Game Theory : The Fox and the Holes

Yueyang Han, Ryan O'Donnell, Gefei Wang, Sara Bazouane

December 2025

1 Setting

Old MacDonald (a farmer) had a farm. On the farm there lived a fox. Unfortunately, the fox keeps destroying the farmer's crops and the farmer has had enough. After much thought, he has asked 4 mathematicians to use techniques from MATH60141/MATH70141 Introduction to Game Theory to help him formalise and solve this problem.

Assume we have a **fox**, n **holes** adjacent to one another, and an unsuspecting **farmer**. The fox arrives at the farm on the first night, eats one of the farmer's crops, and then picks one of the n holes to hide in at random, unbeknownst to the farmer. On subsequent nights, the fox pops out of its hole, eats a crop, and then moves to an adjacent hole. Each morning, the farmer inspects a hole of his choosing to see if the fox is in it. If the fox is in the chosen hole, congratulations: the farmer has found the fox, the fox is wrung and the game is over. If not, the process repeats itself the following night until the fox is caught.

Our primary goal is to answer the question: what is the strategy the farmer can employ to **guarantee** catching the fox AND also minimising the number of crops lost? We will consider this as the number of holes n varies, using theoretical results aided by computational simulation.

2 Formalising using Game Theory

2.1 Modelling Assumptions for the Game

We aim to give a formal and rigorous setup of the setting we described in Section 1. We can model the scenario as a sequential zero-sum **pursuit-evasion** game G played on a linear graph with n nodes (holes). There are two players:

- Player A (*The Fox*): Wants to evade Player B (for as long as possible)
- Player B (*The Farmer*): Wants to catch Player A

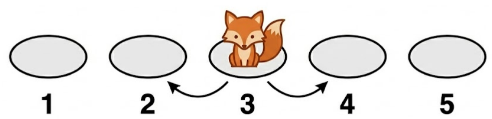


Figure 1: Fox in a Hole problem with $n = 5$ holes [1]

The game then proceeds in discrete time steps $t \in \mathbb{N} = \{1, 2, 3, ...\}$. These represent nights for the fox, and days for the farmer, i.e. the fox hides in a random hole on night $t = 1$, and the farmer chooses their first hole on day $t = 1$. We denote the set of n holes as $H = \{1, 2, ..., n\}$.

Position of the fox: We define the initial position of the fox as $h_1 \in H$, and all subsequent positions analogously as $h_t \in H$ for $t \in \mathbb{N}$.

Cost function: We define the number of crops eaten by the time the game ends to be the **cost** to the farmer g_B - this corresponds to the number of days t taken for the farmer to catch the fox. With this setup, the farmer comes into play on $t = 1$ after the fox has destroyed a crop and moved to an adjacent hole, so the lowest cost possible is $g_B = 1$. Note that this is also the payoff of the fox g_A, creating a zero-sum game. We will simply write $g := g_A = g_B$ for simplicity. The farmer aims to minimise this cost and the fox aims to maximise this payoff.

2.2 The Role of Imperfect Information and Perfect Recall

A key element of this game is **imperfect information**. This means that we initially model the farmer as not knowing the location of the fox h_t at any time t until the fox is caught, and vice versa for the fox. In a game of imperfect information, any mixed strategies we define can also be thought of as **behaviour strategies**: a randomised choice of a move for each information set of the game [2], where each move made by each player game can be modelled as a decision node, and each new day gives rise to a new information set. In addition, we assume both players have **perfect recall**, and know which holes they have previously checked, regardless whether this affects their decisions or not.

2.3 A formal model of the rules of movement of the fox

Initially, we assume that on each night the fox moves to an adjacent hole, and that it cannot remain in the same hole on two consecutive nights. In particular, if the fox is in hole i at night t, then at night $t+1$ it must move to one of the adjacent holes in $\{i-1, i+1\}$, unless the fox is at one of the boundary holes 1 or n, in which case the fox is forced to go to hole 2 or $n-1$ respectively. These are non-absorbing boundaries: the fox must go back if it reaches one of the last holes.

For the non-boundary holes, define $\alpha_{t,i} \in [0, 1]$ and $1 - \alpha_{t,i} \in [0, 1]$ as the probability of moving left and right from hole i at time t respectively. This can vary depending on the hole the fox is in (i) and the time t. The fox begins at hole 1 chosen randomly based on the distribution:

$$\mathbb{P}(h_1 = i) = \pi_i, \quad i \in H, \quad \text{where } \pi_i \in [0, 1], \quad \sum_{i=1}^{n} \pi_i = 1$$

and for all $t \geq 2$ follows the transition probabilities:

$$\mathbb{P}(h_{t+1} = j \mid h_t = i) = \begin{cases} \alpha_{t,i}, & \text{if } 1 < i < n \text{ and } j = i - 1, \\ 1 - \alpha_{t,i}, & \text{if } 1 < i < n \text{ and } j = i + 1, \\ 1, & \text{if } i = 1 \text{ and } j = 2, \\ 1, & \text{if } i = n \text{ and } j = n - 1, \\ 0, & \text{otherwise.} \end{cases}$$

In particular, we have $\mathbb{P}(h_{t+1} = h_t \mid h_t = i) = 0$ for all i and t, ensuring that the fox always moves to a different hole each night but can return to a previously visited hole on later nights.

