

# The paranoid watchman: a search problem on graphs

Dominik Brugger, Ulm University  
Christine Cheng, University of Wisconsin-Milwaukee  
Janet Dalzell, Ivy Tech Community College  
David Leach, University of West Georgia  
Matt Walsh, Indiana-Purdue University Fort Wayne

## Abstract

We consider the problem of a watchman or team of watchmen guarding a sealed facility, modelled by a simple graph, in which there may be an intruder. Upon detection the intruder is considered to be caught; a watchman can detect an intruder anywhere in his (closed) neighbourhood. Given a fixed graph  $G$  and that watchmen are permitted to freely traverse  $G$ , how many watchmen are required in order to guarantee that no intruder is hiding in  $G$ , even if the intruder anticipates the watchers' movements? We partially characterize those graphs for which a single watchman is sufficient and gives bounds for the broader problem in terms of other search parameters.

**Keywords:** Graph searching, pursuit-evasion, interval graphs, cops and robbers, domination search.

## 1 The problem

Consider the following scenario: a watchman is tasked with guarding a facility, and wishes to ensure that no intruder has entered. Unless the watchman is lucky enough to be posted someplace where he can perceive the entire structure, he must periodically walk through to ensure the integrity of the facility. Let us model the facility's layout with a simple graph  $G$ : each vertex will represent a room, with edges between adjacent rooms. Note that hallways, under this model, would be represented as vertices rather than edges; doorways and similar portals would be the real-world analogue of edges.

We imbue our watchman with two characteristics: perspicuity and paranoia. For the first, we assume that if the watchman is at a given vertex  $v$ ,

then he will detect an intruder anywhere in  $N[v]$ , the closed neighbourhood of  $v$ . If this seems unreasonable given our model, then we can refine the model so that vertices represent portions of rooms rather than entire rooms. The second trait is expressed in our (and his) assumptions about the intruder: namely, that if there is an intruder in the facility, she is capable of anticipating the watchman's moves perfectly, and further is capable of moving arbitrarily fast through the graph so long as she does not enter into the watchman's neighbourhood.

The watchman's walk can be modelled as a function  $w : \mathbb{R}^+ \rightarrow V \cup E$  where his position at time  $t$  is  $w(t)$ ; for any  $v \in V$ , the pre-image  $w^{-1}(v)$  is a union of open intervals, while the pre-image  $w^{-1}(e)$  of any edge  $e$  is a union of singletons. Further, the sequence of successive vertices and edges over any interval yields a walk in  $G$ : if  $w(t) = e_{ij}$  then for some  $\epsilon > 0$  either  $w(t - h) = v_i$  and  $w(t + h) = v_j$  for every positive  $h < \epsilon$  or vice-versa. Our question is this: which graphs  $G$  admit a function  $w$  with these properties, such that if there is an intruder in the graph, she will be seen? We shall call such a graph *watchable*.

To further formalize the notion of watchability, let us define an *intruder function*  $i : \mathbb{R}^+ \rightarrow V \cup E$  under the same terms as the watchman's function. The watchman catches (that is, perceives) the intruder if, for some positive time  $t$ , both  $w(t)$  and  $i(t)$  are vertices and  $i(t) \in N[w(t)]$ . A watchable graph is one where there exists a watchman function  $w$  such that for any intruder function  $i$  there exists a real  $t$  such that the watchman catches the intruder.

For graphs that are not watchable in this sense, the natural question is then how many watchmen (each with their own walk functions, in coordination with each other) are necessary to ensure that a given graph is clear of intruders. We say that a graph is  $k$ -watchable if there is a collection of  $k$  walk functions  $w_1, \dots, w_k$  that ensure that any intruder will be caught by one of the watchmen; the *watchman number*  $w(G)$  of a graph  $G$  is the least  $k$  such that  $G$  is  $k$ -watchable.

Alternatively, the problem can be formulated in purely discrete terms as a two-player game. Here the watchman and the intruder are assigned some initial positions, and the two players alternate in making moves. The watchman's move consists of taking a new position at an adjacent vertex to his previous position; the intruder then moves along a path of arbitrary length (including length 0), with the limitation that she must stop upon entering a vertex in the watchman's neighbourhood. (Note that the intruder can start from such a vertex, but must stop if she remains in or returns to the neighbourhood.) The intruder is aware of the watchman's position at all

times. The game ends if the watchman and the intruder occupy the same vertex; a watchable graph is one where the watchman has a strategy that guarantees termination.

While this problem has not previously been studied, it does have several antecedents and relatives in the literature of both searching and pursuit-evasion. The study of searching and sweeping problems in graphs was initiated by Parsons in [9, 10]; surveys by Alspach [1] and Fomin and Thilikos [5] provide an excellent overview of the field. A central concept in the searching and sweeping literature is that of contamination; in our parlance, a vertex is *contaminated* if it is possible that an intruder might be hiding there. This gives a third description of the watchman problem: as in the previous model, the watchman has an initial position and makes a sequence of moves along edges. At all times the watchman’s neighbourhood is uncontaminated; a vertex becomes recontaminated if there is a path between that vertex and a contaminated one that avoids the watchman’s neighbourhood. A graph is watchable if there is a sequence of moves that decontaminates all vertices.

Of particular note is the *domination search* problem, introduced by Fomin, Kratsch, and Müller [4]. Here a co-ordinated team of searchers are attempting to decontaminate a graph; the team alternatively places searchers at vertices (decontaminating their neighbourhoods) and removes them (possibly recontaminating vertices, under the same rules for recontamination as in the watchman problem). The domination search number  $ds(G)$  of a graph  $G$  is the smallest number of searchers required to decontaminate  $G$ . While quite similar in spirit and description, the watchman and domination search problems are distinct; one obvious difference is that the only graphs  $G$  with  $ds(G) = 1$  are those such that  $\gamma(G) = 1$  while the class of watchable graphs is significantly broader.

Perhaps the most celebrated pursuit-evasion game in graph theory is “cops and robbers”; this is a two-player game of perfect information where one player controls a set of cops (searchers), while the other controls a robber. Players alternately move their tokens along edges, where the cop player can move as many of their tokens as they like on a given turn, and either player can choose to pass by not moving their token(s). The game ends when a cop occupies the same vertex as a robber; the cop number of a graph is the smallest number of cops required to guarantee termination. (In [1] this model is referred to as the *basic pursuit-evasion* (BPE) model of searching graphs, with the corresponding parameter called the *search number*.) A graph with cop number 1 is called *cop-win*; other graphs, where the robber can forever evade a single cop, are *robber-win*. The class of cop-win graphs was characterized by Quilliot [11] and independently by Nowakowski and

Winkler [8]: a graph is cop-win iff its vertices can be ordered  $v_1, \dots, v_n$  such that for every  $i < n$  there is some  $j > i$  such that  $N_i[v_i] \subseteq N_i[v_j]$ , where  $N_i[x] = N[x] - \bigcup_{k < i} \{v_k\}$ . Such an ordering is called a *dismantling ordering*, and the notion of dismantling will come into play in our analysis of watchable graphs.

The structure of the rest of this paper is as follows. In the next section, we give a formal demonstration of the equivalence of the three models of the paranoid watchman problem. In section 3 we discuss some possible conditions for watchable graphs, and demonstrate some classes of watchable graphs. Section 4 examines relationships between the watchman problem and other graph search problems (and other parameters). Section 5 gives some results on the watchman number of a tree. Section 6 considers the question of monotonicity; a search model is monotone if there exists a search programme on any graph that never allows a vertex to be recontaminated. Many of the previously studied graph searching models are monotone, but we show that the watchman problem is not. We close by briefly discussing some other possible variations of the problem, such as allowing a watchman's view to extend beyond his immediate neighbourhood.

## 2 Models of the paranoid watchman problem

We start with formalizing a discrete search model for the watchman problem, which is much more convenient both for comparisons with other (discretely-defined) search problems and for algorithmic purposes. A *k-watchmen search program*  $\Pi$  on graph  $G = (V, E)$  is a sequence of pairs

$$\Pi = (W_0, A_0), (W_1, A_1), \dots, (W_r, A_r)$$

such that :

- (i) for  $i = 0$  to  $r$ ,  $W_i = (v_1^i, v_2^i, \dots, v_k^i)$  where  $v_j^i \in V$  is the location of watchman  $j$  at step  $i$  for  $j = 1, \dots, k$ ;
- (ii) for  $j = 1, \dots, k$ ,  $i = 0, \dots, r - 1$ , either  $v_j^i = v_j^{i+1}$  or  $\{v_j^i, v_j^{i+1}\} \in E$  (i.e., at each step, a watchman either stays put or moves to an adjacent vertex);
- (iii)  $A_0 = \bigcup_{j=1}^k N[v_j^0]$  (i.e., the set of cleared vertices at the end of step 0 is the union of the closed neighbourhoods of the vertices containing a watchman at step 0);

- (iv) for  $i = 1$  to  $r$ ,  $A_i = \tilde{A}_{i-1} \cup \bigcup_{j=1}^k N[v_j^i]$  where  $\tilde{A}_{i-1} \subseteq A_{i-1}$  such that every path from a vertex in  $V - A_{i-1}$  to a vertex in  $\tilde{A}_{i-1}$  contains a non-starting vertex in  $\bigcup_{j=1}^k N[v_j^i]$  (i.e.,  $\tilde{A}_{i-1}$  is the set of cleared vertices at the end of step  $i-1$  that are “protected” by the  $k$  watchmen at step  $i$ . A vertex  $u \in \tilde{A}_{i-1}$  is protected because any path from a contaminated node  $w$  at the end of step  $i-1$  to  $u$  has to pass through the neighbourhood of some  $v_j^i$ .)

