A Graph Coloring Result and Its Consequences For Polygon Guarding Problems

Frank Hoffmann*
Klaus Kriegel*

B 93–08
June 1993

Abstract

We prove the following graph coloring result: Let G be a 2–connected bipartite planar graph. Then one can triangulate G in such a way that the resulting graph is 3–colorable.
This result implies several new upper bounds for guarding problems including the first non–trivial upper bound for the rectilinear Prison Yard Problem:

1. $\left\lfloor \frac{n}{3} \right\rfloor$ vertex guards are sufficient to watch the interior of a rectilinear polygon with holes.
2. $\left\lfloor \frac{5n}{12} \right\rfloor + 3$ vertex guards resp. $\left\lfloor \frac{n+4}{3} \right\rfloor$ point guards are sufficient to watch simultaneously both the interior and exterior of a rectilinear polygon.

Moreover, we show a new lower bound of $\left\lfloor \frac{5n}{16} \right\rfloor$ vertex guards for the rectilinear Prison Yard Problem and prove it to be asymptotically tight for the class of orthoconvex polygons.

*Institut für Informatik, Freie Universität Berlin, Takustr. 9, D-14195 Berlin, *hoffmann@tcs.fu-berlin.de*, *kriegel@tcs.fu-berlin.de*. Both authors have been partially supported by the ESPRIT Basic Research Action project ALCOM II and the Wissenschaftler–Integrationsprogramm Berlin.

1 Introduction

The original Art Gallery Problem raised by V. Klee asks how many guards are sufficient to watch the interior of an n-sided simple polygon. In 1975 V. Chvátal [1] gave the answer proving that $\lfloor n/3 \rfloor$ guards are always sufficient and sometimes necessary. Since then many results have been published studying variants of the problem or analyzing algorithmic aspects, see [9], [10], [12] for a detailed discussion.
One of the main open questions in this field is the so called Prison Yard Problem for simple rectilinear polygons (comp. [10]), i.e. one wants to determine the minimal number of vertex guards sufficient to watch simultaneously both the interior and exterior of any n-sided simple rectilinear polygon.
The Prison Yard Problem for general simple polygons has been completely settled by Z. Füredi and D. Kleitman proving that $\lceil \frac{n}{2} \rceil$ vertex guards for convex and $\lfloor \frac{n}{2} \rfloor$ vertex guards for any non–convex simple polygon are sufficient, see [3]. As mentioned in [3] this does not imply new bounds for the rectilinear case. Here, the only upper bound known has been the trivial $\lfloor \frac{7n}{16} \rfloor + 5$–bound (see [9]) which can be obtained by combining the $\lfloor \frac{n}{4} \rfloor$–result for the interior (see [4]) with the $\lceil \frac{n}{4} \rceil + 1$ vertex guards for the exterior of an n–sided rectilinear polygon.
Below we are going to derive several new bounds for the original rectilinear Prison Yard Problem as well as for the stronger "Prison Problem" where the guards have to watch not only the inside and outside of the yard but also all cells of the prison. The key tools to prove them are coloring and multicoloring arguments. Especially the new graph coloring result shown in Section 2 is probably also of some independent interest. It says that one can triangulate a 2–connected bipartite planar graph in such a way that the resulting graph is 3–colorable.
In Section 3 and Section 4 we apply this result to guarding problems by a suitable modeling of the rectilinear polygons. Next in Section 5 we establish lower bounds for the vertex guard number in staircase–like and in orthoconvex prison yards. In Section 6, we use a new multicoloring technique to prove these bounds to be asymptotically tight for the described polygon classes. The following table summerizes the upper bounds on guard numbers shown in this paper for rectilinear polygons, see [9] for previous bounds:

polygon	problem	guard type	previous bound	new bound
simple	*prison yard*	*vertex*	$\lfloor \frac{7n}{16} \rfloor + 5$	$\lfloor \frac{5n}{12} \rfloor + 2$
simple	*prison yard*	*point*	$\lfloor \frac{7n}{16} \rfloor + 5$	$\lfloor \frac{n+4}{3} \rfloor$
staircase	*prison yard*	*vertex*	–	$\lfloor \frac{3n}{10} \rfloor + 2\,(tight)$
orthoconvex	*prison yard*	*vertex*	–	$\lfloor \frac{5n}{16} \rfloor + 2\,(tight)$
h *holes*	*prison*	*vertex*	–	$\lfloor \frac{5n-4h}{12} \rfloor + 2$
$h \geq n/6$ *holes*	*art gallery*	*vertex*	$\lfloor \frac{n+2h}{4} \rfloor$	$\lfloor \frac{n}{3} \rfloor$

We close in Section 7 with posing a few related open questions and with discussing algorithmic aspects of our results.

2 A Result on 3–Colorable Planar Graphs

This paragraph is devoted to the proof and the discussion of the following theorem on 3–colorings of planar graphs.

Theorem 2.1: *Let G be a planar, 2–connected, and bipartite graph. Then there exists a triangulation of G such that the triangulation graph is 3–colorable.*

The proof of the Theorem consists of two lemmata. The first one is due to Whitney and can be proved by standard induction arguments, for an elegant proof see [6].

Lemma 2.2: *A planar triangulated graph is 3–colorable iff all vertices have even degree.*

A triangulation of a planar graph G will be called *even* if in the triangulated graph any vertex has even degree.

Lemma 2.3: *Let G be a 2–connected, bipartite, and planar graph then there exists an even triangulation of G.*
Proof: Since G is 2–connected any face of it is bounded by a cycle. By adding chords to facial cycles of length > 4 we can assume that all faces of G are bounded by 4–cycles only. Let Q denote the set of these faces and Q_v be the set of all faces with a given vertex v on the boundary. Consider an auxiliary 2–coloring of G with colors *red* and *blue*. For any face $q \in Q$ we define the main diagonal to be that one connecting the vertices of q colored *red*. Furthermore, we introduce a $\{0,1\}$–valued variable x_q which will be set 1 if we choose the main diagonal in q and 0 if the other diagonal is chosen.

If v is a vertex of q we define $\epsilon_{q,v}$ to be 0 if v is colored *red* and 1 if v is colored *blue*. Obviously, $x_q \oplus \epsilon_{q,v}$ describes the increase of the degree of v by the diagonal of q chosen with respect to x_q. Here and in the following $\oplus$ denotes the addition modulo 2. It is easy to see that the existence of the desired triangulation is equivalent to the condition that the following system of equations has a solution:

$$deg(v) \oplus \bigoplus_{q \in Q_v} (x_q \oplus \epsilon_{q,v}) \;=\; 0 \quad (\forall v \in V)$$

or, equivalently,

$$\bigoplus_{q \in Q_v} x_q \;=\; deg(v) \oplus \bigoplus_{q \in Q_v} \epsilon_{q,v} \quad (\forall v \in V)$$

The left side of the second system forms the homogeneous part of the system. It is well known that such a system has a solution iff the rank of the homogeneous part is equal to the rank of the full system or, equivalently, any linear dependence of rows in the homogeneous part is also a dependence of rows in the full system. Taking into account that over $GF(2)$ the only linear combinations of rows are $\oplus$-sums it is sufficient to prove the following

Claim: *If for some set of vertices* $W \subseteq V$ *holds* $\bigoplus_{v\in W}\bigoplus_{q\in Q_v} x_q \equiv 0$ *then*

$$\bigoplus_{v\in W}(deg(v) \oplus \bigoplus_{q\in Q_v} \epsilon_{q,v}) = 0.$$

Here the symbol $\equiv$ in the first sum means that all x_q are understood as free variables, i.e $\bigoplus_{v\in W}\bigoplus_{q\in Q_v} x_q$ has to be zero for any $0,1$–assignment of the variables. Since a variable x_q occurs only in the four equations corresponding to the vertices of the face q, it follows that for any $q \in Q$ the number of vertices of q which are also in W must be even, i.e. it is $0, 2$ or 4. Now we will prove the claim by showing that the sums $\Sigma_1 = \bigoplus_{v\in W} deg(v)$ and $\Sigma_2 = \bigoplus_{v\in W}\bigoplus_{q\in Q_v} \epsilon_{q,v}$ are both zero.

