# How many landmark colors are needed to avoid confusion in a polygon?

Lawrence H. Erickson and Steven M. LaValle
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801 USA
{lericks4, lavalle}@uiuc.edu

***Abstract*— Suppose that two members of a finite point guard set $S$ within a polygon $P$ must be given different colors if their visible regions overlap, and that every point in $P$ is visible from some point in $S$. The *chromatic art gallery problem*, introduced in [7], asks for the minimum number of colors required to color any guard set (not necessarily a minimal guard set) of $P$.**

**We study two related problems. First, given a polygon $P$ and a guard set $S$ of $P$, can the members of $S$ be efficiently and optimally colored so that no two members of $S$ that have overlapping visibility regions have the same color? Second, given a polygon $P$ and a set of candidate guard locations $N$, is it possible to efficiently and optimally choose the guard set $S \subseteq N$ that requires the minimum number of colors? We provide an algorithm that solves the first question in polynomial time, and demonstrate the NP-hardness of the second question. Both questions are motivated by common robot tasks such as mapping and surveillance.**

## I. Introduction and Problem Definition

Suppose a robot is navigating in a planar region containing a number of partially distinguishable landmarks. These landmarks could be radio towers (some of which may broadcast at the same frequency), colored signs (some of which may share the same color), or something else. The robot's motion primitives are defined with respect to these landmarks ("Move toward the red sign" or "Move away from the radio tower broadcasting at 102.3 MHz"), perhaps similar to the model studied in [22]. If the robot were operating in a region where two of the landmarks were indistinguishable, then the navigation primitives could become ambiguous ("Move toward the red sign" is confusing if the robot can see two red signs). Therefore, the question arises: How many classes of partially distinguishable landmarks are needed to cover a given bounded region? We assume that every point in the region must be visible from one of the landmarks, that two landmarks must be assigned separate classes if their visible regions overlap, and that the environment boundary blocks visibility.

Partially distinguishable landmarks have been heavily studied in robotics. They are obstacles in localization and mapping tasks, where many papers have been written about how to best disambiguate or work around them [3], [5]. A common issue in simultaneous localization and mapping (SLAM) is the data association problem, determining whether two sensor readings from a sensor suite with limited distinguishing ability correspond to the same physical object [10], [16], [17]. Partially distinguishable landmarks were also studied for the purpose of using the non-presence of landmarks for localization in [9].

There are many reasons why one would wish to minimize the number of landmark classes. There could be a fee for the use of each individual radio frequency, or laws limiting the number of frequencies on which a single entity has permission to broadcast. Discriminating amongst more landmark classes may also make the robotic sensors more complicated (it is easier to build a camera that can reliably distinguish among ten colors than it is to build a camera that can reliably distinguish among a thousand). It was noted in [12] that cameras that could recognize only a relatively low amount of colors were best for human iris scanning (using too many colors tended to falsely increase the difference between different pictures of the same person's iris). This is also related to [18], where a robot was assigned to track a target, but the robot's sensors were deliberately weakened in order to keep sensitive data about the target from becoming public if the robot's sensor data was accessed by a third party.

Let a *polygon* $P$ be a closed, simply connected, polygonal subset of $\mathbb{R}^2$ with boundary $\partial P$. A point $p \in P$ is *visible* from point $q \in P$ if the closed segment $\overline{pq}$ is a subset of $P$. The *visibility polygon* $Vis(p)$ of a point $p \in P$ is defined as $Vis(p) = \{q \in P \mid q \text{ is visible from } p\}$. Let a *guard set* $S$ be a finite set of points in $P$ such that $\bigcup_{s \in S} Vis(s) = P$. The members of a guard set are referred to as *guards*. A pair of guards $s, t \in S$ is called *conflicting* if $Vis(s) \cap Vis(t) \neq \emptyset$. Conflicting guards cannot have the same color (see Figure 1).

In [7], the goal was to find $\chi_G(P)$, the minimum number of colors that *any* guard set of $P$ required. In this paper, we will instead focus on two related problems. First, given a polygon $P$ and guard set $S$, can one efficiently find the minimum number of colors needed to color $S$? Section II provides an algorithm that finds the minimum number of colors required to color $S$ in polynomial time. Second, given a polygon $P$ and a finite set of candidate guard locations $N \subset P$, can one efficiently choose the guard set $S \subseteq N$ that minimizes the number of colors required? In Section III, this problem is shown to be NP-hard. Section IV will discuss directions of future research.

