Hunter vs. Mole

Natasha Komarov* and Peter Winkler†

September 9, 2019

Abstract

We consider a variation of a cops and robbers game [4, 5] in which the cop—here referred to as "hunter"—is not constrained by the graph but must play in the dark against a "mole." We characterize the graphs—which we will call "hunter-win"—on which the hunter can guarantee capture of the mole in bounded time. We also define an optimal hunter strategy (and consequently an upper bound on maximum game time on hunter-win graphs) and note that an optimal hunter strategy need not take advantage of the hunter's unconstrained movement! This game comes from a puzzle of unknown origin which was told to the authors by Dick Hess [3].

1 Introduction

In the original "cops and robbers" game, the cop and robber move alternately and with full information about each other's current positions, from vertex to adjacent vertex on a connected, undirected graph. A "move" in this game is considered to be a step by the cop followed by a step by the robber. The cop moves so as to minimize the number of moves until capture—that is, until both players occupy the same vertex at the same time, either because the cop has moved to occupy the robber's position or because the robber has been forced to move onto the cop's—while the robber moves with the intention of maximizing this quantity. Graphs on which capture will occur, under optimal play by both parties, are called "cop-win." Cop-win graphs have been characterized [4, 5] and the number of moves required to capture has been found [1, 2] to be at most $n-4$ for all $n \geq 7$.

In the variation presented here, the role of the cop is filled by a "hunter" who is chasing a "mole" on a graph. We will allow the hunter unconstrained access from a vertex to any other vertex in the graph and force the mole to go from vertex to adjacent vertex only. We assume that the players move simultaneously on a connected, undirected graph. We will first discuss the case

*Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh PA 15213, USA; nkom@cmu.edu.
†Department of Mathematics, Dartmouth, Hanover NH 03755-3551, USA; peter.winkler@dartmouth.edu.

of graphs containing no loops (in particular, graphs on which the mole must move to a different vertex at each turn). Multi-edges add nothing to our discussion, so we will, for the sake of nicer figures, assume that there are none. We characterize the class of graphs on which a single hunter is guaranteed to catch the mole in bounded time. We will call such graphs hunter-win. Any graphs that are not hunter-win will be called mole-win.

We proceed by first considering a family of graphs called lobsters and showing that they are hunter-win. We will then see that all graphs containing cycles are mole-win, and will further narrow down the possible hunter-win trees by showing that they may not contain as a subgraph a rather innocuous-seeming spider. Finally, we will show that graphs are lobsters if and only if they do not contain this spider, completing our characterization: a graph is hunter-win if and only if it is a lobster.

Our proofs will rely heavily on the fact that the graph contains no loops. We will see that there is good reason for this, as a graph on which the mole is free to remain at a vertex—even if only at a single vertex—may be mole-win, and a graph with at least two loops is completely hopeless for the hunter. We characterize the hunter-win graphs with loops as well.

2 A Characterization of Hunter-Win Graphs

Note first that since a winning strategy for the hunter must **guarantee** capture in bounded time (against any possible trajectory undertaken by the mole), it is equivalent to consider this game played by a hunter and an omniscient, adversarial mole—i.e. a mole who makes all the "worst-case scenario" moves (from the hunter's perspective) to maximally increase the length of the game. We will be considering this adversary in the remainder of our study of the hunter and mole game.

2.1 A Hunter-Win Graph

We begin by showing that certain kinds of trees, called lobsters, are hunter-win. We will see in Theorem 2.9 that these are the only hunter-win graphs, and give a forbidden subgraph characterization of them.

Definition 2.1. *A* **lobster** *is a tree containing a path P such that all vertices are within distance 2 of P. A path satisfying this condition will be called a* **central path**. *A* **knee** *of a lobster with respect to P is a non-leaf vertex at distance exactly 1 from P. For a knee $w = w(G, P)$ in a lobster*

G, define its associated **hip** *to be the vertex v such that $v \in N(w) \cap P$. Any leaf adjacent to a knee is called a* **foot**.

An example of a lobster is presented in Figure 1.

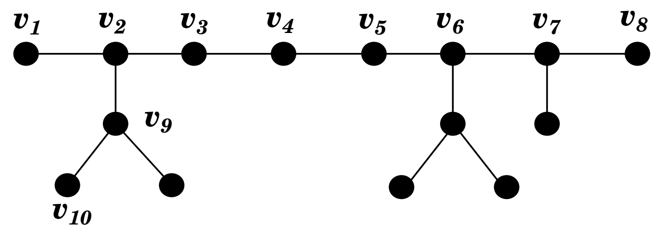

Figure 1: An example of a lobster with a central path $P = v_1 v_2 \ldots v_8$, two knees, two hips, and four feet.

Note that the central path P defined in the example in Figure 1 is not a maximal central path, as the path $P' = v_{10}v_9v_2v_3 \ldots v_8$ is a longer path with all vertices at distance at most two from it. We will often be interested in taking a maximal central path in the proofs that follow.

Lemma 2.2. *All lobsters are hunter-win.*

Proof. Let G be a lobster. We will define a notion of an odd mole (even moles are defined analogously). Let $P = v_1, \ldots, v_m$ be a longest central path on G. We define an **odd mole** on G to be a mole who begins either

(a) on an odd vertex of P,

(b) on a knee adjacent to an even hip of P,

or, (c) on a foot at distance 2 from an odd hip of P.