We say that  $\Pi$  is a *winning*  $k$ -watchmen search program if  $A_r = V$ .

Let  $f_1, f_2, \dots, f_k$  be the movement functions of  $k$  watchmen in graph  $G = (V, E)$ . Set  $t_0 = 0$ , and recursively define  $t_i$  as the first time after  $t_{i-1}$  that some watchman moved. Hence, for watchman  $j$ ,  $f_j(t_i)$  is either a vertex or an edge and  $f_j(t)$  is fixed throughout the interval  $(t_i, t_{i+1})$  for  $i \in \{0\} \cup \mathbf{Z}^+$ . Thus, we can think of the time line as being partitioned into intervals  $[t_0, t_1), [t_1, t_2), \dots, [t_{i-1}, t_i), [t_i, t_{i+1}), \dots$  as illustrated in Figure 1.

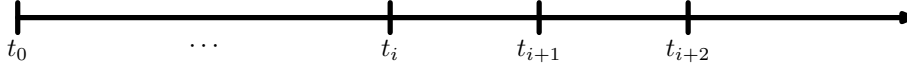


Figure 1: Time line

For each  $j$ , create another movement function  $f'_j$  from  $f_j$  so that the watchman's movement in the interval  $[i, i+1)$  is a time-scaled version of his movement under  $f_j$  in the interval  $[t_i, t_{i+1})$ , for  $i \in \{0\} \cup \mathbf{Z}^+$ . In other words,

$$f'_j(t') = f_j(t) \text{ where } t' - i = \frac{t - t_i}{t_{i+1} - t_i}.$$

**Claim 2.0.1.** *The  $k$  watchmen using  $f_1, \dots, f_k$  can catch any intruder in  $G$  if and only if they can do so using  $f'_1, \dots, f'_k$ .*

*Proof.* Let  $g$  be the movement function of an intruder in  $G$ . Define another intruder movement function  $g'$  so that his movement in the interval  $[i, i+1)$  is also a time-scaled version of his movement under  $g$  in the interval  $[t_i, t_{i+1})$ ,  $i \in \{0\} \cup \mathbf{Z}^+$ . Notice that  $k$  watchmen using  $f_1, \dots, f_k$  cannot catch an intruder using  $g$  if and only if  $k$  watchmen using  $f'_1, \dots, f'_k$  cannot catch an intruder using  $g'$ . The lemma follows immediately.  $\square$

Given  $f_1, f_2, \dots, f_k$ , let  $A(t)$  denote the set of vertices in  $G$  that the watchmen are sure to be intruder-free at time  $t$ . That is, if  $v \notin A(t)$ , then  $v$  is a potential hiding place for an intruder because there is some intruder

function  $g$  so that an intruder using  $g$  is not caught by any of the watchmen by time  $t$  and  $g(t) = v$ . Similarly, define  $A(t^-)$  and  $A(t^+)$  as the set of vertices in  $G$  that the watchmen are sure to be intruder-free just prior to time  $t$  and just after time  $t$  respectively.

Suppose there exists  $t_i$  and  $t_{i'}$ ,  $i < i'$  so that the positions of each watchman is the same in the intervals  $(t_i, t_{i+1})$  and  $(t_{i'}, t_{i'+1})$  and  $A(t_{i+1}^-) = A(t_{i'+1}^-) \neq V$ . For each  $j$  create another movement function  $f'_j$  from  $f_j$  by omitting the portion from  $[t_{i+1}, t_{i'+1})$ . That is, set  $f'_j(t)$  to  $f_j(t)$  when  $t \in [0, t_{i+1})$  and to  $f_j(t + t_{i'+1} - t_{i+1})$  when  $t \in [t_{i+1}, \infty)$ .

**Claim 2.0.2.** *The  $k$  watchmen using  $f_1, f_2, \dots, f_k$  can catch any intruder in  $G$  if and only if they can do so using  $f'_1, f'_2, \dots, f'_k$ .*

*Proof.* Suppose an intruder in  $G$  is moving according to  $g$ , and  $k$  watchmen using  $f_1, \dots, f_k$  cannot capture this intruder. Let  $v$  denote the location of the intruder just prior to  $t_{i'+1}$ ; that is, if  $g(t_{i'+1})$  is a vertex then  $v = g(t_{i'+1})$  and if  $g(t_{i'+1})$  is an edge then  $v$  is the vertex which the intruder is moving from. In either case,  $v \notin A(t_{i'+1}^-)$  because the watchmen cannot catch the intruder. Since  $A(t_{i+1}^-) = A(t_{i'+1}^-)$ , there is another intruder movement function  $g'$  so that the intruder eludes capture by the watchmen using  $f_1, \dots, f_k$ , and the intruder is at vertex  $v$  just prior to  $t_{i+1}$ . Define another intruder function  $g''$  as follows: let  $g''(t) = g'(t)$  in the interval  $[0, t_{i+1})$ , and let  $g''(t) = g(t + t_{i'+1} - t_{i+1})$  in the interval  $[t_{i+1}, \infty)$ . Since  $f'_1, \dots, f'_k$  was obtained from  $f_1, \dots, f_k$  by simply omitting the  $[t_{i+1}, t_{i'+1})$ -portion of these functions, it is now easy to see that  $k$  watchmen using  $f'_1, \dots, f'_k$  cannot capture an intruder using  $g''$ .

On the other hand, suppose some intruder is moving according to some function  $h$ , and  $k$  watchmen using  $f'_1, \dots, f'_k$  cannot capture this intruder. As in the previous paragraph, let  $u$  denote the location of the intruder just prior to  $t_{i+1}$ . Since the functions  $f'_1, \dots, f'_k$  behave exactly like  $f_1, \dots, f_k$  in the interval  $[0, t_{i+1})$ ,  $u \notin A(t_{i+1}^-)$ . But  $A(t_{i+1}^-) = A(t_{i'+1}^-)$  so there is an intruder function  $h'$  so that the intruder using  $h'$  eludes capture by the watchmen using  $f_1, \dots, f_k$ , and the intruder is at vertex  $u$  just prior to  $t_{i'+1}$ . Then consider the intruder function  $h''$  so that  $h''(t) = h'(t)$  in the interval  $[0, t_{i'+1})$  and  $h''(t) = h(t - (t_{i'+1} - t_{i+1}))$  in the interval  $[t_{i'+1}, \infty)$ . Again, it is easy to verify that an intruder using  $h''$  cannot be captured by  $k$  watchmen using  $f_1, f_2, \dots, f_k$ .  $\square$

Let us denote a movement function on  $G$  as *integer-stepped* if and only if a watchman moves at integer time steps.

**Theorem 2.1.** *Let  $G = (V, E)$  be a graph with  $n$  vertices. Then  $G$  is  $k$ -watchable if and only if there exists  $k$  integer-stepped movement functions on  $G$  so that by time  $T = n^k 2^n + 1$ , any intruder in  $G$  is captured by  $k$  watchmen using these functions.*

*Proof.* If  $k$  watchmen can catch any intruder in  $G$  by time  $T$ ,  $G$  is clearly  $k$ -watchable. So let us assume the converse –  $G$  is  $k$ -watchable. Hence, there exists  $k$  movement functions  $f_1, f_2, \dots, f_k$  so that they can capture any intruder in  $G$ . According to Claim 2.0.1, these functions can be made integer-stepped without compromising the watchmen's ability to capture intruders in  $G$ . Furthermore, according to Claim 2.0.2, they can also be trimmed so that whenever each watchman is occupying exactly the same position at intervals  $(i, i + 1)$  and  $(i', i' + 1)$  with  $i' > i$ , then  $A((i + 1)^-) \neq A((i' + 1)^-)$ . Now, there are  $n^k$  different sequences of vertices that the watchmen can occupy in an interval  $(i, i + 1)$ . There are also at most  $2^n$  possible sets for  $A((i + 1)^-)$ . So there are at most  $n^k 2^n$  distinct  $(i, i + 1) - A((i + 1)^-)$  pairs. Hence, once the  $k$  watchmen have gone through  $n^k 2^n$  different intervals, they must surely have captured the intruder.  $\square$

**Theorem 2.2.**  *$w(G) \leq k$  if and only if there exist  $k$  watchman movement functions  $f_1, f_2, \dots, f_k$  such that for some time  $t < \infty$ , the set of vertices the watchmen know for certain are intruder free at time  $t$  is  $V$ ; i.e.  $A(t) = V$ .*

*Proof.* ‘ $\Rightarrow$ ’: According to Theorem 2.1,  $A(n^k 2^n + 1) = V$ .  
‘ $\Leftarrow$ ’: Suppose  $\exists k$  watchman movement functions  $f_1, f_2, \dots, f_k$  such that for some time  $t < \infty$ ,  $A(t) = V$ . Let  $g$  be some intruder function. If the intruder has not been caught by time  $t$ , then the vertices occupied by the intruder at time  $t' \geq t$  are not intruder-free. But this contradicts the assumption that  $A(t) = V$ .  $\square$

Let us now convert the continuous model to the discrete model.

