LE MATEMATICHE
Vol. LIX (2004) – Fasc. I–II, pp. 5–37

SEARCHING AND SWEEPING GRAPHS: A BRIEF SURVEY

BRIAN ALSPACH

This papers surveys some of the work done on trying to capture an intruder in a graph. If the intruder may be located only at vertices, the term searching is employed. If the intruder may be located at vertices or along edges, the term sweeping is employed. There are a wide variety of applications for searching and sweeping. Old results, new results and active research directions are discussed.

1. Introduction.

For the most part, the graph theory terminology of [24] is employed. Nevertheless, for the sake of clarity, the following terminology is specifically mentioned. The term *reflexive multigraph* is used when both loops and multiple edges are allowed, the term *multigraph* means that multiple edges are allowed, and the term *graph* means that neither loops nor multiple edges are allowed. The term *valency* is used for the number of edges incident with a vertex, where a loop contributes two to the valency of the appropriate vertex. We use $\mathrm{val}(u)$ to denote the valency of the vertex u. An edge with end vertices u and v is denoted uv.

An edge has no direction so that reflexive multigraphs, multigraphs and graphs do not have *arcs* (directed edges). The appropriate terms when arcs are employed are *reflexive multidigraphs, multidigraphs*, and *digraphs*. An arc from

This research supported by NSERC under Grant A-4792 and by MITACS.

vertex u to vertex v is denoted (u, v). The terms corresponding to valency are *out-valency* and *in-valency*.

We discuss only finite graphs in this paper. Because of this convention, we shall omit the adjective finite, but it is always present and necessary in some of the proofs.

Graphs and digraphs, and their more general cousins, have become a standard modelling device for many applications. This survey paper deals with applications for which it is natural to employ some kind of graph or digraph as a model with an intruder present in the graph. The two main problems arising when employing this model are detecting and capturing the intruder. This survey deals with capturing an intruder.

There are two distinct contexts for the problem. If an intruder may be located only at vertices, then the process of attempting to capture an intruder is called *searching*. If an intruder may be located at vertices or along edges (or arcs), then the process of attempting to capture an intruder is called *sweeping*. In the case of sweeping, it is convenient to think of the graph or digraph as embedded in 3-space so that the points of the edges or arcs have geometric realizations. However, the algorithms actually used to sweep graphs and digraphs do not require that the graphs and digraphs be embedded in 3-space. Thus, even for sweeping it is possible to think of graphs and digraphs as abstractions.

The problem initially was motivated by the spelunking community and the first publication dealing with searching for someone lost in a cave system appears to be [7]. A mathematician at Pennsylvania State University, Tory Parsons, was approached by local spelunkers in the mid 1970s to see if he had any ideas about improving their searching techniques. Parsons immediately formulated the problem in terms of graph theory. He wrote two papers [19]. [20] about sweeping graphs. His work initiated the study of searching and sweeping graphs.

There are other problems, besides trying to find someone lost or hiding in a cave system, that are modelled naturally by sweeping graphs. Searching a road system for a vehicle that is either moving or parked somewhere is another such problem. Another problem that can be modelled as a graph sweeping problem, though it may not be obvious at first sight, is the problem of clearing a complex system of interconnected pipes that is contaminated by some noxious gas. The process of clearing the system of the noxious gas uses essentially the same algorithms used for sweeping a graph. So we may think of this as a graph sweeping problem. There are many similar problems that may be viewed in one of these two ways.

The connection between looking for an intruder who can hide at vertices or at a point in an edge, and clearing a graph of a noxious gas arises via sweeping algorithms. Even though one is searching for an intruder in that model, the sweepers conducting the sweep do so by clearing an edge at a time. This corresponds to expelling the contaminant along the edge. This will be made clear when the concept of *intruder territory* is defined.

There are problems for which searching a graph serves as a natural model. For example, if one is trying to find a piece of software in a computer network, one needs to check the computers corresponding to the vertices of the network. It does not make sense to think of the software as residing in the lines linking the computers together. There are other instances where search problems form the natural model.

Since this is intended to be a brief survey, proofs are omitted. We do provide a quick outline for proofs that are short.

2. Sweeping Models.

Almost all of the existing literature on sweeping deals with five sweeping models. We now describe these models. The first model is what Parsons originally presented.

Let X be a connected graph embedded in E^3 so that the vertices of X are distinct points in E^3, and each edge of X is represented by a line segment whose endpoints are the corresponding vertices. It is known this can be done so that the interiors of all line segments corresponding to edges are pairwise disjoint.

We let X inherit the usual topology of E^3. Imagine there may be an intruder in X who can be located at any point of X, that is, anywhere along an edge or at a vertex. We wish to capture any intruder or establish that the graph is free of an intruder. We describe what we mean by a general sweep strategy.

Definition 2.1. Let X be a connected graph embedded in E^3 as described above. For each positive integer k, let $\mathcal{C}_k(X)$ be the set of all families $F = \{f_1, f_2, \dots, f_k\}$ of continuous functions $f_i : [0, \infty) \to X$. A *general sweep strategy* for X is a family $F \in \mathcal{C}_k(X)$ such that for every continuous function $h : [0, \infty) \to X$, there is a $t_h \in [0, \infty)$ and $f_i \in F$ satisfying $h(t_h) = f_i(t_h)$. We say that f_i *captures* the intruder at time t_h.

There is an obvious extension of the preceding definition to reflexive multigraphs. The required alteration arises because multiple edges and loops cannot be embedded in E^3 as line segments. Loops are simply drawn as closed curves and multiple edges are drawn as internally disjoint simple curves. These can be done with no overlap of points other than at vertices.

Arbitrary continuous functions can exhibit unusual behavior making the notion of a general sweep strategy not always easy to work with. Consequently, we introduce a more discrete sweeping model that is considerably easier to work with. It involves placing some restriction on sweepers, but placing no restrictions on an intruder. We introduce discrete time intervals, and we lose no generality by assuming each time interval has length 1. The basic idea is that for each time interval, only one sweeper moves from a vertex to an adjacent vertex. The following definition formalizes the concept.

Definition 2.2. Let X be a connected graph embedded in E^3 as described earlier. A general sweep strategy $F = \{f_1, f_2, \dots, f_k\}$ of continuous functions in $\mathcal{C}_k(X)$ is simply called a *sweep strategy* if the following conditions are satisfied:

(i) $f_i(t) \in V(X)$ for every integer $t \in [0, \infty)$, $1 \leqq i \leqq k$;
(ii) there is an integer $\mathbb{N}(F) > 0$ such that $f_i(t)$ is constant for $t \in [\mathbb{N}(F), \infty)$, $1 \leqq i \leqq k$;
(iii) for every nonnegative integer $n < \mathbb{N}(F)$, there exists a unique $i_n \in \{1, 2, \dots, k\}$ such that for $i \neq i_n$, $1 \leqq i \leqq k$, f_i is constant on $[n, n+1]$, while f_{i_n} moves uniformly along the edge joining $f_{i_n}(n)$ to $f_{i_n}(n+1)$.

The next sweeping model allows sweepers to move in a discontinuous way while maintaining the restriction that an intruder moves according to some continuous function.

Definition 2.3. Let X be a connected graph embedded in E^3. A family $F = \{f_1, f_2, \dots, f_k\}$ of functions satisfying $f_i : [0, \infty) \to X$, $i = 1, 2, \dots, k$, is called a *wormhole* family if the following conditions are satisfied:

(i) $f_i(t) \in V(X)$ for every integer $t \in [0, \infty)$, $1 \leqq t \leqq k$;
(ii) there is an integer $N(F) > 0$ such that $f_i(t)$ is constant for $t \in [N(F), \infty)$, $1 \leqq i \leqq k$;
(iii) for every nonnegative integer $n < N(F)$, there exists a unique $i_n \in \{1, 2, \dots, k\}$ such that for $i \neq i_n$, $1 \leqq i \leqq k$, f_i is constant on $[n, n+1]$, while f_{i_n} either moves uniformly from $u = f_{i_n}(n)$ to a neighboring vertex v along the edge uv on the interval $[n, n+1]$, or is constant on the interval $[n, n+1)$ and then $f_{i_n}(n+1) = v \neq u$. We denote all the wormhole families on X with k functions by $\mathcal{W}_k(X)$. A *wormhole sweep strategy* for X is a family $F \in \mathcal{W}_k(X)$ such that for every continuous function $h : [0, \infty) \to X$, there is a $t_k \in [0, \infty)$ and $f_i \in F$ satisfying $h(t_k) = f_i(t_k)$. We say that f_i *captures* the intruder at time t_k.

It is easy to see that the collections of general sweep strategies and wormhole sweep strategies are not the same. In fact, the intersection of the two collections is the family of sweep strategies.

Another way to look at a wormhole sweep strategy is to observe that it is an extension of a sweep strategy in that sweepers are allowed an extra action. Sweepers are only allowed to traverse edges of the graph X from their present location to an adjacent vertex in sweeping, whereas, they are allowed to jump to any vertex of X in wormhole sweeping.

The last two sweeping models differ in that the notion of capturing an intruder is not the same. In the models already described, capture takes place only by having a sweeper occupy the same point occupied by the intruder at the same time.

Definition 2.4. Let X be a connected graph embedded in E^3. A family $F = \{f_1, f_2, \ldots, f_k\}$ of functions satisfying $f_i : [0, \infty) \to X$, $i = 1, 2, \ldots, k$, is called a *laser* family if the following conditions are satisfied:

(i) $f_i(t) \in V(X)$ for every integer $t \in [0, \infty)$, $1 \leqq t \leqq k$;
(ii) there is an integer $N(F) > 0$ such that $f_i(t)$ is constant for $t \in [N(F), \infty)$, $1 \leqq i \leqq k$;
(iii) for every nonnegative integer $n < N(F)$, there exists a unique $i_n \in \{1, 2, \ldots, k\}$ such that for $i \neq i_n$, $1 \leqq i \leqq k$, f_i is constant on $[n, n+1]$, while f_{i_n} is constant on the interval $[n, n+1)$ and then $f_{i_n}(n+1) = v \neq u$. We denote all the laser families on X with k functions by $\mathcal{L}_k(X)$. A *laser sweep strategy* for X is a family $F \in \mathcal{L}_k(X)$ such that for every continuous function $h : [0, \infty) \to X$, there is a $t_k \in [0, \infty)$ and $f_i, f_j \in F$ for which $h(t_k)$ lies in an edge joining $f_i(t_k)$ and $f_j(t_k)$. We say that the intruder is *captured* at time t_k.