This paper is also closely related to research on art gallery coverage [2], [13], [15], [19], other minimal polygon cover questions [4], [20], and work on visibility graphs

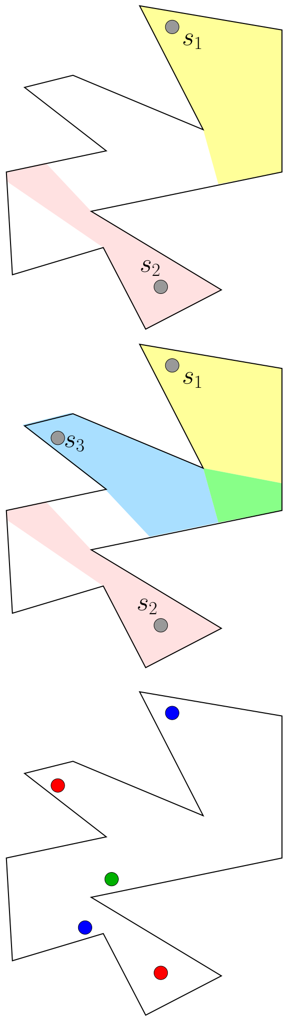


Fig. 1. [top] The visibility polygon of $s_1$ (the yellow region) does not overlap the visibility polygon of $s_2$. Therefore, $s_1$ and $s_2$ do not conflict, so they can be given the same color. [middle] The visibility polygon of $s_1$ (the yellow region) intersects the visibility polygon of $s_3$ (the blue region). The area of intersection is the green region. Since the two visibility polygons overlap, $s_1$ and $s_3$ conflict, and may not be assigned the same color. [bottom] A guard set for the polygon with a proper coloring.

[8]. The chromatic art gallery problems highlight an important issue in visibility work. Standard art gallery problems essentially ask for the minimum number of star-shaped polygons required to cover a given region. Minimum cover problems tend to be very difficult to solve, so many art gallery proofs convert the problem to a partition question (through triangulation, convex quadrilateralization, or some other decomposition), which tend to be much easier. The chromatic art gallery variations discussed in [7] and this paper are not easily converted into partition questions. In standard art gallery problems it is not important to control the exact number of guards that can see any particular point, one must simply ensure that each point is covered by at least one guard. Therefore, the cover problem can be approximated as a partition problem. In the chromatic variants, overlap between visibility regions of the guards cannot be ignored, so approximation as a partition problem is generally infeasible.

## II. Coloring a Guard Set

A given guard set in a polygon can be efficiently colored. To demonstrate this, we will show that the conflict relationships between guards can be represented by a certain family of graphs that are themselves easy to color.

Let $S$ be a guard set of $P$. Two guards $s, t \in S$ must be assigned different colors if $Vis(s) \cap Vis(t) \neq \emptyset$. The function $\chi_F(S, P)$, is defined as the minimum number of colors required to color $S$.

Given a polygon $P$ and a guard set $S$, let $G(S, P)$ be the *2-link visibility graph of $S$ in $P$*. The size of the vertex set $V$ of $G(S, P)$ is equal to $|S|$. Let $f : V \rightarrow S$ be a bijection between the members of $V$ and $S$. Let $s_i = f(v_i)$. The vertices $v_i, v_j \in V$ are adjacent if and only if $s_i$ and $s_j$ conflict. The chromatic number of $G(S, P)$ is equal to $\chi_F(S, P)$.

*Theorem 1:* Given a polygon $P$ and guard set $S$, the graph $G(S, P)$ can be constructed in $O(n|S|^2)$, where $n$ is the number of vertices of $P$.

*Proof:* Each visibility polygon $Vis(s_i)$ can be computed in $O(n)$ time [11], so generating all $|S|$ of them will take $O(n|S|)$ time. Each visibility polygon can have $O(n)$ vertices, and [1] showed that computering the intersection between two polygons with $O(n)$ vertices each takes at most $O(n)$ time (though the algorithm is complicated and difficult to implement). Therefore, pairwise testing of all the visibility polygons (to determine the edges of $G(S, P)$) will take $O(n|S|^2)$ time. Therefore, constructing $G(S, P)$ takes $O(n|S|) + O(n|S|^2) = O(n|S|^2)$ time. ∎

These 2-link visibility graphs are closely related to the $CN$-complexes in [14]. The $CN$-complex is a simplicial complex that represents a camera network. Each vertex in the complex is associated with a camera, and an $n$-simplex is drawn between $n + 1$ vertices if there is a region where all $n + 1$ of the visible ranges overlap. This is somewhat different from the 2-link visibility graph in that the 2-link visibility graph is only concerned with pairwise overlap of visibility regions. Additionally, there are situations in the $CN$-complex where a single camera is represented by multiple vertices. The $CN$-complexes were used for topologically based coordinate-free tracking and navigation.