**Theorem 2.3.** *Let  $f_1, f_2, \dots, f_k$  be the movement functions of  $k$  watchmen on graph  $G$ . If there exists  $t < \infty$  such that  $A(t) = V$  then there is a winning discrete  $k$ -watchman search program  $\Pi$ .*

*Proof.* Again, set  $t_0 = 0$  and let  $t_{i+1}$  be the first time after  $t_i$  that one of the  $k$  watchmen moves. Hence, as before, we think of the time line as being partitioned into intervals  $[t_0, t_1), [t_1, t_2), \dots, [t_{i-1}, t_i), [t_i, t_{i+1}), \dots$ . Once the watchmen have settled into their respective vertices at  $t_i$ , the intruder is free to roam the graph.

First, let us investigate what happens in the continuous model when the watchmen move from their vertices  $v_j^i$  to  $v_j^{i+1}$  at time  $t_{i+1}$ . Then the vertices  $\bigcup_j N[v_j^{i+1}]$  are guaranteed to be intruder-free. As the intruder can only move from one vertex to an adjacent one at time  $t_i$ , vertices on the border to the not intruder-free area that are not in the neighborhood of the watchmen, i.e. vertices in

$$\begin{aligned} R_{\{t_{i+1}\}} &:= \left\{ v \in A(t_{i+1}^-) - \bigcup_j N[v_j^{i+1}] \mid \exists \{u, v\} \in E, u \in V - A(t_{i+1}^-) \right\} \\ &= \left\{ v \in A(t_{i+1}^-) \mid \exists \{u, v\} \in E, u \in V - A(t_{i+1}^-) \right\} - \bigcup_j N[v_j^{i+1}] \end{aligned}$$

are also no longer guaranteed to be intruder-free.

Now let us also consider the time frame  $(t_{i+1}, t_{i+2})$ , i.e. after the watchmen have settled at the vertices  $v_j^{i+1}$  but before they move again. During this time, no new vertices are identified as intruder-free, but every vertex in the set

$$R_{(t_{i+1}, t_{i+2})} := \left\{ v \in A(t_{i+1}^+) \mid \exists \text{ path from } u \in R_{\{t_{i+1}\}} \text{ to } v \text{ not intersecting with } \bigcup_j N[v_j^{i+1}] \right\}$$

is no longer considered intruder-free.

Therefore  $A(t_{i+1}^+) = \left( A(t_{i+1}^-) \cup \bigcup_j N[v_j^{i+1}] \right) - R_{\{t_{i+1}\}}$  and  $A(t_{i+2}^-) = A(t_{i+1}^+) - R_{(t_{i+1}, t_{i+2})}$ .

For  $i \in \{0\} \cup \mathbf{Z}^+$  define  $W_i = (v_1^i, v_2^i, \dots, v_k^i) = (f_1(t_i), f_2(t_i), \dots, f_k(t_i))$  as illustrated in Figure 2. Recall that in the discrete model  $A_i$  is the set of cleared vertices at step  $i$ .

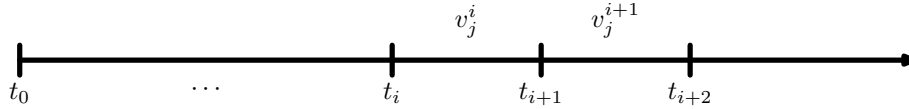


Figure 2: Location of watchman  $j$  on the time line

**Claim 2.3.1.** For every  $i \in \{0\} \cup \mathbf{Z}^+$ ,  $A_i = A(t_{i+1}^-)$ .

*Proof.* For  $i = 0$ ,  $A_0 = \bigcup_j N[v_j^0] = A(t_1^-) = A(t_{0+1}^-)$  ✓

Now assume  $A_i = A(t_{i+1}^-)$  for some  $i \in \{0\} \cup \mathbf{Z}^+$ . We have to show:  $A_{i+1} = A(t_{i+2}^-)$

*Proof:* “ $\subseteq$ ”: Let  $v \in A_{i+1} = \bigcup_j N[v_j^{i+1}] \cup \tilde{A}_i$ .

We need to show:  $v \in A(t_{i+2}^-)$ , i.e.  $v \in A(t_{i+1}^+) - R_{(t_{i+1}, t_{i+2})}$ . If  $v \in \bigcup_j N[v_j^{i+1}]$  then the statement is true. Now let  $v \in \tilde{A}_i - \bigcup_j N[v_j^{i+1}]$ . Then  $v \in A_i - \bigcup_j N[v_j^{i+1}]$  such that every path from  $u \in V - A_i$  to  $v$  contains a non-starting vertex in  $\bigcup_j N[v_j^{i+1}]$ .

Assume  $v \in R_{\{t_{i+1}\}}$ , then as  $A_i = A(t_{i+1}^-)$ , there exists  $u \in V - A_i$  such that there is an edge  $\{u, v\} \in E$ .

Therefore  $v \in \bigcup_j N[v_j^{i+1}]$  as  $v$  is the only non-starting vertex in this  $(u, v)$ -path. This is a contradiction to  $v \in \tilde{A}_i - \bigcup_j N[v_j^{i+1}]$ . Thus  $v \notin R_{\{t_{i+1}\}}$  and therefore  $v \in A(t_{i+1}^+)$ .

We still need to show that  $v \notin R_{(t_{i+1}, t_{i+2})}$ . Therefore, assume to the contrary that there exists a path  $P$  from  $u \in R_{\{t_{i+1}\}}$  to  $v$  not intersecting with  $\bigcup_j N[v_j^{i+1}]$ . Then there exists  $\tilde{u} \in V - A_i$  with  $\{u, \tilde{u}\} \in E$  (otherwise  $u$  would not be in  $R_{\{t_{i+1}\}}$ ). Then the path  $\tilde{P} = \tilde{u}P$  from  $\tilde{u}$  to  $v$  does not contain a non-starting vertex in  $\bigcup_j N[v_j^{i+1}]$  which is again a contradiction to the assumption that  $v \in \tilde{A}_i - \bigcup_j N[v_j^{i+1}]$ . Therefore  $v$  must *not* be in  $R_{(t_{i+1}, t_{i+2})}$  and we proved that  $A_{i+1} \subseteq A(t_{i+2}^-)$ .

Now let us argue the other inclusion.

“ $\supseteq$ ”: Let  $v \in A(t_{i+2}^-)$ . Then  $v \in A(t_{i+1}^+) - R_{(t_{i+1}, t_{i+2})}$ . We need to show that  $v \in A_{i+1} = \bigcup_j N[v_j^{i+1}] \cup \tilde{A}_i$ . Let us assume the contrary, then  $v \notin \bigcup_j N[v_j^{i+1}]$  and  $v \notin \tilde{A}_i$ . Thus  $v \in A(t_{i+1}^-) = A_i$  and  $v$  was recontaminated, i.e. there exists  $u \in V - A_i$  and a path  $P$  from  $u$  to  $v$  that does not contain a non-starting vertex in  $\bigcup_j N[v_j^{i+1}]$ . Then there exists  $\tilde{v}_2 \in P \cap (V - A_i)$  with  $d_G(\tilde{v}_2, A_i - \tilde{A}_i) = 1$  and there also exists  $\tilde{v}_1 \in P \cap (A_i - \tilde{A}_i)$  such that  $\{\tilde{v}_1, \tilde{v}_2\} \in E$ . Then  $\tilde{v}_1 \in R_{\{t_{i+1}\}}$  as  $A_i = A(t_{i+1}^-)$  and there exists a path from  $v$  to  $\tilde{v}_1$  not intersecting with  $\bigcup_j N[v_j^{i+1}]$ , in contradiction to  $v \notin R_{(t_{i+1}, t_{i+2})}$ .

Therefore,  $v$  must be in  $A(t_{i+2}^-)$  and we also proved  $A_i \supseteq A(t_{i+1}^-)$ . Together with  $A_i \subseteq A(t_{i+1}^-)$  we have the equality  $A_i = A(t_{i+1}^-)$  and the claim follows by induction.  $\square$

$\square$

On the other hand, we can also convert the discrete model to a continuous version.

**Theorem 2.4.** *Suppose  $\Pi$  is a winning  $k$ -watchman search program. Then there exist movement functions  $f_1, f_2, \dots, f_k$  such that for some time  $t < \infty$ ,  $A(t) = V$ .*

*Proof.* Assume we are given  $\Pi$  by  $\Pi = (W_0, A_0), (W_1, A_1), \dots, (W_r, A_r)$  with  $W_i = (v_1^i, v_2^i, \dots, v_k^i)$  for  $i = 0, 1, \dots, r$ . The corresponding continuous model can be constructed as follows.

