J. geom. 72 (2001) 188 – 198
0047–2468/01/020188 – 11 \$ 1.50 + 0.20/0

Journal of Geometry

A generalized fortress problem using k-consecutive vertex guards

S. M. Yiu

Abstract. The *fortress problem* was posed independently by Joseph Malkelvitch and Derick Wood to determine the number of guards sufficient to cover the exterior of an n-vertex polygon. O'Rourke and Wood showed that $\lceil n/2 \rceil$ vertex guards are sometimes necessary and always sufficient. Yiu and Choi considered a variation of the problem by allowing each guard to patrol an edge (called an *edge guard*) of the polygon and obtained a tight bound of $\lceil n/3 \rceil$ edge guards for general polygons. In this paper, we unify and generalize both results by considering the number of *k-consecutive vertex guards* that are required to solve the fortress problem. A tight bound of $\lceil n/(k+1) \rceil$ is obtained.

Mathematics Subject Classification (2000): 68U05.
Key words: Computational Geometry, Fortress Problem, Consecutive Vertex Guard.

1. Introduction

Given a simple polygon, a point x, exterior or interior to the polygon, is said to be visible from (or *covered by*) a point y if the line segment joining them does not intersect the boundary of the polygon. The definition is extended to the visibility of a point from an edge. A point x is visible from an edge uv if there exists a point y on uv such that x is visible from y.

The *art gallery problem* asks how many guards are sometimes necessary and always sufficient to cover the interior of an n-vertex simple polygon. The problem was solved by Chvátal and Fisk [3]. Among the different variations of the problem, the *fortress problem* requires the guards to cover the exterior instead of the interior of the polygon. For an excellent description of these problems, refer to [3, 4]. O'Rourke and Wood [3] solved the fortress problem for vertex guards. Yiu and Choi [6] solved the corresponding problem for *edge guards*. In this paper, the power of *k-consecutive vertex guard* in the fortress problem is investigated.

A *k-consecutive vertex guard* is a set of vertex guards located at k consecutive vertices on the boundary of the polygon while a *k-consecutive edge guard* is a mobile guard which is allowed to patrol k consecutive edges. This paper shows that $\lceil n/(k+1) \rceil$ k-consecutive vertex guards are sometimes necessary and always sufficient to cover the exterior of any n-vertex simple polygons for any fixed $k < n$. In [6], it was shown that the power of an edge guard is equivalent to that of a 2-consecutive vertex guard in the worst case for general simple polygons with respect to the fortress problem. In this paper, a different proof is used to further generalize the result by showing that the power of allowing each guard to patrol k consecutive edges is equivalent to that of placing guards at $(k + 1)$-consecutive vertices. The problem of finding the minimum value of k such that a single k-consecutive guard can cover the interior of the polygon is solved in [1] for both vertex and edge guards. Other related problems are found in [2, 5].

A simple n-sided convex polygon requires $\lceil n/(k+1) \rceil$ k-consecutive vertex guards to cover its exterior for any fixed $k < n$. If $k > n/2$, two guards may overlap on some of the vertices, but it will not affect the resulting number of guards. Thus we have

LEMMA 1. $\lceil n/(k+1) \rceil$ *k-consecutive vertex guards are sometimes necessary to cover the exterior of a simple n-sided polygon.*

The same example can be used to establish

LEMMA 2. $\lceil n/(k+1) \rceil$ $(k-1)$*-consecutive edge guards are sometimes necessary to cover the exterior of a simple n-sided polygon.*

The sufficiency proof for k-consecutive vertex guards will be presented in the following section. Since the sufficiency for k-consecutive vertex guards implies also the sufficiency for $(k - 1)$-consecutive edge guards, so the power of k-consecutive vertex guards is shown to be equivalent to that of $(k - 1)$-consecutive edge guards in this context.