2.4 Strategies and Strategy Profile of the Fox

We can think of the transition probabilities of the fox's movement as inducing a **strategy profile** $\mathbb{F}$, containing all possible strategies $\mathbf{f}$ that satisfies the fox's movement restrictions. Each strategy has the realisation:

$$\mathbf{f}_{\pi_\mathbf{i}, \alpha_\mathbf{t,i}} = (h_1, h_2, h_3, h_4, ...)$$

where this infinite dimensional vector **f** is a shorthand with each entry representing the hole chosen at each information set (i.e. each night), with the h_i defined as before, and the probabilities $\pi_i, \alpha_{t,i}$ unique to each strategy (this is the behavioural component). We can also think of this as a mixed strategy as it can be represented as probabilistic combinations of pure strategies, where pure strategies have $\pi_i,\ \alpha_{t,i} \in \{0,1\}$ for all $i \in H, t \geq 2$. Imagine that the farmer knows that the strategy profile $\mathbb{F}$ from which the fox's strategies are drawn from (i.e. the possible moves the fox is allowed to make), but does NOT know which strategy the fox is using (i.e. the values of $(\pi_i, \alpha_{t,i})$), until the fox is caught.

An important point is that if the fox is found on day T, the game ends, so we can reduce the strategy to a finite vector $(h_1, ..., h_T, *)$, with $*$ indicating that the game is over by day T.

Note: $f_{\pi_i,\alpha_{t,i}}$ are strictly speaking the realisation of the strategies generated by each $(\pi_i, \alpha_{i,t})$ combination for all $i \in H, t \geq 2$. For convenience, we will refer to these as the strategies themselves going forward.

2.5 Strategies and Strategy Profile of the Farmer

The farmer is at liberty to pick any one each day in any order he likes, with no restriction. Using the same notation as for the fox, the farmer has a strategy profile $\mathbb{S}$ containing all strategies **s** (and all mixed strategies associated with these) with realisation:

$$\mathbf{s} = (x_1, x_2, x_3, x_4 ...), \quad \text{where } x_t \in H \quad \forall\ t \geq 1$$

Having fixed the fox's and farmer's strategy profiles, we are ready to analyse some of the farmer's responses.

3 The Irrational Farmer

We first imagine that the farmer, seething with rage, is unable to make rational decisions, and simply chooses a hole $x_i \in H$ at random to check on each day. We therefore model the farmer as having the behaviour strategy $\mathbf{s_r} = \{x_1, x_2, x_3, ...\}$ with x_t modelled by discrete uniform random variables such that:

$$\mathbb{P}(x_t = i) = \frac{1}{n}, \qquad \text{for all } t \in \mathbb{N} \text{ and } i \in H.$$

Again, we can reduce the strategy to a finite-dimensional vector using the $*$ notation if the game ends on day $t = T$.

A first question comes to mind: with this strategy, what is the expected number of crops that will be lost before the fox is found, i.e. the **expected cost** to the farmer?

3.1 Theoretical Analysis of Expected Payoff

Claim: Under the farmer's "random" behaviour strategy $\mathbf{s_r}$, the farmer's expected cost is:

$$g(\mathbf{f}, \mathbf{s_r}) = n \quad \text{for all fox strategies} \quad \mathbf{f} \in \mathbb{F}$$

Proof: Fix any day $t \in \mathbb{N}$, assume that the fox has not yet been caught before day t. On day t the fox is in some hole $h_t \in H$. Observe from the farmer's behaviour strategy $\mathbf{s_r}$, the distribution is independent of the farmer's current choice x_t, as the distribution

$$\mathbb{P}(x_t = i) = \frac{1}{n} \quad \text{for all } i \in H, \text{ and all } t \in \mathbb{N},$$

is independent of the fox's movements and strategy **f**. Note for a capture to occur on day t, x_t must equal h_t.

Then, conditional on h_t, the probability that the farmer catches the fox on day t is

$$\mathbb{P}(\text{fox captured on day } t \mid h_t) = \mathbb{P}(x_t = h_t \mid h_t) = \frac{1}{n},$$

because x_t is chosen uniformly at random from H and is independent of h_t. Therefore, irrespective of which hole the fox is in, the conditional probability of capture on day t (given that the fox has not been before t) is

$$\mathbb{P}(\text{fox captured on day } t \mid \text{fox not captured before day } t) = \frac{1}{n},$$

and the conditional probability of the fox surviving that day is

$$\mathbb{P}(\text{fox survives day } t \mid \text{fox not captured before day } t) = 1 - \frac{1}{n} = \frac{n-1}{n}.$$

Since the farmer's choices x_t are independent over time, these conditional probabilities are the same on each day t. So we can consider the random variable T denoting the time at which the fox is captured, with distribution:

$$\mathbb{P}(T = t) = \left(1 - \frac{1}{n}\right)^{t-1} \frac{1}{n}, \qquad t \in \mathbb{N},$$