We will refer by V_j^1 to the set of knees adjacent to hip v_j, and similarly by V_k^2 to the set of feet adjacent to a knee in V_k^1.

We will suppose first that the hunter is playing against an odd mole. The hunter starts on v_1. When she reaches a vertex v_i for the first time, she does one of the following things: if $V_i^2 = \emptyset$ then she moves to v_{i+1}, and otherwise, she moves to a random knee in V_i^1. She then checks the knees of hip v_i in random order, alternating between v_i and a knee in V_i^1 at each step. When all the knees in V_i^1 have been checked, she goes back to v_i and then heads to v_{i+1}.

Let M_t be the set of vertices to which the odd mole may move at time t in order to avoid being caught in one move. We proceed by induction on t to show that when the hunter is on v_i for the first time (say at time t), M_t consists of the vertices in

$$\begin{aligned} &(a) \quad \{v_j : j \in \{i{+}2, i{+}4, \dots\}\}, \\ &(b) \quad V_k^1 \text{ for some } k \in \{i{+}1, i{+}3, \dots\}, \\ \textit{and} \quad &(c) \quad V_\ell^2 \text{ for some } \ell \in \{i, i{+}2, \dots\} \end{aligned}$$

When $t = 1$, we know that $i = 1$ and this follows from the definition of an odd mole. Now suppose that for some t, the hunter is at v_i and M_t is as described in the claim above.

If $V_i^2 \neq \emptyset$, then the hunter moves to a vertex in V_i^1 at time $t{+}1$ to check that the mole is not on this branch. If the mole is in V_i^2, then the hunter's strategy defined above keeps the mole confined to this branch until the hunter has checked all the knees (since when she checks V_i^1, the mole also goes to a vertex in V_i^1, and therefore when the hunter goes to v_i, the mole must go to a vertex in V_i^2). Note that this also means that the mole cannot switch knees, so once the hunter searches a vertex in V_i^1, there is no possibility of recontamination. Therefore the number of moves that the hunter spends searching through V_i^1 is equal to $2k_i$ where $k_i = |V_i^1|$. We can assume the mole was not in this branch, as he would otherwise have been caught by this point. If the mole was at v_{i+2} at time t, he can move to v_{i+1} or v_{i+3}; even while the hunter is busy checking her current branch, he cannot get back to v_i since the hunter is in V_i^1 when the mole is on v_{i+1}, and so she heads to v_i at the mole's only opportunities to go there. If the mole was in V_{i+1}^1, then he can move to v_{i+1} or to a vertex in V_{i+1}^2. Finally, if the mole was in V_{i+2}^2, he moves to a vertex in V_{i+2}^1. Then the hunter returns to v_i at time $t + 2k_i$ and $M_{t+2k_i} = M_t \backslash V_i^2$. At time $t + 2k_i + 1$, the hunter moves to v_{i+1} and following the same arguments as above, we can see that M_{t+2k_i+1} consists of (a) $v_{i+3}, v_{i+5}, \dots$, (b) $V_{i+2}^1, V_{i+4}^1, \dots$, and (c) $V_{i+1}^2, V_{i+3}^2, \dots$.

If instead $V_i^2{=}\emptyset$ then the hunter moves to v_{i+1} immediately after visiting v_i. If at time t the mole were at v_{i+2}, he must now move to v_{i+3} or a vertex in V_{i+2}^1 to avoid immediate capture. If he was in V_{i+1}^1, he must move into V_{i+1}^2 (therefore if $V_{i+1}^2{=}\emptyset$ then $V_{i+1}^1 \not\subset M_t$ since he would be captured at time $t + 1$ on v_{i+1}). And finally, if he was in V_{i+2}^2, he must move into V_{i+2}^1. Therefore, M_{t+1} consists of (a) $v_{i+3}, v_{i+5}, \dots$, (b) $V_{i+2}^1, V_{i+4}^1, \dots,$, and (c) $V_{i+1}^2, V_{i+3}^2, \dots$.

If the hunter gets to vertex v_{m-1} using this scheme without having won, she knows that her adversary was an even mole, and the scheme repeats identically in the opposite direction (with the hunter waiting one more step at v_{m-1} so that the mole is once again at an even distance from her), ending in capture at v_2. □

Note that this strategy takes the hunter from vertex to adjacent vertex, and so does not make use of her ability to move about the graph freely. In fact, we will see in Section 3 that, surprisingly, this is an optimal strategy for the hunter.

2.2 Mole-win Graphs

We have seen that lobsters are hunter-win. In order to complete our characterization, we will show that no other graphs are hunter-win. We begin with two simple but important lemmas, the proofs of which are straightforward and left to the verification of the reader.

Lemma 2.3. *Let H be any mole-win graph. Then any graph G containing H as a subgraph is also mole-win.*

Lemma 2.4. *The cycle C_n is mole-win for all $n \geq 3$.*

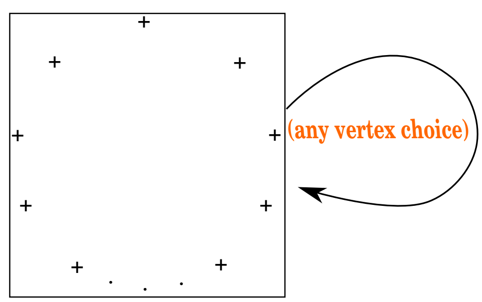

Figure 2: A diagram of the mole's choices on C_n given any hunter move sequence

Figure 2 shows a "move diagram"—a visualization of the proof of Lemma 2.4. In the case of the cycle, it is rather trivial, but will be useful in helping to visualize the argument in Lemma 2.6. We label the vertices with "+" if that vertex is a possible choice for the mole, and "0" otherwise. The arcs represent move sequences that the hunter can choose. In the case of the cycle, this diagram shows that no matter what choice the hunter makes for her move at time t, the situation for the mole at time $t+1$ will be the same as at time t.