2. Sufficiency proof

From now on, a k-consecutive *vertex* guard is simply referred to as a guard. A vertex of the polygon is called a *guarded vertex* if one of the guards is assigned to it. That is, at least one of the vertex guards in any of k-consecutive vertex guards is located on that vertex. Other vertices are called *unguarded*. In the worst case, the number of guarded vertices is at least as many as that of unguarded vertices. One might wonder whether by leaving only every $(k+1)$th vertex unguarded, the exterior will always be covered. There will be at most n different guard placements depending on which vertex one starts with according to the above method. However, it can be shown that this simple strategy does not work, but it can be used as the basis for the sufficiency proof. Before presenting the sufficiency proof, it is useful to see why this simple method fails.

Figures 1 and 2, obtained by modifying Figure 6.2 of [1], show that this simple strategy does not work for $k = 2$ or 3. In Figure 1, $k = 2$, and placing guards on vertices labeled 1 and 2, 2 and 3, or 3 and 1 will leave one of the regions labelled Q unguarded. A similar situation occurs in Figure 2, where $k = 3$. The figures can be easily generalized for any value of k.

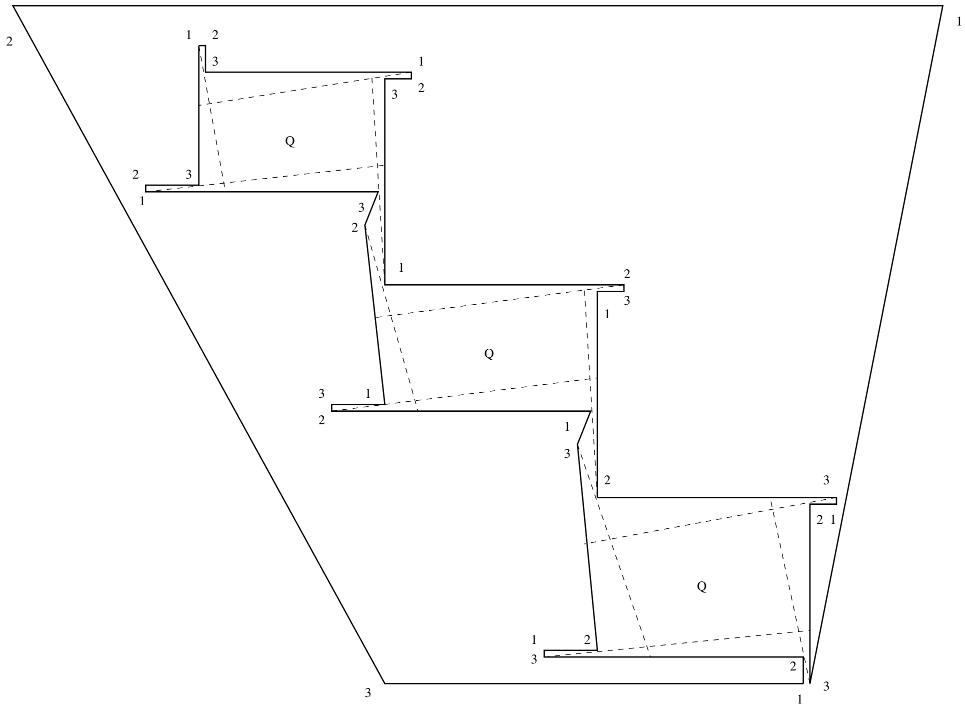

Figure 1 Simple guard placement strategy does not work for $k = 2$

The reason that this simple method does not work is the following. Given a polygon, each connected region inside its convex hull but exterior to the polygon is called a *pocket*. A *triangulation graph* of a pocket is a graph whose embedding is a triangulation of the pocket. Using the above simple strategy, there may exist some triangular faces (*unguarded triangles*) whose vertices are all unguarded. Such triangles are not guaranteed to be covered by the guards. On the other hand, if all faces of the triangulation are guarded, the pocket will be guarded. For example, in Figure 3, where $k = 2$, a part of each of the two unguarded triangles is not covered. The following will show how the positions of the guards can always be shifted in such a way that all unguarded triangles become guarded, thus gives a sufficiency proof.