Definition 2.5. A *mixed sweep strategy* is the same as a wormhole sweep strategy except that now capture takes place by either a sweeper and the intruder occupying the same point at the same time, as in earlier sweep strategies, or for two sweepers to occupy adjacent vertices u and v at the same time t, and for the intruder to be located on an edge joining u and v, including multiple edges, at time t.

Definition 2.6. We say that a reflexive multigraph X has been *cleared* following the conclusion of one of the various sweep strategies on X.

What we are calling wormhole sweeping has been called edge searching, our laser sweeping has been called node searching, and our mixed sweeping has been called mixed searching. We believe the distinction we make between sweeping and searching will lead to less confusion regarding terminology.

3. Sweep Numbers.

Once the various sweep models have been presented, it is natural to ask questions about the existence of sweep strategies, how many sweepers are needed to clear a graph, and relationships between the numbers of sweepers required to carry out various sweep strategies. The first result establishes the basis for the next sequence of definitions.

Theorem 3.1. *If X is a connected reflexive multigraph, then for each of the five sweeping models, there exists a sweep strategy using a finite number of sweepers.*

It is easy to prove this by stationing a sweeper on each vertex of X. Any intruder is immediately captured under laser sweeping and mixed sweeping. One additional sweeper can clear X.

Given a reflexive multigraph X, the preceding theorem tells us it makes sense to consider the minimum number of sweepers required to capture any intruder in X for each of the five sweeping models. The next theorem tells us, in fact, that the minimum number for general sweeping, sweeping and wormhole sweeping is the same. Parsons [20] promised to produce a proof for general sweeping and sweeping, but never published a proof. Subsequently, a proof has appeared in [2]. The next lemma [19], of interest in itself, is required for the proof of the theorem.

Lemma 3.2. *If $F = \{f_i : 1 \leqq i \leqq k\}$ is a general sweep strategy for the reflexive multigraph X, then there exists a constant C such that any intruder in X is captured at some time less than C.*

Theorem 3.3. *If X is a reflexive multigraph, then the minimum number of sweepers required to carry out a general sweep strategy, a sweep strategy, or a wormhole sweep strategy on X is the same.*

Theorem 3.3 justifies the next definition.

Definition 3.4. The *sweep number* of a connected reflexive multigraph X is the smallest k such that there exists a general sweep strategy, sweep strategy, or wormhole sweep strategy for X. The sweep number is denoted $\mathrm{sw}(X)$. We shall see later there are reasons to have a distinct notation for the wormhole sweep number and we use $\mathrm{wsw}(X)$.

Definition 3.5. The *laser sweep number* of a connected reflexive multigraph X is the smallest value of k such that there exists a laser sweep strategy for X. We denote the laser sweep number of X by $\ell\mathrm{sw}(X)$.

Definition 3.6. The *mixed sweep number* of a connected reflexive multigraph X is the smallest value of k such that there exists a mixed sweep strategy for X. We denote the mixed sweep number of X by $\mathrm{xsw}(X)$.

Theorem 3.7. *If X is a reflexive multigraph, then*
(i) $\mathrm{sw}(X) - 1 \leqq \ell\mathrm{sw}(X) \leqq \mathrm{sw}(X) + 1$,
(ii) $\mathrm{sw}(X) - 1 \leqq \mathrm{xsw}(X) \leqq \mathrm{sw}(X)$, *and*
(iii) $\ell\mathrm{sw}(X) - 1 \leqq \mathrm{xsw}(X) \leqq \ell\mathrm{sw}(X)$.

The preceding theorem is easy to prove so we take a look at the extremes. If P denotes a path of length at least 1, then $\mathrm{xsw}(P) = \mathrm{sw}(P) = 1$ but $\ell\mathrm{sw}(P) = 2$. If $3K_2$ denotes the multigraph of order 2 with an edge of multiplicity 3, then $\ell\mathrm{sw}(3K_2) = \mathrm{xsw}(3K_2) = 2$ and $\mathrm{sw}(3K_2) = 3$. For K_4, all of the values are the same. Thus, for each of the inequalities, there exists a multigraph for which equality holds.

4. Search Models.

We now move to searching as opposed to sweeping. Recall that the essential diference is that the intruder and the searchers may be located only at vertices.

Definition 4.1. Let X be a reflexive multigraph. There may be an intruder in the graph and his location may be any vertex. There is some number of searchers in the graph all of whom also must be located at vertices. The intruder and the searchers move according to some set of rules, but without exception a move is a movement from vertex to vertex (staying put is allowed). Capture takes place when a searcher and the intruder occupy the same vertex at the same time. The problem of attempting to capture the intruder is known as *searching* the multigraph X.

There are an immense variety of rules and restrictions we may put in place for searching a graph. The following model has been the most widely studied.

Definition 4.2. Let X be a reflexive multigraph. First distribute k searchers at the vertices of X, where more than one searcher may be located at a vertex. Then place the intruder at a vertex of X. The searchers and the intruder move at alternate ticks of a clock. Any subset of searchers may move at even values of t, and the intruder may move at any odd value of t. A move consists of going from a vertex to an adjacent vertex in X or staying at the current vertex. All participants have complete knowledge of the location of all other participants. This model is known as the *basic pursuit-evasion model with* *complete information*. We refer to this model as the BPE model. The *search number* of X, denoted $\text{sn}(X)$, is the smallest k such that k searchers can capture any intruder using the BPE model.

There is a vast gap between sweeping a graph and searching a graph using the BPE search model. We shall return to search numbers later in the paper, but include one result about search numbers in order to indicate the extent of the aforementioned gap. We do so using the family of trees. In the next section, we shall see that there are trees with arbitrarily large sweep number. The next result tells us that every tree has search number 1 under the BPE search model.

Theorem 4.3. *If T is a tree, then* $\text{sn}(T) = 1$.

The proof of this theorem follows easily by always having the searcher move towards the intruder along the unique shortest path between them. Since trees are finite, capture eventually takes place.

5. Intruder Territory.

There have been several rather colorful ways to describe the processes of searching and sweeping. Some authors have phrased all of their work in terms of cops and robbers, where the robber is, of course, an intruder, and the cops are the searchers or sweepers. As a matter of fact, the cops and robber terminology has been used so far in the literature only in the case of BPE searching, but I suspect it will spread to other models.

The language in this paper uses intruder, searcher, and sweeper. In this case, capture is thought of in terms of stumbling upon the intruder or two sweepers simultaneously seeing him in an edge.

Another way of looking at the problem involves thinking of edges in a graph as being contaminated. Instead of capturing an intruder, what we now want the sweepers to do is to remove the contamination along edges, and being finished when no more edges of the graph are contaminated. This fits precisely with the notion of sweeping an edge. One sweeper γ_1 is located at a vertex u and a second sweeper γ_2 traverses the edge uv from u to v. Upon reaching the vertex v, γ_2 either captures an intruder or forces an intruder to vacate the edge uv. What we know at this point is that the edge uv is not a possible location for an intruder. As long as γ_1 remains at u and γ_2 remains at v, an intruder cannot slip back into the edge uv.

Note that exactly the same process is involved when we think of this in terms of contamination. As γ_2 is traversing the edge from u to v, he is removing the contamination as he goes. Upon reaching the vertex v, the edge uv is no longer contaminated. As long as γ_1 and γ_2 remain at u and v, respectively, the edge uv cannot become recontaminated.

The notion that ties the intruder-sweeper view together with the contamination view is the following notion of intruder territory for sweeping.

Definition 5.1. Let X be a reflexive multigraph embedded in E^3. Suppose X is being swept according to some sweep strategy. At any given time t, let Y be the set of points of X where an intruder cannot be located assuming capture has not taken place. The subset $X \setminus Y$ is defined to be the *intruder territory* at time t. The set Y is called the *cleared set* at time t. Any edge uv entirely contained in Y is said to be *cleared* or *clear* depending on the context. Similarly, we say that a vertex v is *cleared* when all edges incident with v are cleared.

There is a corresponding notion of intruder territory for graph searching.

Definition 5.2. Let X be a reflexive multigraph. Suppose X is being searched according to some search strategy. At any given time t, let Y be the set of vertices of X where an intruder cannot be located assuming capture has not taken place. The subgraph of X induced by $V(X) \setminus Y$ is defined to be the *intruder territory* at time t. The subgraph induced by X on Y is called the *cleared subgraph* at time t.

Note the difference in defining the intruder territory for sweeping and searching. When a reflexive multigraph is being swept, sweepers may be located in the interior of edges. This implies that segments of an edge may be free of an intruder. So it makes sense to talk about the set of points that are cleared. On the other hand, when searching a graph, the intruder and the searchers are located at vertices. The role of the edges is to establish which vertices are allowed as the next location of a searcher or the intruder on a given move. Thus, it makes sense to talk about induced subgraphs.

In terms of the definition of intruder territory for sweeping, the intruder territory consists of the points of the multigraph where the intruder may be located. On the other hand, this is the same set as the set of points that are still contaminated. Hence, if one prefers thinking in terms of contamination, then the portion of the multigraph that is still contaminated is precisely the same as the intruder territory.

