenz, Stamm and Widmayer), implying that no PTAS can exist for AGP.

Literature Survey - Hardness Results

Definition (Decision Version of the Art Gallery Problem)

Given a polygon P and a number k as input, can the polygon P be guarded with k or fewer guards?

Hardness results known for the decision version of AGP:

- Proved to be NP-complete for vertex guards (Lee and Lin).
- Proved to be NP-complete for point guards (Aggarwal).
- Proved to be APX-complete (Eidenbenz, Stamm and Widmayer), implying that no PTAS can exist for AGP.
- Specifically for polygons with holes, AGP cannot be approximated to within a factor of $\Omega(\ln n)$ (Eidenbenz, Stamm and Widmayer).

Literature Survey - Approximation Algorithms

For computing the minimum number of guards, the following approximation algorithms exist:

- $\mathcal{O}(\log n)$-approximation algorithm for vertex and edge guards by Ghosh in 1987 via a reduction to set cover.

Literature Survey - Approximation Algorithms

For computing the minimum number of guards, the following approximation algorithms exist:

- $\mathcal{O}(\log n)$-approximation algorithm for vertex and edge guards by Ghosh in 1987 via a reduction to set cover.
- $\mathcal{O}(\log OPT)$-approximation pseudopolynomial time algorithm for point guards and perimeter guards by Deshpande et al.

Literature Survey - Approximation Algorithms

For computing the minimum number of guards, the following approximation algorithms exist:

- $\mathcal{O}(\log n)$-approximation algorithm for vertex and edge guards by Ghosh in 1987 via a reduction to set cover.
- $\mathcal{O}(\log OPT)$-approximation pseudopolynomial time algorithm for point guards and perimeter guards by Deshpande et al.
- $\mathcal{O}(\log OPT)$-approximation randomized algorithm with fully polynomial expected running time by Efrat and Har-Peled.

Literature Survey - Approximation Algorithms

For computing the minimum number of guards, the following approximation algorithms exist:

- $\mathcal{O}(\log n)$-approximation algorithm for vertex and edge guards by Ghosh in 1987 via a reduction to set cover.
- $\mathcal{O}(\log OPT)$-approximation pseudopolynomial time algorithm for point guards and perimeter guards by Deshpande et al.
- $\mathcal{O}(\log OPT)$-approximation randomized algorithm with fully polynomial expected running time by Efrat and Har-Peled.
- $\mathcal{O}(\log \log OPT)$-approximation algorithm for perimeter guards by King and Kirkpatrick in 2011 by using ϵ-nets.

Literature Survey - Approximation Algorithms

For computing the minimum number of guards, the following approximation algorithms exist:

- $\mathcal{O}(\log n)$-approximation algorithm for vertex and edge guards by Ghosh in 1987 via a reduction to set cover.
- $\mathcal{O}(\log OPT)$-approximation pseudopolynomial time algorithm for point guards and perimeter guards by Deshpande et al.
- $\mathcal{O}(\log OPT)$-approximation randomized algorithm with fully polynomial expected running time by Efrat and Har-Peled.
- $\mathcal{O}(\log \log OPT)$-approximation algorithm for perimeter guards by King and Kirkpatrick in 2011 by using ϵ-nets.

Conjecture (Ghosh (1987))

There exist polynomial time algorithms with a constant approximation ratio for vertex guarding polygons without holes.

Summary of Our Results

- We obtain a 6-approximation algorithm, which has running time $\mathcal{O}(n^2)$, for vertex guarding polygons that are weakly visible from an edge and contain no holes. This result settles Ghosh's conjecture for a special class of polygons.

Summary of Our Results

- We obtain a 6-approximation algorithm, which has running time $\mathcal{O}(n^2)$, for vertex guarding polygons that are weakly visible from an edge and contain no holes. This result settles Ghosh's conjecture for a special class of polygons.
- We prove that the above approximation ratio can be improved to 3 for the special class of polygons without holes that are orthogonal as well as weakly visible from an edge.

Summary of Our Results

- We obtain a 6-approximation algorithm, which has running time $\mathcal{O}(n^2)$, for vertex guarding polygons that are weakly visible from an edge and contain no holes. This result settles Ghosh's conjecture for a special class of polygons.
- We prove that the above approximation ratio can be improved to 3 for the special class of polygons without holes that are orthogonal as well as weakly visible from an edge.
- Through a reduction from the Set Cover problem, we prove that, for the special class of polygons containing holes that are weakly visible from an edge, there cannot exist a polynomial time algorithm for the vertex guard problem with an approximation ratio better than $((1-\epsilon)/12)\ln n$ for any $\epsilon > 0$, unless NP $=$ P.

Summary of Our Results

- We obtain a 6-approximation algorithm, which has running time $\mathcal{O}(n^2)$, for vertex guarding polygons that are weakly visible from an edge and contain no holes. This result settles Ghosh's conjecture for a special class of polygons.
- We prove that the above approximation ratio can be improved to 3 for the special class of polygons without holes that are orthogonal as well as weakly visible from an edge.
- Through a reduction from the Set Cover problem, we prove that, for the special class of polygons containing holes that are weakly visible from an edge, there cannot exist a polynomial time algorithm for the vertex guard problem with an approximation ratio better than $((1-\epsilon)/12)\ln n$ for any $\epsilon > 0$, unless NP = P.
- We prove that the point guard problem for weak visibility polygons is NP-hard by showing a reduction from the decision version of the minimum line cover problem.

Visibility Polygons

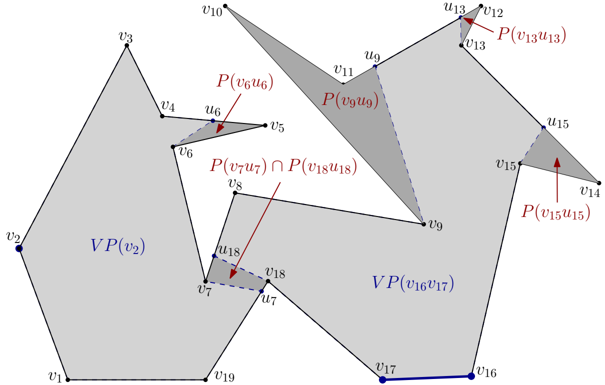


Figure: Figure showing visibility polygon $\mathcal{VP}(v_2)$ and weak visibility polygon $\mathcal{VP}(v_{16}v_{17})$, along with several pockets created by constructed edges belonging to both. Observe that the boundary of $\mathcal{VP}(z)$ consists of polygonal edges and *constructed edges*. Note that one point of a constructed edge is a vertex of P, while the other point lies on $bd(P)$.