Let the vertices of the polygon be indexed as $v_1, v_2, \ldots, v_n$ in a counterclockwise order. That is, if we walk along the boundary, the interior of the polygon is always on the left. Consider the convex hull of the polygon, the regions to be covered are the one exterior to the hull and those pockets. Starting from any vertex, apply the simple strategy of leaving every $(k + 1)$th vertex unguarded as described above. If no two consecutive hull vertices

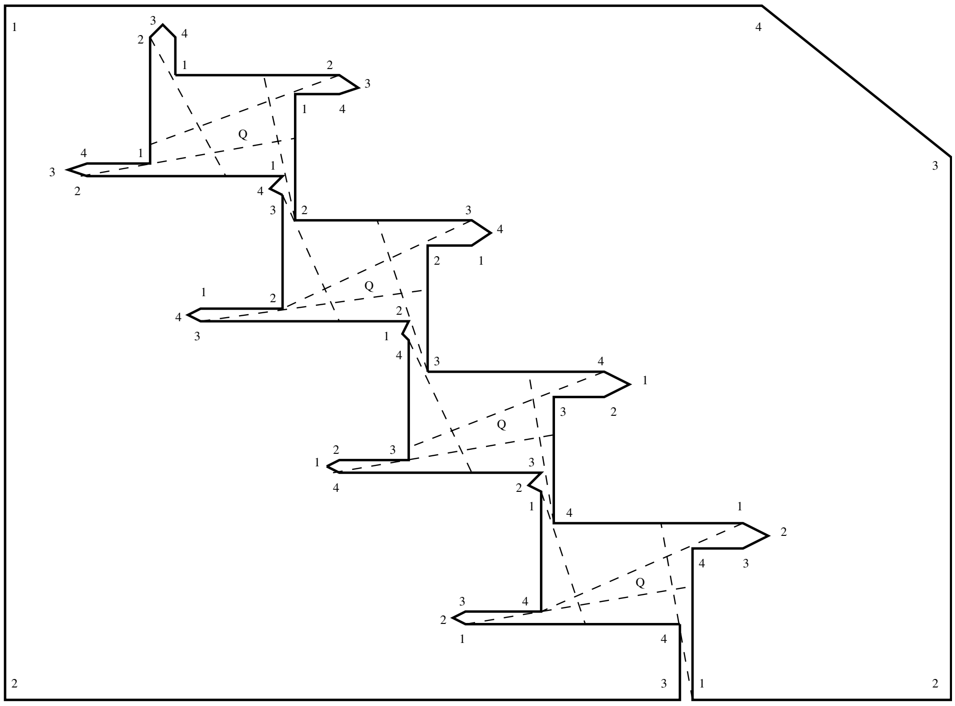

Figure 2 Simple guard placement strategy does not work for $k = 3$

are unguarded, the region exterior to the hull will be covered, otherwise, adjustment can be made to this initial placement of guards to ensure that every other hull vertex is guarded (see Lemma 5). If all pockets have fewer than $(2k+3)$ vertices including two hull vertices, this initial placement of guards suffices to cover the exterior of the whole polygon, so we assume there exists at least one pocket of more than $(2k+2)$ vertices. Triangulate these pockets. In fact, the problem mentioned in the above will only occur inside the pockets. The idea of the proof is to shift some of the guards by *at most* one vertex in a counterclockwise direction in order to cover all those unguarded triangles.

A guard located at vertices $v_i, v_{i+1}, \ldots, v_{i+k-1}$ can be defined by two vertices (v_i, v_{i+k-1}). Moving it one step to the *left* is the same as shifting it from (v_i, v_{i+k-1}) to (v_{i+1}, v_{i+k}). A guard at (v_i, v_{i+k-1}) is said to be on the right of vertex v_{i+k}. Every unguarded triangle is defined by three vertices. For two unguarded triangles, (v_l, v_m, v_n) and (v_p, v_q, v_r) where $l < m < n$ and $p < q < r$, there are only three possible cases if $n \geq r$: (1) $p \geq m$ (Figure 4a), (2) $m \geq r$ and $p \geq l$ (Figure 4b), and (3) $l \geq r$ (Figure 4c). In other words, two unguarded triangles will be completely *disjoint* as in case (3) or one is completely *enclosed*