Suppose a sweeper γ is located on a vertex u, where there are cleared edges incident with u and there are contaminated edges incident with u. If γ jumps to another vertex in the multigraph and there are no other sweepers left on u, then it is clear that an intruder hiding in one of the contaminated edges incident with u may now move into any edge incident with u. In other words, all of the edges incident with u now become part of the intruder territory. In the language of contamination, we say that any cleared edge incident with u becomes recontaminated. We should mention, that if e is a cleared edge incident with u and there is another sweeper located at an interior point of e, then only the half-open segment of e from u to the interior point becomes recontaminated.

The material in this brief section allows us to jump back and forth freely from intruder terminology to contamination terminology.

6. Sweeping Trees.

After seeing Theorem 4.3, it may come as a bit of surprise that sweeping trees is so much more complicated. The principle reasons for the difference are lack of information about the intruder's location, and the freedom of movement the intruder has in sweeping models. The next lemma plays a key role in the development of sweeping trees.

Lemma 6.1. *Let T_1, T_2, and T_3 be vertex-disjoint trees each having at least one edge, and let v_j be a vertex of valency 1 in T_j, $j = 1, 2, 3$. Let T be the tree obtained by identifying the vertices v_1, v_2, v_3 as a single vertex v. If* $\mathrm{sw}(T_j) = k$, $j = 1, 2, 3$, *then* $\mathrm{sw}(T) = k + 1$.

Here are the essential ideas for proving this useful result. First, note that there is a sweep strategy for $k + 1$ sweepers. Station a single sweeper at vertex v and use k sweepers to clear each of the three subtrees one at a time. The sweeper stationed at v forces any intruder to stay in the subtree where he is initially located.

The harder part of the proof is to show that there is no sweep strategy for k sweepers. The point is that in order to clear any one of the subtrees, all k sweepers must be in the subtree away from the vertex v. At such a time, an intruder can move freely back and forth between the other subtrees. Thus, k sweepers have no way of clearing a subtree and making certain it remains cleared when they move into another subtree.

Lemma 6.1 can be used to produce the following upper bound on the sweep number of a tree.

Lemma 6.2. *If T is a tree of order n, then the sweep number satisfies*

$$\mathrm{sw}(T) \leqq 1 + \log_3(n - 1).$$

We are going to give an important theorem concerning the complexity of sweeping a tree and provide no proof. However, we shall present some structural aspects of trees that are used to prove the theorem. Recall that a branch of a tree

T at vertex v is a subtree obtained by taking one of the components arising from the deletion of v and reattaching v and the edge from v to the component . Let $B_T(vu)$ denote the branch of T at vertex v containing the edge $vu \in E(T)$.

Definition 6.3. Let T be a tree. A vertex $v \in V(T)$ is called a *hub* if all branches of T at v have sweep number less than $\mathrm{sw}(T)$. A path $v_1 v_2 \cdots v_t$, $t > 1$, of T is called an *avenue* if each of the two terminal vertices has exactly one branch with sweep number $\mathrm{sw}(T)$ and each interior vertex v_i, $1 < i < t$, has exactly two branches with sweep number $\mathrm{sw}(T)$. In fact, this implies $\mathrm{sw}(B_T(v_1 v_2)) = \mathrm{sw}(B_T(v_t v_{t-1})) = \mathrm{sw}(T)$, and $\mathrm{sw}(B_T(v_i v_{i-1})) = \mathrm{sw}(B_T(v_i v_{i+1})) = \mathrm{sw}(T)$, $1 < i < t$.

If T has a hub, we can place one sweeper on the hub and use $\mathrm{sw}(T) - 1$ sweepers to clear all branches at the hub. Similarly, if T has an avenue $v_1 v_2 ... v_t$, we can place one sweeper on v_1 and use $\mathrm{sw}(T) - 1$ sweepers to clear all the non-avenue branches at v_1. We then move all sweepers along $v_1 v_2$ to v_2. Repeating this process at v_2, we can clear the avenue and all branches at the vertices of the avenue. We conclude that avenues and hubs are important structure in trees as far as sweeping is concerned. The next result tells us that they also are plentiful. This then gives us a foundation for determining an algorithm to sweep trees.

Theorem 6.4. *Every tree has either a hub or a unique avenue, but not both.*

We use the divide-and-conquer method to find the sweep number of a tree T. The idea is to divide T into two subtrees of smaller order, recursively compute the sweep number and certain corresponding information for each subtree, and then merge this information for the subtrees to produce a solution for T. Each time we divide a tree into two subtrees, the two subtrees share only one common vertex in $V(T)$. Some of the information we keep is about hubs and avenues in the various trees that arise.

Using the above ideas, one can prove the following theorem. This result was first given in [15].

Theorem 6.5. *The sweep number of a tree can be computed in linear time.*

Knowing the number of sweepers required to sweep a tree depends on cretain structural features of the tree, and this is what is used to determine the sweep number. However, this is not the same problem as actually producing a strategy to capture any intruder in the tree. The following two results give us information about what is required in terms of producing a strategy. Some of this was discussed in [15].

Theorem 6.6. *Under the wormhole sweep model, a sweep strategy for a tree T, using $\mathrm{sw}(T)$ sweepers, can be computed in $O(n \log n)$ time.*

Theorem 6.7. *Under the sweep model, a sweep strategy for a tree T, using* $\mathrm{sw}(T)$ *sweepers, can be computed in $O(n^2 \log n)$ time.*

7. Complexity of Sweeping.

Earlier we defined cleared edges in the context of intruder territory. There is a natural intuitive notion of clearing an edge when discussing sweeping. Both sweeping and wormhole sweeping lead to simple interpretations because in both of these models, there is a single sweeper who moves for a given unit interval. The move itself consists of a sweeper traversing an edge from one end vertex u to the other end vertex v. If there is no way for the intruder to gain access to the vertex u while a sweeper is moving along the edge from u to v, upon reaching the vertex v, we know the intruder is not located at any point of the edge uv. For the other sweeping models, there are times for which it is impossible for the intruder to be located at any point of uv. This is how to view what it means by an to edge be cleared.

Definition 7.2. Given a set E of edges in a graph X, a vertex u of X is said to be *exposed* with respect to E if u is incident with at least one edge from E and at least one edge not in E. The set of exposed vertices with respect to E is denoted $\mathrm{exv}(E)$.

If E is the set of cleared edges at some point in the midst of sweeping a graph, then the set of exposed vertices with respect to E is a lower bound on the sweep number. There must be a sweeper stationed on each vertex of $\mathrm{exv}(E)$. There are situations for which it is desirable to not allow edges to become recontaminated once they have been cleared. We have a special name for sweeps in which edges do not become recontaminated.

Definition 7.2. Let $\mathcal{I}(n)$ denote the intruder territory at time n for a given sweep strategy on a graph X. We say that a sweep strategy, wormhole sweep strategy, laser sweep strategy or mixed sweep strategy is *monotonic* if $\mathcal{I}(j) \subseteq \mathcal{I}(j+1)$ for all integers j over the length of the sweep strategy.

A convenient way of thinking about a monotonic sweep strategy is to say that once an edge is cleared it remains clear. That is, recontamination never occurs. An important step in examining the complexity of sweeping is establishing that there are monotonic sweep strategies with the minimum number of sweepers. This tells us that each edge needs to be cleared only once. This result was proved first in [13] and an alternate proof was given in [6]. The proof in [13] is for wormhole sweeping and the elegant proof in [6] is for mixed sweeping.

Theorem 7.3. *If the sweep number of a connected multigraph X is k, then there is a monotonic wormhole sweep strategy using k sweepers.*

Since determining the sweep number of a multigraph X is an optimization problem, in that we are looking for the smallest k such that k sweepers can clear X, we look at a corresponding decision problem when discussing complexity. Accordingly, we make the following definition.

Definition 7.4. A graph X is k-sweepable if $\mathrm{sw}(X) \leqq k$.

The corresponding decision problem is to determine whether X is k-sweepable.

Problem: Wormhole Sweeping
Instance: Multigraph $X = (V(X), E(X))$, positive integer k.
Question: Is X k-sweepable under the wormhole sweeping model?

Theorem 7.3 implies that the Wormhole Sweeping Problem is in NP. The problem was proved to be NP-complete [15] by using a reduction from the following problem which is known to be NP-complete [11].

Problem: Min-Cut into Equal-Cardinality Subsets
Instance: Graph $X = (V(X), E(X))$ of even order, positive integer k.
Question: Is there a partition of $V(X)$ into two subsets V_1 and V_2 with $|V_1| = |V_2| = |V(X)|/2$ such that $|\{uv \in E(X) : u \in V_1, v \in V_2\}| \leqq k$?

Theorem 7.5. *The Wormhole Sweeping problem is NP-complete.*

The complexity of sweeping and wormhole sweeping is the same. The complexity of wormhole sweeping depends heavily on Theorem 7.3, but this theorem is not valid for sweeping. There are graphs with sweep number k for which there is no monotonic sweep with k sweepers. We shall discuss this later.

8. Bounds.

In the cases for which we actually want to determine the sweep number of a multigraph X, a method that sometimes works is to show how to clear X with k sweepers and then show that $\mathrm{sw}(X) > k-1$. The latter task typically is difficult because, in general, we do not have effective good lower bounds on $\mathrm{sw}(X)$. We now consider lower bounds for graphs with special structure. Some of them may not seem particularly meaningful, but it turns out that they are useful for graphs that are constructed in special ways to answer particular questions.

Proposition 8.1. *If the graph X has minimum valency $k \geqq 3$, then*

$$\mathrm{sw}(X) \geqq k + 1.$$

We use Theorem 7.3 to prove this result, by assuming there is a monotonic wormhole sweep strategy for X using k sweepers. One considers the first cleared vertex and how this forces the sweepers to be deployed. It quickly leads to a contradiction. It is not difficult to produce a strategy for $k+1$ sweepers that works.