Lemmas 2.3 and 2.4 immediately yield the following corollary.

Corollary 2.5. *Any graph containing a cycle is mole-win.*

We now show that by far not all trees will be hunter-win, either. In the remainder of this section, let $S_{3,3}$ be the spider graph with three legs of length three. For notational convenience, we will denote the hub vertex by 0, and the vertices on the legs by v_1, v_2, v_3 where v is one of the symbols a, b, c, and for all v, $0 \in N(v_1)$, $v_1 \in N(v_2)$, and $v_2 \in N(v_3)$. The graph $S_{3,3}$ is represented in Figure 3.

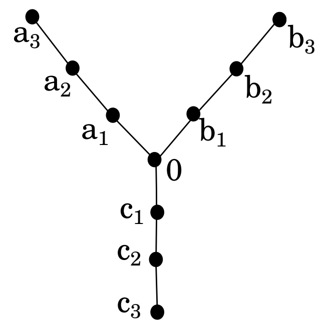

Figure 3: The graph $S_{3,3}$

Lemma 2.6. $S_{3,3}$ *is mole-win.*

Proof. We will show that the hunter cannot make progress against an even mole (that is, a mole who starts at vertices $0, a_2, b_2$, or c_2 at time 1). To do this, we will go through every possible situation that may arise, and show that in every move sequence of the hunter, a previously occuring situation will repeat, showing that the mole never runs out of possible locations. (Note that the previously defined adversarial mole is caught at time t if and only if at time t, there are no possible places for a random mole to be such that he will not be guaranteed to be caught immediately by the hunter.)

We start with the mole at one of the following vertices: $0, a_2, b_2, c_2$. The hunter has (up to symmetry) two distinct choices that will further limit the mole's possible locations: to move to 0 or to move to (without loss of generality) a_2.

Sequence 1. The hunter moves to 0. Then if he is not to be caught immediately, the mole may start on a_2, b_2, or c_2. Therefore the mole may, on his next turn, be at a_1, a_3, b_1, b_3, c_1, or c_3. The hunter again has two choices (up to symmetry): she moves to a_1 or to a_3. In either case, the mole will be at one of the vertices $0, a_2, b_2, c_2$ on the next turn and we have repeated the initial scenario, showing that the hunter cannot win by initially choosing to be at vertex 0.

Sequence 2. The hunter moves to a_2. Then the mole's initial position is at $0, b_2$, or c_2. Therefore the mole will be at a_1, b_1, b_3, c_1, or c_3 next. The hunter now has three choices:

(a) The hunter goes to a_1.

(b) The hunter goes to b_1.

(c) The hunter goes to b_3.

Both options (b) and (c) result in the possible positions a_1, b_1, c_a, c_3 for the mole, and therefore he will next be at $0, a_2, b_2$, or c_2, resulting in the initial situation again.

If the hunter chooses option (a), then the mole must be at vertices b_1, b_3, c_1, or c_3, and therefore will next appear at vertices $0, b_2$, or c_2. The hunter's move sequence may take one of two turns now:

(i) The hunter goes to 0.

(ii) The hunter goes to b_2.

In move sequence (i), the mole must have been at b_2 or c_2 and therefore will next be at b_1, b_3, c_1, or c_3. The hunter may then move to either b_1 or b_3, but in either case, the mole will next be at $0, b_2$,or c_2, creating the same situation (where the hunter must choose between (i) and (ii)), so no progress has been made. In move sequence (ii), the mole may be at a_1, b_1, c_1, or c_3. The hunter can then go to a_1, so that the mole's next position is at $0, b_2$, or c_2 and again the hunter must choose between (i) or (ii), or the hunter can go to b_1 or b_3, both resulting in the mole appearing at one of $0, a_2, b_2, c_2$, and the hunter chooses between sequence 1 or 2.

So we see that for every choice that the hunter can make in determining her move sequence, she will eventually repeat a previously occuring situation, showing that she cannot get the mole into a position where he has no safe choices. □

As in the case of Lemma 2.4, we can represent this with a move diagram (Figure 4).