Visibility Polygons

Definition (Visibility Polygon)

The *visibility polygon* of P from a point z, denoted as $\mathcal{VP}(z)$, is defined to be the set of all points of P that are visible from z. In other words, $\mathcal{VP}(z) = \{q \in P : q \text{ is visible from } z\}$.

Visibility Polygons

Definition (Visibility Polygon)

The *visibility polygon* of P from a point z, denoted as $\mathcal{VP}(z)$, is defined to be the set of all points of P that are visible from z. In other words, $\mathcal{VP}(z) = \{q \in P : q \text{ is visible from } z\}$.

Definition (Weak Visibility Polygon)

A point q of P is said to be *weakly visible* from bc if there exists a point $z \in bc$ such that q is visible from z. The set of all such points of P is said to be the *weak visibility polygon* of P from bc, and denoted as $\mathcal{VP}(bc)$.

Weakly Visible Polygons

Definition (Weak Visibility Polygon)

A point q of P is said to be *weakly visible* from bc if there exists a point $z \in bc$ such that q is visible from z. The set of all such points of P is said to be the *weak visibility polygon* of P from bc, and denoted as $\mathcal{VP}(bc)$.

Weakly Visible Polygons

Definition (Weak Visibility Polygon)

A point q of P is said to be *weakly visible* from bc if there exists a point $z \in bc$ such that q is visible from z. The set of all such points of P is said to be the *weak visibility polygon* of P from bc, and denoted as $\mathcal{VP}(bc)$.

Definition (Weakly Visible Polygon)

If $\mathcal{VP}(v_i v_{i+1}) = P$ for a polygonal edge $v_i v_{i+1}$, then P is called a *weakly visible polygon*.

Euclidean Shortest Path Tree

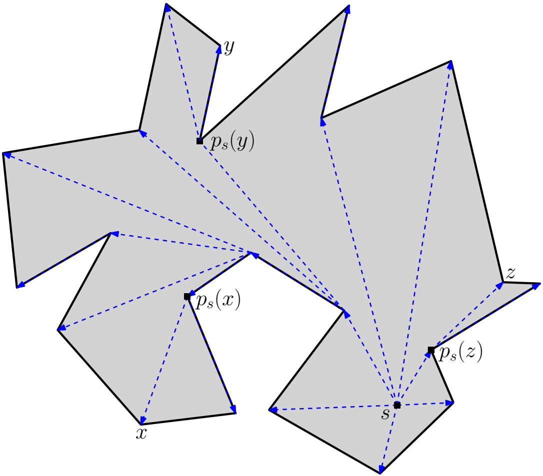
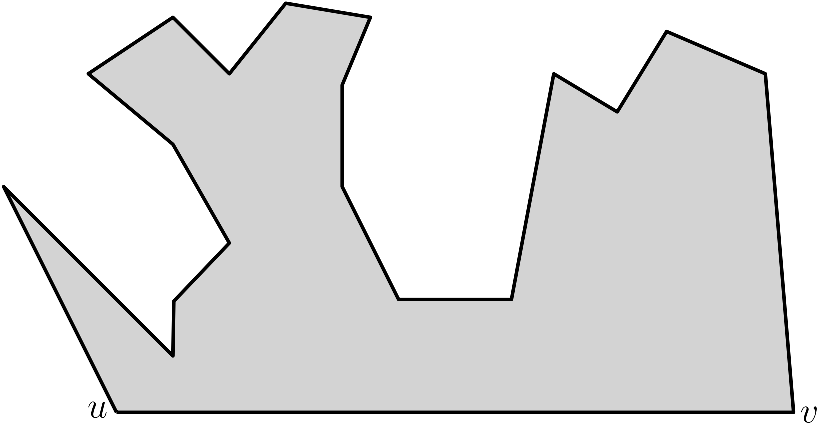


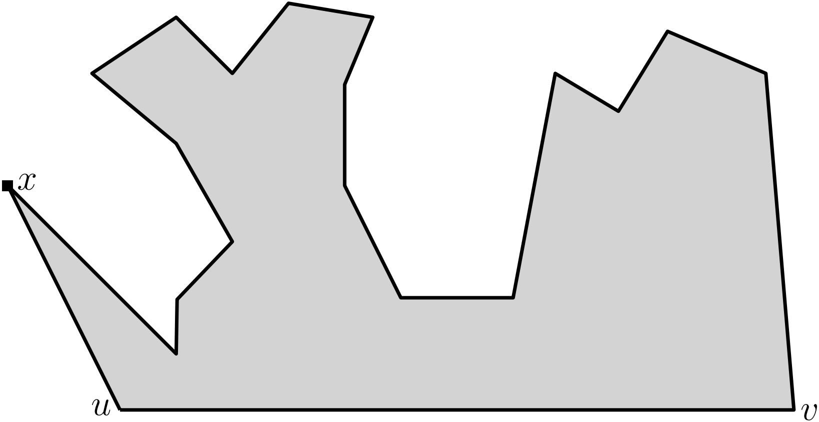
Figure: Euclidean shortest path tree rooted at s. The parents of vertices x, y and z in $SPT(s)$ are marked as $p_s(x)$, $p_s(y)$ and $p_s(z)$ respectively.