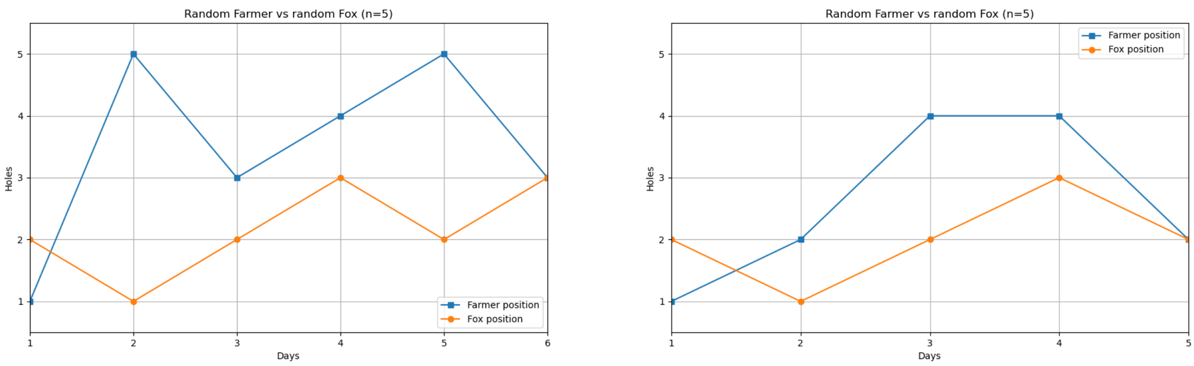
which is exactly the pd.f. of the geometric distribution with parameter $p = \frac{1}{n}$, with expectation $\mathbb{E}[T] = \frac{1}{p} = \frac{1}{1/n} = n$.

Therefore, under the farmer's random strategy $\mathbf{s_r}$, the expected number of crops lost before the fox is caught is exactly n.

Intuitively this makes sense: if there are n holes, no matter where the fox is each day and what strategy it uses, there is a $\frac{1}{n}$ chance of the farmer randomly catching the fox. So an average the fox will be caught after n days, and will have time to reap a payoff of n crops.

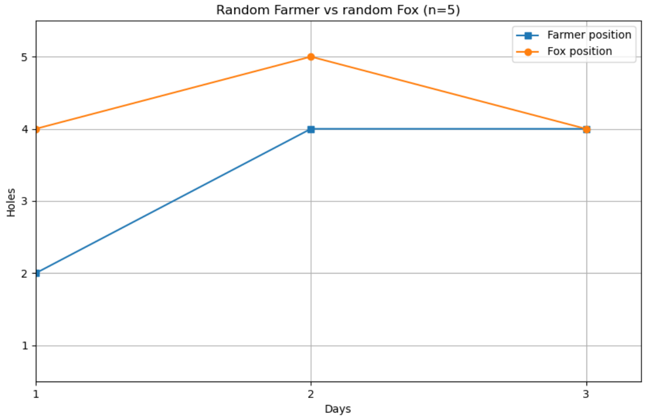
3.2 Computational validation

Before validating the theoretical result obtained in the previous section, we first illustrate how the game unfolds in practice. In the following, we simulate the strategy $f_r \in \mathbb{F}$, with parameters $\pi_i = \frac{1}{n}\ \forall\ i \in H$ and $\alpha_{t,i} = \frac{1}{2}\ \forall\ i \in H, t \geq 2$. This corresponds to the fox initiating a random walk on the holes, with equal probability of moving left or right (where possible).



(a) Fox captured on day 6

(b) Fox captured on day 5



(c) Fox captured on day 3

Figure 2: Three independent simulations of the game for $n = 5$ under random farmer and random fox behaviour.

From the three illustrative simulations above, we observe that the capture time varies from one play to another, but the values obtained (6, 5, and 3) remain reasonably close to the theoretical expected payoff of n (in this case 5). This already suggests that the analytical result is meaningful in practice.

To validate this behaviour more rigorously, we performed a large-scale computational experiment. For each value of n from 1 to 100, we simulated 10,000 independent plays of the game under the random farmer and random fox strategies. For each n, we computed the empirical mean capture time T and compared it to the theoretical expected value n. The results, shown in Fig. 3, reveal an excellent agreement between the empirical averages and the line $T = n$, thereby confirming the theoretical prediction (for number of holes $1 \leq n \leq 100$).

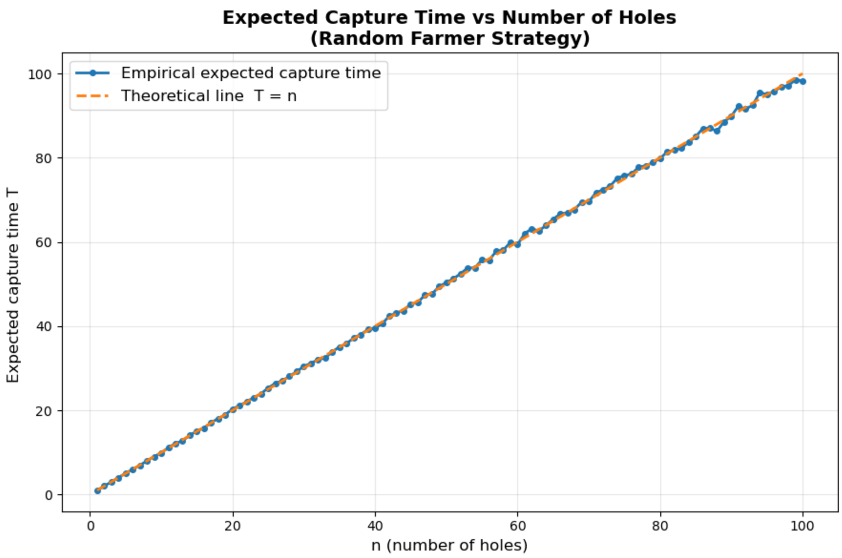


Figure 3: Empirical expected capture time T compared to the theoretical prediction $T = n$.