In Figure 4, the top left leg is the leg containing the a_i vertices, and the bottom leg is the leg containing the c_i vertices. In this figure we include all of the possible hunter moves from any given position (including the ones that we did not discuss in Lemma 2.6 because they represent the hunter checking a vertex on which she knows the mole will not be). With this visualization, it is

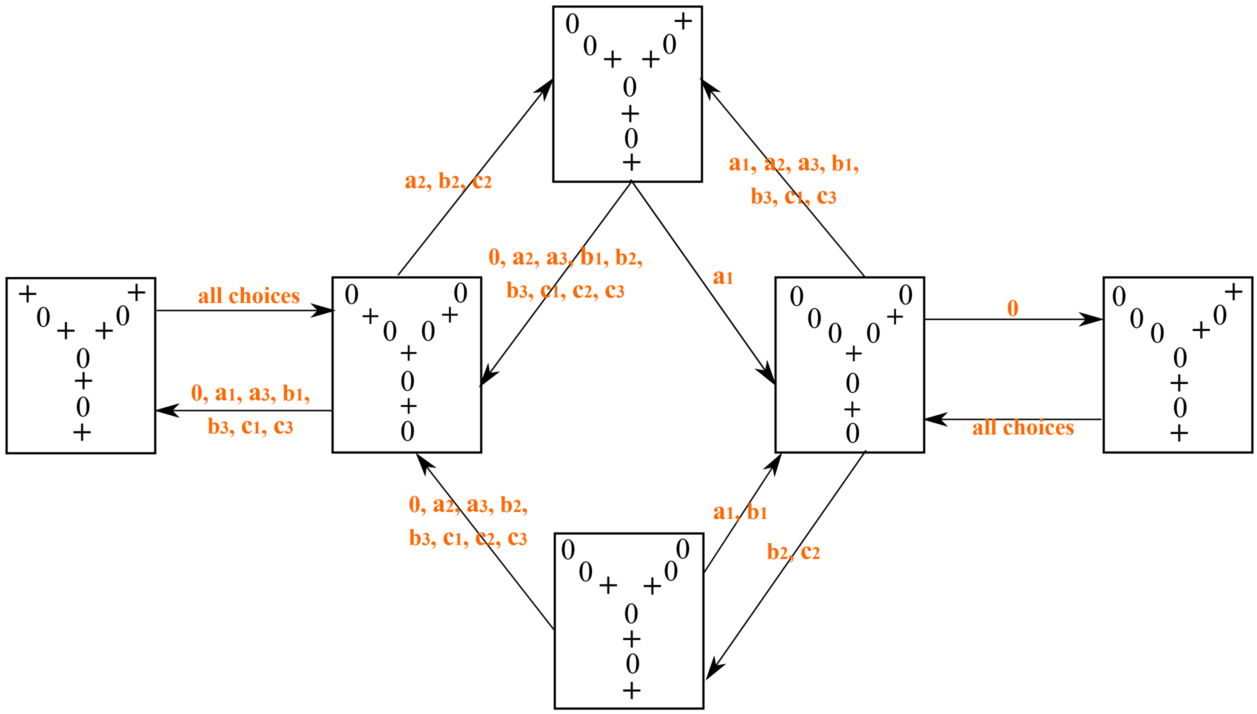

Figure 4: A diagram of the mole's choices in $S_{3,3}$ given any hunter move sequence

easy to see that there are only six distinct positions in which the hunter can find herself, and none of them include a guaranteed capture of the mole.

Lemmas 2.3 and 2.6 immediately yield the following restriction on potential hunter-win trees.

Corollary 2.7. *Any tree containing S as a subgraph is mole-win.*

2.3 Characterization

We are now ready to give a forbidden subgraph characterization of lobsters, and to characterize all hunter-win graphs.

Lemma 2.8. *A tree is a lobster if and only if it does not contain $S_{3,3}$ as a subgraph.*

Proof. Let G be a lobster with a maximal central path P. Suppose, for contradiction, that G contains $S_{3,3}$ as a subgraph and let $v \in V(G)$. If $v \in P$ then v can be adjacent to at most two vertices that are also on P. Since all vertices are within distance 2 of P, v is not on a length three path disjoint from P, and so v cannot be the hub vertex of $S_{3,3}$. If instead $v \notin P$ then either v is a leaf (and therefore not the hub of $S_{3,3}$) or v is at distance exactly 1 from P, and then all but at most one of its neighbors are leaves (and again, v could not be the hub of $S_{3,3}$). Hence, no vertex of G is the hub of $S_{3,3}$, which is a contradiction.

Suppose now that G is any tree not containing $S_{3,3}$ as a subgraph. Let P be a path in G of length $\operatorname{diam}(G)$ (i.e. a longest path in G) and label its end points x and y. Let v be any vertex in G that is not on P and let w be the vertex on P that is on the path connecting v to P. If $d(x,w) \leq 2$ then $d(v,w) \leq 2$ since otherwise the v—y path is longer than P. (Similarly, we have $d(v,w) \leq 2$ if $d(y,w) \leq 2$.) If both $d(w,x) > 2$ and $d(w,y) > 2$, then if $d(v,w) > 2$, $S_{3,3}$ would be a subgraph of G (with w as its hub). Therefore $d(v,w) \leq 2$ and so all vertices are within distance 2 of P, which makes G a lobster. □

Therefore, we have our desired characterization of hunter-win graphs.

Theorem 2.9. *A loop-free graph is hunter-win if and only if it is a lobster.*

3 Optimality of the Hunter Strategy

In Lemma 2.2, we described a strategy for the hunter which checked for (without loss of generality) an even mole first, and then for an odd mole. On a lobster with a longest central path P_m on m vertices, containing k knees, the strategy took time $2(m-2) + 4k$. It may seem, a priori, that considering these two types of moles separately (and furthermore, not making use of her ability to move about the graph unrestricted) may not be the most intelligent strategy for the hunter. However, we shall see that the hunter does not, in fact, have a faster strategy for capturing the mole.