A Naive Algorithm for Guarding All Vertices



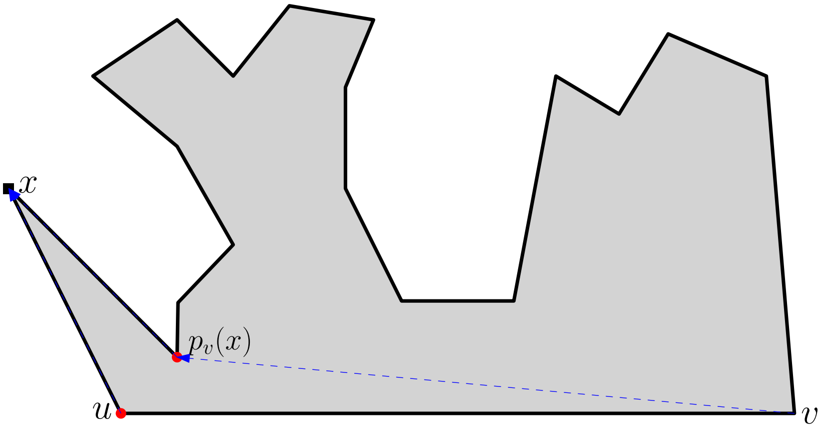
$$A = \{\}; S_A = \{\}$$

A Naive Algorithm for Guarding All Vertices



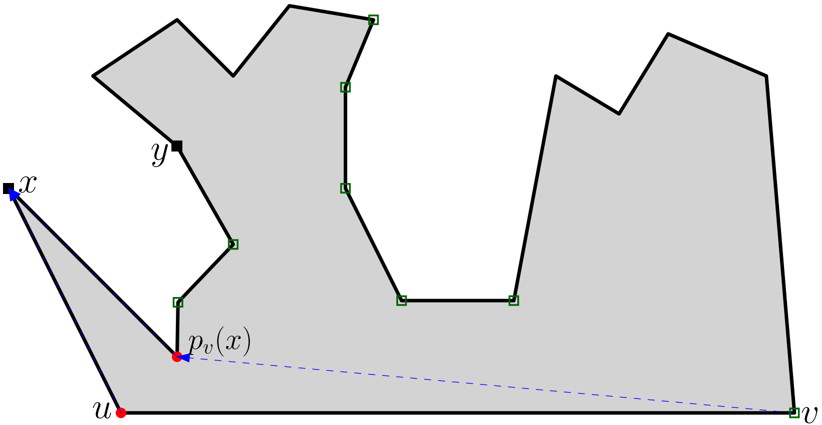
$A = \{x\}$; $S_A = \{\}$

A Naive Algorithm for Guarding All Vertices



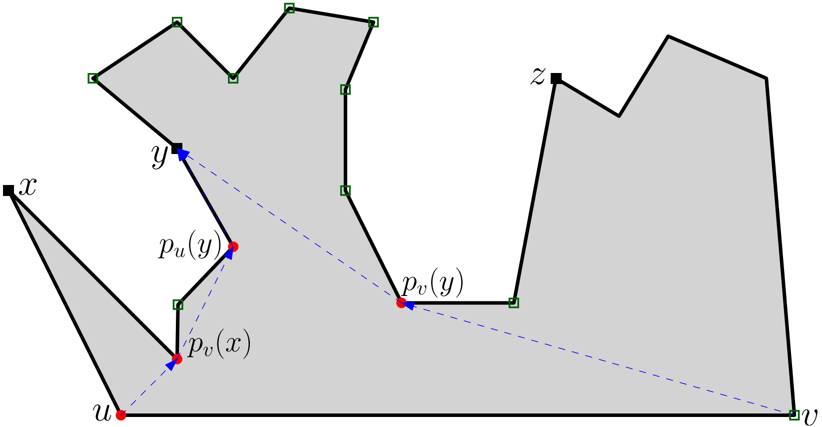
$$A = \{x\} \; ; \; S_A = \{u, p_v(x)\}$$

A Naive Algorithm for Guarding All Vertices



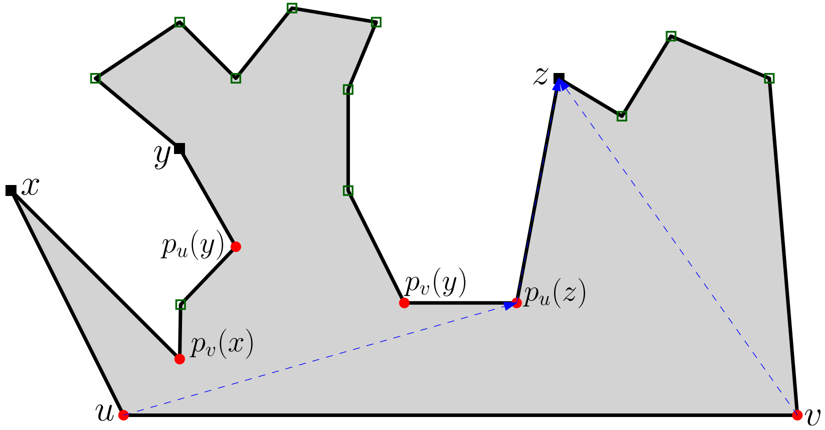
$$A = \{x\} \; ; \; S_A = \{u, p_v(x)\}$$

A Naive Algorithm for Guarding All Vertices



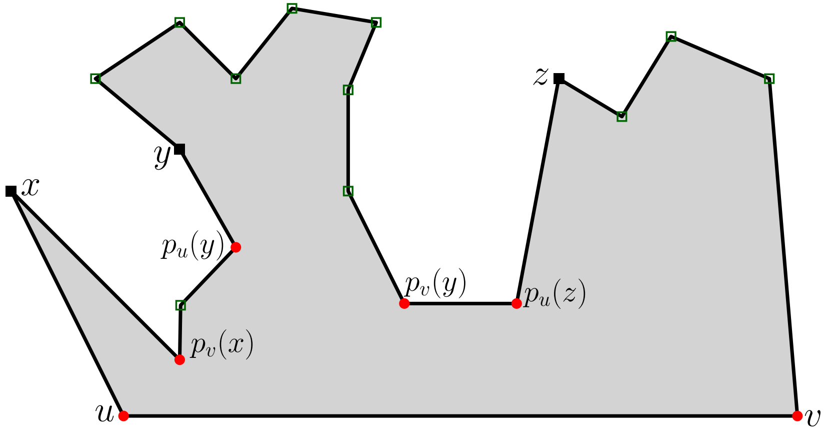
$$A = \{x, y\} \; ; \; S_A = \{u, p_v(x), p_u(y), p_v(y)\}$$