4 Finding an optimal strategy: The farmer develops a brain

The farmer's random strategy $\mathbf{s_r}$ has a clear drawback: in finite time, it is possible that the fox evades capture entirely, i.e. there is no upper bound on the fox's payoff. The farmer is not satisfied by this: we can only imagine that as much as he wants to minimise crop loss, his primary goal is actually to capture the fox with certainty, and as fast as possible. This results in a new goal for the farmer:

Goal for Farmer: Find a strategy $\hat{s}$ that guarantees capture of the fox in finite time. If there are multiple strategies that achieve this, pick the one that guarantees catching the fox in the least number of days. We denote the day of capture as day T, where $T \in \mathbb{N}$.

4.1 Optimal Strategy for $n \leq 5$ holes: A systematic approach

We first investigate the problem for small n to formulate a pattern. It is trivial to see that for $n = 1$, we have $H = \{1\}$ and $h_1 = 1$ with the farmer's only choice for a strategy being $\hat{s}_1 = (1, *)$, so the game ends on day 1. For $n = 2$, consider the simple pure strategy $\hat{s}_2 = (1, 1, *)$ i.e. check hole 1 on day 1 and then again on day 2. So if the fox's starting position $h_1 = 1$, the fox is caught on day 1 and if $h_1 = 2$ that means $h_2 = 1$ and the fox is caught on day 2. We can see that this is an optimal (minimax) strategy since no strategy can guarantee a capture on day 1.

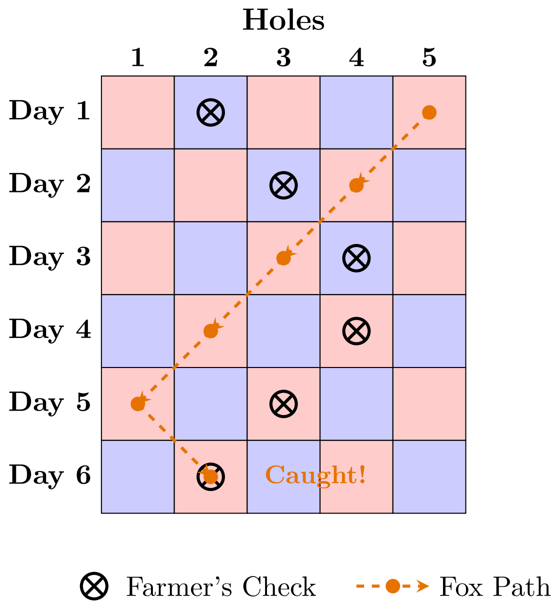
The smallest non-trivial variation of the game is with $n = 3$. The fox has only four possible moves: $1 \to 2, 2 \to 1, 2 \to 3, 3 \to 2$. Now, we consider the pure strategy $\hat{s}_3 = (2, 2, *)$. We can see that this strategy is guaranteed to catch the fox within 2 days due to the moveset of the fox (the fox must occupy hole 2 during one of the first two days). Thus, this is also optimal (no strategy can guarantee a capture on day 1).

A similar approach can be taken for the case $n = 4$. Listing all possible moves of the fox : $1 \to 2, 2 \to 3, 3 \to 4, 4 \to 3, 3 \to 2, 2 \to 1$, we can verify that the pure strategy $\hat{s}_4 = (2, 3, 3, 2, *)$ is optimal. Using this strategy, if the fox is not found on day 1, the set of possible holes the fox could have been at is $h_1 \in \{1, 3, 4\}$, and therefore on day 2, $h_2 \in \{2, 3, 4\}$. If the fox is then not found on day 2, the fox must have been at $h_2 \in \{2, 4\}$, so $h_3 \in \{1, 3\}$. Checking hole 3 again on day 3 and if the fox is not found means $h_3 = 1$, and therefore $h_4 = 2$, whereupon the fox is caught. The maximal payoff for the fox is therefore 4, and $\hat{s}_4$ cannot be improved as no strategy guarantees capture of the fox in 3 or fewer days (see checkerboard argument in 4a).

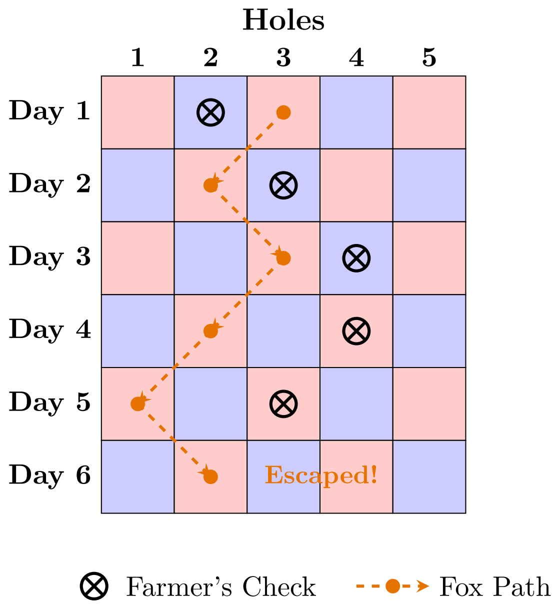
Now we move on to the classical variation of the Fox and the Holes problem with $n = 5$. We first find a strategy for the farmer that is guaranteed to catch the fox and show that it is optimal. Our proposed optimal strategy is $\hat{s}_5 = (2, 3, 4, 4, 3, 2, *)$. To better understand this strategy, it is helpful to visualise the game on a grid with the x-axis as the holes $\{1, ..., 5\}$ and the y-axis as time $t = 1, 2, ...$ as shown in Fig. 4a.