Lemma 3.1. *Let G be a lobster with k knees and with a longest central path P_m with m vertices. The hunter cannot capture the mole in fewer than $2(m-2) + 4k$ steps.*

Proof. First we will show that each internal vertex on P_m and each knee must be visited at least twice in a winning hunter strategy. Suppose that there is a knee or internal vertex v_j that is never visited by the hunter on (without loss of generality) an odd turn. The vertex v_j has at least two neighbors. We will call one of the neighbors v_{j-1} and another v_{j+1}. Then consider the strategy ρ for the mole defined as follows

$$\rho_s = \begin{cases} v_j & \text{if } s \text{ is odd} \\ v_{j-1} & \text{if } s \text{ is even and } \sigma_s \neq v_{j-1} \\ v_{j+1} & \text{if } s \text{ is even and } \sigma_s = v_{j-1} \end{cases}$$

(where ρ_s is the mole's location at time s). Clearly the mole with this strategy cannot be caught by a hunter that fails to visit v_j on an odd turn.

Now let v be a hip in a lobster G with k_v adjacent knees. We claim that any winning hunter strategy $\sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_M\}$ must visit v at least $2k_v+2$ times. Suppose that v has knees $a_1, a_2, \ldots, a_{k_v}$. By u and w we will refer to the neighbors of v that are on P_m. Now suppose that the hunter visits v no more than $2k_v+1$ times. Then without loss of generality, v is visited at most k_v times on odd turns. Suppose that the first odd visit to v occurs at time $2r+1$. Then for all $s < 2r+1$ we define

$$\rho_s = \begin{cases} v & \text{if } s \text{ is odd} \\ u & \text{if } s \text{ is even and } \sigma_s \neq u \\ w & \text{if } s \text{ is even and } \sigma_s = u \end{cases}$$

Now suppose that we have defined ρ_s for all $s < 2m+1$ for some m where $\sigma_{2m-1} \neq v$ and for some j, $\sigma_{2m+1} = \sigma_{2m+3} = \cdots = \sigma_{2m+2j-1} = v$ (i.e. a string of j sequential odd-turn visits to v begins at time $2m+1$). Note that $N(v) \backslash \{\sigma_{2m+1}, \sigma_{2m+3}, \ldots, \sigma_{2m+2j-1}\} \neq \emptyset$ since $|N(v)| \geq k+2$ and $j \leq k$. Let x be a vertex in this set, and let $y \neq x$ be neighbor of x.

Then for all $s \in [2m+1, 2m+2j+1]$ we define

$$\rho_s = \begin{cases} x & \text{if } s \text{ is even} \\ y & \text{if } s \text{ is odd and } s \neq 2m+2j+1 \\ v & \text{if } s = 2m+2j+1 \end{cases}$$

This yields a mole strategy ρ which beats hunter strategy σ since by the definition of ρ, we have $\rho_s \neq \sigma_s$ for all s. □

Lemma 3.1 tells us that every knee must be visited twice, any internal path vertex must be visited twice, and any hip requires an additional number of visits equal to twice the number of knees adjacent to it. Therefore,the optimality of our hunter strategy is an immediate corollary, as summarized in the following theorem.

Theorem 3.2. *The hunter strategy defined in Lemma 2.2 is an optimal strategy.*

4 The Sneakier Mole

In our proof of Lemma 2.2, we relied heavily on the fact that the mole was forced to move at each step by making use of the fact that the mole has a fixed parity. What if this were no longer the case? We represent this variation by keeping the rules of the game the same, but adding loops to the graph at vertices at which the mole is allowed to sit.

This addition is quite unfortunate for the hunter, as can be seen in Lemma 4.1, below.

Lemma 4.1. *Any graph containing at least two loops is mole-win.*

Proof. First fix $n \geq 2$ and consider the graph P_n with a loop at both of its endpoints, with vertices labeled $1, 2, \ldots, n$ from left to right. Its diagram will behave very similarly to the diagram in Figure 2—any choice of first move for the hunter will return the mole to his initial situation. By Lemma 2.3, any graph containing as a subgraph P_k with a loop at each end point (for any $2 \leq k \leq n$) is mole-win. □

We can therefore restrict our attention to graphs containing a single loop. We will see that the mole can make good use of a single loop, but only in certain placements.

Lemma 4.2. *Any path P_n with a loop at a vertex that is at distance less than or equal to 2 from an endpoint is hunter-win (for all $n \geq 1$).*

Proof. Label the vertices of P_n from 1 to n moving left to right, and first place the loop at the vertex labeled "1." We claim that the following sequence of moves for the hunter will guarantee capture of the mole in $2(n-1)$ steps:

$$\mathcal{H} = \{n-1, n-2, \ldots, 2, 1, 1, 2, 3, \ldots, n-1\}.$$

In her movement from right to left, the hunter is progressively taking away options. For his first move, the mole can be anywhere but at n; for his second move, anywhere but at $n-1$; for his third move, anywhere but n and $n-2$; for his fourth, anywhere but $n-1$ and $n-3$; and so on. By the time she gets to 1, the mole's options are therefore only the odd vertices (that is, the diagram looks like $\boxed{+0+0+\cdots}$. By waiting one more turn at 1, the hunter turns the mole's diagram into $\boxed{0+0+0\cdots}$ and so as she moves back along the path to the right, she is taking away (at every other step) one more position of the appropriate parity, ensuring that the situation is just like in the original game on P_n with no loop.