A Naive Algorithm for Guarding All Vertices



$$A = \{x, y, z\} \; ; \; S_A = \{u, p_v(x), p_u(y), p_v(y), p_u(z), v\}$$

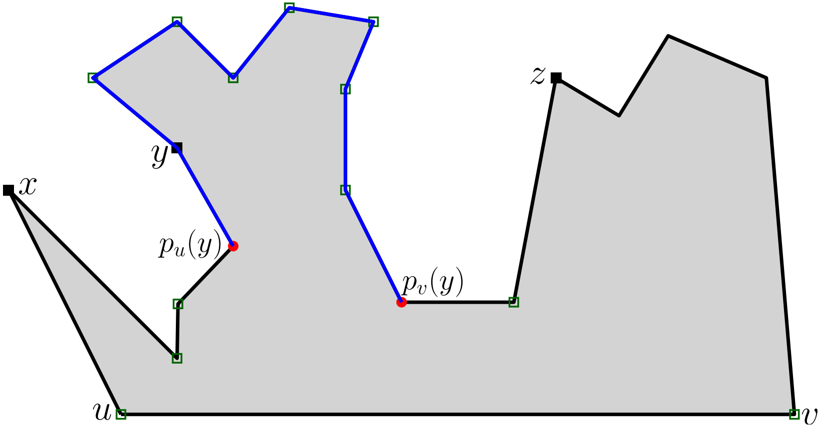
A Naive Algorithm for Guarding All Vertices



$A = \{x, y, z\}$; $S_A = \{u, p_v(x), p_u(y), p_v(y), p_u(z), v\}$

N.B. - $|S_A| = 2|A|$

Performance Guarantee under a Special Condition



N.B. - The vertex $y \in A$ is such that every vertex lying on the clockwise boundary between $p_u(y)$ and $p_v(y)$ (henceforth denoted as $bd_c(p_u(y), p_v(y))$) is visible from $p_u(y)$ or $p_v(y)$.

Performance Guarantee under a Special Condition

Lemma

If each vertex $z \in A$ is such that every vertex of $bd_c(p_u(z), p_v(z))$ is visible from $p_u(z)$ or $p_v(z)$, then $|S_A| \leq 2|S_{opt}|$.

Performance Guarantee under a Special Condition

Lemma

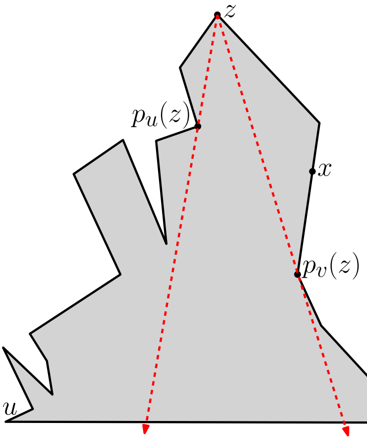
If each vertex $z \in A$ is such that every vertex of $bd_c(p_u(z), p_v(z))$ is visible from $p_u(z)$ or $p_v(z)$, then $|S_A| \leq 2|S_{opt}|$.

Proof.

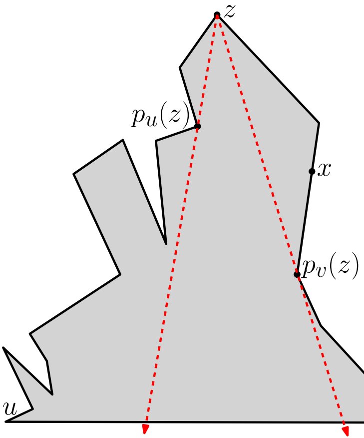
- $|S_A| = 2|A|$
- $|A| \leq |S_{opt}|$ (to be shown next)
- Therefore, $|S_A| = 2|A| \leq 2|S_{opt}|$

□

Location of an Optimal Guard for Vertex z



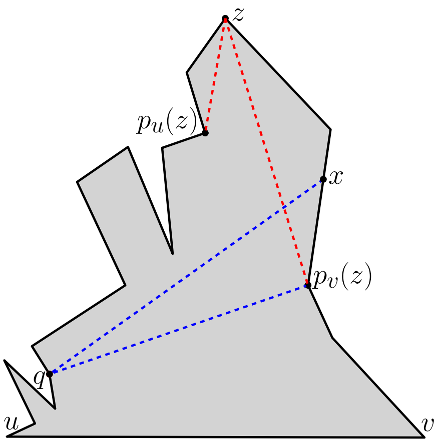
Location of an Optimal Guard for Vertex z



Lemma

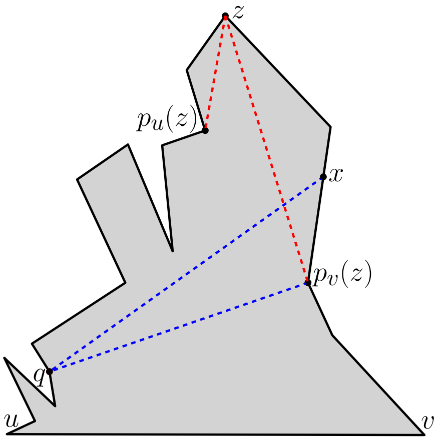
Any guard $x \in S_{opt}$ that sees z must lie on $bd_c(p_u(z), p_v(z))$.

Proof sketch of $|A| \leq |S_{opt}|$



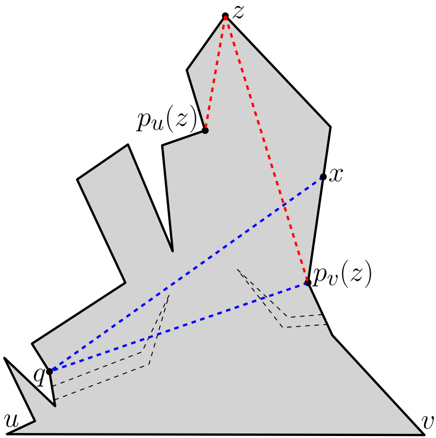
- All vertices of $bd_c(p_u(z), p_v(z))$ are visible from $p_u(z)$ or $p_v(z)$.