A graph is a *chordal graph* if every cycle of length $4$ or greater contains a chord. It was shown in [21] that a chordal graph can be optimally colored in $O(v + e)$ time, where $v$ is the number of vertices and $e$ is the number of edges (recall that in this case $v = |S|$). Since the number of edges can be at most $O(v^2)$, a chordal graph can be colored in $O(|S|^2)$ time.

*Theorem 2:* For any polygon $P$ and any guard set $S$, $G(S, P)$ is a chordal graph. [1]

*Proof:* Let $[v_1, v_2, \ldots, v_f]$ be a chordless cycle of length $4$ or greater in $G(S, P)$. In that case, there must be a Eucliden shortest distance closed walk $W \subseteq P$ such that

[1]This theorem is erroneous. A counterexample was provided by Subhash Suri, Luca Foschini, and Andreas Baertschi. In fact, 2-link visibility graphs are not necessarily perfect graphs.

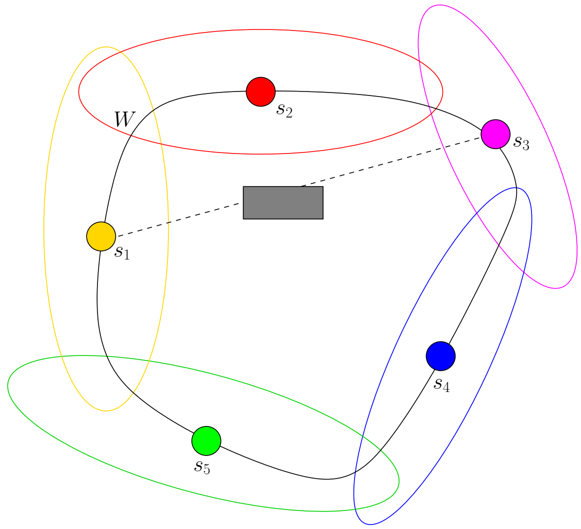


Fig. 2. The visibility regions of five guards are arranged in such a way that their 2-link visibility graph contains a chordless cycle of length 5. Since $\overline{s_1 s_3}$ must pass through an obstacle, the region cannot be simply connected.

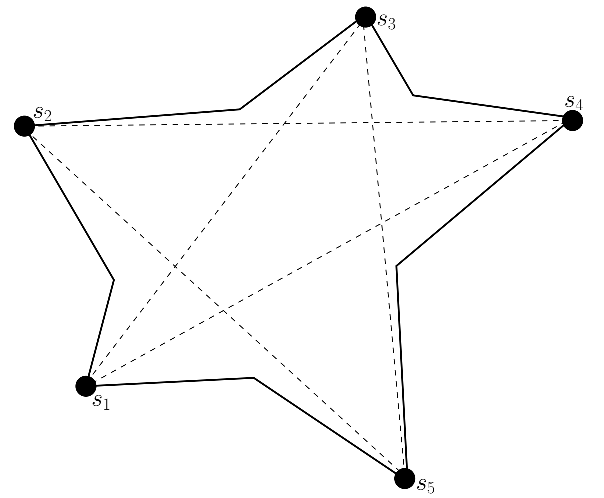


Fig. 3. A polygon adapted from [8]. Five landmarks $s_1$,$s_2$,$s_3$,$s_4$,$s_5$ form a chordless cycle in a standard, 1-link visibility graph. In the 2-link visibility graph, the five landmarks' corresponding vertices would form a clique.

$s_1, s_2, \ldots, s_f \in W$. Because $v_1$ does not conflict with $v_3$ (because $G(S,P)$ is chordless), $\overline{s_1 s_3}$ must contain a point $p$ that is not contained in $P$ (see Figure 2). Since $p \notin P$, the path along $W$ from $s_1$ to $s_3$ that passes through $s_2$ must not be homotopic to the path along $W$ from $s_1$ to $s_3$ that passes through $s_f$. However, this violates the assumption that $P$ is simply connected. Therefore, $G(S,P)$ cannot contain a chordless cycle of length $4$ or greater, which means that $G(S,P)$ is chordal.