1) Since G is planar and 2–connected for any vertex v the degree equals the cardinality of the set Q_v. Using this fact and changing the order of summation we get:

$$\Sigma_1 = \bigoplus_{v\in W} |Q_v| = \bigoplus_{v\in W}\bigoplus_{q\in Q_v} 1 = \bigoplus_{q\in Q}\bigoplus_{v\in q\cap W} 1 = \bigoplus_{q\in Q} |q\cap W|$$

As we have already mentioned all summands are even numbers and consequently $\Sigma_1 = 0$.

2) We start as above changing the order of summation.

$$\Sigma_2 = \bigoplus_{v\in W}\bigoplus_{q\in Q_v} \epsilon_{q,v} = \bigoplus_{q\in Q}\bigoplus_{v\in q\cap W} \epsilon_{q,v}$$

Since for any $q \in Q$ the number $|q\cap W|$ is even we can subdivide Q according to this cardinality into Q_0, Q_2 andQ_4. Furthermore, we subdivide Q_2 according to the property whether the two vertices in $q \cap W$ lie on a diagonal or on an edge of q. So we get:

$$\Sigma_2 = \bigoplus_{q\in Q_0}\bigoplus_{v\in q\cap W} \epsilon_{q,v} \oplus \bigoplus_{q\in Q_2^{diag}}\bigoplus_{v\in q\cap W} \epsilon_{q,v} \oplus \bigoplus_{q\in Q_2^{edge}}\bigoplus_{v\in q\cap W} \epsilon_{q,v} \oplus \bigoplus_{q\in Q_4}\bigoplus_{v\in q\cap W} \epsilon_{q,v}$$

Obviously the first sum is zero and can be deleted. We also delete the sum over Q_2^{diag} since any summand has either the form $1 \oplus 1$ or $0 \oplus 0$. Analogously, the sum over Q_4 is zero, but instead of deleting it, we will add it once more to S_2 and we obtain:

$$\Sigma_2 = \bigoplus_{q\in Q_2^{edge}}\bigoplus_{v\in q\cap W} \epsilon_{q,v} \oplus \bigoplus_{q\in Q_4}\bigoplus_{v\in q\cap W} \epsilon_{q,v} \oplus \bigoplus_{q\in Q_4}\bigoplus_{v\in q\cap W} \epsilon_{q,v}$$

Consider the subgraph of G induced by W and denote its edge set by E_W. We will prove that the number of 1's in the sum above is equal to $2 \cdot |E_W|$ (or equivalently

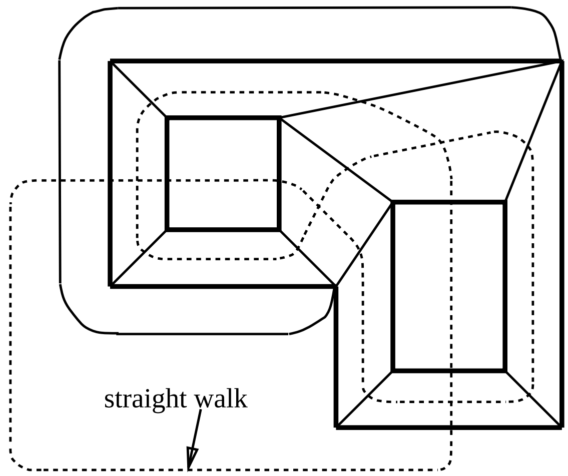


Figure 1

to the number of directed edges in E_W) what will finish the proof of the claim. Let us identify each face q with the *directed* cycle obtained by running around the region in counterclockwise order. Now the sum over Q_2^{edge} can be seen as a representation of all directed edges (u,v) in E_W such that their corresponding face q is in Q_2. Note that for any such edge the representing summand $\epsilon_{q,u} \oplus \epsilon_{q,v}$ has the form $1 \oplus 0$ or $0 \oplus 1$. Analogously, the first (resp. second) sum over Q_4 can be seen as a representation of those directed edges in E_W whichs corresponding face is in Q_4 and which are directed from red to blue (resp. from blue to red) vertices. Here any summand represents two directed edges. It is easy to observe that this representation is one–to–one what completes the proof. $\square$

For the remaining part of this section we assume that $G = (V, E)$ is 2–connected and maximal planar bipartite, i.e. all faces of G are 4–cycles. Obviously, any even triangulation can be described by a $GF(2)$ vector $(a_q)_{q \in Q}$ which is a solution of the linear system of equations in the proof above, and vice versa.
Now, our aim is to characterize the subspace of solutions of this system or, equivalently, to characterize all even triangulations of G. This will be essential for the efficiency analysis of the 3–coloring algorithm in section 7.
We consider the dual graph $G^* = (V^*, E^*)$ and identify its vertex set V^* with the set Q of faces of G. For any vertex $q \in V^*$ the edges incident with it (in counterclockwise order) correspond to the edges of G which form the facial cycle of q (traversing it in counterclockwise order). Clearly, G^* is 4–regular. A maximal path p in G^* with the property that if (q, q') is an edge on p then its successor is the next but one of the edges incident with q' in counterclockwise order will be called

a straight walk path or shortly an S–path. The maximality of S–paths implies that they are closed paths. However S–path are not necessarily simple cycles since some $q \in V^*$ may occur twice on an S–path. Figure 1 shows an example of graph with a nonsimple S–path. The set of all S–paths will be denoted by **S**.
In the following we introduce an operation which formalizes a walk on an S–path with flipping the diagonal in each step. Let $(a_q)_{q\in Q}$ be a vector over $GF(2)$ and p be an S–path. Then we define $\varphi_p \circ (a_q)_{q\in Q} = (b_q)_{q\in Q}$ by:

$$b_q = \begin{cases} a_q \oplus 1 & if\ q\ occurs\ exactly\ once\ on\ p \\ a_q & if\ q\ is\ not\ on\ p\ or\ if\ q\ occurs\ twice\ on\ p \end{cases}$$

Lemma 2.4: *If the vector $(a_q)_{q\in Q}$ represents an even triangulation of G and p is an S–path then the vector $(b_q)_{q\in Q} = \varphi_p \circ (a_q)_{q\in Q}$ also represents an even triangulation.*
Proof: We consider the operation $\varphi_p \circ (a_q)_{q\in Q}$ as a walk around p flipping the diagonals in each step. Let $(q, q') \in E^*$ be an edge on p. The corresponding edge $(u, v) \in E$ is the common edge of the faces q and q'. Remark that flipping the diagonal in q the degree of both u and v will be changed by 1 or -1 as well as it will be changed once more flipping the diagonal in q' in the next step. Thus in the end the parity of the degrees is unchanged. Extending this argument to the whole path p proves the lemma. ◻

The following theorem states that flipping diagonals along S–paths is the only way to get other even triangulations.

Theorem 2.5: *Let $(a_q)_{q\in Q}$ and $(b_q)_{q\in Q}$ be two vectors over $GF(2)$ representing even triangulations of G. Then there is a collection $(p_1, p_2, \ldots, p_k)$ of S–paths in G^* such that*

$$(b_q)_{q\in Q} = \varphi_{p_1} \circ \varphi_{p_2} \ldots \circ \varphi_{p_k} \circ (a_q)_{q\in Q}$$

.

Proof: Similar as in the proof of Lemma 2.3 the problem will be translated into a system of linear equations, but the combinatorial arguments to show that the system has a solution will be completely different. First we define a vector $(\delta_q)_{q\in Q} = (a_q \oplus b_q)_{q\in Q}$ which indicates all faces with different diagonals according to $(a_q)_{q\in Q}$ and $(b_q)_{q\in Q}$. Let $\{y_e | e \in E\}$ be a set of $\{0, 1\}$–valued variables and assume that for any face q the edges on the facial cycle of q are numbered $e_1^q, e_2^q, e_3^q, e_4^q$ in counterclockwise order. It is sufficient to prove that the following system of linear equations has a solution.

$$\begin{aligned} y_{e_1^q} \oplus y_{e_3^q} &= 0 \quad (\forall q \in Q) \\ y_{e_2^q} \oplus y_{e_4^q} &= 0 \quad (\forall q \in Q) \\ y_{e_1^q} \oplus y_{e_2^q} &= \delta_q \quad (\forall q \in Q) \end{aligned}$$

Indeed, the first and the second group of equations ensure that in any solution of the system the 1–valued variables represent the (dual) edges of a collection of

S–paths in G^*. The third group of equations guaranties that the diagonals differ exactly for those faces which occur exactly once as a vertex in this collection of S–paths. Applying the rank method as in the proof of Lemma 2.3 this system has a solution if and only if the following holds.