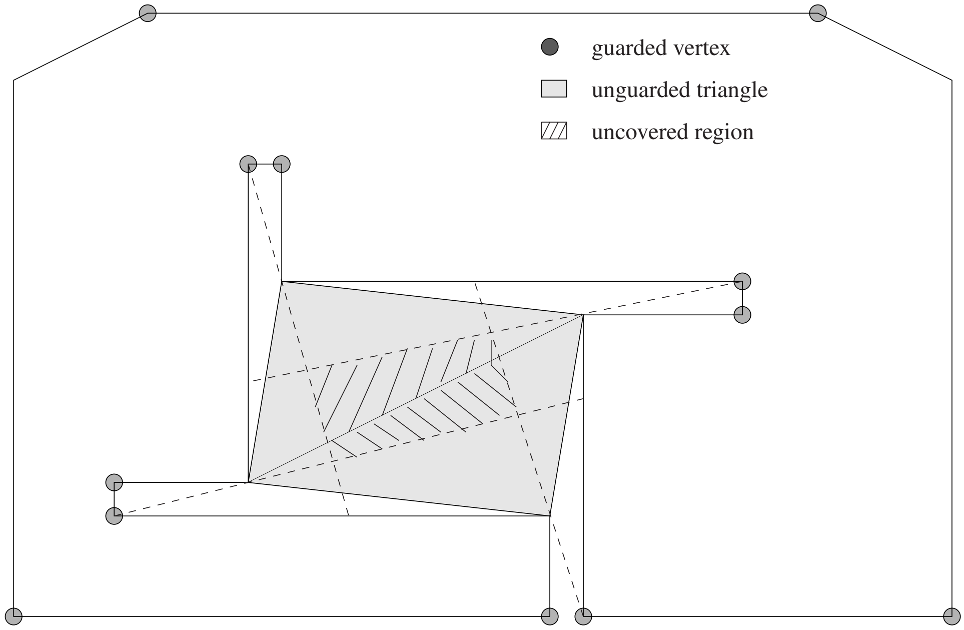

Figure 3 Example of unguarded triangle

by the other as in cases (1) and (2). (Please note that for the purpose of ease drawing, the unguarded triangles in Figures (a), (b), and (c) are clearly visible to the guards, but there are cases that the unguarded triangles cannot be covered by the guards. Figure 4d shows a real situation for case (1).)

The level of an unguarded triangle is 1 if there is no other unguarded triangle completely enclosed by it. The ones with one layer of triangles completely enclosed by it is of level 2. Higher level triangles are defined similarly.

The order of removing the unguarded triangles follows the level number of the triangles. Triangles of level i will be removed before triangles of level $(i + 1)$. If there are more than one unguarded triangles of the same level, they can be processed in any order.

Before giving the details of the proof, we sketch how the guards will be moved. Consider the lowest level, for each original unguarded triangle (v_p, v_q, v_r), the guard at (v_{r-k}, v_{r-1}), i.e., the one on the right of the highest indexed vertex of the triangle, is moved one step to the left. This may introduce one or more unguarded triangle(s), see Figure 5. If there are more than one such unguarded triangles due to the movement of the guard, consider each of them following the prescribed order. For each of these newly unguarded triangles, (v_a, v_b, v_c), move the guard at (v_{b-k}, v_{b-1}) which is on the right of the middle vertex one step to the