Note that Proposition 8.1 is stated for graphs and not multigraphs. The simple explanation for this is that the proposition is not true for multigraphs. The multigraph $3K_2$ has minimum valency 3 but has sweep number 3. Whether Proposition 8.1 is true for multigraphs for which every edge has multiplicity at most 2 is an open problem.

The next result is an immediate corollary of Proposition 8.1.

Corollary 8.2. *If the graph X is k-connected and $k \geqq 3$, then* $\mathrm{sw}(X) \geqq k + 1$.

The following result also is useful for some situations. The subsequent result is then a corollary of Proposition 8.1 and Theorem 8.3.

Theorem 8.3. *If the graph Y is a minor of the graph X, then* $\mathrm{sw}(Y) \leqq \mathrm{sw}(X)$.

Corollary 8.4. *If the graph X has clique number $k \geqq 4$, then* $\mathrm{sw}(X) \geqq k$. *In particular,* $\mathrm{sw}(K_n) = n$ *for* $n \geqq 4$.

Definition 8.5. Let X be a multigraph with vertex set V and edge set E. A *path-decomposition* of X is a sequence $V_1, V_2, \ldots, V_t$ of subsets of V such that $V_1 \cup V_2 \cup \cdots \cup V_t = V$, every edge of X appears in some $\langle V_i \rangle$, for every vertex $u \in V$, the set $\{i : v \in V_i\}$ form a segment in $\{1, 2, \ldots, t\}$. The *width* of a path-decomposition is $\max_i\{|V_i| - 1\}$, $1 \leqq i \leqq t$. The *pathwidth* of X is the smallest width taken over all path-decompositions of X.

Theorem 8.6. *If X is a multigraph of pathwidth k, then $k - 1 \leqq \mathrm{sw}(X) \leqq k + 1$.*

9. Graphs with Small Sweep Number.

In this section we describe the reflexive multigraphs whose sweep numbers are either 1, 2 or 3. The description involves homeomorphic reductions. We next give two definitions to make it perfectly clear what we mean by homeomorphically reducing a reflexive multigraph. Homeomorphic reduction is an operation that potentially takes us out of the class of graphs. That is, there are graphs whose homeomorphic reduction has both loops and multiple edges.

Definition 9.1. Let X be a reflexive multigraph. Let $V' = \{u \in V(X) : \text{val}(u) \neq 2\}$. A *suspended path* in X is a path of length at least 2 joining two vertices of V' such that all internal vertices of the path have valency 2. A *suspended cycle* in X is a cycle of length at least 2 such that exactly one vertex of the cycle is in V' and all other vertices have valency 2.

Definition 9.2. Let X be a reflexive multigraph. Let $V' = \{u \in V(X) : \text{val}(u) \neq 2\}$. The *homeomorphic reduction* of X is the reflexive multigraph X' obtained from X with vertex set V' and the following edges. Any loop of X incident with a vertex of V' is a loop of X' incident with the same vertex. Any edge of X joining two vertices of V' is an edge of X' joining the same two vertices. Any suspended path of X joining two vertices of V' is replaced by a single edge in X' joining the same two vertices. Any suspended cycle of X containing a vertex u of V', is replaced by a loop in X' incident with u. Finally, any suspended cycle of X that has only vertices of valency 2 is a component of X containing no vertices of V'. We replace this cycle by a new vertex in X' with a single isolated loop incident with the vertex.

It is not difficult to prove that the homeomorphic reduction of a reflexive multigraph is unique to within isomorphism.

Lemma 9.3. *If X is a graph and Y is its homeomorphic reduction, then* $\text{sw}(X) = \text{sw}(Y)$.

This result is proved by showing how to go from a wormhole sweep strategy on X with $\text{sw}(X)$ sweepers to a wormhole sweep strategy on Y with the same number of sweepers, and vice versa.

It is easy to see how to prove the next result.

Theorem 9.4. *A multigraph X is 1-sweepable if and only if X is a path.*

Theorem 9,5. *A reflexive multigraph X is 2-sweepable if and only if its homeomorphic reduction consists of a path $u_1u_2 \ldots u_n$ such that there are an arbitrary number of loops incident with u_1 and u_n, and an arbitrary number of loops and pendant edges incident with each vertex u_i, $2 \leqq i \leqq n-1$, and the multiplicity of any non-loop edge is at most 2.*

The preceding theorem is fairly easy to prove. If we restrict ourselves to the class of graphs, then we can see how to get 2-sweepable graphs from Theorem 9.5. Any loops must be subdivided at least twice so that they become cycles with one vertex of valency bigger than 2. Whenever we find an edge of multiplicity 2, at least one of the edges must be subdivided at least once in order to remove multiple edges thereby producing a graph.

The description of 3-sweepable multigraphs requires the introduction of a special family of multigraphs we now describe. Recall that an *outerplanar* multigraph X is a multigraph that can be embedded in the plane so that every vertex of X lies in the boundary of the infinite face. In other words, the boundary of the infinite face is a Hamilton cycle C of X. All edges of X not in C are chords of C lying in the interior of the region bounded by C.

Definition 9.6. Let X be an outerplanar graph embedded in the plane so that C is the Hamilton cycle bounding the infinite face. As we traverse the cycle C clockwise, let $u_1u_2\dots u_p$ be a subpath of C. We say that the chord u_iu_j, $1 \leqq i < j \leqq p$, spans each edge in the subpath $u_iu_{i+1}\dots u_j$. Two such chords are *nested* if there is an edge of $u_1u_2\dots u_p$ spanned by both of them. We say that two boundary edges e_1 and e_2 are *opposing poles* if neither of the two subpaths comprising $E(C) \setminus \{e_1, e_2\}$ has a pair of nested chords. We allow either one of the two opposing poles to be a single vertex. If an outerplanar embedding of X has a pair of opposing poles, then we say that X is *bipolar*.

The Figure 1 shows a bipolar outerplanar graph with opposing poles u_1v_1 and u_7v_5.

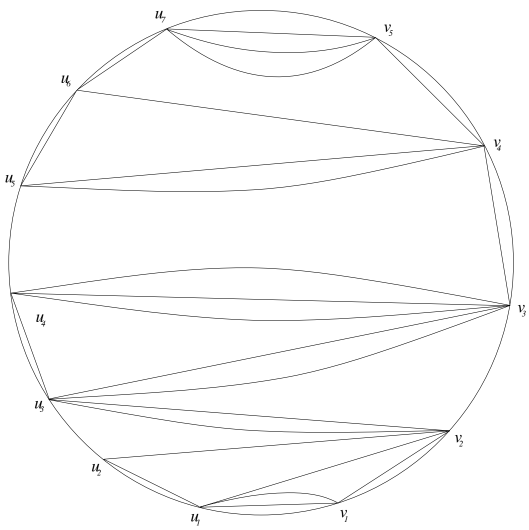

Figure 1: A bipolar outerplanar graph

Theorem 9.7. *A homeomorphically reduced, 2-connected reflexive multigraph X is 3-sweepable if and only if X is outerplanar and bipolar.*

A description of the 3-sweepable reflexive multigraphs is going to involve ways for combining multigraphs, with bipolar outerplanar graphs playing a basic role. We are going to use 1- and 2-sweepable reflexive multigraphs, and 2-connected 3-sweepable reflexive multigraphs as our basic building blocks. There is a common method for combining multigraphs that appears now and we describe it.

Definition 9.8. Let X_1 and X_2 be vertex-disjoint reflexive multigraphs. Choose a vertex $u_1 \in V(X_1)$ and $u_2 \in V(X_2)$. The *amalgamation* of X_1 and X_2 by identifying u_1 and u_2 is the graph X we obtain in the following way. The vertex set $V(X)$ consists of $(V(X_1) \setminus \{u_1\}) \cup (V(X_2) \setminus \{u_2\})$ together with a new vertex u. The edge set $E(X)$ consists of the edges of X_1 joining two vertices of $V(X_1) \setminus \{u_1\}$, the edges of X_2 joining two vertices of $V(X_2) \setminus \{u_2\}$, together with an edge from u to any vertex that was adjacent to either u_1 or u_2 in X_1 or X_2, respectively.

We also may amalgamate three or more vertex-disjoint graphs by identifying them at a single vertex chosen from each. It is done in the obvious way suggested by the amalgamation of two graphs as described in the preceding paragraph. (One may also amalgamate along subgraphs bigger than a single vertex, but we are not concerned with that in this material.)

Proposition 9.9. *Let X be the amalgamation of reflexive multigraphs X_1, $X_2, \ldots, X_t$ by identification of a single vertex from each X_i. If X_i is 2-sweepable for $1 \leqq i \leqq t$, then X is 3-sweepable.*

The proof of Proposition 9.9 follows from stationing a single sweeper at the amalgamated vertex of X which allows two sweepers to clear X.

Consider a block B of a homeomorphically reduced 3-sweepable reflexive multigraph X. The block B must be 3-sweepable, but B itself need not be homeomorphically reduced since it may contain 2-valent vertices belonging to other blocks of X. It follows from Theorem 9.7 that when B is homeomorphically reduced, the resulting reflexive multigraph B' must be outerplanar and bipolar. So we need to extend the definitions of boundary edges and opposing poles from B' to B itself. We say that an edge e of B *reduces* to an edge e' of B' if $e = e'$ or e is in a suspended path of B which becomes the edge e' when B is homeomorphically reduced to B'. Given that e of B reduces to e' of B', we say that e is a *boundary edge* of B if e' is a boundary edge of B', and we say that two boundary edges of B are *opposing poles* for B if they reduce to opposing poles for B'. If uu' and vv' are a pair of opposing poles for B, then u and v are called *opposing*l vertices of B.

For purposes of discussions involving blocks, we consider a loop to be a block by itself.