Claim 1: *If for some* $Q_1, Q_2, Q_3 \subset Q$ *we have that*

$$\Omega(Q_1, Q_2, Q_3) \;:=\; \bigoplus_{q\in Q_1}(y_{e_1^q} \oplus y_{e_3^q}) \oplus \bigoplus_{q\in Q_2}(y_{e_2^q} \oplus y_{e_4^q}) \oplus \bigoplus_{q\in Q_3}(y_{e_1^q} \oplus y_{e_2^q}) = 0$$

then

$$\bigoplus_{q\in Q_3} \delta_q = 0.$$

Suppose that there are sets $Q_1, Q_2, Q_3 \subset Q$ such that the assumption of the claim is fulfilled. Since we consider any $y_e = y_{e_i^q}$ in this equation as a free variable, it is clear that for any $e \in E$ the occurence of y_e in $\Omega(Q_1, Q_2, Q_3)$ is even. If $Q_3 = \emptyset$ there is nothing to prove, so we can assume that $Q_3 \neq \emptyset$. The following procedure decomposes $\Omega(Q_1, Q_2, Q_3)$ into smaller sums (each one representing a closed path in G^* but not necessarily an S–path) and possibly a subsum $\Omega(Q_1', Q_2', Q_3')$ with $Q_3' = \emptyset$:

1. Set for a new path $Q_1' = Q_2' = Q_3' = 0$.
   Choose some $q \in Q_3$ which contributes $(y_{e_1^q} \oplus y_{e_2^q})$ to $\Omega(Q_1, Q_2, Q_3)$.
   Construct a new path in G^* which starts with $(e_1^q)^*$ as the first and $(e_2^q)^*$ as the second edge.
   Delete q from Q_3 and insert it into Q_3'.

2. Denote by e^* the last edge chosen so far on the current path.

3. If e^* is identical with the first edge of the current path
   then stop the construction of this path (it is closed!) and
   if $Q_3 \neq \emptyset$ then return to 1, else stop the decomposition
   else goto 4.

4. Find some q in $Q_1 \cup Q_2 \cup Q_3$ such that y_e (e is the dual of the last edge e^*) is one of the two summands contributed by q and denote the other summand by $y_{e'}$. (Remark that such a q exists by the fact that any variable y_e occurs in $\Omega(Q_1, Q_2, Q_3)$ an even number of times.)
   Set $(e')^*$ to be the next edge on the current path.
   Delete q from that set Q_i it was chosen from and insert it into the corresponding Q_i'. Goto 2.

We remark that one possibly can find a degenerate situation in a decomposition path, namely a loop. This can happen if some q is in Q_1 (resp. Q_2) and also in Q_3. Reaching q on the current path via $(e_3^q)^*$ one can continue choosing $(e_1^q)^*$ and inserting q to Q_1'. Then one can choose $(e_2^q)^*$ to be the next edge and insert q to Q_3'.

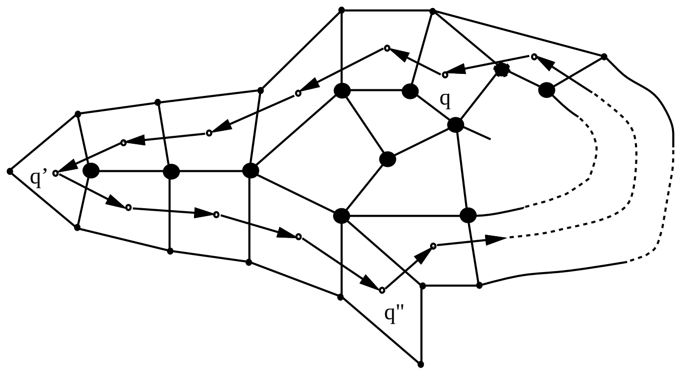


Figure 1:

Figure 2

Now, $(e_1^q)^*$ represents a loop at q instead of an edge in G^*. If Q_1', Q_2' and Q_3' are subsets of Q_1, Q_2 and Q_3 corresponding to a decomposition path then they fulfil the assumptions of the claim, i.e. we have

$$\Omega(Q_1', Q_2', Q_3') = \bigoplus_{q \in Q_1'} (y_{e_1^q} \oplus y_{e_3^q}) \oplus \bigoplus_{q \in Q_2'} (y_{e_2^q} \oplus y_{e_4^q}) \oplus \bigoplus_{q \in Q_3'} (y_{e_1^q} \oplus y_{e_2^q}) = 0$$

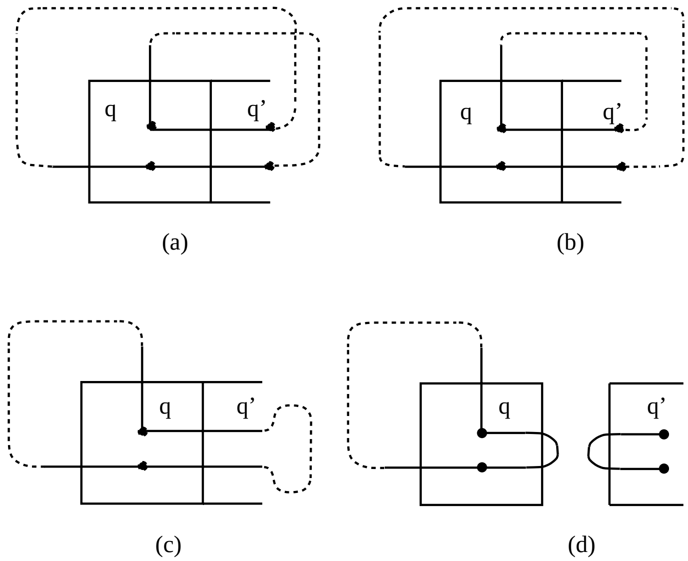
Obviously, it is sufficient to prove the claim for decomposition paths. This will be done in two steps: First for the case that the decomposition path is simple, i.e. Q_1', Q_2' and Q_3' are pairwise disjoint, and then for the general case.

Claim 2: *If Q_1', Q_2' and Q_3' are pairwise disjoint subsets of Q_1, Q_2 and Q_3 corresponding to a decomposition path p then $\bigoplus_{q \in Q_3} \delta_q = 0$.*

Let us return to the given even triangulations of G represented by the vectors $(a_q)_{q \in Q}$ and $(b_q)_{q \in Q}$. We denote by G_a (resp. G_b) the triangulation graphs and by deg_a (resp. deg_b) the degrees in these graphs. Furthermore, let $V_{int(p)}$ be the set of vertices in G which lie on the left side of the cycle p in G^* (see Fig.2, the fat points). Since both triangulations are even we have

$$\Sigma := \bigoplus_{v \in V_{int(p)}} (deg_a(v) \oplus deg_b(v)) = 0.$$

We will prove the second claim by showing that $\bigoplus_{q \in Q_3} \delta_q$ is equal to the sum Σ. To do this we define Q' to be the sum $Q_1' \cup Q_2' \cup Q_3'$ and $Q_{int(p)}$ to be the set of vertices of the dual graph G^* (i.e. faces of G) which lie on the left side of the cycle p. Now, Σ will be computed once more by counting for all edges of G_a and G_b



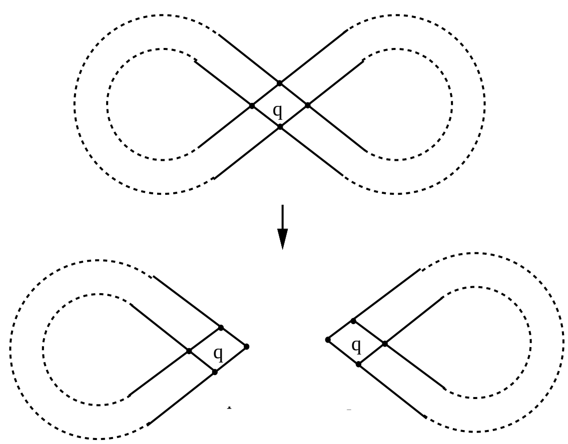
their contributions to Σ. Each such edge is either an edge of the original graph G or a