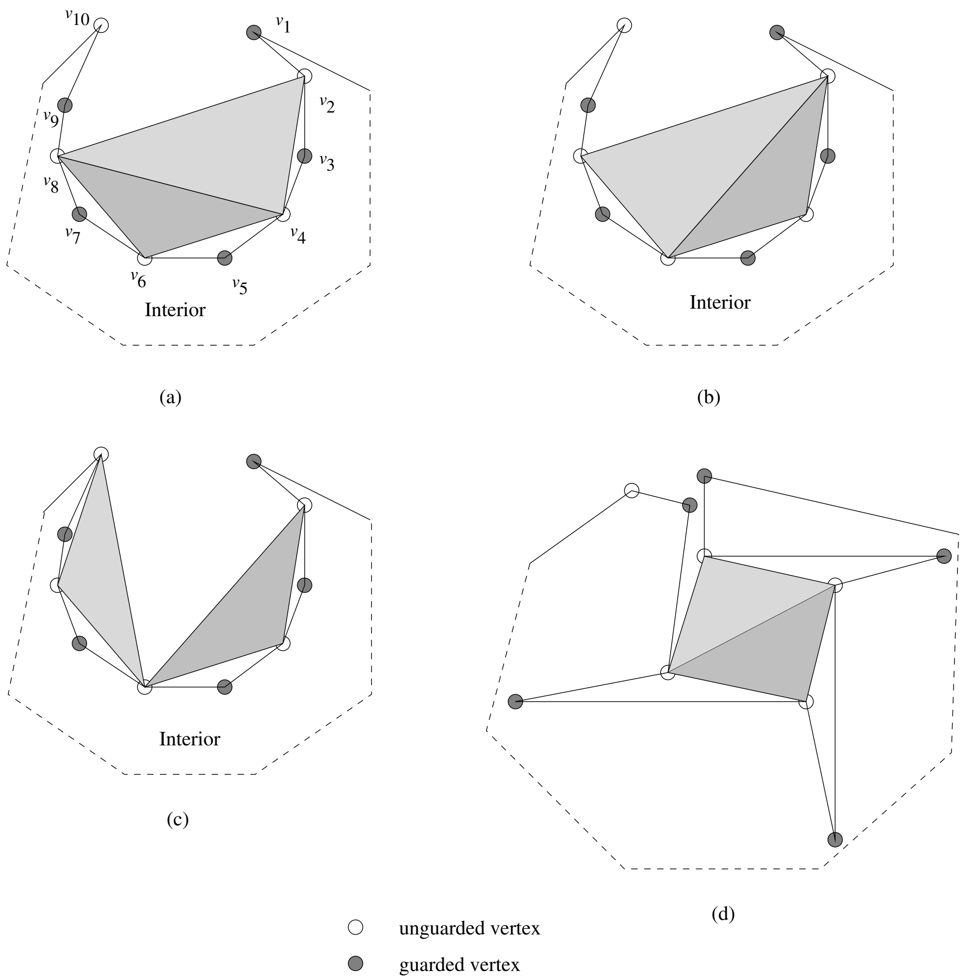

Figure 4 Relationship between two unguarded triangles

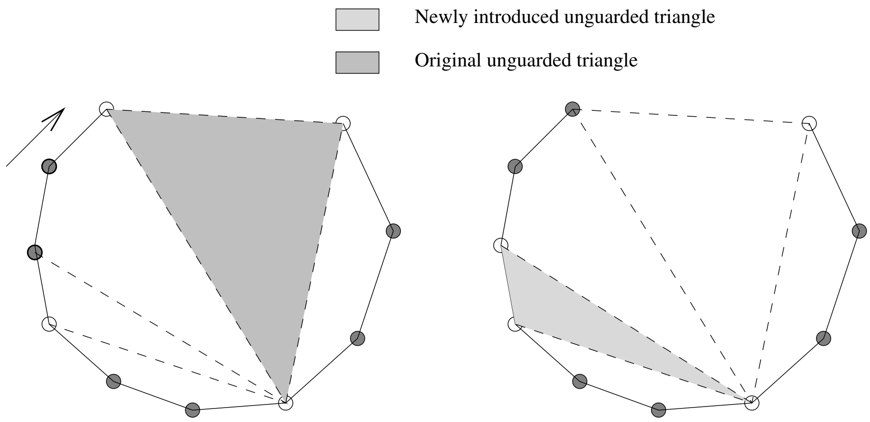

Figure 5 Unguarded triangles introduced by the algorithm

left. Again, this may introduce more unguarded triangles. Repeat the same procedure recursively. Lemma 3 guarantees that the procedure will stop and all unguarded level-one triangles together with those newly introduced unguarded triangles will be covered. Then, Lemma 4 will show that a number of properties are satisifed by which allow it to be applied to higher levels.

