CSI 4900: Honours Project Final Report

Wei Hu

whu061@uottawa.ca

University of Ottawa — September 8, 2020

Contents

1 Introduction

An Erdös-Selfridge-Spencer (ESS) game consists of an attacker A, a defender D, and a game state represented with an array S of size k, where $S[i] \geq 0$ is the number of pieces on the board that is i levels away from the top of the board.

At the beginning of each turn, the attacker partition S into two arrays S_1 and S_2 such that for all $0 \leq i < k$, we have $S[i] = S_1[i] + S_2[i]$, with $S_1[i] \geq 0$ and $S_2[i] \geq 0$.

Definition 1.1. Let v be a value function, which maps a level i to a real number representing the value of a piece at level i. A value function can be represented by an array $(w_0, w_1, w_2, ..., w_k)$ where:

$$v(i) = w_i.$$

Definition 1.2. Let S be board state with k levels. The value function applied to S is defined as:

$$v(S) = \sum_{i=0}^{k} S[i]v(i).$$

A defender would apply its value function to the each of S_1, S_2. It would then destroy the set deemed more valuable by the value function. The pieces in position 0 of the surviving set is said to have gained *tenure* and are removed from the board. We then left shift the surviving array and set S equal to it.

The game ends when S is an empty array (note the game ends in at most k turns). The $score$ is defined to be the total number of pieces that have gained tenure when the game ends.

Note: the board is zero-indexed. Thus, a piece on level 0 will get tenure and yield a point for the attacker (unless it is destroyed by the defender this turn).

Theorem 1.1. *The optimal value function v_* for the defender , which minimizes the score, is defined as:*

$$v_*(i) = \frac{1}{2^{i+1}}. \quad (1)$$

The proof is found in [0]. The proof relies on what we call the *splitting lemma*, which we will prove in the next section.

Lemma 1.2. *Given a finite multiset S consisting strictly of powers of $\frac{1}{n}$, for some $n \in Z^*$, let $s = \sum_{e \in S} e$. If $s \geq 1$, then $\exists S' \subset S$ such that $\sum_{e \in S'} e = 1$.*

The original ESS game declares the attacker the winner if $score > 0$ at the end of the game. We define a “score-keeping" version of the game where:

- the attacker wins if $score > v_*(initial\_state)$

- the defender wins if $score < v_*(initial\_state)$
- we declare a draw if $score = v_*(initial\_state)$

1.1 Biased Defenders

We observe the optimal value function satisfies $v(i) = 2v(i+1)$. By deviating from this equality, we can create sub-optimal defenders. These defenders either overvalue pieces that are close to the top of the board, or those that are far away. This bias is dependent on the direction of the inequality.

Definition 1.3. A farsighted defender is a one whose value function satisfies:

$$v(i) < 2v(i+1). \tag{2}$$

for all $0 \leq i < k$, where k is the length of the board.

Definition 1.4. A nearsighted (myopic) defender is a one whose value function satisfies:

$$v(i) > 2v(i+1). \tag{3}$$

for all $0 \leq i < k$, where k is the length of the board.

Intuitively, we can think of a farsighted defender as one which overvalues pieces that are close to the bottom of the board, while a nearsighted one would overvalue pieces that are near to the top of the board.

1.2 Challenges

In the original paper [0], the focus was on training defenders. Defenders have an action space of size two; attackers, however, have an $O(2^n)$ action space.

Although the paper presents a method for training attackers, it forces the attacker's action space to be made linear. This reduction increases the proportion of optimal moves in the search space (making the game easier for the attacker). Furthermore, it is only guaranteed that the reduced space contains the optimal move for the optimal defender (and we have shown the optimal move for the optimal defender is not always optimal for a suboptimal defender).

Since we only get feedback whenever a piece gets tenure, rewards are sparse. Most of the attacker's actions are bad and lead to no reward, thus it is difficult to get good training feedback.

2 Proof of Theorems

Lemma 2.1 and 2.2 can be used to give alternate proof of the splitting lemma (of which [0] makes use for proving theorem 1.1). Furthermore, these lemmas will be the foundation in our proof of the optimal strategy for playing biased defenders.