Proof sketch of $|A| \leq |S_{opt}|$



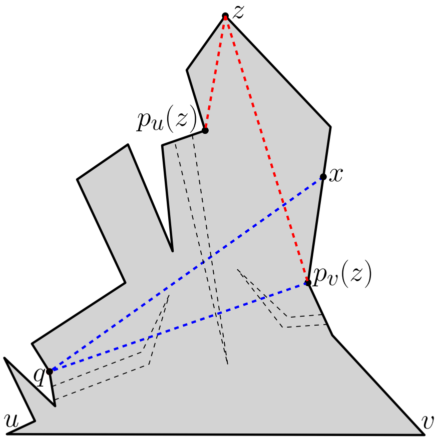
- All vertices of $bd_c(p_u(z), p_v(z))$ are visible from $p_u(z)$ or $p_v(z)$.
- If q is visible from x, then q must be visible from $p_u(z)$ or $p_v(z)$.

Proof sketch of $|A| \leq |S_{opt}|$



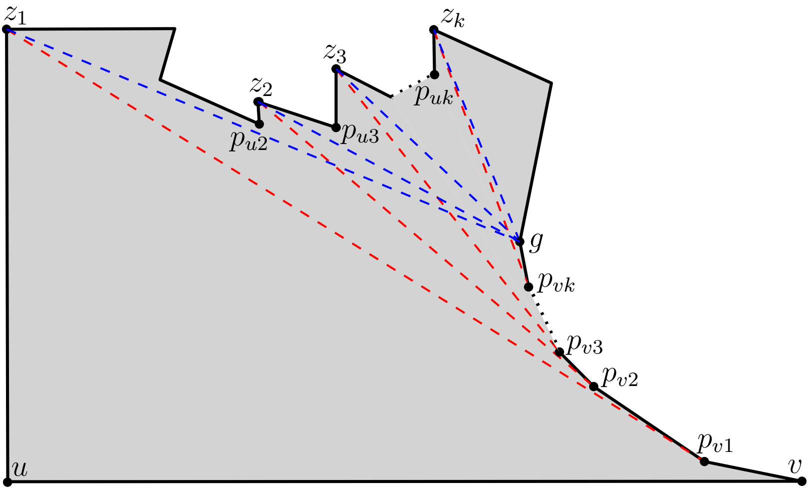
- All vertices of $bd_c(p_u(z), p_v(z))$ are visible from $p_u(z)$ or $p_v(z)$.
- If q is visible from x, then q must be visible from $p_u(z)$ or $p_v(z)$.

Proof sketch of $|A| \leq |S_{opt}|$



- All vertices of $bd_c(p_u(z), p_v(z))$ are visible from $p_u(z)$ or $p_v(z)$.
- If q is visible from x, then q must be visible from $p_u(z)$ or $p_v(z)$.

A Bad Input Polygon for the Naive Algorithm



For this input instance, $|S_A| = 2k$, whereas $S_{opt} = \{u, g\}$.

A Better Strategy for Guarding All Vertices

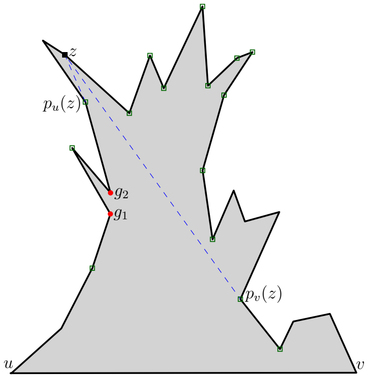
$$B = \{\} \; ; \; S = \{\}$$

Improved Strategy - Skip some unmarked vertices along the clockwise scan and choose vertices to include in B more carefully!

Invariant - If z is the current vertex under consideration along the clockwise scan, then every vertex of $bd_c(u, z)$ is visible from some guard in $S \cup \{p_u(z), p_v(z)\}$.

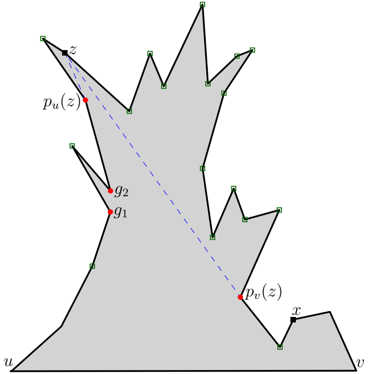
A Better Strategy for Guarding All Vertices

Case 1 - Every vertex lying on $bd_c(z, p_v(z))$, except z itself, is either visible already from guards currently in S or becomes visible if new guards are placed at $p_u(z)$ and $p_v(z)$.



A Better Strategy for Guarding All Vertices

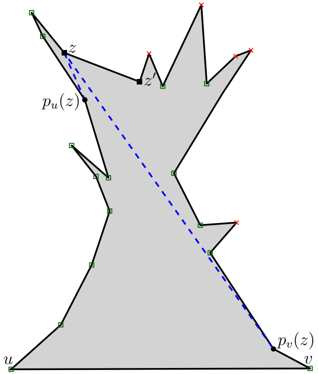
Case 1 - Every vertex lying on $bd_c(z, p_v(z))$, except z itself, is either visible already from guards currently in S or becomes visible if new guards are placed at $p_u(z)$ and $p_v(z)$.



$$B = B \cup \{z\} \ ; \ S = S \cup \{p_u(z), p_v(z)\} \ ; \ z = x$$

A Better Strategy for Guarding All Vertices

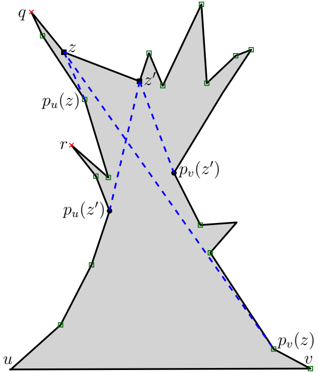
Case 2 - There exist some vertices lying on $bd_c(z, p_v(z))$, not visible already from guards currently in S, such that they do not become visible even if new guards are placed at $p_u(z)$ and $p_v(z)$.