Define the  $k$  watchmen's movement functions  $f_1, f_2, \dots, f_k : [0, \infty) \rightarrow V \cup E$  by

$$f_j(t) = \begin{cases} v_j^0 & t = 0 \\ v_j^i & t \in (i, i+1), \quad i = 0, 1, \dots, r-1 \\ v_j^r & t > r \\ \{v_j^{i-1}, v_j^i\} & t = i, \quad i = 1, 2, \dots, r \end{cases}$$

Then as above, it can be proved similarly that for every  $i \in \{0\} \cup \mathbf{Z}^+$ ,  $A_i = A((i+1)^-)$ . As  $\Pi$  is winning,  $A_r = V$  and so is  $A(r+1) = V$ .  $\square$

Hence, according to Theorem 2.2, 2.3 and 2.4,  $w(G) \leq k$  if and only if there exists a winning discrete  $k$ -watchman search program. We can now restrict our discussion to the discrete version and, in line with the continuous one, we denote the *watchman number of  $G$*  by  $w(G)$ , as the smallest integer  $k$  so that  $G$  has a winning  $k$ -watchman search program. A graph  $G$  is said to be  *$k$ -watchable* if there is a winning  $k$ -watchman search program for  $G$ .

We note that the discrete model can also be understood as a two-player game, with one player controlling the watchmen and the other an intruder. The contaminated set represents the possible positions that the intruder can move to following each watchman player's move. It is clear that in this formulation it does not matter whether the intruder is visible to the watchman player.

### 3 Some 1-watchable graphs

To determine which graphs are (1-)watchable, let us first note that not all vertices are significant in the sense that visiting them on the watchman's tour will be necessary. An easy example of this is a pendant vertex: if  $v$  is adjacent only to  $u$ , then there is no need to enter  $v$  since everything we could see from  $v$  we could see perfectly well from  $u$ . Contrariwise, it is necessary

that our walk enters  $u$  at some point to guard against an intruder hiding in  $v$ .

Let us codify this condition. If  $u$  and  $v$  are vertices such that  $N[v] \subset N[u]$ , then we shall call  $v$  *subordinate* to  $u$ . If  $N[u] = N[v]$ , then  $v$  is *interchangeable* with  $u$ . Note that subordination is a strict partial order on the vertices of  $G$ , while interchangeability is an equivalence relation. If  $v$  is subordinate to  $u$ , then as in the above example no walk needs to take the watchman to  $v$  because there's nothing to see there that can't be seen from  $u$ . By a similar line of reasoning, the watchman only needs to visit one of a set of mutually interchangeable vertices, since the view from any one of them will be identical to that from the others.

Therefore, we define the *dominant subgraph*  $G^*$  of a graph  $G$  as the result of the following two operations:

1. Let  $C_1, \dots, C_k$  be the equivalence classes of vertices under interchangeability. Delete all but a single representative of each of these classes. (Equivalently, apply a homomorphism that reduces each of the cliques  $C_i$  to a single vertex  $c_i$ , but that otherwise preserves adjacency and non-adjacency.)
2. If  $v$  is subordinate to any vertex  $u$ , then delete  $v$ .

(Note that these operations occur simultaneously; if  $u$  is subordinate to  $v$  in  $G$  and  $v$  is subordinate to  $w$  in  $G - u$  but not  $G$ , then  $G^*$  may still contain  $v$ .)

Then the above arguments show:

**Lemma 3.1.** *If  $G$  is a watchable graph, then there exists a walk function with its range restricted to  $G^*$ .*

For now, let us restrict our attention to the dominant subgraph. Note that it may be required that the watchman visits all of the vertices in  $G^*$ , since any one of them might have subordinate vertices in  $G$ .

There are two obvious scenarios that our watchman might wish to avoid, both staples of bedroom farces, Marx Brothers' movies, and the like: the run-around and the shell game.

*The run-around:* suppose that  $G^*$  contains an induced cycle of size 4, with vertices  $a, b, c, d$  in order around the cycle. If the watchman is standing at  $a$ , he cannot know whether there is an intruder at  $c$ ; he can determine this by moving to  $b$ , say. But doing so gives him no satisfaction, for while he can perceive that  $c$  is now empty, that might be because the intruder anticipated his movement and went to  $d$  at the exact same moment as the watchman's

move. Hence, having any such structure in  $G^*$  may be detrimental to the watchman's peace of mind. The same game can clearly be played on larger induced cycles. As such, we might wish to avoid graphs having induced cycles on four or more vertices: in other words, any graphs which are not chordal.

*The shell game:* suppose that  $x, y, z$  are mutually non-adjacent vertices of  $G^*$  with the property that there is a path between each pair of them that avoids the neighbourhood of the third. Then if the watchman finds himself at  $x$ , an intruder who had been hiding in  $y$  has the opportunity to move to  $z$ . If the watchman proceeds to  $y$ , then the intruder could slip behind his back, as it were, into  $x$ . The watchman therefore finds it difficult to convince himself that no intruder is present if such a triple exists. This structure is more usually called an *asteroidal triple*, and so we may prefer graphs which are "AT-free".

In the paper [6] in which they introduced the concept, Lekkerkerker and Boland showed that the interval graphs are precisely those graphs which are chordal and free of asteroidal triples. Therefore, given the above discussion the (connected) interval graphs seem to have no obvious barriers to watchability. In this case, our instincts are correct.

**Theorem 3.2.** *If  $G$  is a connected interval graph, then  $G$  is watchable.*

*Proof.* Let  $V(G) = \{v_1, \dots, v_n\}$  such that  $G$  admits an interval representation on  $(0, T)$  where the interval  $I_j = (l_j, u_j)$  corresponds to the vertex  $v_j$  and that  $l_i < l_j$  whenever  $i < j$ ; without loss of generality we assume that  $l_1 = 0$  and  $u_n = T$ . We assume that the sets  $U = \{u_i : 1 \leq i \leq n\}$  and  $L = \{l_i : 1 \leq i \leq n\}$  are disjoint and have cardinality  $n$ ; let  $B = (L \cup U) - \{0, T\}$ .

Define a function  $f : (0, T) - B \rightarrow V$  as follows:  $f(t) = v_j$  where  $j$  is the largest integer such that  $t \in I_j$ . We can extend  $f$  to a walk function  $f^*$  by defining  $f^*(t)$  for any  $t \in B$  to be the edge  $e_{ij}$ , where  $f(t - \epsilon) = v_i$  and  $f(t + \epsilon) = v_j$ . We claim that  $f^*$  is a watchman's tour of  $G$ .

To show this, suppose that there is an intruder in  $G$ . We can find an intruder function  $g : (0, T) \rightarrow V \cup E$  to model her travels in the graph. Let us call a continuous function  $h : (0, T) \rightarrow (0, T)$  *compatible* with  $g$  if the following conditions hold:

1. If  $g(t) = v_j$  then  $h(t) \in I_j$ .
2. If  $g(t) = e_{jk}$  then  $h(t) \in I_j \cap I_k$ .

Note that such a function can easily be constructed by choosing values for  $h$  at each edge-point in the interval (that is, at each point  $t$  where  $g(t)$  is

an edge) and interpolating with linear splines; for the outermost intervals, assume that the function  $h$  is constant. For simplicity, we may assume that  $h$  is defined over the closed interval  $[0, T]$  by letting  $h(0) = \lim_{t \rightarrow 0} h(t)$  and  $h(T) = \lim_{t \rightarrow T} h(t)$ .

Now clearly at the beginning of the time interval – that is, at  $t = \epsilon$  – the intruder cannot be in the neighbourhood of  $v_1$ ; this implies that any function  $h$  compatible with  $g$  must satisfy  $h(0) > 0$ . Likewise, at the end of the time interval the intruder is outside of the neighbourhood of  $v_n$ , requiring  $h(T) < T$ . By the Intermediate Value Theorem, there must be some  $t^* \in (0, T)$  such that  $h(t^*) = t^*$ . If both the watchman and the intruder are on vertices at time  $t^*$ , then by construction they must either be in the same vertex of adjacent vertices, since their corresponding intervals both contain  $t^*$  and hence have non-empty intersection. If one or the other (or both) are on edges at  $t^*$ , then for some  $\epsilon > 0$  we can see that at either  $t^* + \epsilon$  or  $t^* - \epsilon$  the watchman and intruder must be in identical or adjacent vertices.  $\square$

Note that the tour described in the above proof visits every vertex in the graph; this is actually excessive, since by Lemma 3.1 we could omit at least the two end-vertices in our interval representation (and possibly more). However, this construction together with Lemma 3.1 lets us expand the list of watchable graphs easily. Recall that the *corona* of a graph  $G$  is constructed by adjoining a pendant edge to every vertex of  $G$ . Let us generalize this as follows: a graph  $H$  is a *coronal graph* of  $G$  if it can be constructed from  $G$  and graphs  $H_1, \dots, H_n$  (where  $n = |V(G)|$ ; note that some or all of the  $H_i$ s may be empty) by joining each vertex  $v_i$  with the corresponding graph  $H_i$ . (In other words,  $v_i$  is adjacent to every vertex in  $H_i$ .) Clearly the dominant subgraph of  $H$  is an induced subgraph of  $G$ .

**Corollary 3.3.** *If  $G$  is a connected interval graph and  $H$  is a coronal graph of  $G$ , then  $H$  is watchable.*

*Proof.* A tour of  $G$  constructed by the method of the proof of the previous result will also serve as a watchman’s tour of  $H$ .  $\square$

A caterpillar may be defined as a coronal of a path, and a lobster as a coronal of a caterpillar.