Lemma 2.1. *Given a finite multiset S consisting strictly of powers of $\frac{1}{n}$, for some $n \in Z^+$. If $\sum_{e\in S} e \geq 1$, then $\exists S' \subset S$ such that $\sum_{e\in S'} e = 1$.*

Proof. We define the concept of a *promotion*. If there exists $A \subset S$ satisfying $|A| > 1$ and $\sum_{e\in A} e = (\frac{1}{n})^k$ for some $k \in Z^*$. Then we would consider $T = (S - A) \cup (\frac{1}{n})^k$ to be a promotion of S. We would call S the *parent* of T. We note $\sum_{i\in T} i = \sum_{j\in S} j$.

Furthermore, $\forall B \subset T, \exists C \subset S$ such that $\sum_{i\in B} i = \sum_{j\in C} j$. (Every element in T that is not $(\frac{1}{n})^k$ has greater or equal multiplicity in S than in T. Thus, unless B contains all copies of $(\frac{1}{n})^k$ in T, we are able to choose $C \subset S$ made up of the same elements. Suppose B contains all copies of $(\frac{1}{n})^k$ in T, then we would use $A \subset S$ to account for one copy of $(\frac{1}{n})^k$. For the remaining elements in B, there will be a corresponding copy in S.)

Every promotion has fewer elements than its parent. Let $S >_p S_1 >_p S_2 >_p ... >_p S_n$ denote a chain of promotions with S_n having no further promotions. Since S is finite, the length of the chain is finite. We observe that $\forall k \geq 1$, the multiplicity of $\frac{1}{n}^k$ in S_n is less than or equal to $n-1$ (If it were n or greater, then we have a promotion by letting A equal n copies of $\frac{1}{n}^k$). Thus, the sum of those elements is strictly upper bounded by $\sum_{i=1}^{\infty} \frac{n-1}{n^i} = (n-1)\sum_{i=1}^{\infty} \frac{1}{n^i} = 1$. Yet we have $s = \sum_{e\in S} e = \sum_{e\in S_n} e \geq 1$, thus there must be at least one element in S_n in the form $\frac{1}{n}^k$ where $k = 0$, that is to say $1 \in S_n$.

Thus, there is a subset of S_n that adds to 1; per our previous observation, we conclude there must also be a subset of S which adds to 1 □

Corollary. *Given a finite multiset S consisting strictly of elements in the form $(\frac{1}{n})^r$, for some $n \in Z^+, r \geq k$. We have $\forall s \leq k$, if $\sum_{e\in S} e \geq \frac{1}{n}^s$, then $\exists S' \subset S$ such that $\sum_{e\in S'} e = \frac{1}{n}^s$.*

The *splitting lemma* is the special case where $n = 2$ and $k = 1$.

Lemma 2.2. *Let S be a board state containing only pieces on levels greater than i. Suppose $v_*(S) \geq v_*(i)$, then $\exists S^{'} \subset S$ such that $v_*(S') = v_*(i)$.*

Proof. Note, we have $v_*(S) = \sum_{j=i+1}^{k} S[j]\frac{1}{2^{j+1}} \geq v_*(i) = \frac{1}{2^{i+1}}$. From there, dividing $\frac{1}{2^{i+1}}$ from both sides we get: $\sum_{j=i+1}^{k} S[j]\frac{1}{2^{j-i}} \geq 1$.

Create a multiset M by including $S[j]$ copies of $\frac{1}{2^{j-i}}$ for each $i+1 \leq j < k$. We can apply lemma 2.1 to M and conclude there is subset M' of M which adds to 1.

Based on how M is defined, we know every element e in M corresponds to a piece on the $v_*^{-1}(e\frac{1}{2^{i+1}})$ level of S. We can form a bijection between M and S

Thus, we know there being a subset S' of S corresponding to $M^{'}$ satisfying:

$v_*(S') = \sum_{e \in M'} e\frac{1}{2^{i+1}} = \frac{1}{2^{i+1}} \sum_{e \in M'} e = \frac{1}{2^{i+1}} = v_*(i)$. □

Theorem 2.3. *When playing against an optimal defender, the optimal approach for the attacker is to make the value according to v_* of the two partitioned sets as close as possible.*

Proof. We sketch a proof:

We can show by modifying the proof from [0] that when playing against an optimal defender, the best score that can be achieved from a starting state S is $\lfloor v_*(S) \rfloor$.