Let z' be the next vertex along the clockwise scan that is not visible from any guard already in S.

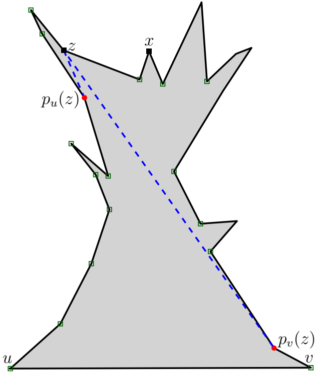
A Better Strategy for Guarding All Vertices

Case 2a - Not every unmarked vertex of $bd_c(p_u(z'), z')$ is visible from $p_u(z')$ or $p_v(z')$.



A Better Strategy for Guarding All Vertices

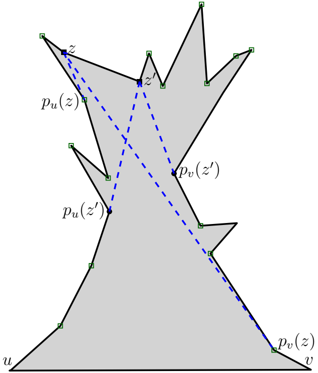
Case 2a - Not every unmarked vertex of $bd_c(p_u(z'), z')$ is visible from $p_u(z')$ or $p_v(z')$.



$$B = B \cup \{z\} \ ; \ S = S \cup \{p_u(z), p_v(z)\} \ ; \ z = x$$

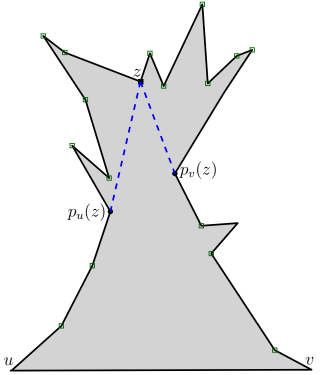
A Better Strategy for Guarding All Vertices

Case 2b - Every unmarked vertex of $bd_c(p_u(z'), z')$ is visible from $p_u(z')$ or $p_v(z')$.



A Better Strategy for Guarding All Vertices

Case 2b - Every unmarked vertex of $bd_c(p_u(z'), z')$ is visible from guards at $p_u(z')$ or $p_v(z')$.



$$B = B \cup \{\} \ ; \ S = S \cup \{\} \ ; \ z = z'$$

Approximation Ratio of our Algorithm

Lemma

$|B| \leq 2|S_{opt}|$.

Approximation Ratio of our Algorithm

Lemma

$|B| \leq 2|S_{opt}|$.

Proof.

There exists a bipartite graph $G = (B \cup S_{opt}, E)$ such that:
(a) the degree of each vertex in B is exactly 1, and,
(b) the degree of each vertex in S_{opt} is at most 2. □

Approximation Ratio of our Algorithm

Lemma

$|B| \leq 2|S_{opt}|$.

Proof.

There exists a bipartite graph $G = (B \cup S_{opt}, E)$ such that:
(a) the degree of each vertex in B is exactly 1, and,
(b) the degree of each vertex in S_{opt} is at most 2. □

Lemma

$|S| \leq 4|S_{opt}|$.

Approximation Ratio of our Algorithm

Lemma

$|B| \leq 2|S_{opt}|$.

Proof.

There exists a bipartite graph $G = (B \cup S_{opt}, E)$ such that:
(a) the degree of each vertex in B is exactly 1, and,
(b) the degree of each vertex in S_{opt} is at most 2. □

Lemma

$|S| \leq 4|S_{opt}|$.

Proof.

- $|S| = 2|B|$
- $|B| \leq 2|S_{opt}|$

Therefore, $|S| = 2|B| \leq 4|S_{opt}|$. □

Insufficiency of Guards in S to Cover all Interior Points

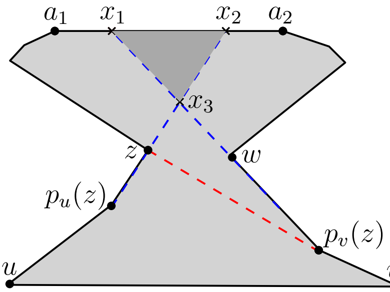


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1x_2x_3$ are invisible.

Insufficiency of Guards in S to Cover all Interior Points

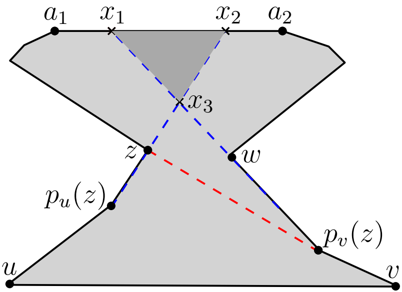


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1x_2x_3$ are invisible.

NOTE: One of the sides x_1x_2 of the triangle $x_1x_2x_3$ is a part of the polygonal edge a_1a_2. In fact, for any such invisible region, one of the sides must always be part of a polygonal edge.

Insufficiency of Guards in S to Cover all Interior Points

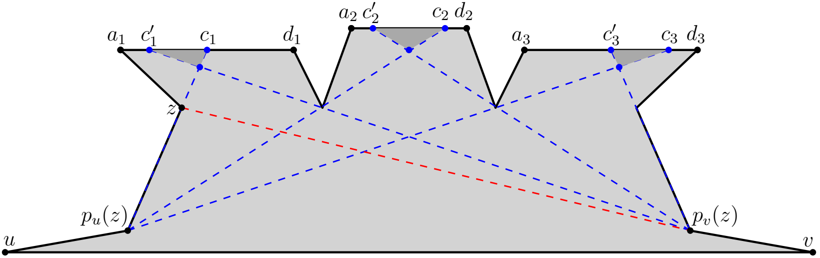


Figure: Multiple invisible regions exist within the polygon that are not visible from the guard set $S = \{p_u(z), p_v(z)\}$.