An important thing to notice is that the starting hole for the fox determines which cells it can reach in its path. Since time increases by one each day, and the fox moves across by one, it stays on the same parity of cell. We have coloured the cells in a checkerboard pattern to emphasise this (so if the fox starts on a red cell, its path is along the red cells).

This is how we come up with this strategy for the farmer. We have set up this "blockade" of checks such that the first half of the checks $(2, 3, 4)$ eliminates all paths of the fox starting in even holes (the "blue paths"), and the second half $(4, 3, 2)$ eliminates all paths of the fox starting in the odd holes (the "red paths"). Fig. 4a shows how one "red path" of the fox starting in hole 5 is caught. No matter the starting position and path the fox takes, this strategy is guaranteed to catch the fox within 6 days, i.e. $\max_{\mathbf{f} \in \mathbb{F}} g(\mathbf{f}, \hat{s}_5) = 6$.



(a) Visualisation of the Optimal Strategy



(b) Escape Path ($h^* = 2$, C = Red)

Figure 4: Comparison of strategies for $n = 5$. (a) The optimal strategy clears both colours. (b) Removing a single check on day 6 creates a safe "Red" cell, allowing the Fox to escape.

Now, to show that this strategy is indeed optimal (i.e. there is no other strategy that is guaranteed to catch the fox in fewer days), we suppose that the farmer uses any strategy with fewer than 6 checks. By the Pigeonhole Principle, one of the colours, label this C, will have fewer than 3 checks. We now consider the three **internal** columns (holes $2, 3, 4$). Since colour C has fewer than 3 checks, there must exist at least one internal column h^* containing no checks of colour C.

We can then construct an escape path for the fox:

1. On days when h^* is colour C, the fox moves (or stays at) hole h^* (Note if $t = 1$, the fox starts here)

2. On days when h^* is the opposite colour, the fox chooses an adjacent hole $\{h^* - 1, h^* + 1\}$ that is not checked (Also if $t = 1$, the fox starts here). This will also be colour C (since parity is conserved).

Since h^* is an internal hole, it has two valid adjacent holes. Also, note that a row (day) cannot have more than one check since the farmer can only check one hole per day. Therefore, the farmer cannot check both adjacent holes simultaneously, and the fox simply moves to the neighbour that is not checked. This forms a path for the fox to escape from the farmer; an example of this can be seen in Fig. 4b.

This shows that 6 checks is the minimum to guarantee capturing the fox, and so $\hat{s}_5 = (2, 3, 4, 4, 3, 2, *)$ is optimal (minimax) for $n = 5$. An analogous argument shows that 4 checks is minimax for $n = 4$.

Note: This is not the unique optimal strategy, e.g. $(2, 3, 4, 2, 3, 4, *)$ is also optimal in this sense for $n = 5$, and we can also illustrate this by the checkerboard argument. However, all other optimal strategies also have a maximal cost of 6.

4.2 General Form of the Optimal Strategy for $n \geq 4$

Claim: The pure "double-sweep" strategy for $n \geq 4$ holes, $\hat{s}_n = (2, 3, ..., n-1, n-1, ..., 3, 2, *)$, is a minimax strategy for the farmer regardless of which strategy the fox plays, i.e.

$$\max_{\mathbf{f} \in \mathbb{F}} g(\mathbf{f}, \hat{s}_n) \;=\; \min_{\mathbf{s} \in \mathbb{S}} \{\max_{\mathbf{f} \in \mathbb{F}} g(\mathbf{f}, \mathbf{s})\}.$$

Proof idea: The proof is a generalisation of the ideas from section 4.1. We observe that $\hat{s}_n$ consists of a first sweep of the non-boundary holes $(2, 3, ..., n-1)$. By the nature of the fox being forced to move to adjacent holes, the fox will be caught in these $n-2$ days if it started on any **even** numbered hole (e.g. these are the blue paths in Figure 4a). If the fox is not caught by day $n-2$, the farmer does another sweep, this time $(n-1, ..., 3, 2)$. The farmer exploits its ability to stay at hole $n-1$ for two consecutive days, thereby catching the fox in the remaining $n-1$ days if the fox started at an odd hole (e.g. red paths in Figure 4a). A similar colouring argument can show that no other strategy that guarantees capture of the fox can do this in fewer days (we can always construct an escape path for the fox using the construction in section 4.1).

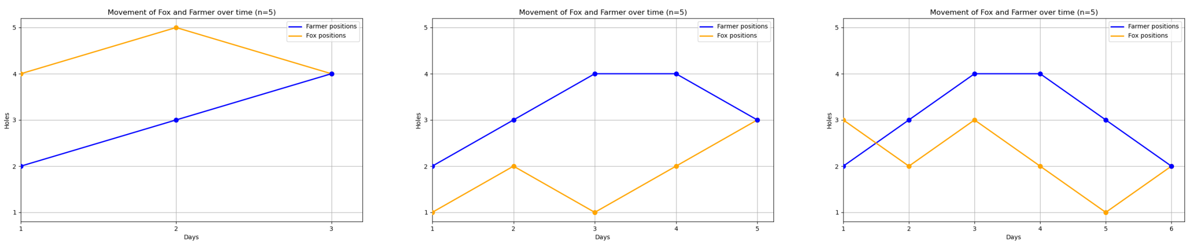
Corollary: The maximum cost to the farmer under the strategy $\hat{s}_n$ is the minimax payoff:

$$\max_{\mathbf{f} \in \mathbb{F}} g(\mathbf{f}, \hat{s}_n) = 2n - 4$$

4.3 Computational Simulation

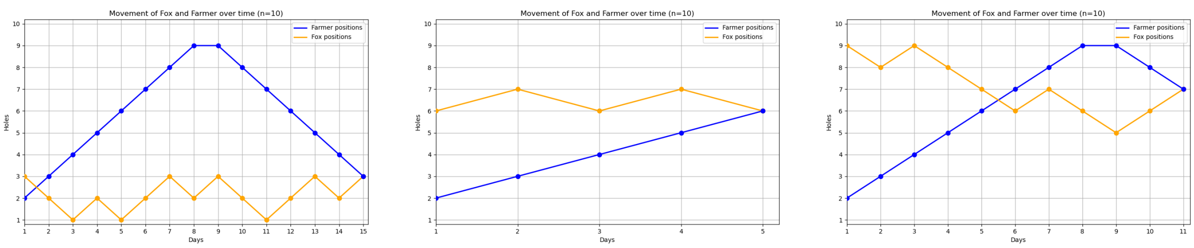
4.3.1 Illustrative plays of the game for $n = 5$ and $n = 10$

Let's visualise how the game unfolds in practice. Below, we present three sample runs of the game for the case $n = 5$ and for $n = 10$, where the farmer uses the optimal deterministic strategy $\hat{s}_n$. To simulate the fox moves, we again use the random walk strategy f_r.



(a) Fox captured on day 3 (b) Fox captured on day 5 (c) Fox captured on day 6

Figure 5: Three independent plays of the game under optimal farmer strategy $\hat{s}_5$ and random fox behaviour f_r (case $n = 5$).



(a) Fox captured on day 15 (b) Fox captured on day 5 (c) Fox captured on day 11

Figure 6: Three independent plays of the game under optimal farmer strategy $\hat{s}_{10}$ and random fox behaviour f_r (case $n = 10$).

4.3.2 Computational Validation of the Optimal Farmer Strategy $\hat{s}_n$

This strategy is optimal in the sense that it guarantees capturing the fox in at most $2n-4$ days for any initial fox position. In other words, the worst–case capture time is bounded by a simple linear function of n, while no strategy can guarantee a uniformly smaller bound. To validate this statement computationally, we performed the following experiment. For each value of $n \in \{4, \ldots, 100\}$, we simulated 10,000 independent plays of the game, computed the capture time (cost) for each run, and recorded the maximum cost observed. Plotting these values against the theoretical upper bound $2n-4$ as in Fig. 7 shows perfect agreement, confirming that the strategy never requires more than $2n-4$ days in practice.

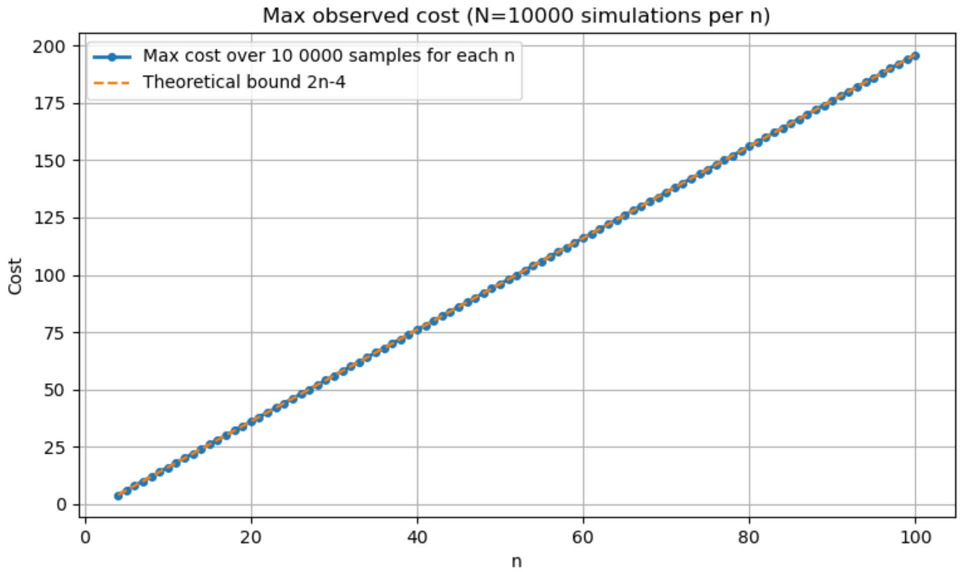


Figure 7: Maximum observed capture time for $\hat{s}_n$ over 10,000 simulations for each $n \in \{4, \ldots, 100\}$, compared to the theoretical bound $2n-4$.

Comparison of expected capture time: Optimal vs. Random Farmer

We now compare the expected capture time achieved by the optimal deterministic farmer strategy $\hat{s}_n$ with that of the random farmer strategy s_r studied earlier. For each $n \in \{1, \ldots, 100\}$, we simulated 10,000 independent plays of the game and computed the empirical mean capture time T under the deterministic strategy (corresponding to the expected cost $g(f_r, \hat{s}_n)$). The resulting curve is plotted against the theoretical prediction of the expected cost using the random farmer strategy, $g(f_r, s_r) = n$ in Fig. 8.