Now suppose that the loop is at vertex 2 and consider the hunter strategy

$$\mathcal{H} = \{n-1, n-2, \ldots, 3, 2, 2, 3, \ldots, n-1\}.$$

An identical argument to the one above shows that by the time the hunter reaches 2 for the first time, the mole's diagram looks like $\boxed{0+0+0\cdots}$, and so by remaining at 2 for one more step and then continuing to the right, the hunter progressively removes one more position of the correct parity at every other step.

Finally, if the loop is at vertex 3, the hunter strategy

$$\mathcal{H} = \{n-1, n-2, \ldots, 3, 2, 3, 3, 2, 3, \ldots, n-2, n-1\}$$

behaves nearly the same way as in the previous arguments. When the cop gets to vertex 3 for the first time, the mole's diagram looks like $\boxed{+\ +\ +0+0+\cdots}$, so that following the move sequence to $3, 2, 3, 3$, the diagram looks as in Figure 5.

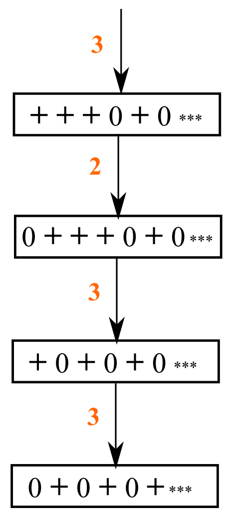

Figure 5: A diagram of the mole's choices after the hunter's first arrival at the loop

Now as in the previous two arguments, the hunter is progressively removing one more position of the correct parity from the mole's list of options at every other step. $\square$

Lemma 4.3. *Any lobster G with a maximal length central path P with a loop off of a vertex v (that is, the loop is either at v, at a knee adjacent to v, or at a foot at distance 2 from v) that is at distance less than or equal to 2 from an endpoint of P is hunter-win.*

Proof. Create two subgraphs, L and R of G, such that L is induced by v and all vertices on the path between it and the nearest path endpoint (and all neighbors of these vertices), and R is $G \backslash L$, as in the example of Figure 6. Since G without the loop is hunter win by Theorem 2.9, R is hunter-win by the contrapositive of Lemma 2.3. Therefore, following the first half of the strategy described in the proof of Lemma 2.2, the hunter can clear R for a mole of the proper parity (without loss of

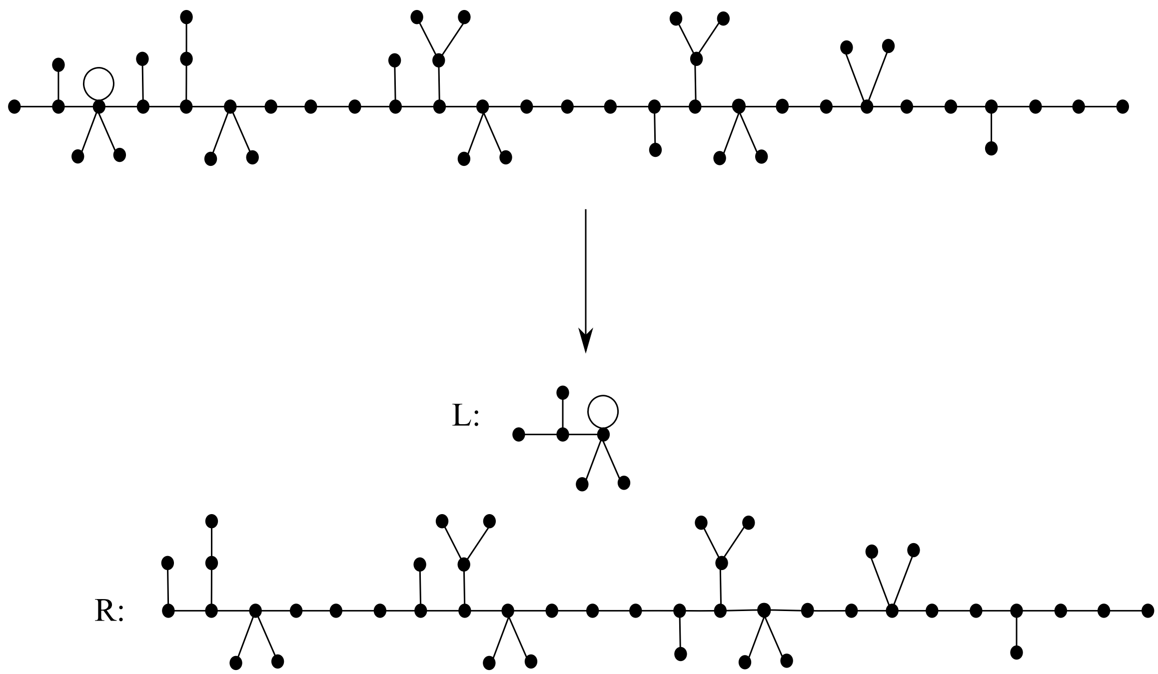

Figure 6: An example of a lobster with a loop, separated into L and R

generality, suppose she clears it for an even mole first, and we call this sequence of moves R_E, and let R_O be the reverse of R_E). Note that by using the loop, the mole can change his parity, but he cannot get into R as an even mole after R has already been checked, so when the hunter gets to the loop for the first time, the mole is either in L or is an odd mole in R.