We know the max value of any piece according to v_* is $\frac{1}{2}$. By repeatedly applying the corollary of lemma 2.1, we can partition S into $2\lfloor v_*(S) \rfloor$ subsets each with value of at least $\frac{1}{2}$ according to v^*.

By evenly distributing those subsets, we can partition S into 2 subsets each with with value of at least $\frac{1}{2}\lfloor v_*(S) \rfloor$ according to v^*.

We can guarantee the value of the surviving subset is greater than or equal to $\frac{1}{2}\lfloor v_*(S) \rfloor$. After the left shift, the value of the surviving set according to v^* is doubled and the tenured pieces are removed from the board. Thus, at every step we have $\Delta score = \Delta v^*(S)$.

We notice the following invariant $current\_score + v_*(current\_state) = \lfloor v_*(start\_state) \rfloor$.

In at most k steps we will have, $v_*(current\_state) = 0$. We thus have $current\_score = \lfloor v_*(start\_state) \rfloor$, which means we achieve the best possible score.

We complete the proof by noting that when we make the value according to v_* of the two partitioned sets as close as possible, we will achieve the necessary condition which is to produce two sets each with value of at least $\frac{1}{2}\lfloor v_*(S) \rfloor$ according to v^*. □

2.1 Farsighted Defenders

Lemma 2.4. *Let A, B be two states, and let v be the value function of a farsighted defender. If $\operatorname{argmax}_x\{B[x] > 0\} < \operatorname{argmin}_x\{A[x] > 0\}$ and $v(B) > v(A)$ then $v_*(B) > v_*(A)$*

Proof. By contradiction, $v*(A) \geq v*(B)$. We can apply lemma 2.2 and map each piece in B to a subset of A with equal value.

Note, we are able to make those subsets of A disjoint. Every time we make a mapping we can "remove from consideration" the pieces used in this mapping to produce an A' and B' satisfying $v*(A') \geq v*(B')$ to which we can reapply lemma 2.2.

This process is repeated until every piece in B is mapped. While v_* deems the pieces on both sides of the mapping to be equal, v will not.

WLOG, suppose a piece at level i in B is mapped to $M \subset A$. Consider any piece in M and suppose it is on level j, we note: $\frac{v(j)}{v(i)} > \frac{v*(j)}{v*(i)}$, since $v(i) < 2v(i+1)$ for any farsighted defender.

Thus, we have $\frac{v(M)}{v(i)} > \frac{v*(M)}{v*(i)} = 1$. We've mapped every piece in B to a disjoint subset of A that is considered more valuable by v. It must be that $v(A) > v(B)$. □

Theorem 2.5. *The following algorithm maximizes the real value of the surviving pieces at the end of each turn against any farsighted defender.*

```
Algorithm 1: Minimizing $v_*$ Value of Destroyed Set Against Nearsighted Defenders
Input: a board position $S$ with $k$ levels, and the value function of a nearsighted defender $v$
Result: a partition of the board $(S_1, S_2)$ which guarantees the subset that the defender will
        destroy has the lowest possible value according to $v_*$
Beginning from the row furthest from tenure, and going from left to right, we start by uniquely
  labeling each piece on the board with an integer starting at one.
$\gamma \leftarrow \frac{v(S)}{2}$;
$resulting\_set \leftarrow$ an array of zeroes of length $k$;
$cumulative \leftarrow$ an array of tuples $(cd, level)$.
The tuple at the $i$-th index has for $cd$ the sum of the biased values of all the pieces up to and
  including the $i$-th piece, and $level$ is the level of that $i$-th piece;
$current\_index \leftarrow cumulative.length - 1$;
while $\gamma > 0$ and $current\_index > 0$ do
    if $cumulative(current\_index - 1).cd < \gamma$ then
        $best\_set[cumulative(current\_index - 1).level]$++;
        $\gamma$-=$v(cumulative(current\_index - 1).level)$;
    end
    $current\_index$--;
end
return $(resulting\_set, (S - resulting\_set))$;
```

Proof. Label each of the pieces with an integer in the same manner that is described in the algorithm. Essentially, the algorithm looks for the piece with the largest index which satisfies the condition that the sum of the biased values of the current piece and all the pieces that come after it is greater than γ.