**Corollary 3.4.** *A tree is watchable if and only if it is a lobster.*

*Proof.* One direction follows from the previous Corollary, together with the fact that caterpillars admit interval representations. Suppose that  $T$  is not a

lobster; let  $T^-$  represent the tree that results from deleting all of the degree 1 vertices in  $T$ . Then consider  $T^{--}$ ; since  $T$  is not a lobster,  $T^{--}$  is not a path and hence must have some vertex  $v_0$  of degree 3; call its neighbours  $v_1, v_2, v_3$ . Then in  $T$  we must be able to find vertices  $u_i$  that are distance two from each respective  $v_i$  such that the path from  $v_0$  to  $u_i$  passes through  $v_i$ . (If we could not find such a  $u_1$ , say, then  $v_1$  would be a leaf in either  $T$  or  $T^-$ .) When the watchman begins his tour, at least two of the  $u_i$ s are contaminated, say  $u_1$  and  $u_2$ . But checking either one for an intruder opens a path for the other to recontaminate  $u_3$ , leaving us in the same situation. Therefore such a  $T$  is not watchable.  $\square$

A subset of the authors had previously believed that a much broader version of Corollary 3.3 held true: namely, that a graph  $G$  was watchable if and only if its dominant subgraph admitted an interval representation. This is true in neither direction, as was first demonstrated in [3]; Figure ?? shows a watchable graph with a dominant subgraph that admits no interval representation, while Figure ?? shows an unwatchable graph with a interval graph for its dominant subgraph. As such, a complete characterization of the watchable graphs remains an open problem.

We now present a few remarks concerning the watchman number of a graph.

**Lemma 3.5.** *For any graph  $G$ ,  $w(G) \leq \gamma(G)$  where  $\gamma(G)$  denotes the domination number of  $G$ .*

*Proof.* Put a watchman on each vertex of a minimum dominating set, with a constant walk function.  $\square$

This suggests a refinement, in that we could separately consider sentries (fixed watchmen) and patrols (moving watchmen): a graph requiring one of each could be viewed as being “more watchable” than one requiring two patrols. (For instance, if the costs of maintaining a sentry are some fraction of those for maintaining a patrol, and our interest is in minimizing cost rather than the size of our force.)

**Lemma 3.6.** *Let  $G$  be a graph. For any  $v \in V(G)$ , if each component of  $G - N[v]$  is  $k$ -watchable, then  $w(G) \leq k + 1$ .*

*Proof.* By putting a sentry at  $v$ , any intruder would be confined to a single component of  $G - N[v]$ ; the remaining  $k$  watchmen can then attack each component serially.  $\square$

A circular-arc graph is an intersection graph obtained from a finite collection of arcs on a circle: Let  $G = (V, E)$  be a graph with  $V = \{v_1, v_2, \dots, v_n\}$  and  $C_i = (\theta_1(v_i), \theta_2(v_i))$  be the circular arc representing  $v_i$  starting at the angle  $\theta_1(v_i)$  and ending at  $\theta_2(v_i)$ , both angles measured counterclockwise from the positive  $x$ -axis. Then  $\{v_i, v_j\} \in E$  if and only if  $C_i \cap C_j \neq \emptyset$  for  $i \neq j$ . (See Figure ?? for an example.)

**Theorem 3.7.** *For every connected circular-arc graph  $G$ ,  $w(G) \leq 2$ .*

*Proof.* Let  $G = (V, E)$  be a connected circular-arc graph with arcs  $\{C_i\}$  representing the vertices  $V = \{v_1, v_2, \dots, v_n\}$ .

We first arbitrarily choose one vertex  $v_s \in V$  and place a *guard*, i.e. a watchman that does not move during the entire search process, on it. We can now restrict our search program to the vertices besides the closed neighborhood of  $v_s$ . Removing  $N[v_s]$  from  $V$  is like removing  $C_s$  together with all other arcs  $C_j$  that overlap with  $C_s$ , i.e.  $C_s \cap C_j \neq \emptyset$ . This results in a circular arc representation that has a hole in it and can be flattened onto a line, making it an (not necessarily connected) interval graph. The subgraph of  $G$  induced by  $V - N[v_s]$  consists of connected components  $G_1, G_2, \dots, G_r$  that are interval graphs.

Now we use another watchman clearing the individual interval graphs  $G_1, G_2, \dots, G_r$  subsequently using a walk constructed in Theorem 3.2. Note that all of those components are connected in  $G$  through  $N[v_s]$  as  $G$  is connected. Also note that they are connected *only* through  $N[v_s]$ , i.e. all paths from a vertex  $v_i \in G_i$  to another vertex  $v_j \in G_j$  ( $i \neq j$ ) contain a vertex in  $N[v_s]$ . Therefore after clearing a whole component  $G_i$ , none of its vertices can get recontaminated from a contaminated vertex of another component  $G_j$ ,  $j > i$  because the one guard is stationed at  $v_s$ . Finally we have to make sure that (ii) of the definition of the watchman program is not violated and we indeed have a walk on  $G$ . This is the case as the second watchman can move from one component  $G_i$  to another  $G_j$  using a path of vertices of  $N[v_s]$  as the subgraph induced by  $N[v_s]$  is connected.  $\square$

## 4 The watchman number of a tree

Let us turn our attention to trees. The following technical lemma will prove quite useful.

**Lemma 4.1.** *Given any graph  $G$  with dominant subgraph  $G^*$ , let  $S = V(G) - V(G^*)$  and let  $R \subseteq S$ . Then  $w(G - R) \leq w(G)$ .*

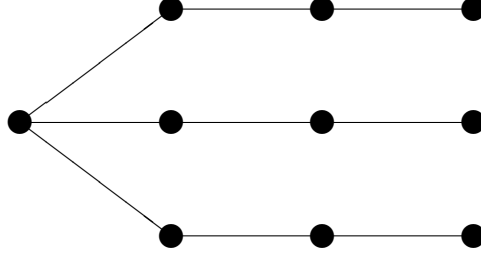


Figure 3: The smallest tree requiring two watchmen. Note that its dominant subgraph contains an asteroidal triple.

*Proof.* Suppose there exists a strategy for  $k$  watchmen to clear  $G$ . If a watchman following this strategy enters  $u \in R$ , then there is some  $v \in V(G - R)$  that they could use instead without altering the efficacy of the strategy. Thus,  $G - R$  must be  $k$ -watchable.  $\square$

**Corollary 4.2.** *If  $T$  is a tree with subtree  $T'$ , then  $w(T') \leq w(T)$ .*

*Proof.* We can build a sequence of trees  $T = T_0, T_1, \dots, T_k = T'$  such that for  $i > 0$ ,  $T_i = T_{i-1} - v_i$  where  $v_i$  is a leaf in  $T_{i-1}$ . Since no leaf in a tree is in that tree's dominant subgraph, we invoke Lemma 4.1 to find  $w(T_0) \geq w(T_1) \geq \dots \geq w(T_k)$ .  $\square$

Every caterpillar is an interval graph, and hence every lobster is 1-watchable. The smallest tree requiring two watchmen is depicted in Figure 3, and forms the basis for the following result.

**Theorem 4.3.** *For any positive integer  $k$ , there exist infinitely many trees  $T$  with  $w(T) = k$ .*

*Proof.* We proceed by recursive construction; certainly there are infinitely many 1-watchable trees. Let  $T$  be a tree with  $w(T) = k$  and let  $v \in V(T)$ ; we construct a tree  $T'$  requiring an extra watchman by taking three copies  $T_1, T_2, T_3$  of  $T$  and attaching to the image  $v_i$  of  $v$  in each  $T_i$  a distinct pendant vertex in a subdivided  $K_{1,3}$ . (See Figure 4.)

Since any one of  $T_1, T_2, T_3$  requires  $k$  watchmen to search it, clearly  $T'$  requires at least that many. Further, if  $T_1$  is being searched by  $k$  watchmen, then an intruder is free to move between  $T_2$  and  $T_3$ , and likewise if the searchers are in one of the other branches instead. Therefore  $k+1$  watchmen are necessary to search  $T'$ ; to show sufficiency, note that while  $k$  watchmen are in a given branch, say  $T_1$ , an additional watchman stationed at  $x_1$  will prevent an intruder in the  $T_2$ -branch from jumping to  $T_3$  or vice-versa.  $\square$

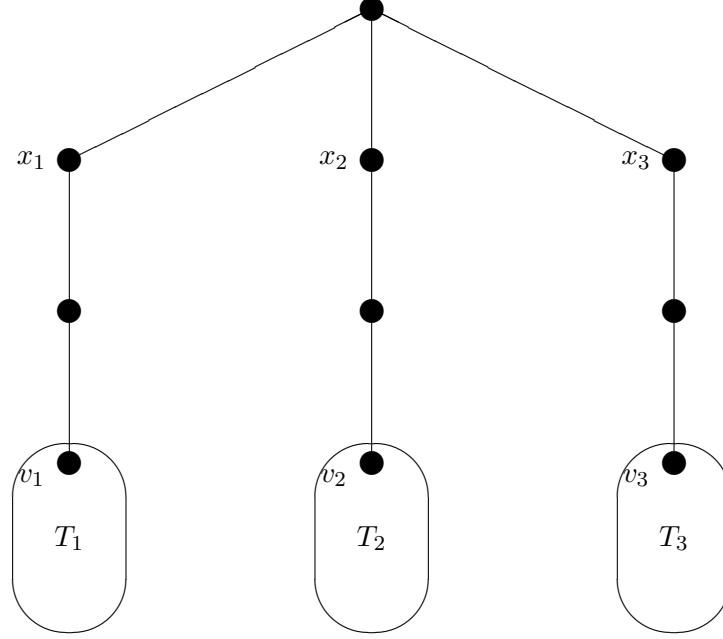


Figure 4: The construction of  $T'$  in Theorem 4.3.

