Research on Foxes and Hares Game

Chunyang Ding, Matthew Hur, and Nate McDermott

Further Mathematics, Interlake High School, Bellevue, Washington

January 15, 2015

The Fox-Hare game is a graph game in which foxes move to try to catch an escaping hare. We analyzed the minimum number of foxes needed to catch a hare, defined as the "fox number," for different graphs varying in planar dimension and loop structure for various numbers of foxes. Starting with a base case of 1 fox and 1 hare, we found that any planar graph containing sub-loops solely of 3 nodes or less have a fox number of 1. Graphs with a fox number of 2 are found to have at least one sub-loop with 4 or greater nodes. We then extrapolated to 3 foxes and found that non-planar graphs with connections forming isomorphic patterns at all boundaries have a fox number of 3 or greater. Through these results, we developed the Fox-Hare Theorem which states that finite graphs with boundary nodes connected to at maximum 2 separate loops must be at maximum a 2-fox graph.

1 Introduction

The Fox-Hare game is a very interesting way of analyzing graphs. It is essentially a simple chase game, but can be adapted to reveal interesting ideas of thinking about graph theory. In order to explore this concept, we will first examine some base cases of the scenario, and gradually add more conditions to expand our views of this idea.

2 Analysis

2.1 Definitions

The Fox-Hare game is a two person game played on any type of graph. In the base scenario, there is 1 hare and n foxes, placed on random spots throughout the graph. The two players take turns moving their characters to adjacent nodes, or not moving for a turn. There are two end scenarios: The foxes can win by "capturing" the hare by occupying the same node as the hare, or there can be stalemate where the hare can always escape regardless of where the foxes go. In

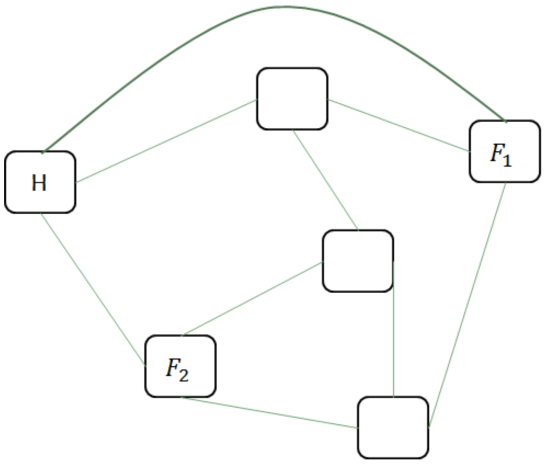

Figure 1: A basic planar graph where the foxes have cornered the hare.

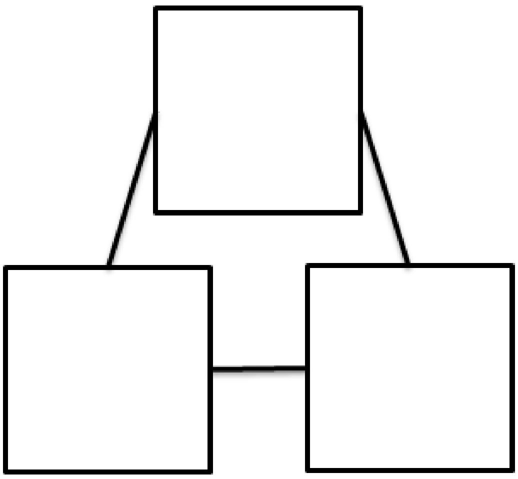

Figure 2: A complete graph where one fox could always capture the hare.

addition, for the base case, we require that all graphs be planar, and that each fox or hare can move at maximum to 1 adjacent node (ie, no “daisy chain” moves). These requirements will be changed as we progress onwards to more advanced cases. Sub-loops of the graphs are loops in which all included nodes are of 2 degree with respect to each other. They are denoted by the number of nodes, e.g. 4-loop for a 4 node graph in the shape of a quadrilateral. See Figure 1 for an example of a standard graph that could be played. Clearly, the foxes will win this game.

When we have such planar graphs, we will seek to find the ”fox number” for it. Fox number is defined as the minimum number of foxes required on an graph to capture the hare. We assume that both the foxes and hares play with optimal strategy, and that the initial positions of the foxes and hares are least optimal for the foxes. Fox number can sometimes be tricky to get, because optimal conditions must be followed.

2.2 1-Fox Games

Before we explore the n-fox game, we will choose to first explore the 1-fox game. We will attempt to identify the characteristics that allow for a graph to be won with only 1 fox. These graphs would seem to be the most basic graphs, but we will fully investigate to confirm these exact traits.

Our first note is that all complete graphs must be a 1-fox graph. A complete graph is defined as a graph where every node is connected to every other node. The example in Figure 2 would therefore be a 4-node complete graph. Clearly, because every node is connected with every other node, a fox occupying any node can capture a hare on any other node.