There are precious few options for how L may look, up to symmetry:

(1) If v is an endpoint of the path, then L=$\{v\}$, and so the hunter wins with strategy $R_E - v - v - R_O$, much like in the first case of Lemma 4.2.

(2) If v is a vertex of the central path at distance 1 from the endpoint, then L is a star with v at its hub. Then once again the strategy $R_E - v - v - R_O$ captures the mole. Notice that if v is on a vertex off of the central path, adjacent to a vertex at distance 1 from the endpoint, then we could redefine the path so that v is on it.

(3) If v is on the path at distance 2 from the nearest endpoint, then either v has an off-path neighbor at distance 2 or it does not.

   (a) v does not have an off-path neighbor at distance 2.

Label the on-path neighbor of v in L by w and the endpoint of the path by x. Then when the hunter goes through R_E and hits v for the first time, the mole could go to x, w, an off-path neighbor of w, or v. Then the hunter goes next to w, so that the mole's options in L become w or an off-path neighbor of v. Now the hunter goes back to v so that the mole's options are x, an off-path neighbor of w, or v, and so when the hunter stays at v for one more step, the mole's options become only w or a vertex in R (of the not-yet-checked parity). Now the hunter simply turns around and heads back into R, removing one vertex of the appropriate parity at each move, just as in Lemma 4.2, and guaranteeing that the mole must be in R, and will be found at the end of the sequence $R_E - v - w - v - v - w - v - R_O$.

(b) The vertex v has an off-path neighbor at distance 2.

Label the k off-path neighbors which are **not** leaves (i.e. they are the middle vertex in a length 2 path starting at v and ending off of the central path) of v by $a_1, a_2, \ldots, a_k$. Notice that by maximality of the central path, all of the off-path neighbors of w are leaves. Now once the hunter completes R_E and hits v for the first time, the mole can go to any vertex in L for his next move. So the hunter can continue to cut off one more vertex of the proper parity at each turn, as before, by doing the following strategy:

$$R_E - v - w - v - a_1 - v - a_2 - v - \cdots - v - a_k - v - v$$

At this point, the mole must be at $a_1, a_2, \ldots$ or a_k. Now by going to $a_1 - v - a_2 - v - \cdots - v - a_k - v - w, v$, the hunter guarantees that the mole is in R, and so captures him during the rest of her sequence: R_O.

If v is a knee adjacent to a hip at distance 2 from the nearest path endpoint, then this situation can be made identical to the one in (2), and if v is on a foot at distance 2 from a hip which is itself at distance 2 from the nearest path endpoint, then a redefinition of the path yields the situation in (1).

Therefore, we have considered every possibility for L, and shown that they are all hunter-win. □

In all of the arguments in the proof of Lemma 4.2, we again make use of the fixed parity of the mole. Even though he can stay at the loop for a while, he cannot change his parity in an effective way (and get “behind” the cop on vertices of the same parity she already checked). Far from all graphs (or even paths) with a single loop are hunter-win, however—and it stands to reason that the graphs which fail to be hunter-win are precisely those on which the hunter cannot clear the graph to one side of the loop while simultaneously keeping the mole from switching his parity in an effective way.

Lemma 4.4. *Any graph containing as a subgraph a copy of one of the following graphs is mole-win.*

G_1*: P_7 with a loop at its middle vertex.*

G_2*: P_7 with one looped non-path vertex, attached to the middle vertex.*

G_3*: P_7 with a path of length 2 attached to the middle vertex, with a loop at its endpoint.*

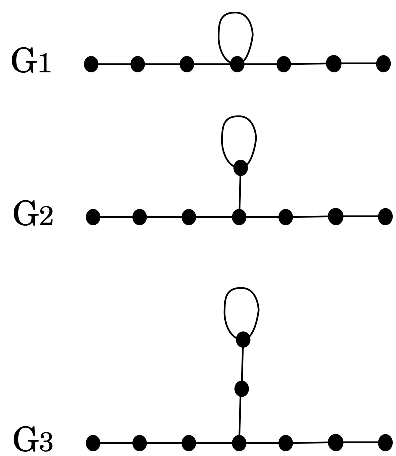

Figure 7: The graphs G_1, G_2, and G_3

Proof. G_1: Label the vertices of G_1 from (without loss of generality) left to right with $1, 2, \ldots, 7$ (so that the loop is at 4). Then the move diagram for G_1 is the one in Figure 8 (with symmetric situations identified on the diagram).

G_2: Label the vertices of G_2 from left to right on the path with $1, 2, \ldots, 7$ again, and label the remaining neighbor of 4 with an 8. The move diagram for G_2 is the one in Figure 9

G_3: Finally, label the vertices of G_3 from 1 to 8 as on G_2 and label the remaining neighbor of 8 with a 9. The move diagram for G_3 is the one in Figure 10.

In all three cases, no move sequence of the hunter can guarantee capture of the mole. And consequently by Lemma 2.3, no graph containing G_1, G_2, or G_3 as a subgraph can be hunter-win.