**Theorem 4.4.** *If  $T$  is a tree of minimum order with watchman number  $k$ , then it must be of the form shown in Figure 4, with each  $T_i$  being a minimum-order tree with watchman number  $k - 1$ .*

*Proof.* Suppose that  $T$  is a tree with  $w(T) = k$ , and that among all trees with watchman number  $k$ ,  $T$  has the minimum order. Then for any  $v \in V(T)$ , each component  $C$  of  $T \setminus N^2[v]$  satisfies  $w(C) \leq k - 1$ .

First we show that there must be some  $v \in V(T)$  such that the graph  $T \setminus N^2[v]$  has exactly three components, each with watchman number  $k - 1$ . Suppose (to the contrary) that for every  $v \in V(T)$ , at most 2 of the components (call them  $A$  and  $B$ ) had watchman number  $k - 1$  (and the others have smaller watchman numbers). If this were the case, then we claim that the entire graph could be cleared by  $k - 1$  watchmen. Let vertices  $u$  and  $v$  be leaves in  $A$  and  $B$  respectively, and let  $L$  be the subgraph of  $T$  induced by all vertices distance 2 or less from the path from  $u$  to  $v$ . Since  $L$  is a lobster, its dominant subgraph is a caterpillar, and thus a single watchman could clear  $T[L]$ . In order to clear  $T$ , we use one watchman to guard  $L$  while additional watchmen serially clear the components  $C_1, \dots, C_\ell$  of  $T \setminus L$ . So  $w(T) \leq 1 + \max_{1 \leq i \leq \ell} \{C_i\}$ . Now, for each  $i$ , either  $C_i$  is one of the original

components of  $T \setminus N^2[v]$  in which case  $w(C_i) \leq k - 2$ , or  $C_i$  is a subgraph of  $A$  or  $B$ . If the latter is the case and  $w(C_i) = k - 1$ , then replacing  $A$  and  $B$  with a copies of  $C_i$  would yield a smaller tree with  $w(T) = k$ , contradicting the assumption that  $T$  is minimum. Thus for all  $i$ ,  $w(C_i) \leq k - 2$ , and so  $w(T) \leq k - 1$ . Therefore there is at least one  $v \in V(T)$  such that  $T \setminus N^2[v]$  has three or more components each with watchman number  $k - 1$ .

We now show that there is exactly one such vertex. Suppose to the contrary that  $u$  and  $v$  are both such vertices. Then  $T \setminus N^2[u]$  has at least three components with watchman number  $k - 1$ . Vertex  $v$  may be in one of these components, but we are guaranteed that there are at least two that do not contain  $v$ . Similarly, there are at least two components of  $T \setminus N^2[v]$  that do not contain  $u$ . Then a smaller graph with watchman number  $k$  can be constructed by amalgamating all ( $\geq 2$ ) vertices in the  $u - v$  path.

So there is exactly one vertex  $v \in V(T)$  such that  $T \setminus N^2[v]$  has three or more components, each with watchman number  $k - 1$ . This graph can be cleared by keeping one watchman in  $N^2[v]$  to act as both a guard to the entryway of each component and a “lookout”. For each component  $C$  of  $T \setminus N^2[v]$ , there is an induced  $P_4$  with one end  $v$  and the other end in  $C$ . Component  $C$  is cleared by moving the lookout into the neighbour of  $v$  along that  $P_4$ . From this point, the lookout can ensure that no intruder enters  $v$  and no intruder enters or leaves  $C$ . The remaining  $k - 1$  watchmen are able to clear  $C$ . The watchmen clear each component in this manner. The lookout is necessary if there are 3 or more such components (if there were only 2, he is not needed, as shown above), and since  $T$  is minimum, it follows that there are exactly three such components, and no smaller components.

Furthermore, the three components of  $T \setminus N^2[v]$  must have the same order. If they did not, then the larger components could be replaced with copies of the smallest one, producing a smaller tree with watchmen number  $k$  and contradicting  $T$  being minimum.

So  $T$  is of the form shown in figure 2, with each  $T_i$  being a minimum-order tree with watchman number  $k - 1$ .  $\square$

**Corollary 4.5.** *Let  $T$  be a tree on  $n$  vertices. Then  $w(T) \leq \log_3(2n + 7) - 1$ .*

Using a similar recursive construction, we are able to obtain an upper bound on  $w(T)$  as a function of the diameter of  $T$ , which we denote as  $d(T)$ .

**Theorem 4.6.** *Let  $T$  be a tree. Then  $w(T) \leq \lceil \log_2 \left( \frac{d(T) + 6}{5} \right) \rceil$*

*Proof.* Since a watchman at  $v$  can see everything in  $N^2[v]$  while keeping an eye on  $v$ , clearly  $w(T) = 1$  for any nonempty tree  $T$  with diameter at most 4.

Suppose that we have a collection of at least three trees  $T_1, \dots, T_k$ , each with watchman number  $w$ . We construct a larger tree  $T$  by joining them together as shown in Figure 5. So  $d(T) \leq 6 + 2 \max\{d(T_1), \dots, d(T_k)\}$ .

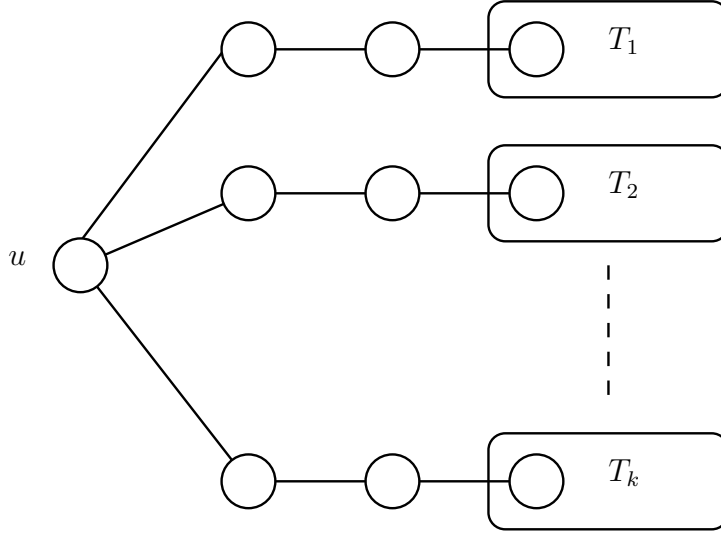


Figure 5: Schematic diagram of the diameter construction

$T$  cannot be watched by  $w$  watchmen, since each  $T_i$  requires  $w$  watchmen, and any three vertices from three distinct  $T_i$ s form an asteroidal triple. So  $d(T) \geq w+1$ . It turns out that  $T$  can be watched by  $w+1$  watchmen. A team of  $w$  watchmen serially clear each  $T_i$ , while the one additional watchman stands guard in the vertex adjacent to  $u$  that is nearest to  $T_i$ . Here he prevents any intruder from slipping through  $u$  or exiting  $T_i$ . So  $d(T) \leq w+1$ , and thus  $d(T) = w+1$ .

If we let  $d(w)$  be the maximum diameter of a tree that is  $w$ -watchable, we get the recurrence  $d(w) = 2d(w-1) + 6$ . Using the initial condition that  $d(1) = 4$ , we get the closed-form solution  $d(w) = 5 \cdot 2^w - 6$  and thus the bound  $w(T) \leq \lceil \log_2 \left( \frac{d(T)+6}{5} \right) \rceil$   $\square$

## 5 Searchers, cops, and watchmen

**Theorem 5.1.** *Let  $c(G)$  denote the cop number of a graph  $G$ . Then  $c(G) \leq w(G)$  for all graphs  $G$ , and there exist graphs  $G$  for which  $w(G) - c(G)$  is arbitrarily large.*

*Proof.* Suppose a strategy exists for  $k$  watchmen to search  $G$ , locating and capturing an intruder if present. This same strategy can be implemented by the cops in cops-and-robbers to capture the robber, since a strategy that works without knowing the robber's position will certainly still work when that position is known. Further, the robber's movements in cops-and-robbers have the same constraint as the intruder's in the watchman problem — both must stop upon entering the neighbourhood of a cop/watchman — and hence the robber's possible strategies are a subset of the intruder's.

To show the final proposition, note that every tree is cop-win but that there exist trees with arbitrarily high watchman number.  $\square$

**Corollary 5.2.** *Every watchable graph is cop-win.*

As an aside: given a graph  $G$ , consider the sequence of subgraphs  $G, G^*, G^{**}, \dots$  where each graph except for  $G$  itself is the dominant subgraph of its predecessor. Clearly this sequence must stabilize; let  $G^\infty$  (the “ultimate dominant subgraph”) denote the terminal graph in this sequence.