Definition 9.10. Let X be a connected, homeomorphically reduced, reflexive 3-sweepable multigraph consisting of 3-sweepable blocks $B_1, B_2, ..., B_r$ such that the block-cut-graph of X is a path. By the definition of block-cut-graph, this means we can write the blocks and cut-vertices as a sequence

$$B_1, a_1, B_2, a_2, \ldots, B_{r-1}, a_{r-1}, B_r$$

so that $a_1, a_2, ..., a_r$ are distinct vertices of X, where $a_j \in V(B_j) \cap V(B_{j+1})$, $1 \leqq j \leqq r-1$. If, in addition, a_j and a_{j+1}, $0 \leqq j \leqq r-1$, are opposing vertices for B_{j+1}, we call $C = (a_0, B_1, a_1, B_2, ..., a_{r-1}, B_r, a_r)$ a 3-*chain* for X.

Definition 9.11. Let X be a connected, homeomorphically reduced, reflexive 3-sweepable multigraph containing a 3-chain of the form

$$C = (a_0, B_1, a_1, B_2, ..., a_{r-1}, B_r, a_r).$$

A *valid set of opposing poles* for C is any sequence of edges

$$a_0x_1, a_1y_1, a_1x_2, a_2y_2, a_2x_3, ..., a_r y_r$$

such that $a_{j-1}x_j, a_j y_j$, $1 \leqq j \leqq r$, are opposing poles for B_j. Define $N(C) = \{x_1, y_1, x_2, y_2, ..., x_r, y_r\}$, $A(C) = \{a_1, a_2, ..., a_{r-1}\}$ and $V(C) = \cup\{V(B_j) : 1 \leqq j \leqq r\}$. For any block B_j and its specified opposing poles $a_{j-1}x_j$ and $a_j y_j$, let P_j be either of the two boundary paths obtained by removing the edges $a_{j-1}x_j$ and $a_j y_j$, and let H_{P_j} denote the subgraph of B_j consisting of P_j and all edges of B_j that reduce to chords whose end vertices belong to P_j when B_j is homeomorphically reduced. We call a vertex of H_{P_j} *free* if it is a cut-vertex of H_{P_j} or it belongs to $A(C) \cup N(C)$. Let X' be a subgraph of X. We say that X' *hangs from*, or is *hanging by*, the vertex $v \in V(C)$ if $v = V(C) \cap V(X')$

Theorem 9.12. *A homeomorphically reduced, connected reflexive multigraph X has* $\mathrm{sw}(G) \leqq 3$ *if and only if X consists of a 3-chain*

$$C = (a_0, B_1, a_1, B_2, \ldots, a_{r-1}, B_r, a_r)$$

with a valid set of opposing poles $a_0x_1, a_1y_1, a_1x_2, a_2y_2, a_2x_3, \ldots, a_r y_r$, along with vertex-disjoint reflexive submultigraphs of the following forms hanging from the vertices in $V(C)$:

(a) an arbitrary number of edges and loops hanging from each free vertex;
(b) an arbitrary number of pinched graphs hanging by their coalesced nodes from each vertex in $A(C)$;
(c) for $1 \leqq j \leqq r$, at most one 2-sweepable graph hanging by one of its end vertices from each of x_j and y_j, or, if $x_j = y_j$, at most two such subgraphs hanging from the single vertex $x_j = y_j$.

10. Restricted Sweeping.

The notion of general sweeping first introduced by Parsons allows considerable freedom for both sweepers and an intruder. The move to sweeping and wormhole sweeping places restrictions on the sweepers in terms of how many may move and the moves themselves. The notion of monotonic is another kind of restriction. The next definition introduces another restriction in which we are interested.

Definition 10.1. Let X be a reflexive multigraph and let E_i denote the set of cleared edges at time i, $i \in \mathbb{Z}$. A sweep strategy for which the submultigraphs $\langle E_i \rangle$ induced by E_i are connected for all i is called *connected.*

Even though the sweep number of a reflexive multigraph is the same for general sweeping, sweeping, wormhole sweeping, and mixed sweeping, once we begin considering monotonicity and connectedness, we shall find that the different sweep models may behave differently. Consequently, we need to introduce extra notation. On the other hand, in this section, we are going to consider only sweeping and wormhole sweeping. Accordingly, we let $\mathrm{msw}(X)$ denote the minimum number of sweepers required for a monotonic sweep strategy for X, we let $\mathrm{mksw}(X)$ denote the minimum number of sweepers required for a monotonic connected sweep strategy for X, we let $\mathrm{mwsw}(X)$ denote the minimum number of sweepers required for a monotonic wormhole sweep strategy for X, and we let $\mathrm{mkwsw}(X)$ denote the minimum number of sweepers required for a monotonic connected wormhole sweep strategy for X. These numbers are called the *monotonic sweep number, monotonic connected sweep number, monotonic wormhole sweep number, and monotonic connected wormhole sweep number* of X, respectively.

From Theorem 7.3, we know that $\mathrm{mwsw}(X) = \mathrm{wsw}(X)$, where we emphasize the fact we are comparing wormhole strategies by using $\mathrm{wsw}(X)$ for wormhole sweeping. The following result was given in [5].

Theorem 10.2. *If X is a connected graph, then* $\mathrm{sw}(X) = \mathrm{wsw}(X) = \mathrm{mwsw}(X) \leqq \mathrm{msw}(X) \leqq \mathrm{kwsw}(X) = \mathrm{ksw}(X) \leqq \mathrm{mkwsw}(X) = \mathrm{mksw}(X)$.

The first two inequalities were proven in [5] together with examples that showed the inequalities could be strict.
Whether the inequality $\mathrm{ksw}(X) \leqq \mathrm{mkwsw}(X)$ could be strict was left as an unsolved problem in [5]. It is shown in [26] that the inequality may be strict. In fact, the difference may be arbitrarily large. The example depends heavily on the way in which cliques must be swept, and that cliques provide lower bounds for sweep numbers. Even though the difference between $\mathrm{mkwsw}(X)$ and $\mathrm{ksw}(X)$ can be made arbitrarily large, the number of vertices grows rapidly. An interesting unsolved problem is whether the ratio of $\mathrm{mkwsw}(X)$ to $\mathrm{ksw}(X)$ is bounded. There is some speculation the ratio may be bounded by 2.

11. Sweeping Digraphs.

Most of the research on sweeping has concentrated on reflexive multigraphs, and little has been done on digraphs. We include a brief description of some recent work dealing with digraphs. Some of this appears in [8]. Since digraphs are equipped with a notion of direction, this should play a role in what it means to sweep a digraph. In fact, we introduce four distinct underlying notions of sweeping digraphs.

Definition 11.1. Let $\overrightarrow{X}$ be a reflexive multidigraph. If both sweepers and the intruder must obey the directions of the arcs of $\overrightarrow{X}$, we call this *directed sweeping*. The minimum number of sweepers required to capture any intruder using directed sweeping on $\overrightarrow{X}$ is denoted $\mathrm{sw}_{1,1}(\overrightarrow{X})$. If both sweepers and the intruder may move with or against the directions of the arcs of $\overrightarrow{X}$, we call this *undirected sweeping*. The minimum number of sweepers required to capture any intruder using undirected sweeping on $\overrightarrow{X}$ is denoted $\mathrm{sw}_{0,0}(\overrightarrow{X})$. If the sweepers may move with or against the directions of the arcs but the intruder must obey the directions of the arcs, we call this *strong sweeping*. The minimum number of sweepers required to capture any intruder using strong sweeping on $\overrightarrow{X}$ is denoted $\mathrm{sw}_{0,1}(\overrightarrow{X})$. Finally, if the sweepers must obey the directions of the arcs but the intruder may move with or against the directions of the arcs, we call this *weak sweeping*. The minimum number of sweepers required to capture any intruder using weak sweeping on $\overrightarrow{X}$ is denoted $\mathrm{sw}_{1,0}(\overrightarrow{X})$.

Of course, performing undirected sweeping on a reflexive multidigraph $\overrightarrow{X}$ is exactly the same as sweeping the reflexive multigraph underlying $\overrightarrow{X}$. The other three modes of sweeping are, in some sense, more interesting for reflexive multidigraphs.

Theorem 11.2. *If $\overrightarrow{X}$ is a reflexive multidigraph, then*

$$\mathrm{sw}_{0,1}(\overrightarrow{X}) \leqq \mathrm{sw}_{1,1}(\overrightarrow{X}) \leqq \mathrm{sw}_{1,0}(\overrightarrow{X}),$$

and

$$\mathrm{sw}_{0,1}(\overrightarrow{X}) \leqq \mathrm{sw}_{0,0}(\overrightarrow{X}) \leqq \mathrm{sw}_{1,0}(\overrightarrow{X}).$$

Since the directed path $\overrightarrow{P_n}$ is easily seen to satisfy

$$\mathrm{sw}_{0,1}(\overrightarrow{P_n}) = \mathrm{sw}_{1,0}(\overrightarrow{P_n}) = 1,$$

we see that we can achieve equality throughout both strings of inequalities. On the other hand, if $\overrightarrow{T_n}$ is the transitive tournament of order n, it is not difficult to see that $\mathrm{sw}_{0,1}(\overrightarrow{T_n}) = 1$. Since the graph underlying $\overrightarrow{T_n}$ is K_n, we have $\mathrm{sw}_{0,0}(\overrightarrow{T_n}) = n$, when $n \geqq 4$, by Corollary 8.4. It is not difficult to show that $\mathrm{sw}_{1,1}(\overrightarrow{T_n}) = \lfloor \frac{n^2}{4} \rfloor$. This shows that the first inequality in each string of inequalities can be strict.

The digraph $\overrightarrow{X}$ with vertex set $\{v_1, v_2, v_3\}$ and arc set $\{(v_2, v_1), (v_2, v_3)\}$ satisfies $\mathrm{sw}_{0,0}(\overrightarrow{X}) = 1$ and $\mathrm{sw}_{1,0}(\overrightarrow{X}) = 2$. The digraph $\overrightarrow{Y}$ with vertex set $\{v_1, v_2, v_3, v_4\}$ and arc set $\{(v_1, v_2), (v_2, v_3), (v_2, v_4), (v_3, v_1), (v_4, v_1)\}$ satisfies $\mathrm{sw}_{1,1}(\overrightarrow{Y}) = 2$ and $\mathrm{sw}_{1,0}(\overrightarrow{Y}) = 3$. Thus, all of the inequalities may be strict in Theorem 11.2.