Figure 3

triangulation diagonal.
Let e be an edge of G then it has a copy e_a in G_a as well as a copy e_b in G_b. Dependent on whether e is incident to $0, 1$ or 2 vertices from $V_{int(p)}$ both copies together contribute $0, 2$ or 4 to Σ. Since we are counting modulo 2 this shows that Σ depends on the triangulation diagonals only.
For any $q \in Q$ we denote by d_a (resp. d_b) the diagonal of q in G_a (resp. G_b). Then,

it is straightforward that for any $q \notin Q' \cup Q_{int(p)}$ neither d_a nor d_b contribute to Σ. If $q \in Q_{int(p)}$ then each of the diagonals contributes 2 to Σ because both incident vertices are in $V_{int(p)}$. Hence Σ depends on triangulation diagonals of faces in Q' only. If $q \in Q'_1 \cup Q'_2$ then it is a part of a straight walk segment of p. Exploring the assumption about the simplicity of p, the face q contains exactly two vertices from $V_{int(p)}$ which lie on an edge of G (see Fig.2 for illustration). Hence any diagonal is incident to exactly one of these two vertices. This implies that d_a and d_b together contribute 2 to Σ. Thus, Σ depends only on the triangulation diagonals of faces $q \in Q'_3$. We note that any such face contains either exactly one (q' and q'' in Fig.2) or exactly three (q in Fig.2) vertices from $V_{int(p)}$. If d_a and d_b coincide (iff $\delta_q = 0$) they clearly contribute together an even number to Σ. Otherwise, if $\delta_q = 1$ one of them contributes an odd and the other one an even number to Σ. This proves our Claim 2 because we have:

$$\bigoplus_{q \in Q'_3} \delta_q = \Sigma = 0$$



Remark: In fact, this proves that for any simple closed path p in G^* the number of faces $q \in Q$ with $\delta_q = 1$ and containing an odd number of vertices from $V_{int(p)}$ is even.

Figure 4

Claim 3: *If Q'_1, Q'_2 and Q'_3 are subsets of Q_1, Q_2 and Q_3 corresponding to a decomposition path p in G^* then $\bigoplus_{q \in Q_3} \delta_q = 0$.*

If there is a q which occurs multiply on p then it must be contained in at least two of the Q_i. Decomposing p into smaller paths, we will reduce such multiplicities (except those ones caused by loops) step by step.

Reduction-type 1: $q \in Q'_1 \cap Q'_3$

If $(e_1^q)^*$ does not represent a loop then it is used twice in p. Denote by q' the neighbor of q reached by $(e_1^q)^*$. There are two possible connection schemes shown in Fig.3a and 3c. The first one can be resolved easily. Exchanging the entries in q' (see Fig.3b) we decompose p into two paths which can be processed independently.

The situation sketched in Fig.3c will be reduced to the degenerate one by cutting both edges between q and q' and replacing them by two loops (see Fig.3d). We note that all subpaths constructed so far one could obtain also immediately from the decomposition procedure making suitable choices in step 1 and step 4.

Reduction-type 2: $q \in Q'_2 \cap Q'_3$ – analogiously to Case 1.

We will apply reductions of type 1 and 2 to p or respectively to the subpaths obtained from p as long as such multiplicities appear in the nondegenerate form.

Reduction-type 3: $q \in Q'_1 \cap Q'_2$

In this situation we split p as indicated in Fig.4. Unfortunately, the resulting paths are not decomposition paths since the turns in q do not correspond to a summand of $\bigoplus_{Q'_1, Q'_2, Q'_3}$. We apply type 3 reductions as long as possible and denote by Q'' the set of all q to which reductions of this type were applied. The result of the whole procedure is a decomposition of p in subpaths which are almost simple (loops are possible). Ignoring all loops we get a family of simple paths $\{p_1, p_2, \ldots, p_k\}$ to which the analysis of the remark on page 9 can be applied. Any $q \in Q''$ occurs in exactly two of these paths. Since if for any path p_j a faces q contains an odd number of vertices from $V_{int(p_j)}$ then either $q \in Q'_3$ or q is one of the two copies of some $q \in Q''$ we finally get

$$0 = \bigoplus_{q \in Q'_3} \delta_q \oplus 2 \cdot \bigoplus_{q \in Q''} \delta_q = \bigoplus_{q \in Q'_3} \delta_q$$

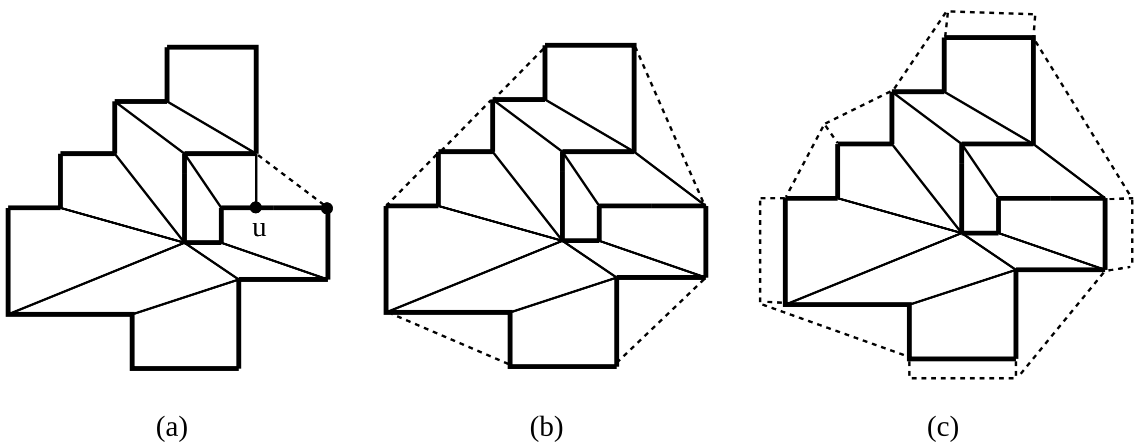
□

3 A Graph Model for the Prison Yard Problem

The idea for our graph model is based on the following nice and simple proof of the classical Art Gallery Theorem due to S. Fisk [2]. Consider any triangulation graph of a given simple polygon. One knows that it is 3–colorable. Clearly, any triangle contains each color and choosing guard position corresponding to the smallest color class one can watch the polygon using $\leq \lfloor \frac{n}{3} \rfloor$ guards.

J. Kahn, M. Klawe, and D. Kleitmann ([4]) applied a similar idea to rectilinear polygons. They proved that any rectilinear polygon (possibly with holes) has a convex quadrilateralization, i.e. a decomposition into convex 4–gons (called quadrilaterals) using only diagonals (here called chords) of the polygon. Moreover it is easy to see that for simple rectilinear polygons the graph consisting of all polygon edges, all chords, and both inner diagonals of all quadrilaterals is 4–colorable. Hence, they obtained an $\lfloor \frac{n}{4} \rfloor$–upper bound for the rectilinear Art Gallery Problem. However, the 4–colorability of this graph does not hold starting with polygons having holes. But now Theorem 2.1 states that one can select one diagonal per quadrilateral such that the graph formed by all polygon edges, all chords, and the selected diagonals is 3–colorable. So we get at least an $\lfloor \frac{n}{3} \rfloor$–upper bound on the vertex guard number for the Art Gallery Problem in the presence of holes. Below we introduce a graph model which allows us to apply this coloring result also to Prison–type Problems.

Given an n–sided rectilinear polygon P (w.l.o.g. in general position, see [4]) we construct its orthoconvex hull $OConv(P)$, i.e. the smallest point set containing P and such that its intersection with any horizontal or vertical line is convex (see Fig.5a). This partitions the exterior region of P into the exterior region of $OConv(P)$ and those connected components of $OConv(P) \setminus P$ which are different from the interior region of P. They will be called pockets. Since all pockets are bounded by rectilinear polygons there is a quadrilateralization of them as well as of the interior of P. We have to pay for this construction by some additional vertices (u in our example). However, using ideas from [7] one can shift these vertices to neighboring polygon corners on the boundary of $OConv(P)$ in such a way that the resulting polygon



$OConv^*(P)$ is also orthoconvex and the quadrilateralizability of the pockets is not destroyed (see Fig.5a, the dashed line).

Fig.5a Fig.5b Fig.5c

Now we have to cover the exterior region of $OConv^*(P)$ with convex sets. Remark that $OConv^*(P)$ is bounded by four extremal edges (northernmost, westernmost, southernmost, easternmost) which are cyclically connected by monotone staircases. So the exterior of $OConv^*(P)$ is covered by four halfplanes defined by the extremal edges and the cones defined by all concave vertices on the staircases.