Next, we note that if a single node is connected to all other nodes, then that graph must be a 1-fox graph. Our reasoning is that once the fox obtains that single node, they would be able to capture the hare that exists on any other node.

Next, we progress to a more complex series of logical determination methods. We use the following algorithm to first simplify a graph in order to better analyze it.

The first step in this algorithm is to eliminate any "dead-end" nodes. If a node is of degree one, meaning that it has only one connection, then it can be identified as a useless node. If the hare ever goes into this node, then the foxes can close off the one escape route and win. Removing the node would therefore not remove any information from the graph.

Afterwards, we repeat this step as many times as necessary until all of those nodes are taken away. We "prune" the graph until it reveals only the viable spots that are connected in several different ways.

Finally, we examine the remaining graph for loops of 4 or more. A loop is defined by a connection of n-nodes of degree 2. If there is a loop of 4 or more nodes, we postulate that the graph must *NOT* be a 1-fox graph.

Our final stipulation implies that any finitely tessellated group of 3-loops in a graph will be a 1-fox graph. Through experimentation, we have shown that any large graph can be solved in this fashion. The optimal strategy for the fox is to chase the hare until the hare reaches the boundary of the graph, at which point the hare must start circling the boundary, and the fox can circle nodes adjacent to the boundary, eventually catching up to the hare. As long as the fox plays with this optimal strategy, it can always limit the possible moves that the hare has. Eventually, the hare is forced of to the side, and is then captured.

With this, we conclude our conditions for 1-fox games. Any graph that can be analyzed with our above conditions will result in capture by a single fox, given our original definitions. The next step seems to be to find the conditions for 2-fox games

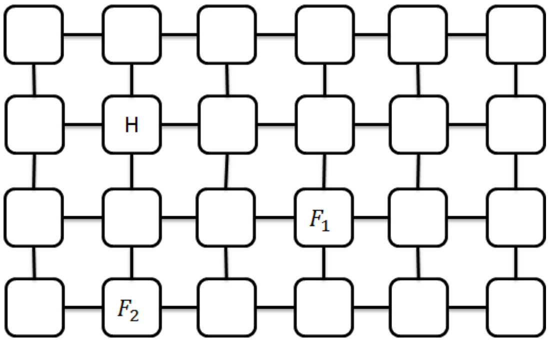

Figure 3: A planar 4-loop graph in which the foxes have cornered the hare.

2.3 2-fox game

To analyze the 2-fox game, we first break our previous conditions of 1-fox games by creating a graph of 4-loops rather than 3-loops. As shown in Figure 3, it requires a minimum of 2 foxes in order to capture the hare. An even simpler case of only one 4-loop can also be considered. Clearly, if there exists only 1 fox, the hare and fox could always go in a loop, and result in a stalemate condition.

It is rather difficult to directly analyze the differences between the 2-fox game and the 1-fox game. It seems that as long as a loop greater than 4 exists, it must be at least 2 foxes. However, there is no upper limit of the number of foxes that would be appropriate for the graph. In order to further analyze, we must compare the 2-fox graph with the hypothetical 3-fox graph. What makes our 2-fox graphs solvable with only 2 foxes?

2.4 3-fox game

Through experimentation, it appears as if there are no planar graphs that allow for the 3-fox stipulation. Therefore, we began to consider other 3D shapes that, when translated into graph theory, might be able to

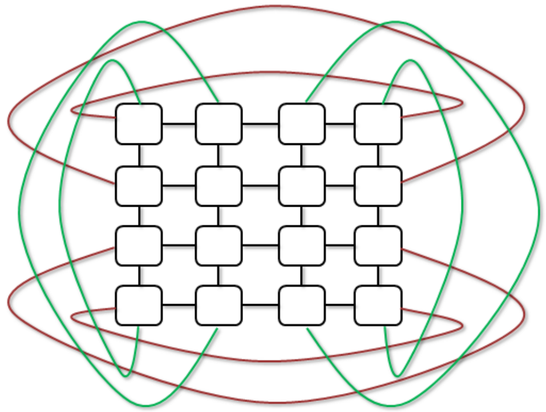

Figure 4: A non-planar graph where the foxes have cornered the hare.

satisfy the conditions. The first one of these that we found was a hollow cylinder that connects the center tube with the outer tube. The graphical representation of this sphere can be found in Figure 4.

We find that on this graph, 2 foxes are not sufficient to capture. This is because there is no possible way for the foxes to corner the hare in any corner or boundary of the graph, and that even if 2 foxes cover all possible nodes where the current hare could escape to, neither fox can move in for the kill without giving up an escape route. Therefore, we find that this cylinder graph is clearly a 3-fox graph.

What differentiates this graph from the 2-fox graph? As mentioned before, this graph is no longer planar as there are connections that cross each other. However, it also means that there is no clearly defined boundary to this graph - any point can be defined as the center of the graph, as it is isomorphic in all directions. There is therefore no "boundary" or "corner" in the graph. Regardless of where the hare and fox moves, it is not possible to differentiate the new position from the previous moves.