Placement of More Guards to Cover all Interior Points

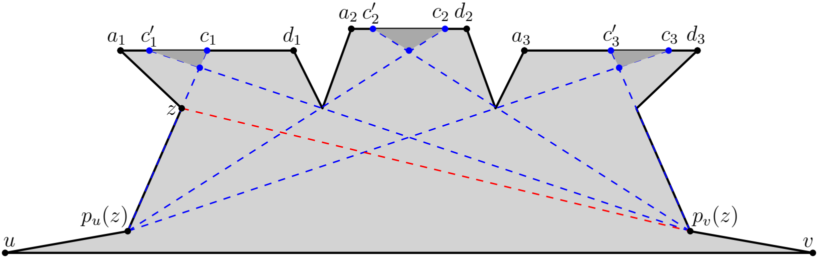


Figure: Multiple invisible regions exist within the polygon that are not visible from the guard set $S = \{p_u(z), p_v(z)\}$.

Lemma

It is possible to choose an additional set of guards S' to cover all invisible regions such that $|S'| \leq 2|S_{opt}|$.

Approximation Ratio of our Algorithm

Theorem

Our algorithm has an approximation ratio of 6.

Proof.

The final guard set returned by our algorithm is $|S \cup S'|$.

$$\begin{aligned} |S \cup S'| &= |S| + |S'| \\ &\leq 4|S_{opt}| + 2|S_{opt}| \\ &= 6|S_{opt}| \end{aligned}$$

□

Running Time of our Algorithm

Theorem

For a weak visibility polygon P having n vertices, the running time of our algorithm is $\mathcal{O}(n^2)$.

Running Time of our Algorithm

Theorem

For a weak visibility polygon P having n vertices, the running time of our algorithm is $\mathcal{O}(n^2)$.

Proof.

- Computation of $SPT(u)$ and $SPT(v)$ takes $\mathcal{O}(n)$ time.
- Computation of guard set S takes $\mathcal{O}(n^2)$ time.
- Computation of guard set S' also takes $\mathcal{O}(n^2)$ time.
- Hence, the overall running time of our algorithm is $\mathcal{O}(n^2)$.

□

Improvement for Orthogonal Weakly Visible Polygons

Lemma

For orthogonal simple polygons weakly visible from an edge, $|S| \leq 2|S_{opt}|$.

Improvement for Orthogonal Weakly Visible Polygons

Lemma

For orthogonal simple polygons weakly visible from an edge, $|S| \leq 2|S_{opt}|$.

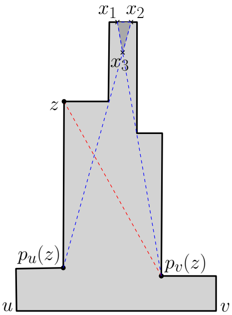


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1x_2x_3$ are invisible.

Improvement for Orthogonal Weakly Visible Polygons

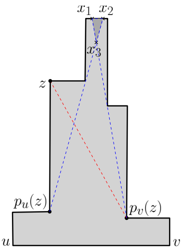


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1 x_2 x_3$ are invisible.

Improvement for Orthogonal Weakly Visible Polygons

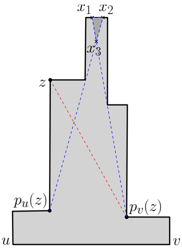


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1x_2x_3$ are invisible.

NOTE: One of the sides x_1x_2 of the triangle $x_1x_2x_3$ is a part of the polygonal edge a_1a_2. In fact, for any such invisible region, one of the sides must always be part of a polygonal edge.

Improvement for Orthogonal Weakly Visible Polygons

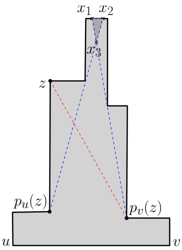


Figure: All vertices are visible from the guard set $S = \{p_u(z), p_v(z)\}$, but all points in the triangular interior region $x_1x_2x_3$ are invisible.

NOTE: One of the sides x_1x_2 of the triangle $x_1x_2x_3$ is a part of the polygonal edge a_1a_2. In fact, for any such invisible region, one of the sides must always be part of a polygonal edge.

Lemma

It is possible to choose an additional set of guards S' to cover all invisible regions such that $|S'| \leq |S_{opt}|$.

Improvement for Orthogonal Weakly Visible Polygons

Lemma

It is possible to choose an additional set of guards S' to cover all invisible regions such that $|S'| \leq |S_{opt}|$.

Improvement for Orthogonal Weakly Visible Polygons

Lemma

It is possible to choose an additional set of guards S' to cover all invisible regions such that $|S'| \leq |S_{opt}|$.

Theorem

For orthogonal simple polygons weakly visible from an edge, our algorithm has an improved approximation ratio of 3.

Proof.

$|S \cup S'| \leq |S| + |S'| \leq 2|S_{opt}| + |S_{opt}| \leq 3|S_{opt}|$ □

A Known Inapproximability Result

Theorem (Eidenbenz, Stamm and Widmayer (1998))

For polygons with holes, there cannot exist a polynomial time algorithm for AGP with an approximation ratio better than $((1-\epsilon)/12)\ln n$ *for any* $\epsilon > 0$*, unless NP* $\subseteq$ *TIME(*$n^{\mathcal{O}(\log\log n)}$*).*

A Known Inapproximability Result

Theorem (Eidenbenz, Stamm and Widmayer (1998))

For polygons with holes, there cannot exist a polynomial time algorithm for AGP with an approximation ratio better than $((1-\epsilon)/12)\ln n$ *for any* $\epsilon > 0$*, unless* $NP \subseteq TIME(n^{\mathcal{O}(\log\log n)})$*.*

The above theorem utilizes the following result by Feige -

Theorem (Feige (1998))

Set Cover cannot be approximated to within a factor of $(1-\epsilon)\ln n$ *for every* $\epsilon > 0$ *unless* $NP \subseteq TIME(n^{\mathcal{O}(\log\log n)})$*.*

Our Inapproximability Result

A modification of their reduction leads us to the following result -

Theorem

For weak visibility polygons with holes, there cannot exist a polynomial time algorithm for the vertex guarding problem with an approximation ratio better than $((1-\epsilon)/12)\ln n$ *for any* $\epsilon > 0$*, unless* $NP \subseteq TIME(n^{\mathcal{O}(\log\log n)})$.