■

Note that this contrasts with the result for standard "1-link" visibility graphs (where two vertices would have an edge between them if their corresponding landmarks were mutually visible). It was shown in [8] that, if the landmarks are simply the polygon vertices, there exist simple polygons that cause a non-chordal visibility graph to arise. This is because it is possible to have five landmarks $s_1, s_2, s_3, s_4, s_5$ in a polygon such that $s_i$ is mutually visible only with $s_{i+2}$ and $s_{i+3}$ (indicies are modulus 5, see Figure 3), which would create a chordless length $5$ cycle. These situations do not cause chordless 2-link visibilty graphs because the creation of these chordless cycles requires the line segment between a pair of mutually visible landmarks $\overline{s_i s_j}$ to intersect a line segment between two other mutually visible landmarks $\overline{s_k s_l}$ without creating a graph edge between $v_i$ or $v_j$ and $v_k$ or $v_l$. In the 2-link visibility graph, $v_i$,$v_j$,$v_k$, and $v_l$ would form a clique. This means that 2-link visibility graphs are likely much more easy to analyze than standard visibility graphs, which have so far resisted most efforts to place them into a well-known graph family.

It takes $O(n|S|^2)$ time to generate the graph, and $O(|S|^2)$ time to optimally color it. Therefore, computing $\chi_F(S,P)$ takes $O(n|S|^2)$ time, where $n$ is the number of vertices in $P$.

## III. Choosing a Guard Set from a Set of Candidates

Let $N$ be a finite set of points in a polygon $P$, such that there exists at least one guard set $S \subseteq N$. Let $G(N) = \{S \subseteq N \mid S \text{ is a guard set}\}$. Let $\chi_C(N,P) = \min_{S \in G(N)} \chi_F(S,P)$. In other words, $\chi_C(N,P)$ is the minimum number of colors that any guard set $S \subseteq N$ requires (see Figure 4). Let $CHROME$ be the decision form of this problem (specifically, "Is $\chi_F(N,P)$ for a given polygon $P$ and a given set of candidate guard locations less than $k \in \mathbb{N}$?").

Let a *vertex guard set* of a polygon $P$ be a finite set of points $S \subset P$ such that for each $s \in S$, $s$ is a vertex of the polygon $P$. Let $VG(P)$ be the set of all vertex guard sets of $P$. Call $Guard(P) = \min_{S \in VG(P)} |S|$ the *minimum art gallery guard number* of a polygon $P$. The *minimum art gallery problem* ($MINART$) asks, for a given polygon $P$, whether $Guard(P) \leq k$ for some integer $k$. This problem was shown to be NP-hard in [13].

We will construct a polygon $P$ and candidate guard list $N$ that encodes a $3SAT$ problem and show that the $3SAT$ expression is satisfiable if and only if $\chi_F(S,P)$ is low enough. The polygon $P$ will be the same polygon used in [13] to demonstrate the NP-hardness of the minimum art gallery guard problem (illustrations are adapted from [19]). This polygon consists of a number of different junctions that encode different parts of the $3SAT$ problem. The first type of junction is a *clause junction*, shown in Figure 5. This type of junction consists of a triangle with three thin notches on its right side. Each notch can be guarded by one of two candidate guards, one of which represents a literal evaluating to the "true" value, and one of which represents a literal evaluating to the "false" value. At least one of the "true" locations must be chosen as a guard location so that the bottom right corner of the triangle is covered.

The second type of junction is a variable junction (see Figure 6). A variable junction consists of two wells, each with a number of spikes in it (the blue regions in Figure 6).