**Lemma 5.3.**  *$G$  is a cop-win graph if and only if  $G^\infty \cong K_1$ .*

*Proof.* We proceed by induction: suppose this holds for all graphs on  $n$  or fewer vertices, and let  $|V(G)| = n + 1$ . If  $G^* = G$  then  $G$  has no two vertices  $u, v$  such that  $N[u] \subseteq N[v]$  and hence no dismantling ordering is possible. Otherwise,  $G^*$  has at most  $n$  vertices, and hence is cop-win precisely if  $G^\infty = (G^*)^\infty \cong K_1$ . If  $G^*$  is cop-win and this isomorphism holds, then we can take a dismantling ordering of its vertices and extend it to an ordering of  $V(G)$  by prepending the vertices of  $G - G^*$ , ensuring that the subordinate vertices precede the equivalent vertices. Clearly  $G$  cannot be cop-win unless  $G^*$  is also, and hence if  $G^\infty = (G^*)^\infty$  is not a single vertex then  $G$  is not cop-win.  $\square$

Let us turn our attention to domination search, which also gives us a bound on the watchman number of a graph. A *domination search program*  $\Pi_{ds}$  on a graph  $G = (V, E)$  is a sequence of pairs (also considered as the *steps* of  $\Pi_{ds}$ )

$$\Pi_{ds} = (D_0, A_0), (D_1, A_1), \dots, (D_{2m-1}, A_{2m-1})$$

such that

1.  $D_i \subseteq V$  and  $A_i \subseteq V$  for all  $i = 0, 1, \dots, 2m - 1$  where  $D_i$  is the multiset of vertices containing a searcher, and  $A_i$  is the set of cleared vertices at step  $i$

2.  $D_0 = \emptyset, A_0 = \emptyset$
3. At the  $(2i - 1)$ -th step we place new searchers and clear vertices:  $D_{2i-2} \subset D_{2i-1}$  and  $A_{2i-1} = A_{2i-2} \cup N[D_{2i-1}]$  for every  $i = 1, 2, \dots, m$ .
4. At the  $2i$ -th step we remove searchers and possibly recontaminate vertices:  $D_{2i-1} \supset D_{2i}$  and  $A_{2i}$  is the subset of  $A_{2i-1}$  such that whenever  $v \in A_{2i}$  every path from  $v$  to a vertex from  $V - A_{2i-1}$  contains a vertex in  $N[D_{2i}]$ .

A domination search program is winning if all vertices are cleared in the end, i.e. if  $A_{2m-1} = V$ . The domination search number  $ds(G)$  is the minimum number  $k$  such that there exists a winning domination search program using at most  $k$  searchers at any given time.

**Theorem 5.4.** *For every connected graph  $G$ ,  $w(G) \leq ds(G)$ .*

*Proof.* Similarly to the previous theorem, suppose that  $G$  admits a domination search using  $k$  searchers. We encode this strategy as a sequence  $D_0, \dots, D_{2m+1}$  of subsets of  $V(G)$  where  $D_i$  represents the vertices containing searchers after  $i$  moves, with  $D_0 = \emptyset$ ; if  $j$  is odd, then  $D_{j-1}$  and  $D_{j+1}$  are subsets of  $D_j$ , and  $k = \max_i |D_i|$ .

Construct a strategy for the watchmen in  $G$  as follows: initially place watchmen on the vertices in  $D_1$ , with any additional watchmen (in case  $|D_1| < k$ ) being placed arbitrarily. We then keep the watchmen on vertices in  $D_2$  fixed, while moving the others until all of  $D_3$  is occupied. Carrying on in this fashion (moving from covering  $D_{2j-1}$  to  $D_{2j+1}$  while keeping watchmen in  $D_{2j}$  fixed) must eventually decontaminate  $G$ , since at the moment watchmen are covering  $D_{2j+1}$  there are no more vertices contaminated then there are at the corresponding moment in the domination search, and at the end of the domination search all vertices are uncontaminated.  $\square$

**Corollary 5.5.** *For every connected graph  $G$ ,  $c(G) \leq ds(G)$ .*

**Theorem 5.6.** *For every connected graph  $G$ ,  $ds(G) \leq w(G) + 1$ .*

*Proof.* Let  $G$  be a graph and  $\Pi = (W_0, C_0), (W_1, C_1), \dots, (W_r, C_r)$  be a winning  $k$ -watchman search program of length  $r$ . Without loss of generality let us assume that at every step, only one watchman is moving (if there were more than one watchmen moving, perform these movements in several consecutive steps individually). Consider the  $i$ -th step and assume that watchman  $j(i)$  is moving from vertex  $v_-^i := v_{j(i)}^{i-1}$  to  $v_+^i := v_{j(i)}^i$ ,

watchman search		$W_0$	$W_1$	$\dots$	$W_r$
domination search	$D_0$	$D_1, D_2$	$D_3, D_4$	$\dots$	$D_{2r+1}, D_{2r+2}$

Table 1: Domination search program  $\Pi_{ds}$  for  $\Pi$

i.e.  $W_i = (v_1^i, v_2^i, \dots, v_k^i) = (v_1^{i-1}, v_2^{i-1}, \dots, v_{j(i)}^i, \dots, v_k^{i-1})$  with  $W_{i-1} = (v_1^{i-1}, v_2^{i-1}, \dots, v_{j(i)}^{i-1}, \dots, v_k^{i-1})$  for  $i = 1, \dots, r$ . Construct a domination search program  $\Pi_{ds} = (D_0, A_0), (D_1, A_1), \dots, (D_{2r+2}, A_{2r+2})$  of length  $2r+2$  that corresponds with  $\Pi$  as shown in Table 1 with

1.  $D_0 = \emptyset, A_0 = \emptyset$
2.  $D_1 = D_2 = \{v_1^0, v_2^0, \dots, v_k^0\}$
3.  $D_{2i-1} = D_{2i-2} \cup \{v_+^{i-1}\}$  for  $i = 2, \dots, 2r$   
i.e. at every odd step add one searcher at the vertex that the corresponding watchman in  $\Pi$  is moving to
4.  $D_{2i} = D_{2i-1} - \{v_-^{i-1}\}$  for  $i = 2, \dots, 2r$   
i.e. at every even step remove one searcher from the vertex where the corresponding watchman in  $\Pi$  is leaving from

Now all we have to show is that  $\Pi_{ds}$  is a winning  $(k+1)$ -domination search program:

At step 1,  $k$  searchers are placed on vertices in  $G$ . None is removed at step 2 and after that, one searcher is added and removed alternately, so  $\max_i |D_i| = k+1$ .

**Claim 5.6.1.** *The set of cleared vertices in the  $k$ -watchmen search program after step  $i$  is a subset of the set of cleared vertices after step  $2i+2$  in the  $(k+1)$ -domination search program, i.e.  $C_i \subseteq A_{2i+2}$ , for every  $i = 0, \dots, r$ .*

We proceed by induction. First,  $C_0 = \bigcup_{j=1}^k N[v_j^0] = N[D_2] = A_2 \checkmark$ . Now assume  $C_i \subseteq A_{2i+2}$  for some  $i$ . Then

- $C_{i+1} = \bigcup_{j=1}^k N[v_j^{i+1}] \cup \tilde{C}_{i+1} = N[W_{i+1}] \cup \tilde{C}_{i+1}$  with  $\tilde{C}_{i+1}$  being the set of protected vertices at step  $i+1$
- $A_{2i+3} = A_{2i+2} \cup N[v_+^{i+1}]$
- $D_{2i+4} = D_{2(i+2)} = W_{i+1}$

- $A_{2i+4} \subseteq A_{2i+3}$  such that for every vertex  $v \in A_{2i+4}$  every path containing  $v$  and a vertex from  $V - A_{2i+3}$  contains a vertex from  $N[D_{2i+4}]$ .

Now let  $v \in C_{i+1}$ . If  $v \in N[W_{i+1}]$  then also  $v \in A_{2i+4}$  as the same vertices are occupied in both the watchman search program and in the domination search program in step  $i+1$  and  $2i+4$  respectively. But if  $v \notin N[W_{i+1}]$  then  $v$  had been protected, i.e.  $v \in C_i$  and every path from a contaminated vertex in  $V - C_i$  had a non-starting vertex in  $N[W_{i+1}]$ . As  $C_i \subseteq A_{2i+2}$ , we also have  $v \in A_{2i+2}$  and even  $v \in A_{2i+3}$  because  $v \notin N[W_{i+1}]$ . Now consider a path  $P$  from a contaminated vertex in  $V - A_{2i+3}$  to  $v$ . As  $V - A_{2i+3} \subseteq V - C_{i+1}$ ,  $P$  has a (non-starting) vertex in  $N[D_{2i+4}] = N[W_{i+1}]$ . Thus  $v \in A_{2i+4}$  and we have  $C_{i+1} \subseteq A_{2(i+1)+2}$ .  $\square$

## 6 Monotonicity

A discrete graph search program  $\Pi$  is called *monotone* if the sequence of cleared vertices  $A_0, A_1, \dots, A_r$  is monotone, i.e. if  $A_i \subseteq A_{i+1}$  for  $i = 0, \dots, r-1$ . In other words, recontamination does not occur.

For the edge search problem, LaPaugh [7], and later Bienstock and Seymour [2], showed that for every graph  $G$  with edge search number  $k$ , there is always a *monotone* strategy that clears the graph using  $k$  searchers. As every edge needs to be cleared only once, this implies that such a strategy requires  $\mathcal{O}(n)$  steps. This allowed Megiddo et al.[?] to prove that determining if the edge search number of a graph is at most a certain integer is NP-complete.