Let $G(P)$ be the planar graph (Fig.5b) over the polygon vertices the edge set of which consists of all polygon edges, all quadrilateralization chords, and all pairs of consecutive convex corners on boundary stairs of $OConv^*(P)$. We say that a subset C of the vertex set dominates $G(P)$ if any quadrilateral, any triangle over a staircase, and any of the four extremal edges contains at least one vertex from C. The Prison Yard Problem in this context now reads: Find a small dominating set for $G(P)$.

Since we want to apply Theorem 2.1. it is neccessary to modify $G(P)$ in such a way that it becomes bipartite, i.e. all convex regions (also the exterior cones and halfplanes!) which have to be dominated are represented by convex quadrilaterals. To do this we need some additional vertices. We start as before constructing $OConv^*(P)$. Then we use 8 new vertices to obtain a copy of all extremal edges as indicated in Fig.5c. Finally, for any monotone boundary staircase of $OConv^*(P)$ which contains more than one convex vertex (we do not count the vertices on the extremal edges) we copy every second one of them. Now it is possible to replace any boundary triangle in $G(P)$ by a quadrilateral in the new graph $G^*(P)$, comp. Fig.5c. The number of additional vertices is bounded by $\lfloor \frac{n-12}{4} \rfloor + 8$ and hence the number of vertices of $G^*(P)$ is bounded by $\lfloor \frac{5n}{4} \rfloor + 5$. Obviously, if a new vertex will be chosen as a guard position this guard can be placed onto the original polygon vertex.

In order to demonstrate the power of Theorem 2.1. we introduce the Prison Problem which is a generalization of the Prison Yard Problem as well as of the Art Gallery Problem for polygons with holes. Let a rectilinear polygon P with h rectilinear holes $P_1, \ldots, P_h$ be given, having in total n vertices. We have to select a set of vertices such that any point in the plane can be watched from one of the selected vertices. A graph $G^*(P, P_1, \ldots, P_h)$ representing this problem can be constructed as follows:
1) quadrilateralize the holes $P_1, \ldots, P_h$,
2) quadrilateralize the interior of P minus the holes,
3) proceed with the exterior of P as in the construction of $G^*(P)$.
Clearly, P has at most $n - 4h$ vertices and hence the number of additional vertices for the construction of $G^*(P)$ is bounded by $8 + \left\lfloor \frac{n-4h-12}{4} \right\rfloor$. Thus, $G^*(P, P_1, \ldots, P_h)$ has at most $\left\lfloor \frac{5n-4h}{4} \right\rfloor + 5$ vertices.

Finally, we generalize the concept of graph coloring to the notion of labellings and multicolorings. Suppose, we have given k different colors. Then a function which labels any vertex of a graph $G(P)$ with a certain set of colored pebbles will be called a k-labelling. It is a k–multicoloring if adjacent vertices are labelled with disjoint color sets. A labelling is called l-uniform if the pebble sets have cardinality l for all vertices. Clearly, any k-coloring is a 1-uniform k-multicoloring. A multicoloring is called dominating if for any color the set of those vertices labelled with a pebble of this color dominates $G(P)$. Hence, a dominating k-multicoloring of $G(P)$ which uses in total $f(n)$ pebbles implies the existence of an $\left\lfloor \frac{f(n)}{k} \right\rfloor$ solution of the Prison Yard Problem for P.

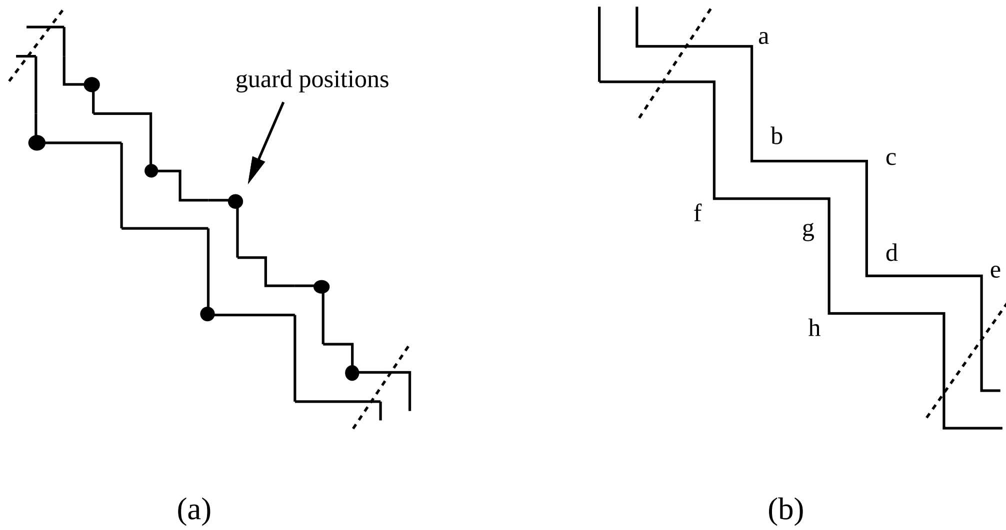
4 General Upper Bounds

We start with a straightforward application of Theorem 2.1 to the classical rectilinear Art Gallery Problem. Consider a rectilinear polygon P with h holes and a total of n vertices. Both the polygon and the holes can be quadrilateralized, so we have the following.

Corollary 4.1: $\left\lfloor \frac{n}{3} \right\rfloor$ *vertex guards are sufficient to solve the Art Gallery Problem for rectilinear polygons with holes.*

Observe, that the guards watch the interior of the holes, too. Further let us remark that this improves the prevously known $3n/8$–bound which was obtained by converting a polygon with holes into a 1–connected one by adding h edges ($2h$ new vertices) and then applying the $n/4$–result of [4]. However, there is some evidence that the $2n/7$–lower bound, see [5], for the vertex guard number is tight.

Conjecture 4.2: *For any quadrilateralized rectilinear polygon possibly with holes there is a 2–uniform dominating 7–coloring.*



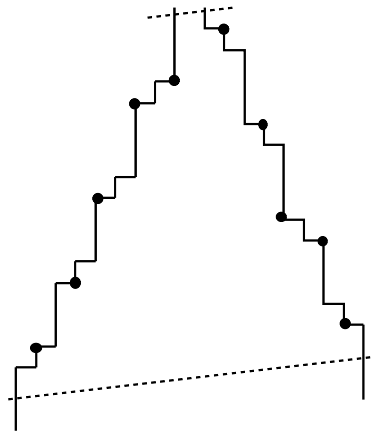
The second application of Theorem 2.1 deals with the weak version of the rectilinear Prison Yard Problem where point guards are allowed, see [9]. Here the spiral polygon gives an $(\lceil \frac{n}{4} \rceil + 1)$–lower bound. However, the best upper bound up to now has been the same as for the vertex guard version: $\lfloor \frac{7n}{16} \rfloor + 5$.

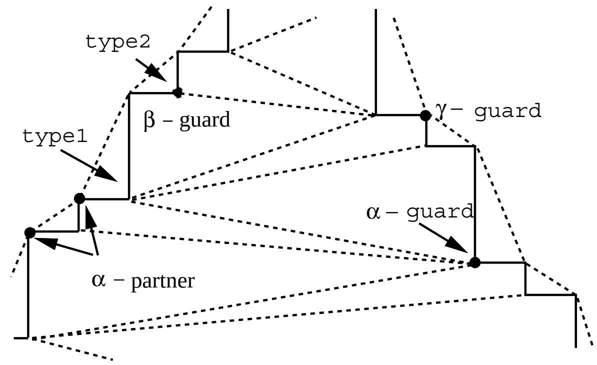
Fig.6a: Dorward's example Fig.6b: $3n/10$ guards are necessary

Corollary 4.3: *For any rectilinear polygon (possibly with holes) P on n vertices $\lfloor \frac{n+4}{3} \rfloor$ point guards are sufficient to solve the Prison Problem.*
Proof: Let R be a rectangle enclosing P. We consider R together with P as a polygon P' having P as a hole. After quadrilateralizing P' as well as the original P the resultig graph fulfills the assumptions of Theorem 2.1 and we can choose guard positions corresponding to the minimal color class of the dominating 3–coloring. □

Observe, that one gets at most two point guards the other guards can be chosen to sit in vertices.