Theorem 11.3. *If $\overrightarrow{X'}$ is a subdigraph of a reflexive multidigraph $\overrightarrow{X}$, then* $\mathrm{sw}_{0,1}(\overrightarrow{X'}) \leqq \mathrm{sw}_{0,1}(\overrightarrow{X})$.

The preceding theorem does not hold for directed sweeping. If we let $\overrightarrow{T}$ be the tournament we obtain by reversing the arc from the source to the sink in the transitive tournament of order 4, then $\mathrm{sw}_{1,1}(\overrightarrow{T}) = 2$. If we then let $\overrightarrow{Y}$ be the subdigraph of $\overrightarrow{T}$ obtained by removing the reversed arc, we have $\mathrm{sw}_{1,1}\overrightarrow{Y} = 3$.

Definition 11.4. The *arc digraph* $L(\overrightarrow{X})$ of a digraph $\overrightarrow{X}$ has its vertices corresponding to the arcs of $\overrightarrow{X}$, and an arc from (u, v) to (x, y) whenever (u, v) and (x, y) are arcs of $\overrightarrow{X}$ such that $v = x$.

Definition 11.5. The *path cover number* of a digraph $\overrightarrow{X}$ is the fewest number of directed paths of $\overrightarrow{X}$ that cover the vertex set of $\overrightarrow{X}$. We denote this by $\mathrm{pc}(\overrightarrow{X})$.

An acyclic digraph is a digraph without directed cycles. The following result is proved in [17].

Theorem 11.6. *If* $\overrightarrow{X}$ *is an acyclic digraph, then* $\mathrm{sw}_{1,0}(\overrightarrow{X}) = \mathrm{pc}(\overrightarrow{X})$.

In contrast to the preceding result for weak sweeping, the next result for strong sweeping has a much different conclusion.

Theorem 11.7. *If* $\overrightarrow{X}$ *is an acyclic digraph, then* $\mathrm{sw}_{0,1}(\overrightarrow{X}) = 1$.

Let $\mathcal{D}$ denote the collection of the following digraphs: The directed cycles of length 2 or more, the directed cycles of length 2 or more with a suspended directed path of length 2 or more joining two distinct vertices of the directed cycle, and the digraphs formed by amalgamating two directed cycles of lengths at least 2 at a single vertex. It is not difficult to verify that any digraph in $\mathcal{D}$ has strong sweep number 1. We say that such digraphs are 1-*strong- sweepable*. The next theorem tells us that they are essentially the only building blocks.

Theorem 11.8. *A digraph* $\overrightarrow{X}$ *is 1-strong-sweepable if and only if every strong component of* $\overrightarrow{X}$ *is either in* $\mathcal{D}$ *or is a single vertex.*

The descriptions of the digraphs with sweep number 1 under the other sweeping modes is simpler. We have $\mathrm{sw}_{0,0}(\overrightarrow{X}) = 1$ if and only if $\overrightarrow{X}$ is an orientation of a path. For directed sweeping and weak sweeping, the only digraph with sweep number 1 is the directed path.

12. Bounding the Search Number.

We now leave sweeping and return to searching. Recall that the search model we are using is BPE search. The first thing we point out is that we restrict ourselves to graphs. Since searching involves only vertex locations and we allow the intruder to not move on any of the intruder's turns, neither multiple edges nor loops play any role in capturing an intruder. Also recall that $\mathrm{sn}(X)$ denotes the search number of X under this search model.

In this section we examine upper and lower bounds on $\mathrm{sn}(X)$. Some of the bounds are crude but still useful in determining search numbers.

Definition 12.1. The *girth* of a graph X is the length of a shortest cycle in X. We let the girth of a tree be 0. We use $\mathrm{gir}(X)$ to denote the girth of X.

The next result by Aigner and Fromme [1] indicates that the girth of a graph is involved in determining the search number. We explore this connection for the rest of this section.

Theorem 12.2. *Let X be a graph with minimum valency at least d. If X contains no 3-cycles or 4-cycles, then* $\mathrm{sn}(X) \geqq d$.

Frankl [10] improved the preceding result as follows.

Theorem 12.3. *If X is a connected graph with* $\mathrm{gir}(X) \geqq 8t-3$ *and minimum valency at least $d+1$, then* $\mathrm{sn}(X) > d^t$.

Frankl [9] also proved the following.

Theorem 12.4. *Let d denote the minimum valency of a connected graph X. If any two vertices of X are connected by at most two paths of length at most 2, then* $\mathrm{sn}(X) d/2$. *Moreover, if X contains no 3-cycles, then* $\mathrm{sn}(X) \geqq (d+1)/2$.

The results in this section so far have dealt with lower bounds for the search number. We conclude the section with results that give upper bounds for certain graphs. The first is a trivial upper bound.

Definition 12.5. Let X be a graph. A *dominating set* of X is a set $V_1 \subseteq V(X)$ such that every vertex of X either belongs to V_1 or is adjacent to a vertex in V_1. The cardinality of a minimum dominating set of X is called the *domination number* of X, and is denoted $\mathrm{dom}(X)$.

Proposition 12.6. *If X is a graph, then* $\mathrm{sn}(X) \leqq \mathrm{dom}(X)$.

Aigner and Fromme [1] obtained the following result on upper bounds.

Theorem 12.8. *Let X be a connected graph with maximum valency at most 3. If any two adjacent edges of X are contained in a cycle of length at most 5, then* $\mathrm{sn}(X) \leqq 3$.

Theorem 14.4 establishes an upper bound on the search number of a planar graph. Since planarity may be characterized in terms of minors, this suggests that minors might be a useful avenue for investigation. The next two theorems by Andreae [3] deal with minors.

Theorem 12.8. *Let u be a vertex of a graph Y such that $Y \setminus u$ has no isolated vertices. If X is a graph with no Y-minor, then* $\mathrm{sn}(X) \leqq |E(Y \setminus u)|$.

Let A and B be the two graphs displayed in Figure 2. Since K_5 and $K_{3,3}$ are both minors of B, a planar graph cannot have a B-minor. Consequently, the next theorem extends Theorem 14.4 given later.

Theorem 12.9. *Let X be a graph. We then have* $\mathrm{sn}(X) \leqq 2$ *if $A \not\preceq X$, and* $\mathrm{sn}(X) \leqq 3$ *if $B \not\preceq X$.*

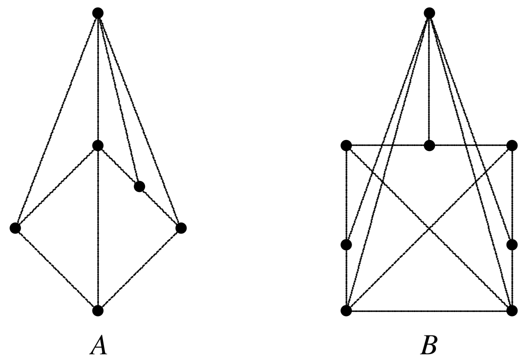

Figure 2

13. Searching Cayley Graphs.

We now move to consideration of search numbers for some special classes of graphs. The first is the class of Cayley graphs. We give the definition first.

Definition 13.1. Let G be a finite group and $S \subset G$, with $1 \notin S$ and $S = S^{-1}$, that is, $s \in S$ implies $s^{-1} \in S$. The *Cayley graph* $\mathrm{Cay}(G; S)$ is the graph with vertex set G and edges joining g and gs, for all $g \in G$ and $s \in S$. We refer to $\mathrm{Cay}(G; S)$ as the *Cayley graph on G with connection set S*.

Theorem 13.2. *If $X(G; S)$ is a connected Cayley graph on the abelian group G, then* $\mathrm{sn}(X(G; S)) \leqq \lceil \frac{(|S|+1)}{2} \rceil$.

Corollary 13.3. *The d-dimensional cube Q_d satisfies*

$$\mathrm{sn}(Q_d) = \lceil \frac{d+1}{2} \rceil.$$

Theorem 13.2 is a consequence of a more general result in [9] that we omit here. The essential contributing factor to the bound appearing in Theorem 13.2 is that Cayley graphs of valency 3 or more on abelian groups have many 4-cycles. This translates into a single searcher being able to reduce the search to a quotient Cayley graph of valency two or three less with one less searcher. This is the basis of an inductive proof.

Corollary 13.3 gives the actual search number for the d-dimensional cube. We know the search number is at most this value from Theorem 13.2. To show that we need this many searchers, note that the intruder is captured when he is located at a vertex v such that every neighbor of v is adjacent to a vertex with a searcher on it, and at least one neighbor of v itself has a searcher on it. When

d is even, it takes at least $d/2$ searchers to cover the d neighbors of the vertex v at which the intruder is located because any two vertices of Q_d have at most two common neighbors. One additional searcher is required to be on a vertex adjacent to v. Thus, it takes at least $1+d/2 = \lceil (d+1)/2 \rceil$ searchers to capture an intruder in Q_d.

When d is odd, it takes at least $(d+1)/2$ searchers to take care of the d neighbors of v. One of them actually may be located at a neighbor of v taking care of v as well. Thus, at least $(d+1)/2 = \lceil (d+1)/2 \rceil$ searchers are required. Hence, we see that equality holds in both cases and the result follows.

One other result specifically dealing with Cayley graphs is the following. It was obtained by Frankl [10] .

Theorem 13.4. *Let* $\mathrm{Cay}(G; S)$ *be a connected Cayley graph on the finite group* G. *If* $gSg^{-1} = S$ *for all* $g \in G$, *then* $\mathrm{sn}(\mathrm{Cay}(G; S))) \leqq |S|$.