Our Inapproximability Result

A very recent result by Dinur and Steurer -

Theorem (Dinur and Steurer (2014))

Set Cover cannot be approximated to within a factor of $(1 - \epsilon) \ln n$ *for every* $\epsilon > 0$ *unless* $NP = P$.

Our Inapproximability Result

A very recent result by Dinur and Steurer -

Theorem (Dinur and Steurer (2014))

Set Cover cannot be approximated to within a factor of $(1-\epsilon)\ln n$ *for every* $\epsilon > 0$ *unless* $NP = P$.

With this strengthening of Feige's quasi-NP-hardness, our inapproximability result gets improved to -

Theorem

For weak visibility polygons with holes, there cannot exist a polynomial time algorithm for the vertex guarding problem with an approximation ratio better than $((1-\epsilon)/12)\ln n$ *for any* $\epsilon > 0$, *unless* $NP = P$.

Hardness for Point Guards in Weakly Visible Polygons

Definition (Minimum Line Cover Problem (MLCP))

Let $\mathcal{L} = \{l_1, \ldots, l_n\}$ be a set of n lines in the plane. Find a set P of points, such that for each line $l \in \mathcal{L}$ there is a point in P that lies on l, and P is as small as possible.

Hardness for Point Guards in Weakly Visible Polygons

Definition (Minimum Line Cover Problem (MLCP))

Let $\mathcal{L} = \{l_1, \ldots, l_n\}$ be a set of n lines in the plane. Find a set P of points, such that for each line $l \in \mathcal{L}$ there is a point in P that lies on l, and P is as small as possible.

Definition (Decision Version of Line Cover Problem (DLCP))

Given $\mathcal{L}$ and an integer $k > 0$, decide whether there exists a line cover of size k.

Hardness for Point Guards in Weakly Visible Polygons

Definition (Minimum Line Cover Problem (MLCP))

Let $\mathcal{L} = \{l_1, \ldots, l_n\}$ be a set of n lines in the plane. Find a set P of points, such that for each line $l \in \mathcal{L}$ there is a point in P that lies on l, and P is as small as possible.

Definition (Decision Version of Line Cover Problem (DLCP))

Given $\mathcal{L}$ and an integer $k > 0$, decide whether there exists a line cover of size k.

DLCP is known to be NP-hard.

Hardness for Point Guards in Weakly Visible Polygons

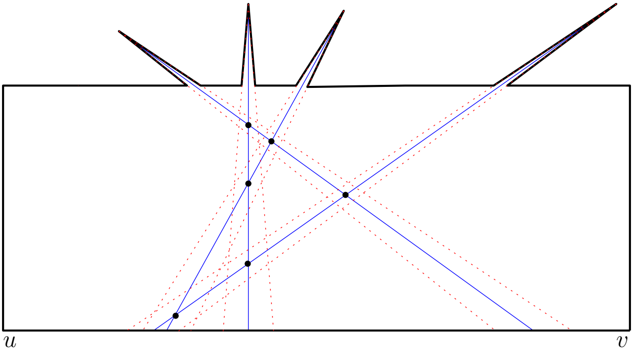


Figure: NP-hardness reduction from DLCP for point guarding polygons weakly visible from an edge

Hardness for Point Guards in Weakly Visible Polygons

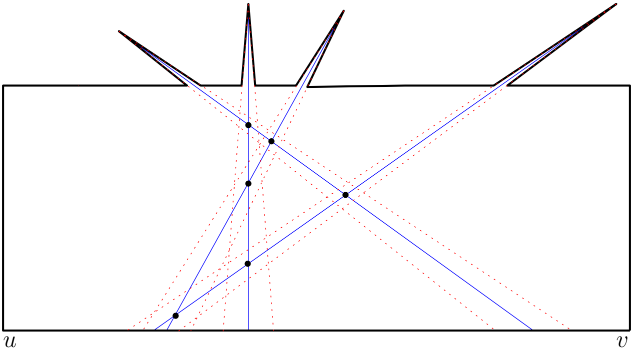


Figure: NP-hardness reduction from DLCP for point guarding polygons weakly visible from an edge

Theorem

The Point Guard problem is NP-hard for polygons weakly visible from an edge.

Roadmap for Future Work

- Designing a constant factor approximation algorithm for vertex guarding all simple polygons, with the intention of proving Ghosh's conjecture to be true even in the most general case.

Roadmap for Future Work

- Designing a constant factor approximation algorithm for vertex guarding all simple polygons, with the intention of proving Ghosh's conjecture to be true even in the most general case.
- Implement our approximation algorithms using the CGAL library in C++, and then perform extensive benchmark testing using our implementation. This should help us accumulate practical evidence regarding how closely the size of the guard sets computed by our algorithms approximates the size of the optimal guard set.

Roadmap for Future Work

- Investigate the problem of vertex guarding in a setting where the guards are allowed to see points within the polygon directly as well as via a single diffuse reflection along one of the edges, which act as mirrors.

Roadmap for Future Work

- Investigate the problem of vertex guarding in a setting where the guards are allowed to see points within the polygon directly as well as via a single diffuse reflection along one of the edges, which act as mirrors.
- Explore natural variations of the problem where certain restrictions are imposed on the guard sets themselves - for example, a guard set may be considered to be valid only when it is, say a hidden set, or perhaps a clique in the visibility graph of the polygon.

Roadmap for Future Work

- Investigate the problem of vertex guarding in a setting where the guards are allowed to see points within the polygon directly as well as via a single diffuse reflection along one of the edges, which act as mirrors.
- Explore natural variations of the problem where certain restrictions are imposed on the guard sets themselves - for example, a guard set may be considered to be valid only when it is, say a hidden set, or perhaps a clique in the visibility graph of the polygon.
- In all these parallel threads of exploration, our objective would be to come up with an approximation algorithm with a reasonable approximation ratio, and also to show the optimality of our algorithm by establishing corresponding inapproximability bounds.

Thank You!