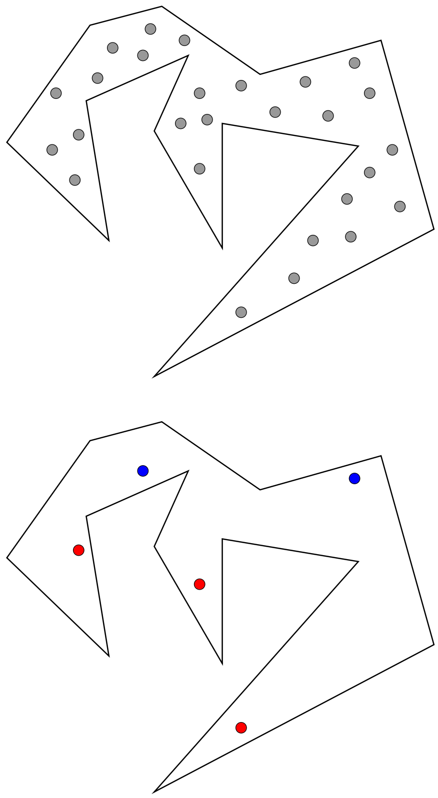

Fig. 4. [top] A polygon $P$ with several candidate guard locations. [bottom] A subset of the candidate guard locations that forms a guard set and requires only two colors. This is the minimum number of colors that any subset of the candidate guard locations requires.

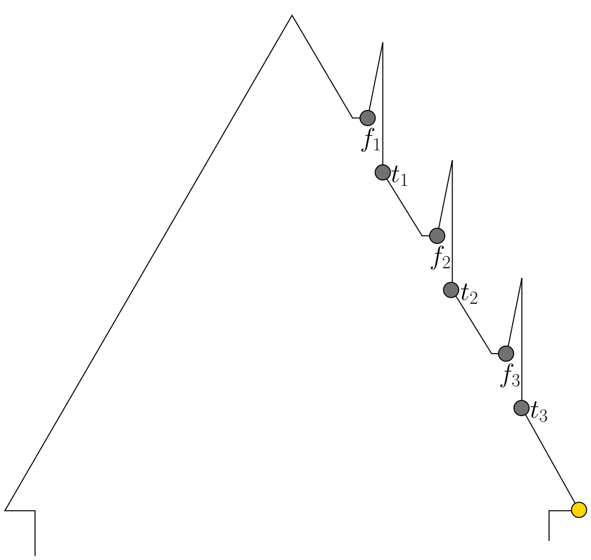


Fig. 5. A clause junction. Each notch represents a literal or a negated literal in the original $3SAT$ problem. Each of these junctions has three $t$ locations, and three $f$ locations. At least one guard must be placed at one of the "true" ($t$) locations, or the yellow vertex on the bottom right will not be guarded.

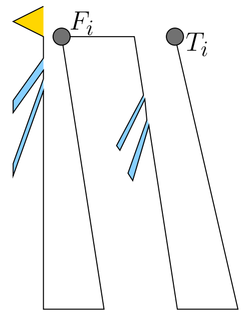


Fig. 6. A variable junction. A guard placed at $F_i$ can guard the leftmost blue spikes, and a guard placed at $T_i$ can guard the blue spikes on the right. A guard must be placed at either $F_i$ or $T_i$ to guard the yellow region.

At least one guard must be placed at the $T_i$ point or the $F_i$ point to guard a notch on the left side of the junction.

These junctions are shown arranged into a complete polygon for the $3SAT$ expression $(\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3})$ in Figure 7. The whole polygon consists of a large rectangular region with clause junctions attached to the upper right portions of the polygon and variable junctions attached to the lower left portions of the polygon. The rightmost opening of a variable junction must be to the left of the leftmost opening into a clause junction. A single clause junction can be completely guarded with three guards, one of which must be at a $t$ vertex. A guard at point $q$ can guard the entire rectangular region and the portions of the variable junction that are not spikes or left-side notches. The remaining portions of a variable junction $x_i$ can be guarded with a single guard at point $T_i$ if each $x_i$ literal in a clause has a guard at its $t$ point, and each $\overline{x_i}$ literal in a clause has a guard at its $f$ point. The remaining portions of a variable junction $x_i$ can be guarded with a single guard at point $F_i$ if each $x_i$ literal in a clause has a guard at its $f$ point, and each $\overline{x_i}$ literal in a clause has a guard at its $t$ point.

It was shown in [13] that these types of polygons require at least three guards for each of $m$ clauses, at least one for each of $n$ variables, and a guard at the $q$ point, and that guarding the polygon with this minimum number of $3m+n+1$ guards was only possible when the underlying $3SAT$ equation was satisfiable.

*Theorem 3:* The problem $CHROME$ is NP-hard.