Can we set this as our key differential between the 2-fox and 3-fox conditions? We note that any planar graph must have a boundary, and therefore a hare can be pushed towards that side. However, what defines an boundary? If we observe our graphs, we realize that any boundary or corner point can be defined as a node that is part of at maximum 2 different loops.

Given all of this information, we conclude with the Fox-Hare theorem:

(Fox-Hare Theorem) Let G be a finite graph. Given that any node in G is connected to at maximum 2 separate loops, G must be at maximum a 2-fox graph.

3 Extension

Up to this point, we have analyzed all of our graphs using our initial conditions. That is, we assume that all graphs have only one hare, and that the hares and foxes can only move to one adjacent node (or choose to maintain position). Therefore, we shall begin to break down these conditions and extend the exploration.

3.1 n-hare Games

Let us change the conditions that there requires to be only 1 hare on the graph. Instead, we shall place n hares on the graph. In order for the fox to capture the hare, we shall define that both fox and hare need to continuously occupy the same location, thus blocking off that location. This results in the obvious condition that for n hares, a minimum n foxes are required. For the foxes to win, there must be sufficient foxes to capture each hare!

At this point, there appear to be three different possibilities that encompass all graphs: an n fox game, an $n + 1$ fox game, and an $n + 2$ fox game. Initially, it seems that it is quite simple for what might have previously been considered an $n+1$ fox game to be turned into an n fox game through clever manipulation by the foxes. If the foxes could force a hare to die in a certain location, the fox might be able to change a graph to be more favorable for the foxes.

However, we must also consider that the hares are very smart as well! The hares always have the option of not moving and staying on a single location. Therefore, the hares can strategically choose the locations where they die, such that at least one hare can survive in a stalemate. In this, we could then apply the same strategies of the 1-hare game to all of the n-hare games.

Of course, there are boundary conditions - if there are more hares than half the number of nodes on the graph, for instance, the foxes could not win. In addition, if the hares could essentially wall off a single hare, preventing any fox from coming through, there can never be a win situation either. Therefore, it is difficult to establish all of the n-hare cases as it is so close between different ideals.

3.2 n-hop Games

Finally, we attempt to analyze the n-hop games. This game requires only 1 hare, but the hare is allowed to travel up to n hops each turns. What could our analysis tell us about the differences of n-hop games on the same graphs as viewed before?

First, we shall analyze the differences between a 2-hop game and an 1-hop game, using the layered 4-loop that can be found in Figure 3. As we can tell, the 2 foxes are no longer sufficient for capturing the single hare, although it was possible in the previous setting. Therefore, we see that we must entirely reevaluate the n-hop games in order to build it upwards.

An interesting extension is that now, we not only must define the hares and foxes, but also need to define the graph. Without a clear definition of the graph, it could quickly be the limiting factor. If we wanted to find all possible 2-fox graphs for a certain n-hop hare, we would need to know the dimensions of the graph. Otherwise, a scaled version of a graph with the same characteristics could become a higher fox number graph just by becoming bigger. In these scenarios, it is common for the hare to find a hole in the line of foxes, and escape through.

For simplicity, we will only analyze 4-loop graphs (found in Figure 3) with an n-hop hare. We will build up our knowledge using this case scenario, and allow the rest of the permutations of cases to be left as exercise for the reader.

3.3 n-hop, 4-loop Games

Considering a n by n 4-loop graph in which only the hare is given n hops, we examine the ability of the hare to escape various numbers of foxes, up to $n-1$.

When there are $n-1$ or fewer foxes, the hare can always create a stalemate by escaping the encroaching foxes. The hare can stay in the corner of the graph and as soon as the foxes approach, escape through the inevitable hole in the line of foxes. For example, with a 3-hop fox on a 4 by 4 graph, with 1 or 2 foxes, the hare can easily outmaneuver the foxes.

When we consider the case of 3 foxes on the same graph, however, we find that the hare cannot escape.

4 Discussion

Through our analysis of variations of this simple graph game, we have found many interesting aspects of graph theory. This analysis has given insight into a novel branch of graph theory concerned with moving entities on nodes. As we examined up to the 3 dimensional non-planar graph, some further extensions to consider include higher dimensional graphs and relationships between numbers of hares and foxes to the "fox number." If we are to extend the problem such that there are values assigned to each connection, we could better model real conditions in which hares have to consider different factors such as underbrush or food availability before moving. By creating algorithms for understanding such graphs, we could extrapolate to algorithmically understand real-life behaviors capable of taking into account many factors.

Thus, the Fox-Hare problem has many facets still left to explore. Our two extension cases, leave us with many more questions on this topic for extension. We have determined several ways to continue extending the problem, including the n-hop game, the n-graph game, and the value assignment to each connection. These are very interesting aspects for future study.