Corollary 4.4: *For any rectilinear polygon on n vertices with h holes* $\lfloor \frac{5n-4h}{12} \rfloor + 2$ *vertex guards are sufficient to solve the Prison Problem.*
Proof: Apply Theorem 2.1 to the graph $G^*(P)$ defined in Section 3. □





5 Lower Bounds

Any simple convex polygon requires $\lceil \frac{n}{2} \rceil$ vertex guards to solve the Prison Yard Problem. What are candidates for lower bound examples in the rectilinear world? Figure 6a shows an example of a rectilinear polygon due to Dorward who claimed that it required $\lceil \frac{n}{3} \rceil$ guards, see [10]. Continuing, however, periodically the guarding positions indicated in Fig.6a one sees that $\lceil \frac{7n}{24} \rceil + 2$ watchmen are sufficient. Let P_0 be the simplest staircase shown in Fig.6b.

Proposition 5.1: *The prison yard* P_0 *requires* $\lceil \frac{3n}{10} \rceil$ *vertex guards.*

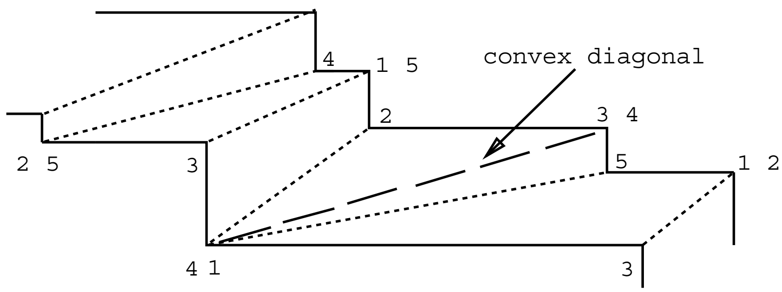
Fig.7: $5n/16$ vertex guards are necessary

Proof: Consider a segment S on 10 vertices as indicated in the figure. We need at least two guards for the triangles abc, cde, fgh . But there are also 4 inner corridors in S to be watched what is impossible with one guard sitting in c and only one more in f,g, or h. Finally, guards placed in disjoint segments cannot help each other watching these triangles and the inner corridors. ◻

We will show in the next section that $\lceil \frac{3n}{10} \rceil + 2$ vertex guards are also sufficient for any strictly monotone rectilinear polygon. Surprisingly, there are other monotone polygons which require even more guards. Let P_1 be the pyramid in Fig.7. Assume that the edge lengths are chosen in such a way that to watch an inner quadrilateral one has to choose one of its vertices as guard position. Again, as indicated $5n/16$ guards are sufficient (up to an additive constant) and we show that we need as many.

Proposition 5.2: *The prison yard* P_1 *requires* $\lceil \frac{5n-10}{16} \rceil$ *vertex guards.*

Proof: We distinguish 3 types of guards, comp.Fig.7. A guard stationed in a concave corner such that he can watch four quadrilaterals is called an α–guard. We remark that any such guard must have at least one "partner" on the other



side watching the opposite triangle and we choose one of them and form an α–pair. An α–guard pair watches together 4 quadrilaterals, 2 type1–triangles, and 1 type2–triangle. β– resp.γ–guards are sitting in concave (convex) corners and they are not part of α–pairs. They watch each 2 (resp.1) quadrilaterals , 0 (resp.1) type1–triangle, and 1(resp.1) type2–triangle. Since we have a total of $(n-2)/2$ quadrilaterals and $(n-2)/4$ triangles of each type we conclude for any guarding set consisting of $g = 2a + b + c$ guards of α–, β–, and γ–type respectively that: $4a + 2b + c \geq (n-2)/2, \quad 2a + c \geq (n-2)/4, \quad a + b + c \geq (n-2)/4$. Adding to the first inequality the second one and then the third multiplied by 2 we get $4g \geq 5(n-2)/4$. But this implies the lower bound. $\square$

Fig.8: 5–multicoloring a strictly monotone polygon

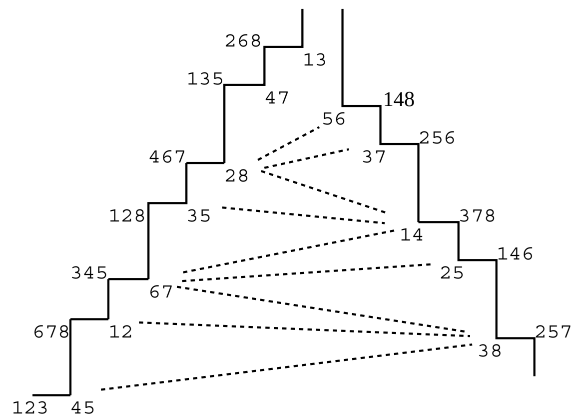
6 Special Upper Bounds

In this section we are going to show that the lower bounds derived for strictly monotone and for orthoconvex rectilinear polygons are tight up to an additive constant. We use the idea from Section 2 to construct a dominating set of vertices in the graph $G(P)$ by a multicoloring.

Theorem 6.1: $\left\lfloor \frac{3n}{10} \right\rfloor + 2$ *guards are sufficient to solve the Prison Yard Problem for strictly monotone rectilinear polygons.*

Proof: Let P denote such a polygon, say with north–west orientation (comp. Fig.8). First we remark that the quadrilateralization of P is unique and its dual is a path $W = q_1, q_2, \ldots q_{(n-2)/2}$; any chord of the quadrilateralization connects a convex with a concave vertex and any quadrilateral has a diagonal connecting two convex vertices (called convex diagonal).

Let $(d_i), i = 1, 2, \ldots n-1$ be the following sequence of polygon edges, diagonals, and chords obtained by traversing W. We start with d_1 the bottom polygon edge in q_1 followed by the convex diagonal of q_1, d_i is the common edge of $q_{(i-1)/2}$ and $q_{(i+1)/2}$ for an odd $1 \leq i$ otherwise it is the convex diagonal of $q_{i/2}$. d_{n-1} is the top



edge of P. The d_i's induce a canonical numbering of the vertices of P. Starting with d_2 each d_i encounters exactly one new vertex, say v_{i+1}. Let Q^i denote the polygon generated by the first i quadrilaterals.
We show that there is a greedy algorithm which following the path W constructs a dominating 5-multicoloring of $G(P)$ with the following properties:

- both the north–westernmost vertex v_n and the south–easternmost vertex v_2 are labelled by 4 pebbles;
- any other convex (resp. concave) vertex is colored by 2 (resp.1) pebbles.

Fig.9: 8–multicoloring the lower bound example

While building this multicoloring we maintain the following invariant: Each convex diagonal contains exactly 3 colors, i.e. there is one common color on both sides of the diagonal. We start as follows. Color the left endpoint v_1 of the bottom edge d_1 by 2 colors the right vertex v_1 by one pebble with a third color. Complete the multicoloring of q_1 by repeating one color from v_1, say c, in v_3 together with a pebble having the fourth color. One pebble with the fifth color is put on v_4.
Assume that we have already colored correctly Q^i. The next vertex to be colored is v_{2i+3}. It closes a triangle already labelled by three different colors, hence it gets the remaining two colors. v_{2i+4} is colored by the fifth color not used before in q_{i+1}.
Eventually, to get a dominating multicoloring we have to put on v_n the remaining 3 colors to dominate both the northern and western extremal edge of P, and similarly 3 more pebbles on v_2.
Why does this scheme work correctly?
Let $\chi(j)$ denote the set of colors put on vertex j. Assuming q_i being colored correctly we have for its convex diagonal (v_k, v_{2i+1}) that $|\chi(k) \cap \chi(2i+1)| = 1$. Now the algorithm colors v_{2i+3} by 2 pebbles such that $|\chi(2i+3) \cup \chi(2i+2) \cup \chi(l)| = 5$, where v_l is that vertex of the i–th convex diagonal which is in a common exterior triangle with v_{2i+3}. But then for the other vertex v_m of the i–th diagonal follows

that $|\chi(2i+3) \cap \chi(m)| = 1$ (so the invariant holds for q_{i+1}) and we can indeed color v_{2i+4} by the fifth color not being element of the set $\chi(2i+2) \cup \chi(2i+3) \cup \chi(m)$ which has cardinality 4.

In total we use $2\frac{n+4}{2} + \frac{n-4}{2} + 4 = \frac{3n+12}{2}$ pebbles. Consequently, there exists a dominating color class of size $\leq \lfloor \frac{3n}{10} \rfloor + 2$. □

In a similar way we prove the following statement.