*Proof:* Let $P$ be a polygon of the kind used in [13] that encodes a $3SAT$ problem. It was shown that a minimal art-gallery guard placement for these kinds of polygons need only place guards at the $T$, $F$, $t$, $f$, and $q$ points shown in Figure 7. Let those points be our set of candidate guard locations for $CHROME$. Let $y_c$ be the lowest $y$-value such that for every point $p \in P$ where $p$ has a $y$-value of $y_c$, $p$ is visible from all candidate guard locations in the variable clauses (the green line in Figure 7). Note that, since the value of $y_c$ is determined by the size of the left-side notches in the variable clauses (the yellow region in Figure 6), $y_c$ can be arbitrarily close to the bottom of the rectangular region of the polygon (as the notches can be arbitrarily small). Since the line at $y_c$ is visible from each candidate guard location in a variable junction, each guard placed in a variable junction will conflict with each other guard in a variable junction. Note that each candidate guard location in a clause junction must be visible from a spike in a variable junction (see the diagonal blue lines in Figure 7). Since the spikes lie below $y_c$, each candidate guard location in a clause junction conflicts with each candidate guard location in a variable junction (note that the diagonal blue lines in Figure 7 intersect the green line). Also, because the opening to the rightmost variable junction lies to the left of the opening of the leftmost clause junction, each candidate guard location in a clause junction must conflict with all clause candidate locations to its left, so each candidate guard location in a clause junction conflicts with every other candidate guard location in a clause (note that for every candidate guard location $p$ in a clause junction in Figure 7, the diagonal blue line coming from $p$ intersects with all the blue lines to the left of $p$). Finally, the point $q$ can see the bottom right edge of the rectangular region of the polygon, meaning that it conflicts with every other candidate guard location (note that the red line in Figure 7 intersects with all the blue and green lines).

Therefore, for this polygon and candidate guard set, $\chi_C(N, P) = Guard(P)$. Since the underlying $3SAT$ expression is satisfiable if and only if $Guard(P) \leq 3m+n+1$, the underlying $3SAT$ expression is also satisfiable if and only if $\chi_C(N, P) \leq 3m + n + 1$. Since $3SAT$ is NP-complete, determining whether $\chi_C(N, P) \leq k$ for some integer $k$ is NP-hard.

■

## IV. Conclusion

We have presented a method for efficiently coloring a guard set of a polygon such that no two guards with overlapping visibility polygons have the same color, and we have shown that the graphs representing the conflict relationships between guards are always chordal. For robotics purposes, this means one could place landmarks randomly around an environment, and then quickly and optimally color them so that the robot can never see two of the same kind. For example, if the robot is using radio signal intensity from a set of pre-placed towers to navigate (in a somewhat idealized setting), one could quickly and optimally assign non-conflicting frequencies to each tower.

We have also shown that the similar problem of choosing a set of guards that requires a minimal amount of colors from a finite candidate list is NP-hard. Even though choosing the optimal subset of candidate guards to minimize the number of required colors is NP-hard, it may be possible to develop a useful approximation algorithm, though related results about the approximability of the minimum art gallery guard problem [6] may indicate limits on the usefulness of approximation algorithms for this problem.

Since a set of guards, once placed, can be efficiently colored, a probabilistic algorithm for placing guards may be feasible for attacking the problem put forth in [7], perhaps by decomposing the polygon into convex subregions and randomly choosing which subregion to place a guard into based on the incidence relationships between the regions. Also, since the 2-link visibility graphs are chordal, two other questions naturally arise. First, are all connected chordal graphs 2-link visibility graphs for some guard set of a polygon $P$? If not, is there a smaller family of graphs that the 2-link visibility graphs belong to?


## Acknowledgements

The authors acknowledge the support of National Science Foundation grants #0904501 (IIS Robotics) and #1035345 (Cyberphysical Systems), DARPA SToMP grant HR0011-05-1-0008, and MURI/ONR grant N00014-09-1-1052.


## References


[1] B. Chazelle, “Triangulating a simple polygon in linear time,” *Discrete and Computational Geometry*, vol. 6, no. 5, pp. 485–524, 1991.

[2] V. Chvatal, “A combinatorial theorem in plane geometry,” *Journal of Combinatorial Theory Series B*, vol. 18, no. 1, pp. 39–41, 1975.

[3] I. J. Cox and J. J. Leonard, “Modeling a dynamic environment using a bayesian multiple hypothesis approach,” *Artificial Intelligence*, vol. 66, no. 2, pp. 311–344, 1994.