LEMMA 3. *Let* (v_p, v_q, v_r) *be a level-one unguarded triangle. All triangles formed with vertices from* v_p *to* v_r *will become guarded by following the above procedure.*

Proof. To remove the original unguarded triangle, move the guard at (v_{r-k}, v_{r-1}) one vertex to the left. This guard must exist since this is a level-one unguarded triangle. Vertex v_{r-k} will be unguarded and this may then introduce one or more unguarded triangles. Since all vertices with indices from $r-k+1$ to r are covered by the moved guard, and the diagonal (v_q, v_r) will prevent vertices v_{q+1} to v_{r-1} from forming triangles with other vertices, all newly introduced unguarded triangles will have v_{r-k} as the vertex with the highest index and will be formed by vertices v_q to v_{r-k} (inclusive). All such triangles will be of different levels.

If there are more than one such new unguarded triangles, order them in the manner described for the initial placement of guards and consider them in this order. Note that vertices v_{r-k-1} and v_{r-k} are both unguarded. Let (v_a, v_b, v_{r-k}) be the first of these triangles to be removed. The guard at (v_{b-k}, v_{b-1}) is moved one vertex to the left. This guard must exist due to the initial placement of the guards. By a similar reasoning, all unguarded triangles introduced in this step will be formed by vertices between v_a to v_{b-k} (inclusive) with v_{b-k} as the vertex with the highest index. The whole procedure is then repeated recursively. Either all new unguarded triangles will be removed or we stop at an unguarded triangle, (v_l, v_m, v_n), with only one guard between v_l and v_m $(m = l + k + 1)$ as shown in Figure 6. In this case, the guard at (v_{l+1}, v_{m-1}) can be moved left by one vertex to (v_{l+2}, v_m). The unguarded triangle will be covered and no more unguarded triangles will be introduced since there are only two unguarded vertices under the diagonal (v_l, v_m). The result follows. It is obvious that no guards outside the diagonal (v_q, v_r) will be moved by this procedure as all unguarded triangles introduced by the procedure are bounded by the diagonal (v_q, v_r). No guards between v_p and v_q need to be moved because the original unguarded triangle, (v_p, v_q, v_r), was at the innermost level. It will be shown that all positions of the guards which are situated between v_p and v_r are final and will not be moved again in all subsequent steps.

□

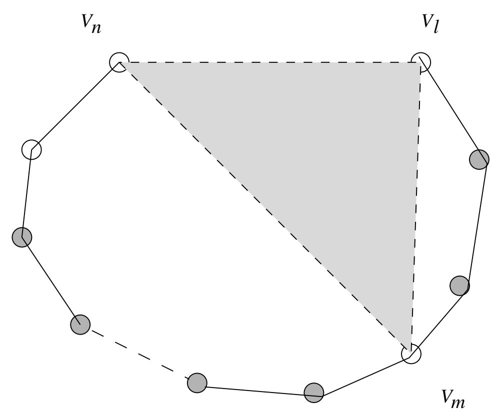

Figure 6 Last unguarded triangle introduced

LEMMA 4. *All higher level unguarded triangles can be removed by following the same procedure described above.*

Proof. Assume all level-one unguarded triangles have been covered. Let (v_p, v_q, v_r) be one of these covered triangles. Before proceeding to the next level, the following properties are guaranteed:

PROPERTY 1. Vertex v_r is guarded.

PROPERTY 2. Vertex v_{r+1} is guarded.

Vertex v_r is guarded by the guard originally at (v_{r-k}, v_{r-1}) which is shifted to cover the unguarded triangle. Since v_r *was* an unguarded vertex, v_{r+1} must be guarded. Even if it is enclosed by another original level-one triangle (v_r, v_s, v_t), the guard at (v_{r+1}, v_{r+k})

was not moved as it is not the guard on the right of the highest indexed vertex. Thus both properties are established.

According to the algorithm, for an original unguarded triangle, the guard to the right of the highest indexed vertex is moved and for a newly introduced unguarded triangle, only the guard to the right of the middle vertex will be moved. From Property 2, vertex v_{r+1} cannot be the vertex of a level-two unguarded triangle. However, if v_{r+1} is not enclosed by another level-one unguarded triangle, the guard at (v_{r+1}, v_{r+k}) may be shifted later to cover an unguarded triangle when handling level-two triangles. The vertex v_{r+1} may then become the *highest* indexed vertex of a newly introduced unguarded triangle. It will never be the middle vertex of a newly introduced unguarded triangle. Also, vertex v_r is guarded (by Property 1), it will not be a vertex of any unguarded triangle. This is the reason why we choose to move the guard to the right of the highest indexed vertex of an original unguarded triangle when resolving it. And if v_p is a vertex of an unguarded triangle, only the guard to the right of it *may* be moved. Therefore, the guards situated between v_p and v_r will never be moved again in all subsequent steps. The part of the pocket between v_p and v_r can thus be cut off as shown in Figure 7.