□

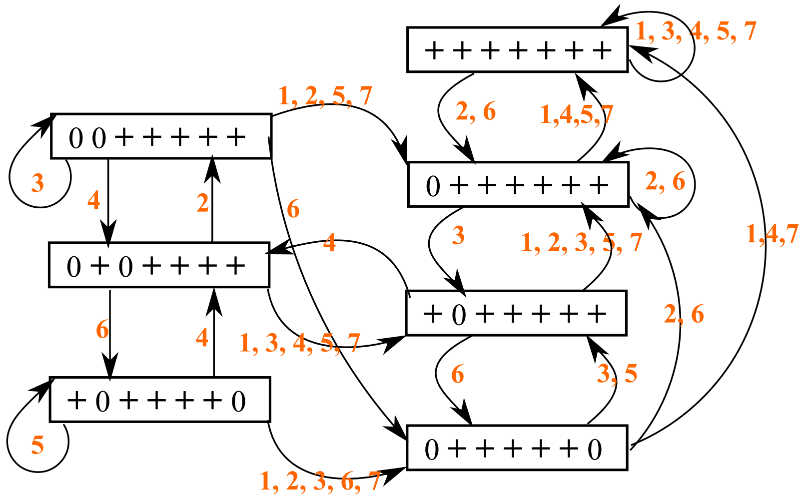

Figure 8: Move diagram for G_1

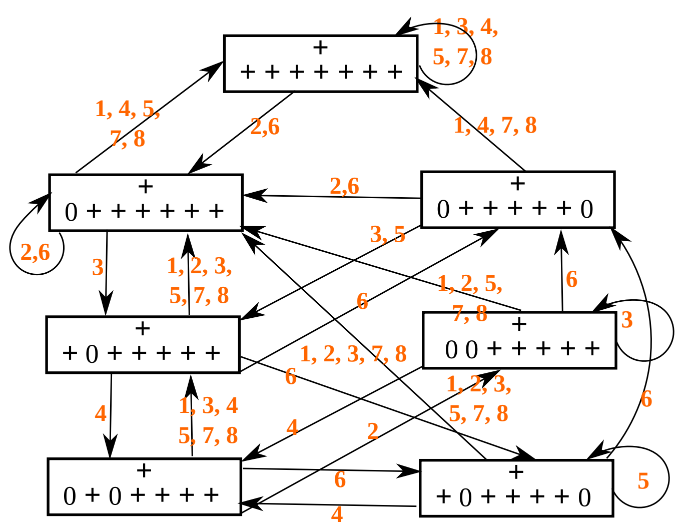

Figure 9: Move diagram for G_2

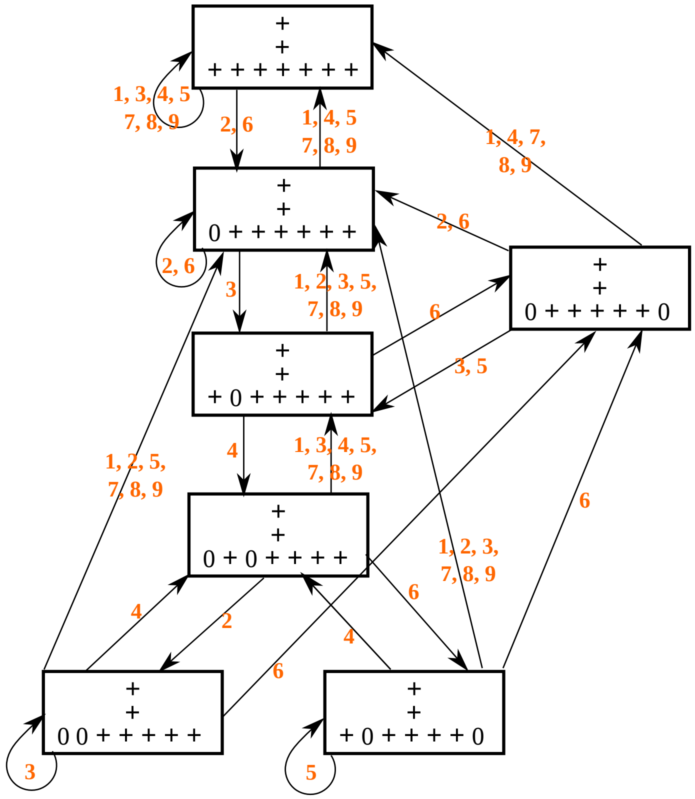

Figure 10: Move diagram for G_3

The previous two lemmas immediately yield the following characterization of hunter-win graphs (in which we allow loops).

Theorem 4.5. *A graph is hunter-win if and only if it satisfies all of the following:*

1. *It is a lobster.*
2. *It contains no more than one loop.*
3. *If it does contain a loop, the loop is either*
   (a) *at a vertex on the central path which is within distance 2 of the nearest endpoint, or*
   (b) *at a vertex off the central path which is a neighbor of such a path vertex.*

References

[1] A. Bonato, P. Golovach, G. Hahn, and J. Kratochvil. The capture time of a graph. *Discrete Math.*, 309:5588–5595, 2009.

[2] T. Gavenčiak. Games on graphs. Master's thesis, Charles University, Prague, 2007.

[3] D. Hess. Private communication, 2012.

[4] R. Nowakowski and P. Winkler. Vertex to vertex pursuit in a graph. *Discrete Math.*, 43:235–239, 1983.

[5] A. Quilliot. *Homomorphismes, points fixes, rétractations et jeux de poursuite dans les graphes, les ensembles ordonnés et les espaces métriques.* PhD thesis, Université de Paris VI., 1983.