We can then be certain that the current piece can be contained in the set that we want the defender to destroy, by showing that there is no reason to include any elements with a smaller index. By contradiction, assume there is some set S that contains one or more element of smaller index and has minimal value according to v_* and is greater than γ according to v.

We will show that we can replace those elements with index smaller than that of the current piece with elements with larger index than the current piece to form a set with equal or less value according to v_* and equal or more value according to v. Let T be a set containing the current element and all elements with larger index than the current element. Subtract $S \cup T$ from S, T to form S' and T' respectively. Since we are claiming S is optimal, then $v(S') \leq v(T')$.

Consider, only a single element inside S' which has index smaller than the current piece... □

2.2 Nearsighted Defenders

Lemma 2.6. *Let v be the value function of a nearsighted defender. Let A, B be two sets of pieces. If every piece in B is further from tenure than any piece in A, and B has greater value according to v, then B has greater value according to v_* as well.*

Theorem 2.7. *The following algorithm maximizes the real value of the surviving pieces at the end of each turn against any nearsighted defender.*

Algorithm 2: `Minimizing` v_* `Value of Destroyed Set Against Nearsighted Defenders`

```
Input: a board position $S$ with $k$ levels, and the value function of a nearsighted defender $v$
Result: a partition of the board $(S_1, S_2)$ which guarantees the subset that the defender will
        destroy has the lowest possible value according to $v_*$
$\gamma \leftarrow \frac{v(S)}{2}$;
$best\_set \leftarrow S$;
$best\_value \leftarrow v_*(S)$;
$current\_set \leftarrow$ an array of zeroes of length $k$;
$current\_value \leftarrow 0$;
for $i \leftarrow 0$ to $k$ do
    for $piece \leftarrow 0$ to $S[i]$ do
        if $v(i) < \gamma$ then
            $current\_set[i]{+}{+}$;
            $\gamma {-}{=} v(i)$;
            $current\_value {+}{=} v_*(i)$;
        end
        else
            if $current\_value + v_*(i) < best\_value$ then
                $best\_value \leftarrow current\_value + v_*(i)$;
                $best\_set \leftarrow current\_set$;
                $current\_set[i]{+}{+}$;
            end
            $break$;
        end
    end
end
return $(best\_set, (S - best\_set))$;
```

Before proving Theorem 4.5, we establish:

Lemma 2.8. *Let there be no pieces on the board which have greater value than γ according to some nearsighted v. Let i be the smallest i such that the i-th level is not empty. To form a subset of M with value of at least γ according to v, while minimizing the value of the subset according to v_*, we are guaranteed to be able to include a piece from level i.*

Proof. Let M not contain one of the pieces on level i. Since $v(S) \geq \gamma \geq v(i)$. Then we can apply lemma 4.3 and conclude $v_*(S) > v_*(i)$. Then, by lemma 4.1, we know there is a subset M' of M with a value of exactly $v_*(i)$. If we were to replace M' in M with a piece from level i, we do not change the value according to v_* but we increase or maintain the value according to v (Modus Tollens of lemma 4.3). □

We now prove Theorem 4.5:

Proof. The idea behind the algorithm is that as we iterate through the board, starting with the more valuable pieces (according to v_*), which we are certain we can add to current_set due to lemma 4.6. Every time we come across a piece whose value exceeds the remaining γ, we consider the value according to v of the set formed by adding that piece to the current_set. If that value is lesser than that of the best_set we have encountered before, that set becomes our new best_set.

"Notice, that lemma 4.6 guarantees one of the optimal sets can be formed with what has been added. Since, we have recursively considered every feasible set that comes after the decisions, one is bound to be optimal" <- needs rewording! □

3 Machine Learning

3.1 Implementation

- Ben should investigate and do a write-up on the microaction approach (by going through the code and understanding the implementation details and the “tricks” that allow the technique to work) and discuss how it addresses the challenges in the previous subsection.

- Ben should tune the trained model, and use 2D convolution as Prof. Mao mentioned.

3.2 Analysis

- Ben should benchmark the model as Prof. Moura mentioned.

- Wei should look into proving convergence as Prof. Mao mentioned.

4 Conclusion

5 Bibliography

Raghu, Maithra, et al. "Can deep reinforcement learning solve erdos-selfridge-spencer games?." *International Conference on Machine Learning*. 2018.