Let (v_a, v_b, v_c) be the first level-two unguarded triangle to be considered. If v_c is a vertex of a level-one unguarded triangle, v_c must be covered (by Property 1) and triangle (v_a, v_b, v_c) will be guarded. Otherwise, Property 2 or the initial placement of guards will guarantee that the guard (v_{c-k}, v_{c-1}) must exist, it can be shifted one vertex to the left and covered the triangle. As in the case of level-one unguarded triangles, some triangle(s) may become unguarded. Let (v_d, v_e, v_{c-k}) be the first such triangle to be considered, the above properties ensure that the guard at (v_{e-k}, v_e) must exist. So, the procedure can be repeated recursively. If all such new unguarded triangles are covered, we are done. Otherwise, a situation similar to the one shown in Figure 6 occurs. By the same argument, all new unguarded triangles together with the original level-two unguarded triangles will be covered. All higher level unguarded triangles are covered similarly. The result of Lemma 4 follows. □

LEMMA 5. *To cover the exterior of an n-vertex simple polygon, $\lceil n/(k+1) \rceil$ k-consecutive vertex guards are always sufficient.*

Proof. If no two consecutive hull vertices are unguarded, the region outside the convex hull must be covered. By Lemmas 3 and 4, the guards can be positioned to cover all triangles inside the pockets. Only $\lceil n/(k+1) \rceil$ k-consecutive vertex guards are used, so the result follows.

Suppose there are two consecutive unguarded hull vertices, v_i and v_j, they must belong to the same pocket. Before proceeding to cover the unguarded triangles, move the guard at (v_{j-k}, v_{j-1}) one vertex to the left to cover the hull vertex v_j. Vertex v_{j-k} will be unguarded and introduce some unguarded triangles. All these triangles will not be enclosed by any of the