Theorem 6.2: $\lfloor \frac{5n}{16} \rfloor + 2$ *guards are always sufficient to solve the Prison Yard Problem for orthoconvex rectilinear polygons.*

Proof: We outline the proof for pyramids. The result follows the for an orthoconvex polygon P by decomposing it into at most 2 pyramids and 1 strictly monotone polygon. For the strictly monotone part of P we extend the 5–multicoloring constructed in Thm. 6.1 to an 8–multicoloring by adding an independent dominating 3–multicoloring. Using Thm. 2.1 this 3–multicoloring can be chosen to be 1–uniform for all vertices not on extremal edges.

Recall that a horizontal pyramid is a rectilinear polygon with a horizontal edge (bottom edge) the length of which equals the sum of the lengths of all other horizontal edges. For such a pyramid P we consider again the dual path $W = q_1, q_2, ..., q_{(n-2)/2}$ of its unique quadrilateralization. We construct a dominating 8–multicoloring of $G(P)$ with the following properties:

- one of the bottom edge vertices has 5 pebbles the other 6 pebbles;
- one of the top edge vertices has 4 pebbles the other 5 pebbles;
- any other convex vertex has 3 pebbles, any concave one gets 2 pebbles;

Again, the existence of such a dominating 8–multicoloring can be shown using a still simple but slightly more complicated greedy algorithm along W. Let d_0 be the bottom edge and d_i for $1 \leq i < (n-2)/2$ denote the common edge of the quadrilaterals q_i and q_{i+1}. We duplicate both bottom edge vertices and introduce on both sides dummy (zero length) horizontal edges. Now we are in a situation where on both sides of any d_i, $i \geq 0$ there are 2 horizontal edges. Let $\bar{d}_i$ denote the path consisting of these 2 horizontal edges with d_i in the middle. $\bar{d}_i$ starts and ends with a convex vertex and has 2 concave middle vertices. The invariant we maintain during the algorithm is the following:

Denoting the colors by 1 thru 8 the color pattern on $\bar{d}_i$ is modulo a permutation of the form $123 - 45 - 16 - 247$, hence one color is missing.

First initialize the coloring on $\bar{d}_0$ using this pattern. Having colored the first i quadrilaterals, $\bar{d}_{i+1}$ has 2 new vertices, one concave and the other convex. The convex one closes an exterior triangle which has already 5 colors because of the color pattern of

$\bar{d}_i$. So it gets the remaining 3 colors. Now it is not hard to see that q_{i+1} has already got 6 different colors, so we can put the remaining 2 on the new concave vertex of $\bar{d}_{i+1}$ which then also fulfills the invariant condition.
Eventually, after having dominated the 2 top exterior triangles on both sides we have to put 3 more pebbles to the top edge and 1 more pebble to $\bar{d}_0$. We end up with a dominating 8–multicoloring providing the claimed bound. □

Let us close this section by a remark on the constructed multicolorings. It seems to be curious to use 5 or 8 dominating color classes when constructing 1 dominating vertex set. The point here is that in our multicoloring all convex (concave) vertices not on extremal edges get the same number of pebbles, so it is trivial to count the pebbles. Recall that the number of convex (concave) vertices does not depend on the special shape of the polygon. Of course, one can try to construct directly, say, in a greedy way a dominating set. However, even in such a regular example like in Fig.9 it would be hard to give a good estimate of its size.

7 Related Open Problems and Algorithmic Aspects

Let us add two more open problems to those mentioned in Section 4. First, we start with a remark concerning Theorem 2.1. It is essential for the result proved there that all inner faces of the graph G are 4–cycles. The result is not true if like in the graph $G(P)$ in Section 3 there are also inner triangles. So one needs some new ideas to prove for example an $\lfloor \frac{n}{3} \rfloor$–upper bound for the rectilinear Prison Yard Problem. We think that replacing the multicoloring argument used in Theorem 6.2 by an 8–labelling one can show that the following conjecture is correct.

Conjecture 7.1: *There is an absolute constant c such that any rectilinear prison yard can be watched by $5n/16 + c$ vertex guards.*

Recall, that in a k–labelled graph it is possible for adjacent vertices to be labelled by pebbles of the same color. Further, it would be interesting to find applications of Theorem 2.1 or of some multicoloring or labelling to non–rectilinear art gallery type problems, compare with [3], [11], and Chapter 5.2 in [9].

Below we are going to analyze several algorithmic aspects of our results. First, let us mention that all upper bound results proved here can be converted into efficient algorithms. Since one can quadrilateralize simple rectilinear polygons in linear time we can guard orthoconvex prison yards also in linear time using the greedy algorithm from Thm.6.2.

A much more interesting algorithmic problem is how to find the 3–coloring of Theorem 2.1 more efficiently than by using a general (superquadratic) method for solving linear systems of equations. Subsequently we describe a quadratic upper bound which clearly implies the same time bound for the algorithmic problems in 4.1, 4.3, and 4.4 .
Let us outline the idea. The main point, it is possible to find an efficient substitution scheme for solving the system of linear equations exploring the facts that the equations are over $GF(2)$ and that the underlying graph is planar. First, we iteratively use simple cycle separators from [8] to obtain a face numbering for which at any moment the boundary (which can be disconnected) between already numbered faces and the remaining faces has total length at most $O(\sqrt{n})$. Then one shows that there is a substitution scheme based on this numbering with the following properties:

- The length of any substitution is bounded by $O(\sqrt{n})$;
- Any substitution is applied to at most $O(\sqrt{n})$ equations (which correspond to boundary points only).

To go into the details let us start with quoting a result from [8].

Theorem 7.2: *If G is an embedded 2–connected planar graph, with an assignment of weights to the vertices which sums to 1, and no face has weight $> 2/3$ then there exists a simple cycle weighted separator of size $2\sqrt{2\left\lfloor \frac{d}{2}\right\rfloor n}$, where d is the maximum face size. Further, this cycle is constructible in linear time.*

Recall, a simple cycle C of G is a weighted separator if both the weight of the interior of C and the weigth of the exterior is $< 2/3$. In our situation we use uniformly distributed weights and $d \leq 4$. Our goal is to define a suitable enumeration of the face set Q of G.
We apply the Theorem to G and the cycle separator defines a partition of Q into two subsets Q_1, Q_2. Now consider the subgraphs G_1^*, G_2^* induced by these sets in the dual graph G^* and define two new graphs by setting $G_1 = (G_1^*)^*$ and $G_2 = (G_2^*)^*$. Observe, that the new graphs are planar and fulfil the assumptions of the Theorem. Especially the maximum face size does not increase and the elements of Q_j are in 1–1 correspondence with the faces of G_j for $j = 1, 2$. We apply recursively this procedure to the new graphs until we eventually end up with graphs of size $O(\sqrt{n})$. This way we obtain a partition tree of Q with depth $O(\log n)$. The set of its leaves defines a partition of Q and we enumerate the faces according to the left–to–right order of the leaves and an arbitrary enumeration within each leaf. Let Q^i be the set of the first i faces in the enumeration and G^i the graph defined by the edges and vertices of Q^i. Then a simple calculation shows that for all i the boundary length of the graph G^i is bounded by $O(\sqrt{n})$. Here the boundary length is defined as the number of edges belonging both to a quadrilateral q_k with $k \leq i$ and some q_l with $l > i$.
Having such an enumeration $q_1, q_2, ..., q_m$ (with $m = n/2 - 1$) of the quadrilaterals

fixed we can describe the algorithm and the underlying substitution scheme as follows.