14. Searching and Genus.

There is no transparent connection between searching and embedding graphs in surfaces. Nevertheless, planar graphs often have nice properties with respect to various graph parameters. Hence, it is no surprise that graphs embedded on surfaces have been considered. In this section we consider only orientable surfaces.

Definition 14.1. We say that a collection $\mathcal{S}$ of searchers *protects* a subgraph Y of X if the searchers in $\mathcal{S}$ can move so that for any sequence of intruder moves leading to the intruder moving to a vertex of Y, the intruder is immediately captured by a searcher in $\mathcal{S}$ upon moving to a vertex of Y.

Lemma 14.2. *Let* v_1, v_t *be distinct vertices of a graph* X *and* $P = v_1 v_2 \ldots v_t$ *be a shortest path from* v_1 *to* v_t. *After a finite number of moves, a single searcher can protect* P.

The lemma is proved by showing that a single searcher γ can reach a vertex of P, after a finite number of moves, so that the distance from γ to any vertex of P is at most the distance from the intruder to the vertex. Once this situation is achieved, it is not hard to see that no matter how the intruder moves, the situation can be maintained by the searcher on the path.

The preceding lemma plays an important role in what follows. It also has surprising consequences. One such consequence is the following corollary.

Corollary 14.3. *Let C be a shortest cycle in a graph X. If there are at least two searchers in X, then after a finite number of moves a single searcher can protext C.*

The corollary is proved by moving a searcher to a vertex u of C. At this point, the intruder cannot use either edge of C incident with u. Apply Lemma 14.2 to the path P obtained by removing an edge e of C incident with u. A single searcher can move so that P is protected in the subgraph $X \setminus \{e\}$. However, the intruder can never use this edge without first reaching a vertex of P. That is, the intruder is forced to move in $X \setminus \{e\}$. Hence, the single searcher protecting P in $X \setminus \{e\}$ is simultaneously protecting C in X.

The following result was obtained in [1], but as pointed out in the discussion of Theorem 12.9, it is subsumed by the earlier result. Nevertheless, we give it here as a separate result because of the special role played by planar graphs. There are planar graphs achieving the upper bound. One such example is the dodecahedron. This is easily seen by observing that no matter the location of the intruder, two searchers cannot cover all three neighbors of the vertex containing the intruder.

Theorem 14.4. *If X is a planar graph, then* $\mathrm{sn}(X) \leqq 3$.

A non-planar graph X is called *toroidal* if it can be embedded in the torus. The following result was given in [22]. What is interesting is that no one knows of a toroidal graph with search number 4.

Theorem 14.5. *If X is toroidal, then* $\mathrm{sn}(X) \leqq 4$.

No one has much of an idea what happens for larger genus. Schroeder [22] has established that $\lfloor \frac{3g}{2} \rfloor + 3$ is an upper bound for the search number of a graph of genus g, and he has conjectured that $g + 3$ is an upper bound. However, there is little evidence to indicate how good the established bound or conjectured bound might be.

15. Graph Products.

For some graph parameters, there is a strong connection between the values of the parameter for graphs in the product, and the value of the parameter for the product itself. In this section, we examine this theme for search numbers. The first product we consider is the cartesian product.

Definition 15.1. Let $X_i = (V_i, E_i)$, $1 \leqq i \leqq n$, be graphs. The *cartesian product* of $X_1, X_2, \ldots, X_n$, denoted $X_1 \square X_2 \square \ldots \square X_n$, has vertex set $V_1 \times V_2 \times \cdots \times V_n$, where two vertices $(u_1, u_2, \ldots, u_n)$ and $(v_1, v_2, \ldots, v_n)$ are adjacent if and only if there exists j satisfying $1 \leqq j \leqq n$ such that $u_j v_j \in E_j$ and $u_i = v_i$ for all $i \neq j$.

Maamoun and Meyniel [14] investigated bounds on the search numbers of cartesian products. The next three results deal with that work.

Theorem 15.2. *If $T_1, T_2, \ldots, T_r$ are trees, then*

$$\mathrm{sn}(T_1 \,\square\, T_2 \,\square\, \cdots \,\square\, T_r) \leqq \lceil \frac{r+1}{2} \rceil.$$

Theorem 15.3. *If X is the cartesian product of r connected graphs $X_i = (V_i, E_i)$, with $|V_i| \geqq 2$, then* $\mathrm{sn}(X) > \lceil \frac{r-1}{2} \rceil$.

Theorems 15.2 and 15.3 give an immediate proof of the following corollary. This corollary is a generalization of Corollary 13.3, in which the search number of the d-dimensional cube Q_d is presented, since the d-dimensional cube Q_d is the cartesian product of d trees.

Corollary 15.4. *If $T_1, T_2, \ldots, T_r$ are trees of order at least 2, then*

$$\mathrm{sn}(T_1 \,\square\, T_2 \,\square\, \cdots \,\square\, T_r) = \lceil \frac{r+1}{2} \rceil.$$

Tošić [23] considered the cartesian product of arbitrary connected graphs. The next theorem is his.

Theorem 15.5. *If X_1 and X_2 are two connected graphs, then*

$$\mathrm{sn}(X_1 \,\square\, X_2) \leqq \mathrm{sn}(X_1) + \mathrm{sn}(X_2).$$

Corollary 15.4 establishes the search number of the cartesian product of any number of trees. We now consider the same problem for the cartesian product of cycles. This is from [16].

Theorem 15.6. *If $X = C_1 \,\square\, C_2 \,\square\, \cdots \,\square\, C_r$, where each C_i is a cycle of length at least 4, then* $\mathrm{sn}(X) = r + 1$.

Definition 15.7. Let $X_1 = (V_1, E_1), X_2 = (V_2, E_2), \ldots, X_r = (V_r, E_r)$ be a collection of r graphs. The *categorical product* of $X_1, X_2, \ldots, X_r$, denoted $X_1 \times X_2 \times \cdots \times X_r$, is defined to be the graph with vertex set $V_1 \times V_2 \times \cdots \times V_r$, where vertex $(u_1, u_2, \ldots, u_r)$ is adjacent to vertex $(v_1, v_2, \ldots, v_r)$ if and only if $u_i v_i \in E(X_i)$ for all i, $1 \leqq i \leqq r$.

We now discuss searching categorical products. In some sense, categorical products behave strangely. For example, unlike cartesian products, the categorical product of two connected graphs need not be connected. It turns out to be easy to determine when connected graphs have connected categorical products. That is the subject of the next well-known result.

Theorem 15.8. *If X_1 and X_2 are connected graphs, then their categorical product $X_1 \times X_2$ is connected if and only if at least one of them is not bipartite.*

Theorem 15.9. *Let X_1 and X_2 be connected non-bipartite graphs such that* $\mathrm{sn}(X_2) \geqq \mathrm{sn}(X_1)$.

(i) If $\mathrm{sn}(X_2)2$*, then*

$$\mathrm{sn}(X_1 \times X_2) \leqq 2\mathrm{sn}(X_1) + sn(X_2) - 1.$$

(ii) If $\mathrm{sn}(X_1) = \mathrm{sn}(X_2) = 1$*, then*

$$\mathrm{sn}(X_1 \times X_2) \leqq 3.$$

The following corollary is an immediate consequence of the preceding theorem.

Corollary 15.10. *Let X_i, $1 \leqq i \leqq r$, be connected non-bipartite graphs.*
If $\mathrm{sn}(X_i) \geqq 2$ *for some i, then*

$$\mathrm{sn}(X_1 \times X_2 \times \cdots \times X_r) \leqq 2(\sum_{i=1}^{r} \mathrm{sn}(X_i)) - \max\ \mathrm{sn}(X_i) - r + 1.$$

If $\mathrm{sn}(X_i) = 1$ *for all i, $1 \leqq i \leqq r$, then*

$$\mathrm{sn}(X_1 \times X_2 \times \cdots \times X_r) \leqq r + 1.$$

The next two theorems are from [16].

Theorem 15.11. *If X is the categorical product of $r \geqq 2$ complete graphs, each of order at least 3, then*

$$\mathrm{sn}(X) \leqq \lceil \frac{r+1}{2} \rceil + 1.$$

Definition 15.12. The *strong product* of $X_1, X_2, \ldots, X_r$, denoted $X_1 \boxtimes X_2 \boxtimes \cdots \boxtimes X_r$, is defined to be the graph with vertex set $V(X_1) \times V(X_2) \times \cdots \times V(X_r)$, where vertex $(u_1, u_2, \ldots, u_r)$ is adjacent to vertex $(v_1, v_2, \ldots, v_r)$ if and only if for each i, $1 \leqq i \leqq r$, either $u_i v_i \in E(X_i)$ or $u_i = v_i$.

Theorem 15.13. *If* X_1 *and* X_2 *are graphs satisfying either* $\mathrm{sn}(X_1) \geqq 2$ *or* $\mathrm{sn}(X_2) \geqq 2$, *then* $\mathrm{sn}(X_1 \boxtimes X_2) \leqq \mathrm{sn}(X_1) + \mathrm{sn}(X_2) - 1$.

The following result is an immediate corollary of Theorem 15.13 since the search number of a cycle is 2.

Corollary 15.14. *If* $C_1, C_2, \ldots, C_r$ *are cycles all of whose lengths are at least 5, then*

$$\mathrm{sn}(C_1 \boxtimes C_2 \boxtimes \cdots \boxtimes C_r) \leqq r + 1.$$

Definition 15.15. Let X_1 and X_2 be two graphs. The *wreath product*, denoted $X_1 \wr X_2$, has $V(X_1) \times V(X_2)$ as the vertex set, where (u_1, u_2) and (v_1, v_2) are adjacent if and only if either $u_1 = v_1$ and u_2 is adjacent to v_2 in X_2, or u_1 and v_1 are adjacent in X_1.