[4] J. C. Culberson and R. A. Reckhow, “Covering polygons is hard,” in *Proc. IEEE Symposium on Foundations of Computer Science*, 1988, pp. 601–611.

[5] M. DiMarco, A. Garulli, A. Giannitrapani, and A. Vicino, “A set theoretic approach to dynamic robot localization and mapping,” *Autonomous Robots*, vol. 16, no. 1, pp. 23–47, 2004.

[6] S. Eidenbenz, C. Stamm, and P. Widmayer, “Inapproximability results for guarding polygons and terrains,” *Algorithmica*, vol. 31, no. 1, pp. 79–113, 2001.

[7] L. H. Erickson and S. M. LaValle, “A chromatic art gallery problem,” University of Illinois at Urbana-Champaign, Urbana, IL, Tech. Rep., 2010.

[8] S. K. Ghosh, “On recognizing and characterizing visibility graphs of simple polygons,” *Discrete and Computational Geometry*, vol. 17, no. 2, pp. 143–162, 1997.

[9] T. Hester and P. Stone, “Negative information and line observations for monte carlo localization,” in *Proc. IEEE International Conference on Robotics and Automation*, 2008, pp. 2764–2769.

[10] D. Hhnel, S. Thrun, B. Wegbreit, and W. Burgard, “Towards lazy data association in SLAM,” in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, P. Dario and R. Chatila, Eds. Springer Berlin / Heidelberg, 2005, vol. 15, pp. 421–431.

[11] B. Joe and R. B. Simpson, “Corrections to Lee’s visibility polygon algorithm,” *BIT Numerical Mathematics*, vol. 27, no. 4, pp. 458–473, 1987.

[12] E. Krichen, M. Chenafa, S. Garcia-Salicetti, and B. Dorizzi, “Color-based iris verification,” in *Advances in Biometrics*, ser. Lecture Notes in Computer Science, S.-W. Lee and S. Li, Eds. Springer Berlin / Heidelberg, 2007, vol. 4642, pp. 997–1005.

[13] D. T. Lee and A. K. Lin, “Computational complexity of art gallery problems,” *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 276–282, 1986.

[14] E. J. Lobaton, P. Ahammad, and S. Sastry, “Algebraic approach to recovering topological information in distributed camera networks,” in *International Conference on Information Processing in Sensor Networks*, 2009, pp. 193–204.

[15] A. Lubiw, “Decomposing polygonal regions into convex quadrilaterals,” in *Proc. Symposium on Computational Geometry*, 1985, pp. 97–106.

[16] M. Montemerlo and S. Thrun, “Simultaneous localization and mapping with unknown data association using FastSLAM,” in *Proc. IEEE International Conference on Robotics and Automation*, 2003, pp. 1985 – 1991.

[17] A. Nuchter, K. Lingemann, J. Hertzberg, and H. Surmann, “6D SLAM with approximate data association,” in *International Conference on Advanced Robotics*, 2005, pp. 242–249.

[18] J. M. O’Kane, “On the value of ignorance: Balancing tracking and privacy using a two-bit sensor,” in *Proc. Workshop on the Algorithmic Foundations of Robotics*, 2009, pp. 235–249.

[19] J. O’Rourke, *Art Gallery Theorems and Algorithms*. Cambridge, UK: Oxford University Press, 1987.

[20] J. O'Rourke and K. Supowit, "Some np-hard polygon decomposition problems," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 181–190, 1983.
[21] R. E. Tarjan and M. Yannakakis, "Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs," *SIAM Journal on Computing*, vol. 13, no. 3, pp. 566–579, 1984.
[22] K. Taylor and S. M. LaValle, "I-bug: An intensity-based bug algorithm," in *Proc. IEEE International Conference on Robotics and Automation*, 2009, pp. 3466–3471.