We can ask the same question for the paranoid watchman problem. Assume that  $G = (V, E)$  is  $k$ -watchable. *Is there a winning monotone  $k$ -watchmen search program?* The answer turns out to be *No*.

We demonstrate it using the example in Figure ???. In order to clear this graph using only one watchman, it is unavoidable to visit the central vertex 0, as all of the vertices  $a$ ,  $b$  and  $c$  need to be visited to clear the other vertices 1, 2 and 3. Afterwards we need to enter one of the branches, let's say, without loss of generality, we visit  $a$  after 0. Then the vertices  $b$  and  $c$  get recontaminated.

Figure 6 shows another example, where a winning 1-watchman search program is described in Table 2. In both cases, the recontamination of the vertices can be thought of as being “accidental”; at time  $t$  when they were cleared, the intent of the programs were to clear other nodes. The programs were going to clear them at some later time  $t' > t$ .

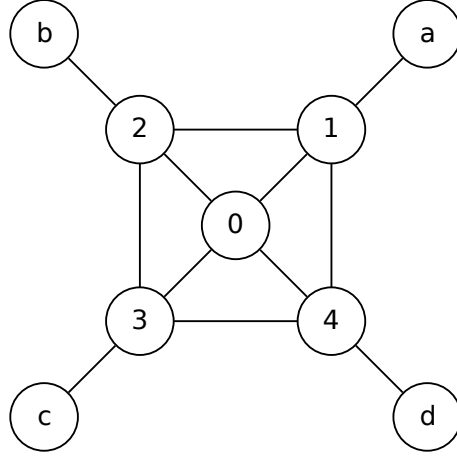


Figure 6: 1-watchable graph  $G$  where recontamination cannot be avoided

i	$W_i$	$A_i$	recontaminated
0	(1)	$\{0, 1, 2, 4, a\}$	$\{\}$
1	(0)	$\{0, 1, 2, 3, 4, a\}$	$\{\}$
2	(2)	$\{0, 1, 2, 3, a, b\}$	$\{4\}$
3	(0)	$\{0, 1, 2, 3, 4, a, b\}$	$\{\}$
4	(3)	$\{0, 1, 2, 3, 4, a, b, c\}$	$\{\}$
5	(4)	$\{0, 1, 2, 3, 4, a, b, c, d\}$	$\{\}$

Table 2: Winning watchman program for Figure 6

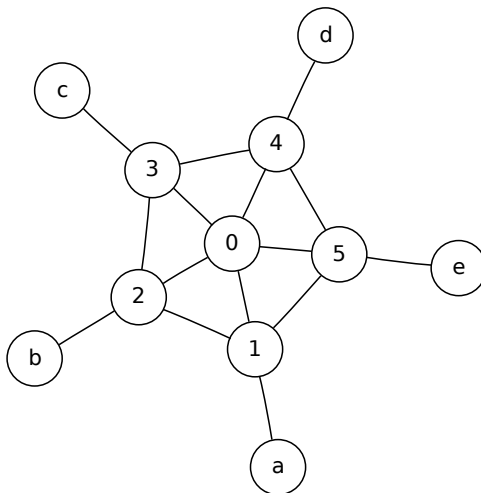


Figure 7: 1-watchable graph  $G$  where recontamination cannot be avoided

This is no longer the case if we use a 5-cycle instead of the 4-cycle as seen in Figure 7. All vertices on the cycle need to be visited since they all have adjacent leaves. Exploiting symmetry, we can assume that vertex 1 is visited before its non-adjacent vertices 3 and 4. Suppose 3 is visited after 1. At that point, 1 gets recontaminated because there is a path from 4 to 1 that does not intersect the neighborhood of 3. A possible winning 1-watchman search program is described in Table 3.

In domination search, all of the three examples above *can* be cleared using 2 searchers without causing any vertex to get recontaminated. Place a searcher on the central vertex, and use another one to clear the leaves. This does not mean that there is always a winning monotone  $ds(G)$ -domination search program as Fomin et al. [4] pointed out the example of the *Dobrev* graph as seen in Figure 8. This graph has domination search number 2 but it is not possible to use a winning *monotone* 2-domination search program.

i	$W_i$	$A_i$	recontaminated
0	(5)	$\{0, 1, 4, 5, e\}$	$\{\}$
1	(1)	$\{0, 1, 2, 5, a, e\}$	$\{4\}$
2	(0)	$\{0, 1, 2, 3, 4, 5, a, e\}$	$\{\}$
3	(3)	$\{0, 1, 2, 3, 4, 5, a, c, e\}$	$\{\}$
4	(4)	$\{0, 3, 4, 5, c, d, e\}$	$\{1, 2, a\}$
5	(0)	$\{0, 1, 2, 3, 4, 5, c, d, e\}$	$\{\}$
6	(2)	$\{0, 1, 2, 3, 4, 5, b, c, d, e\}$	$\{\}$
7	(1)	$\{0, 1, 2, 3, 4, 5, a, b, c, d, e\}$	$\{\}$

Table 3: Winning 1-watchman search program for Figure 7

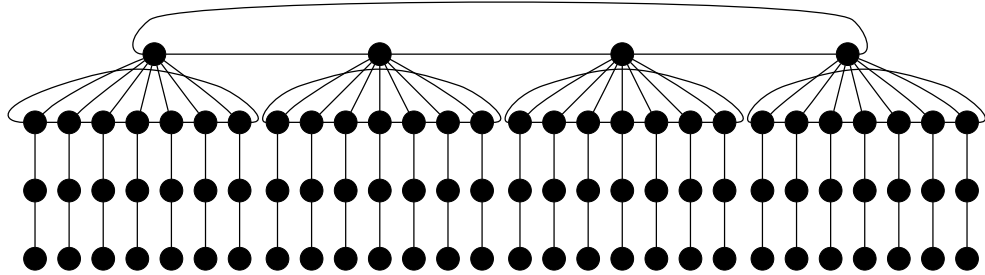


Figure 8: The Dobrev graph

## 7 Critiquing the model

As a model for real-world searching, there are several refinements or extensions that we might want to consider.

- *Windows and curtains.* We have assumed that visibility and accessibility are coincident, but this is not necessarily the case. A window might provide a vista into another region without allowing passage; conversely, a curtain between rooms might block sight-lines but still allow access. A more realistic model might allow representation of these features.
- *Mouseholes.* Similarly, one can imagine the watchmen and intruders having access to different sets of edges; if the “intruder” is a mouse or a squirrel that we want to catch, then we probably won’t be able to use the same passages that it could. (This can be combined with the previous refinement: we might not be able to follow a mouse into its hole, but we could still see or hear through such a hole.)
- *Entrances and exits.* We have assumed that the facility being guarded is effectively a closed system, with no way in or out. This is not typical of buildings, and hence we might want to designate certain vertices as entrances or exits, and require that these get special attention.
- *One-way doors.* We can translate the entire problem from graphs into digraphs, allowing some passages (a door that locks behind you) or vistas (a camera/screen combination) to only go in one direction.
- *Visibility.* One final variation would be to extend the watchman’s perceptions, so that an intruder within distance  $d$  of the watchman’s position will be spotted. The distance metric here could be the typical one in graphs, or for a more nuanced model we could assign weights to the edges of the graph and define the distance between two vertices as the minimum sum of weights on a path between them.

Any of these considerations could be used to motivate further development on this topic.

## References

- [1] B. Alspach, Searching and sweeping graphs: a brief survey. *Le Matematiche* 59 #1–2 (2004), 5–37.

- [2] D. Bienstock and P. Seymour, Monotonicity in graph searching. *J. Algorithms* 12 #2 (1991), 239–245.
- [3] D. W. Brugger, On the paranoid watchman problem. Masters Thesis, University of Wisconsin-Milwaukee, 2009.
- [4] F. V. Fomin, D. Kratsch, and H. Müller, On the domination search number. *Discrete Applied Mathematics* 127 #3 (2003), 565–580.
- [5] F. V. Fomin and D. Thilikos, An annotated bibliography on guaranteed graph-searching. Retrieved July 31st, 2008, from <http://www.ii.uib.no/~fomin/fedor/articles1/abgs.pdf>
- [6] C. G. Lekkerkerker and J. Ch. Boland, Representation of a finite graph by a set of intervals on the real line. *Polska Akademia Nauk. Fundamenta Mathematicae* 51 (1962-3), 45–64.
- [7] A. S. LaPaugh, Recontamination does not help to search a graph. *J. of the ACM* 40 #2 (1993), 224–245.
- [8] R. Nowakowski and P. Winkler, Vertex-to-vertex pursuit in a graph. *Discrete Mathematics* 43 (1983), 235–239.
- [9] T. D. Parsons, Pursuit-evasion in a graph. Theory and Applications in Graphs, Y. Alavi and D. R. Lick, eds., Springer Berlin (1976), 426–441.
- [10] T. D. Parsons, The search number of a connected graph. *Congressus Numerantium* 19 (1978), 549–554.
- [11] A. Quilliot, Thèse de 3<sup>o</sup> Cycle. Université de Paris VI, 1978.