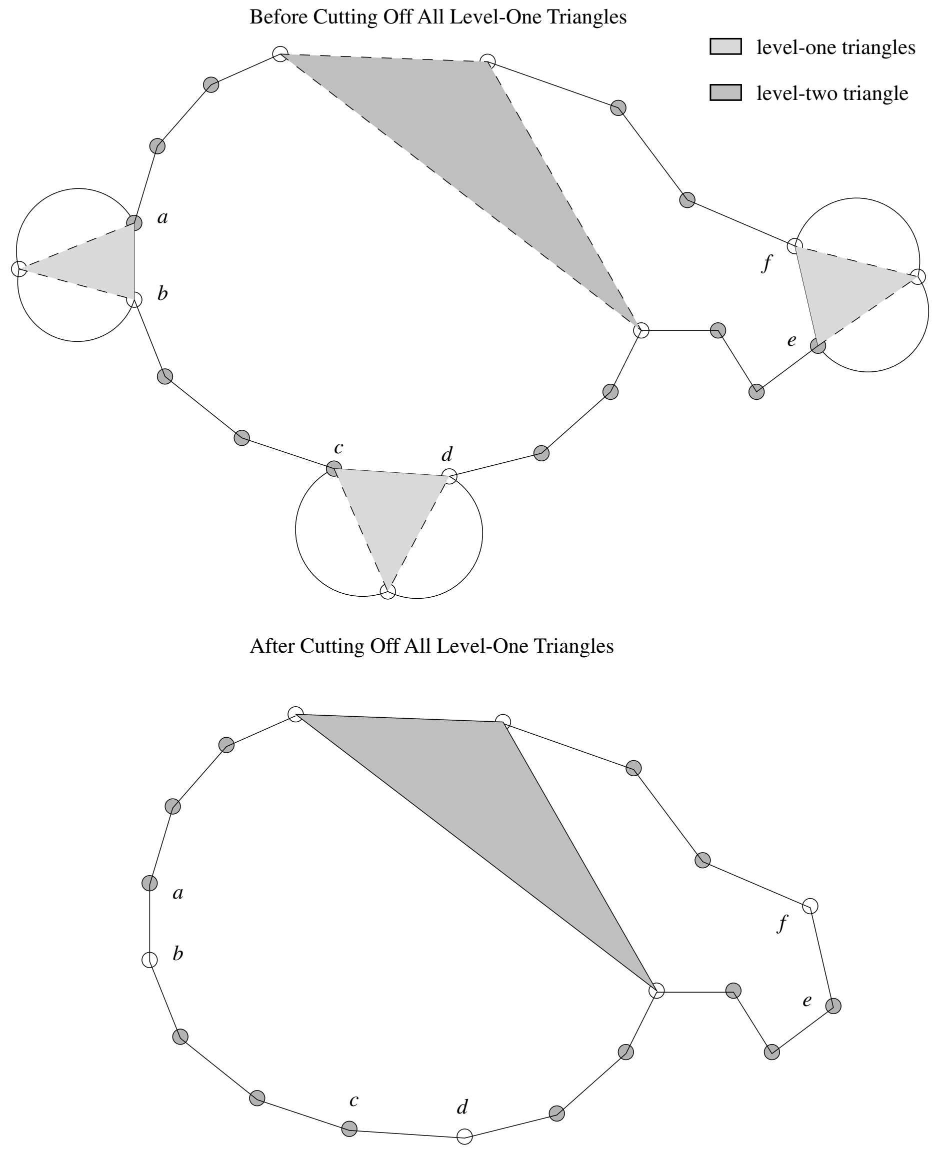

Figure 7 Cutting off covered level-one triangles

original unguarded triangles. It does not cause any problem to the proofs of Lemmas 3 and 4 except when we start to cover these triangles. Let (v_p, v_q, v_{j-k}) be one of these triangles. Instead of moving the guard at (v_{j-2k}, v_{j-k-1}), move the guard at (v_{q-k}, v_{q-1}) because the guard at (v_{j-2k}, v_{j-k-1}) does not exist. After all unguarded triangles introduced by the algorithm are covered, vertex v_{j-k} will remain unguarded. That is, it violates Property 1. However, since these triangles are the last to be removed, the subsequent steps will not depend on this property. So, it will not affect the correctness of the algorithm. □

THEOREM 1. $\lceil n/(k+1) \rceil$ *k-consecutive vertex guards are sometimes necessary and always sufficient to cover the exterior of an n-vertex simple polygon.*

Proof. By Lemmas 1 and 5, the theorem follows easily. □

COROLLARY 1. $\lceil n/(k+1) \rceil$ $(k-1)$*-consecutive edge guards are sometimes necessary and always sufficient to cover the exterior of an n-vertex simple polygon.*

Proof. By Lemmas 2 and 5, the corollary follows easily. □

References

[1] Abellanas, M., García-López, J. and Hurtado, F., *Consecutive guards*, Proceedings of the 6th Canadian Conference on Computational Geometry, 1994, 393–398.

[2] Gyori, E., Hoffmann, F., Kriegel, K. and Shermer, T., *Generalized guarding and partitioning for rectilinear polygons*, Proceedings of the 6th Canadian Conference on Computational Geometry, 1994, 302–307.

[3] O'Rourke, J., *Art Gallery Theorems and Algorithms*, Oxford University Press, Oxford 1987.

[4] Shermer, T., *Recent results in art galleries*, Proceedings of the IEEE **80** (1992), 1384–1399.

[5] Shermer, T., *Covering and guarding polygons using L_k-sets*, Geometriae Dedicata **37** (1991), 183–203.

[6] Yiu, S. M. and Choi, A., *Edge guards on a fortress*, Proceedings of the 6th Conference on Computational Geometry, 1994, 296–301.

S. M. Yiu
Department of Computer Science
and Information Systems
The University of Hong Kong
Pokfulam Road
Hong Kong
e-mail: smyiu@csis.hku.hk

Received 21 July 1999; revised 23 March 2000.