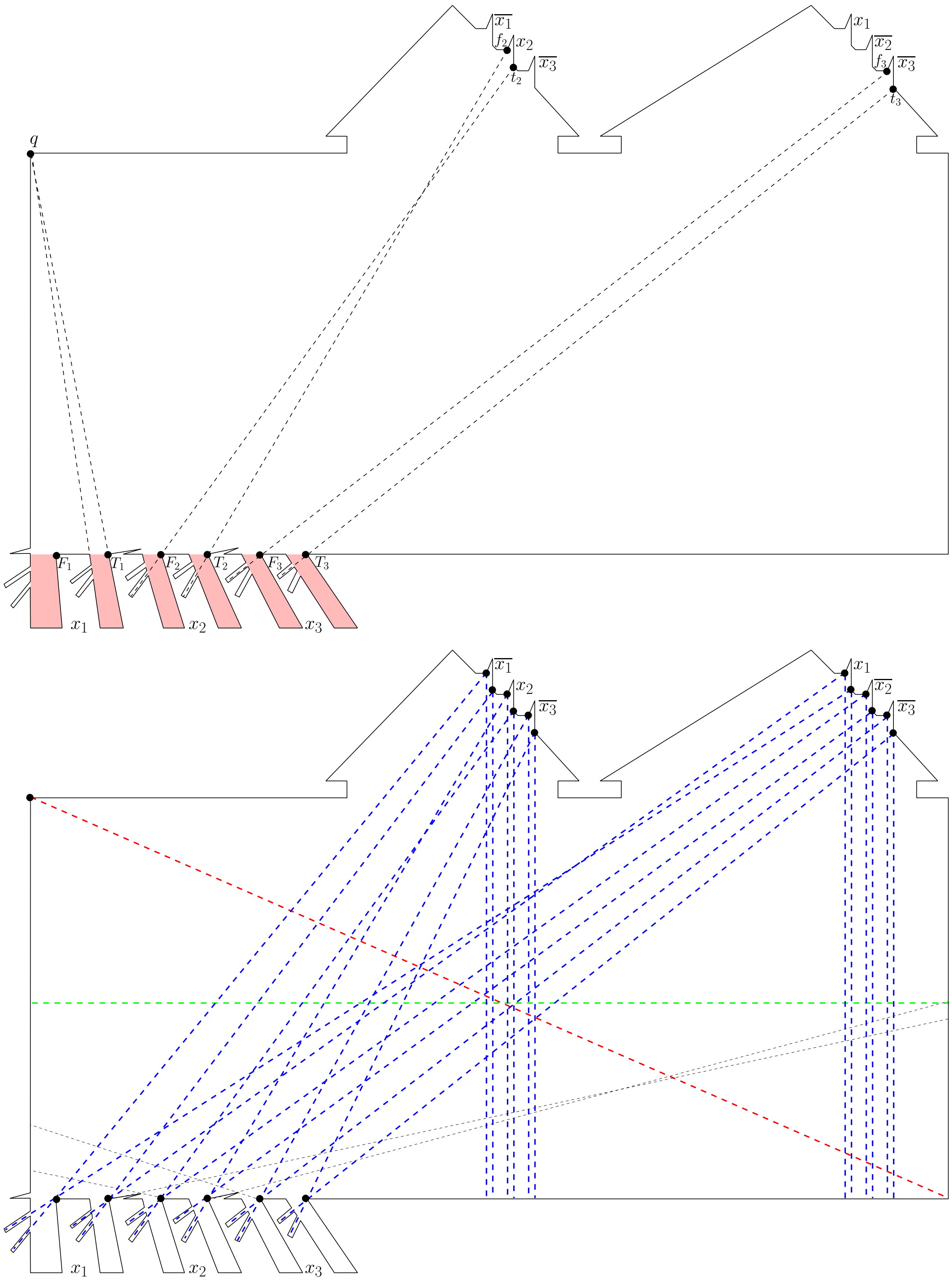


Fig. 7. [top] The whole polygon for the $3SAT$ expression $(\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3})$. If a literal exists in a clause in non-negated form (see $x_2$ in the first clause), then a spike is made that is collinear with the $f$ vertex of that literal in the clause region and the $T$ vertex in the corresponding variable region. A second spike is made that is collinear with the $t$ vertex of the literal in the clause region and the $F$ vertex in the corresponding variable region. If a literal exists in a clause in negated form (see $\overline{x_3}$ in the second clause), then a spike is made that is collinear with the $f$ vertex of the literal in the clause region, and the $F$ vertex in the corresponding variable region. A second spike is made that is collinear with the $t$ vertex of the literal in the clause region, and the $T$ vertex in the corresponding variable region. A guard placed at point $q$ can see all the pink regions. [bottom] The polygon for the $3SAT$ expression $(\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3})$ with candidate guard locations. Each guard placed on one of the candidate locations will conflict with every other guard placed on a candidate location.