The results demonstrate that the deterministic strategy also performs better on average than the random farmer when considering expected capture times for all given n, while also providing a strict worst–case guarantee.

	$n=5$	$n=6$	$n=7$	$n=8$	$n=9$	$n=10$
Random Farmer Expected Payoff $g(f_r, s_r)$	5	6	7	8	9	10
Optimal Farmer Expected Payoff $g(f_r, \hat{s}_n)$	3.550	4.360	5.523	6.339	7.511	8.333

Table 1: Expected capture time/payoffs for the random and optimal strategies for the values $5 \leq n \leq 10$ based on 10^8 simulations. We observe $g(f_r, \hat{s}_n) < g(f_r, s_r)$ for all n.

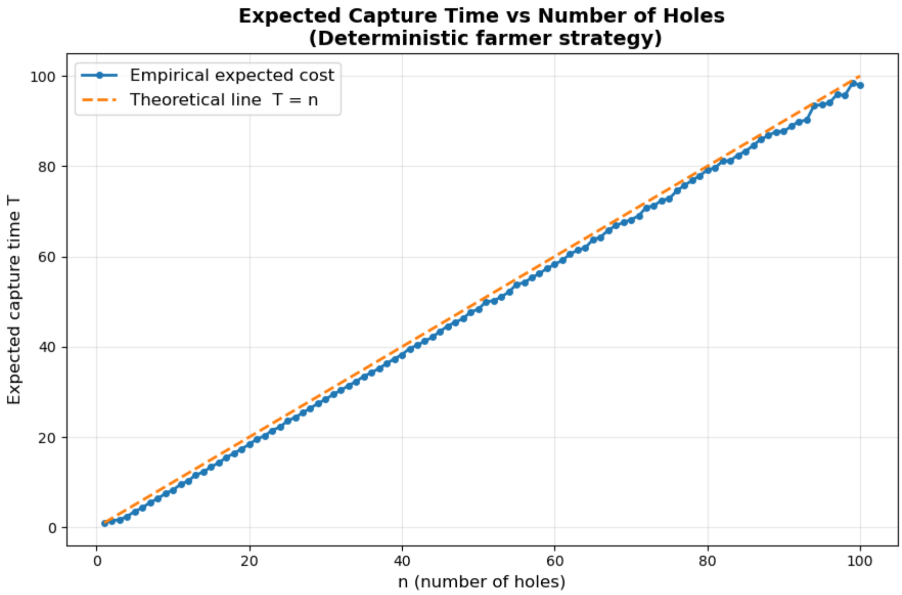


Figure 8: Comparison of the expected capture time T under the optimal deterministic farmer $\hat{s}_n$ strategy with the theoretical prediction $T = n$ of the random farmer strategy s_r.

5 The Psychic Fox

We now consider the scenario where the fox is now able to know the farmer's strategy $\mathbf{s}$ and the holes he picks (possibly because the farmer is very disruptive when checking the holes each day), but the farmer still doesn't know anything about the position of the fox, only its possible movement rules as defined earlier given by the strategy profile $\mathbb{F}$. Hence, the fox now has **perfect information** about the farmer, while the farmer does not. Let's restrict ourselves to analysing the **best response** of the fox (out of its possible strategies in $\mathbb{F}$) to the farmer's strategies $\mathbf{s_r}$ (random) and $\mathbf{\hat{s}_n}$ (optimal).

We first consider the farmer playing their optimal strategy $\mathbf{\hat{s}_n} = \{2, \ldots, n-1, n-1, \ldots, 2, *\}$ for $n \in \mathbb{N}$ holes. We established that this is a minimax strategy for the farmer with maximal cost $g(\mathbf{f}, \mathbf{\hat{s}_n}) = 2n - 4$. Knowing that the farmer will play this strategy, the fox ought to find a best response strategy that achieves the maximum payoff of $2n-4$. This is indeed possible, and a possible solution is given by the pure strategy: $\hat{\mathbf{f}} := (1, \ldots, n-3, n-2, n-3, \ldots, 1, 2)$, where the fox is guaranteed to evade capture until day $t = 2n - 4$ on hole 2. Other best responses with the same payoff are also possible.

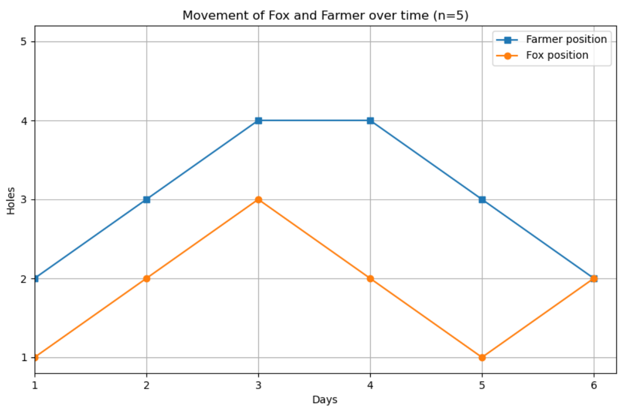


Figure 9: Evolution of farmer and fox positions over time under optimal strategies (case $n = 5$).