The Algorithm:

Data Structures: First we store attributes associated with the variables x_q in an array of length m according to the fixed enumeration. Such an attribute is from the set $\{$*free,0,1,subst*$\}$. Here *free* means that up to a given stage of the algorithm there has been no restriction to the value of the variable in a final solution of the system. The attributes 0 and 1 describe the situation that the algorithm determines at a given stage that in each solution of the system a variable will have value 0 resp. 1. Finally, *subst* stands for the fact that the variable has been substituted at some moment by the sum of other variables. Initially all attributes are set *free*.
Furthermore we need an $m \times (m+1)$–array to store for any variable which is not *free* either its value (in the last column) or its substitution formula. A substitution formula is a $\oplus$–sum of some variables and a last summand which is 0 or 1. The set of summands which form a substitution formula are encoded by a 0-1–vector of length $m+1$ in a row of the array. In an third array we store attributes of vertices, these are $\{$*passive, active, dead*$\}$. Let Q^i denote the set of the first i quadrilaterals. Then *passive* (resp. *dead*) means for a vertex v at stage i that $Q^i \cap Q_v = \emptyset$ resp.$Q_v \subset Q^i$, otherwise the vertex is *living*. Initially all vertices are *passive*; at some stage each vertex starts to be *living*, and eventually all are *dead*.
We have an $(4m) \times (m+1)$–array to store in its rows the "current potential"$f_{q,v}$ of all the corners. A corner is just a pair (q,v), where v is a vertex of q. The current potential of a corner is a $\oplus$–sum of some variables x_p and a last summand which is 0 or 1. The initial potential of any corner (q,v) is 0.
Finally, in a fifth $n \times (m+1)$–array we similarly maintain the potentials f_v of the vertices, initially set to be $deg(v)$.

Stage 1 – Construction of a Substitution Scheme: The algorithm works in a greedy way. After the described initialization step we add one quadrilateral after the other and (possibly) manipulate during each step the potentials of the newly added corners as well as the potentials of some living vertices. Recall that all living vertices are on the boundary. We maintain during the algorithm the following invariant:
If v, w are neighboring vertices in some $q \in Q^i$ then for the current potentials we have $f_{q,v} = f_{q,w} \oplus 1$. Moreover, for each dead vertex v holds $f_v \equiv 0$.
Assume we have already processed i quadrilaterals and our invariant holds. Let q be the $(i+1)$st quadrilateral.
Vertices of q which were *passive* become *living*. Consider the set of (at most) 4 *living* vertices which become *dead* by adding q to Q^i. If this set is empty we update each corner potentials by setting $f_{q,v} = x_q \oplus \epsilon_{q,v}$ and then we add this $f_{q,v}$ to f_v.
If there is exactly one new dead vertex v we set $f_{q,v} := f_v$ and $f_{q,w} := f_v \oplus 1$ for each of the 2 neighboring corners in q and, consequently, $f_{q,v}$ for the fourth vertex in q

opposite to v. We update the vertex potentials by adding the corresponding corner potential. Observe, that for the dead vertex v we have then $f_v \equiv 0$. Finally, we give the label *subst* to x_q and insert the formula $x_q = f_{q,v} \oplus \epsilon_{q,v}$ in the substitution array. What happens if we have more than 1 new dead vertices? In this case we first choose an arbitrary vertex of this set, say v, and proceed as before. This way we can garuantee that after the updating $f_v \equiv 0$. But what about the other dead vertices, say some w in q. Its corner potential in q is either f_v or $f_v \oplus 1$. We want that $f_w := f_{q,w} \oplus f_w \equiv 0$. There are 2 trivial cases, namely that this holds automatically or if we can set some variable 0 or 1 to fulfil this condition. Otherwise, we have to choose a free variable, say x_p, occuring in f_w and substitute it by $f_w \oplus x_p$ (hence the x_p's cancel out). We insert this substitution of x_p in the substitution array and apply it in all f_z of still living vertices. (Remark: We do not (!) substitute x_p in the dead vertices.) This procedure is applied to all of the remaining dead vertices of q.

Stage 2 – Computation of the Solution: After the first stage of the algorithm it remains to denest the substitution formulae. This problem can be represented by a directed graph with m nodes, each one corresponding to a variable. If a variable x_q has the attribute *subst* then we draw arcs from its node to all nodes of variables which occur in the substitution formula of x_q. By the construction of the substitution scheme (substitutions are applied to the potential of all living vertices) this graph is acyclic and all sinks correspond to variables with the attributes *free, 0* or *1*. We remark that for any evaluation of the *free* variables we get one solution of the system. For simplicity we set all *free* variables to be *0*. Now all sinks in the acyclic graph are labelled *0* or *1*. We choose any node such that all its sucessors are sinks and evaluate the corresponding variable according to its substitution formula. Deleting all arcs drawn from this node it becomes a sink. We repeat this procedure until all variables are evaluated. Clearly, one can modify this method without evaluating the *free* variables in advance.

Analysis of the Running Time: Since one can find cyclic separators in linear time, we can compute the enumeration of the faces in $O(nlogn)$ time.
For the first stage of the algorithm we will show that any step (adding the $(i+1)$st face) can be done in linear time. Here, the only critical point is the application of a substituiton to the potentials of all living vertices. Clearly, the number of applications is bounded by $O(\sqrt{n})$ whereas, the cost of one application is equal to the number of variables occuring in the substitution formula. We note that these variables come from the potential of the new dead vertex. Hence, it is sufficient to show that after i–th step of the algorithm the total set of variables occuring in the potentials of living vertices is bounded by $O(\sqrt{n})$. Let us denote this set by X.
Recall that the total boundary length of the current graph G^i is bounded by $O(\sqrt{n})$. The boundary can be disconnected, but any component will be an even cycle because G^i is a subgraph of a bipartite graph. Obviousely, one can add edges to G^i (inside the boundary faces) in such a way that the resulting graph H is 2–connected and maximal planar bipartite. Thus, we can apply our algorithm also to H, using the

same enumeration up to the i–th face. Since G^i contains already all vertices of H, the number of vertices which get dead in the further work of the algorithm is bounded by $O(\sqrt{n})$. Consequently, if Y denotes the set of variables which get label *subst* , *0* or *1* after the i–th step of the algorithm then its cardinality is also bounded by $O(\sqrt{n})$. Let Z be the set of variables labelled free after finishing the first stage of the algorithm (on H). Then we have

$$X \subset Y \cup (X \cap Z).$$

It remains to show that the number of variables in $X \cap Z$ is bounded by $O(\sqrt{n})$. Since a free variable x_q can generate different solutions of the system it corresponds to a collection of S–paths by Theorem 2.5, i.e. x_q is in the corner potential of those faces whichs occurence on S–paths of the collection is odd. Moreover, if such a variable is also in X then one of its S–paths crosses the boundary of G^i. Otherwise for any boundary vertex v the number of faces in Q^i which touch v and contain x_q in their corner potential would be even and this would be a contradiction to $x_q \in X$. Finally, we remark that the number of S–paths crossing the boundary of G^i is not greater than one half of the boundary length. This implies the required bound for $|X \cap Z|$ and hence the number of variables in any substitution formula is bounded by $O(\sqrt{n})$. We conclude that the running time of the first stage of the algorithm is bounded by $O(n^2)$
It is straightforward that the second stage of the algorithm can be executed in $O(n^2)$ time. Taking into account the fact that any substitution formula is not longer than $O(\sqrt{n})$ one can solve this stage in $O(n^{3/2})$ time.
Therefore the total running time of our algorithm is $O(n^2)$.

References

[1] V. Chvátal, *A combinatorial theorem in plane geometry*, J. Combin. Theory Ser. B, 1975, Vol. 18, pp. 39-41.

[2] S. Fisk, *A short proof of Chvátal's watchman theorem*, J. Combin. Theory Ser. B, 1978, Vol. 24, p. 374

[3] Z. Füredi and D. Kleitmann, *The Prison Yard Problem*, to appear in Combinatorica,

[4] J. Kahn and M. Klawe and D. Kleitman, *Traditional galleries require fewer watchmen*, SIAM Journal of Alg. Disc. Math., 1983, pp. 194-206.

[5] F. Hoffmann, *On the rectilinear art gallery problem*, Proc. ICALP'90, LNCS 443, 1990, pp. 717-728.

[6] L. Lovász, *Combinatorial Problems and Exercises*, North Holland, Amsterdam, 1979.

[7] A. Lubiw, *Decomposing polygonal regions into convex quadrilaterals*, Proc. 1st ACM Symp.Comp.Geometry, 1985, pp. 97-106.

[8] G. Miller, *Finding small simple cycle separators for 2–connected planar graphs*, Journal of Comp. and System Sciences, 1986, pp. 265-279.

[9] J. O'Rourke, *Art gallery theorems and algorithms*, Oxford University Press, New York, 1987.

[10] J. O'Rourke, *Computational Geometry Column 15*, SIGACT News 23:2, 1992, pp. 26-28

[11] T. Shermer, *Triangulation graphs that require extra guards*, NYIT, Computer Graphics tech. report No.3D-13, 1984.

[12] T. Shermer, *Recent results in art galleries*, Simon Fraser University, Technical report 1990, TR 90-10.