The easy way to picture the wreath product $X_1 \wr X_2$ is to replace each vertex of X_1 with a copy of X_2 and make two vertices in distinct copies of X_2 adjacent if and only if the corresponding vertices are adjacent in X_1. The wreath product of more than two graphs is defined in the obvious way since this product is associative.

Theorem 15.16. *Let* X_1 *and* X_2 *be two graphs. If* $\mathrm{sn}(X_1) \geqq 2$ *or* $\mathrm{sn}(X_1) = \mathrm{sn}(X_2) = 1$, *then* $\mathrm{sn}(X_1 \wr X_2) = \mathrm{sn}(X_1)$. *If* $\mathrm{sn}(X_1) = 1$ *and* $\mathrm{sn}(X_2) \geqq 2$, *then* $\mathrm{sn}(X_1 \wr X_2) = 2$.

16. 1-Searchable Graphs.

Definition 16.1. If X is a graph and $\mathrm{sn}(X) \leqq k$, then we say that X is *k-searchable*.

The purpose of this section is to present the nice characterization of 1-searchable graphs as given by Nowakowski and Winkler [18]. It also was obtained independently by Quilliot [21].

One characterization involves an order relation defined on the vertex set of a graph. We use the notation $\overline{N(u)}$ to denote the closed neighborhood of u, that is, the neighbors of u together with u itself. We define a relation $\mathbb{R}_0$ on $V(X)$, where X is a graph, by letting $\mathbb{R}_0 = \{(u, u) : u \in V(X)\}$. In other words, $\mathbb{R}_0$ is just the diagonal relation. Assume that the relation $\mathbb{R}_k$ has been defined for each integer $k \in [0, t]$. Define $\mathbb{R}_{t+1}$ by saying $(u, v) \in \mathbb{R}_{t+1}$ if and only if for every $w \in \overline{N(u)}$, there is a vertex $w' \in \overline{N(v)}$ such that $(w, w') \in \mathbb{R}_k$ for some $k \in [0, t]$.

We now observe that $\mathbb{R}_k \subseteq \mathbb{R}_{k+1}$. If we choose a pair (u, u) from the diagonal, then for each $w \in \overline{N(u)}$, we choose the same w and have $(w, w) \in \mathbb{R}_0$. This implies that $(u, u) \in \mathbb{R}_{k+1}$. If $(u, v) \in \mathbb{R}_k$, where $u \neq v$, then whatever satisfies the definition to include $(u, v) \in \mathbb{R}_k$ also works to include $(u, v) \in \mathbb{R}_{k+1}$. It follows that $\mathbb{R}_k \subseteq \mathbb{R}_{k+1}$.

It is easy to see that whenever we reach a t such that $\mathbb{R}_t = \mathbb{R}_{t-1}$, then $\mathbb{R}_k = \mathbb{R}_{t-1}$ for all $k \geqq t$. On the other hand, the graph X is finite so that it has only finitely many ordered pairs of vertices. Since $\mathbb{R}_k \subseteq \mathbb{R}_{k+1}$, there must be a smallest t for which $\mathbb{R}_t = \mathbb{R}_{t-1}$. We then define the relation $\mathbb{R}$ to be $\mathbb{R}_{t-1}$.

The relation just defined is not particularly intuitive, but it does lead to the nice Theorem 16.4 below.

Definition 16.2. The order $\mathbb{R}$ will be called *complete* if every ordered pair (u, v) belongs to $\mathbb{R}$.

Lemma 16.3. *If the graph X is either a tree or a complete graph, then $\mathbb{R}$ is the complete order on $V(X)$.*

Theorem 16.4. *A graph X is 1-searchable if and only if the order $\mathbb{R}$ on X is complete.*

Definition 16.5. A graph X is said to be *bridged* if for every cycle C of length at least 4 in X, there is a bridge B with vertices u, v of attachment such that the shortest u, v-path in B has length strictly less than the length of either of the two (u, v)-paths along C.

From the definition, we see that any cycle of length 4 or 5 in a bridged graph must have a chord. The following result provides a non-trivial class of 1-searchable graphs.

Theorem 16.6. *Bridged graphs are 1-searchable.*

17. Research Directions.

This brief survey omits a considerable amount of material that has been produced during the course of investigating searching and sweeping graphs. Given the length of this survey, the scarcity of details, and the just mentioned omissions, one might conclude that the problems have been thoroughly investigated. I believe this to be far from the case. In fact, I would say that the problem has been barely touched. Following are some directions that future research might take.

There has been very little work done on digraphs. Section 11 gives some information on new work regarding sweeping digraphs. Searching digraphs is considered in [4], [12], [17]. Special classes of digraphs, such as tournaments, seem like a rich source of new problems.

All of the work on sweeping considered in this survey assumes that no information on the location of an intruder is available. Indeed, it isn't even known whether an intruder is present in the graph. Another wide open area for new research is to allow various levels of information. Yang [25] has obtained results on allowing sweepers to be able to see all the points of their closed 1-neighborhoods; that is, all the points of the incident edges together with the vertices on the other end of the incident edges. There are a variety of approaches to partial information on the intruder's location.

Other than general sweeping, only one sweeper at a time is allowed to move in the sweeping models discussed. For most practical applications, it is apparent that there would be many instances where simultaneous moves would be desirable. What happens to the time performance of sweeping if we allow more than one sweeper to move at a time?

Suppose we associate costs with sweepers, edge traversals, jumps and time until completion. What can be said about some notion of optimization if we allow extra sweepers? Again, this has practical implications.

BPE search has complete information available for both the intruder and the searchers. What happens to search numbers if we restrict the information? For example, it seems natural to consider zero information searching. Will some of the characteristics of sweeping carry over to searching with little or no information?

What happens to both searching and sweeping if we build the graph as we go? Essentially all of the work on both searching and sweeping assume the graph is known ahead of time. Having the graph unfold as vertices are reached will change the number of sweepers required dramatically. This also seems to have practical importance.

REFERENCES

[1] M. Aigner - M. Fromme, *A game of cops and robbers,* Discrete Appl. Math., 8 (1984), pp. 1–12.

[2] B. Alspach - X. Li - B. Yang, *Searching Graphs and Directed Graphs,* preprint (2004).

[3] T. Andrae, *On a pursuit game played on graphs for which a minor is excluded,* J. Combin. Theory Ser. B, 41 (1986), pp. 37–47.

[4] J. Bar'at, *Directed path-width and monotonicity in directed graph searching,* Graphs Combin., to appear.

[5] L. Barrière - P. Fraigniaud - N. Santoro - D. Thilokis, *Searching is not jumping,* in *Proc. 29th Workshop on Graph Theoretic Concepts in Computer Science (WG 2003)*, Lecture Notes in Computer Science, 2880 (2003), pp. 34–45.

[6] D. Bienstock - P. Seymour, *Monotocity in graph searching,* J. Algorithms, 12 (1991), pp. 239–245.

[7] R. L. Breisch, *An intuitive approach to speleotopology,* Southwestern Cavers, 6 (1967), pp. 72–78.

[8] Daniel Dyer, *Sweeping Graphs and Digraphs,* Ph.D. Thesis, Simon Fraser University, 2004.

[9] P. Frankl, *On a pursuit game on Cayley graphs,* Combinatorica, 7 (1987), pp. 289–295.

[10] P. Frankl, *Cops and robbers in graphs with large girth and Cayley graphs,* Discrete Applied Math., 17 (1987), pp. 301–305.

[11] M. Garey - D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness,* Freeman, San Francisco, 1979.

[12] Y. Hamidoune, *On a pursuit game on Cayley digraphs,* Europ. J. Combin., 8 (1987), pp. 289–295.

[13] A.S. LaPaugh, *Recontamination does not help to search a graph,* Journal of ACM, 40 (1993), pp. 224–245.

[14] M. Maamoun - H. Meyniel, *On a game of policemen and robber,* Discrete Appl. Math., 17 (1987), pp. 307–309.

[15] N. Megiddo - S. L. Hakimi - M. Garey - D .Johnson - C.H. Papadimitriou, *The complexity of searching a graph,* Journal of ACM, 35 (1988), pp. 18–44.

[16] S. Neufeld - R. Nowakowski, *A game of cops and robbers played on products of graphs,* Discrete Math., 186 (1998), pp. 253–268.

[17] R. Nowakowski, *Search and sweep numbers of finite directed acyclic graphs,* Discrete Appl. Math., 41 (1993), pp. 1–11.

[18] R. Nowakowski - P. Winkler, *Vertex-to-vertex pursuit in a graph,* Discrete Math., 43 (1983), pp. 235–239.

[19] T.D. Parsons, *Pursuit-evasion in a graph,* Theory and Applications of Graphs.

Y. Alavi and D. R. Lick, eds., Springer, Berlin, (1976),pp. 426–441

[20] T.D. Parsons, *The search number of a connected graph,* Proc. Ninth Southeastern Conf. Combinatorics, Graph Theory and Computing, Congress. Num. XXI, Winnipeg, 1978, pp. 549–554.

[21] A. Quilliot, *Thèse de* 3° *Cycle,* Université de Paris VI, 1978.

[22] B. Schroeder, *The copnumber of a graph is bounded by* $\lfloor \frac{3}{2} genus(G) \rfloor + 3$*,* Trends Math., (2001), pp. 243–263.

[23] R. Tošić, *On cops and robber game,* Stud. Sci. Mat. Hungar., 23 (1988), pp. 225–229.

[24] D.B. West, *Introduction to Graph Theory,* Prentice-Hall, 1996.

[25] B. Yang, *Sweeping, visibility, and tree decompositions,* preprint, (2004).

[26] B. Yang - D. Dyer - B. Alspach, *Sweeping graphs with large clique number,* preprint.

Department of Mathematics and Statistics,
University of Regina,
Regina, SK (CANADA) S4S 0A2
*e-mail:*alspach@math.uregina.ca