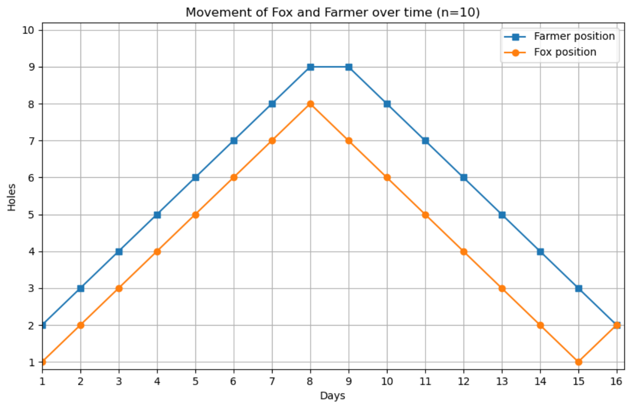


Figure 10: Evolution of farmer and fox positions over time under optimal strategies (case $n = 10$).

Now consider the fox's best response against the farmer's random strategy $\mathbf{s_r}$. We recall from our earlier claim in section 3.1 that the farmer's expected cost when playing $\mathbf{s_r}$ was $g(\mathbf{f}, \mathbf{s_r}) = n$ for all fox strategies $\mathbf{f} \in \mathbb{F}$, so the same holds for the fox's payoff (zero-sum game). We can directly conclude that all of the fox's strategies are payoff-equivalent best responses in this scenario, with expected payoff n - the fox cannot strategically optimise their response against the random farmer!

This leads to the interesting conclusion that for $n > 4$ holes (where $g(\hat{\mathbf{f}}, \hat{\mathbf{s}}) = 2n - 4 > n = g(\mathbf{f}, \mathbf{s_r})$), the farmer would lose more on average if they played their "optimal" strategy. Therefore if the farmer is aware that the fox has perfect information about their every move, and wants to minimise expected crop loss, they should pick randomly rather than "optimally", so that the fox cannot capitalise on the fact that they know information about the farmer.

6 Conclusion, Advice for Farmer and Model Limitations

There is good news for the farmer. For $n \geq 5$ holes, f the fox's moves are restricted by our assumption and the fox has imperfect information, then the farmer can use the **"double-sweep"** strategy $\hat{\mathbf{s}}$. This guarantees capture of the fox in the least amount of time (within $2n - 4$ days), and also has a reasonable expected crop loss from our simulations.

However, were the fox to know the farmer's strategy in advance, the farmer has an argument to avoid the optimal strategy, and instead use the random guessing strategy. Against the optimal strategy, the fox can construct a best response that always hits the maximum possible payoff $2n - 4$, higher than the expected payoff n of the random strategy, but the fox can't pre-empt randomness. Although the farmer cannot guarantee capture of the fox in finite time with this strategy, the probability of the fox evading the farmer for large t is extremely small, and as $t \to \infty$, this probability tends to 0, so the fox will eventually be caught.

Note that in this analysis, we have only thoroughly considered two main strategies for the fox and farmer, i.e. random and optimal. There are many other possible strategies in the profiles $\mathbb{S}$ and $\mathbb{F}$, such as assigning different probabilities to going left and right for the fox, and choosing the boundary nodes with a reduced probability compared to the middle node as a random fox spends less time at the boundaries. This makes the scenario fascinatingly complex and warrants thorough analysis.

However, it could be worse for the farmer! If we relax the movement constraint of the fox such that we allow it to stay in its current hole, i.e. $h_{t+1} = h_t$, opening up a whole new profile of strategies, then the farmer cannot guarantee catching the fox in finite time with an optimal strategy $\hat{s}$. In this case, the random farmer strategy once again proves the most versatile (see simulations). This reflects real life scenarios, where if the your opponent knows your strategy in advance, it is often better play completely randomly, so your moves are unpredictable.

Appendix

A Links to simulations

Graphical illustrations of expected payoff with different fox and farmer strategies: https://github.com/bazouanes-ICL-Ensimag/Game-Theory-Fox-and-Holes.git
Values for the expected payoff with different strategies, including allowing the fox to stay in its own hole (for interest): https://github.com/Ryan7od/GameTheoryCW

B Motivating ideas from MATH60141/70141 and other modules

In this report, we used many concepts from the Game Theory report. This includes a rigorous formalisation of the scenario as a zero-sum game using notions such as strategies and strategy profiles, best responses, pure strategies and mixed strategies, as well as ideas from Chapter 8 (Mastery) involving incomplete information and considering mixed strategies as behaviour strategies with respect to decision nodes. We looked at the idea of expected payoff and cost and ran simulations to support this. Care was taken to make sure all notation was rigorous but also concise, which resulting in some simplifications where necessary (such as interchanging a strategy and its realisation as a sequence of moves). Justification was always given in these cases. Similarly, any notions of domination of strategies and equilibria were deliberately not considered in this report due to the game itself and strategy profiles not being finite: there are too many strategies to consider. A simplified version of the game may warrant analysis of equilibria - this would certainly be interesting.

In addition, the scenario lends itself well to being modelled as a Discrete Time Markov Chain. We therefore used some terminology and notation from *MATH60045/MATH70045 Applied Probability* such as transition probabilities and random walks. A more Markovian approach to the problem could be worth investigating in the future.

C Use of Generative AI

Generative AI was used sparingly in this report to plot tables in Overleaf. All game-theoretic ideas and computational experiments were original or motivated by further reading.

References

[1] LOGICALLY YOURS. Seemingly impossible fox puzzle —— fox in a hole —— asked in google interview. YouTube video, https://www.youtube.com/watch?v=0Prp9n7XfP8, 2025. Accessed: 2025-12-07.

[2] Sam Brzezicki. Introduction to game theory module : Math60141/70141. Lecture Notes, 2025.