Analysis of Symmetry and Conventions in Off-Belief Learning (OBL) in Hanabi

William Zhang, Fan Pu Zeng
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, Pennsylvania, 15213

Abstract

We investigate if policies learnt by agents using the Off-Belief Learning (OBL) algorithm in the multi-player cooperative game Hanabi in the zero-shot coordination (ZSC) context are invariant across symmetries of the game, and if any conventions formed during training are arbitrary or natural. We do this by a convention analysis on the action matrix of what the agent does, introduce a novel technique called the Intervention Analysis to estimate if the actions taken by the policies learnt are equivalent between isomorphisms of the same game state, and finally evaluate if our observed results also hold in a simplified version of Hanabi which we call Mini-Hanabi .

1 Introduction

AI systems are becoming increasingly sophisticated, thanks to improvements in technology, advances in AI and ML research, and increasing availability of data. As more AI systems are deployed for real-world usage, it becomes necessary for AI agents to be able to cooperate with one another and avoid pathological interactions (Conitzer and Oesterheld). This is the chief motivation for designing cooperative AI agents.

To improve our understanding in this area, Hanabi has been proposed as the new frontier for developing strategies in cooperative AI (Bard et al. 2020). Hanabi is a cooperative game with imperfect information that is played between two to five players. Successful gameplay relies heavily on making use of the theory of mind, which refers to understanding why other players made the actions that they did in order to infer information that you do not have. As such, it provides a suitable testbed to evaluate the performance of algorithms in this space. To this end, the Hanabi Learning Environment has been developed by DeepMind to provide a uniform framework for benchmarking (Bard et al. 2020).

In this paper, we present three results related to a recent multi-agent reinforcement learning algorithm called Off-Belief Learning (OBL) introduced by (Hu et al. 2021).

Our first result is an analysis into what kinds of conventions OBL learns. This is determined by the degree of correlations between subsequent actions. For conventions that are found, we investigate whether they are natural, or represent a problem for the zero-shot coordination setting. We demonstrate that the policies learnt do exhibit correlation between certain actions, but show that these are naturally-occurring conventions due to the nature of the game which are not arbitrary, and that no other arbitrary conventions arises.

Our second result is whether OBL learns policies that are equivalent among isomorphisms of the game. In the case of Hanabi, these are permutations in the colors of the cards that are dealt, which results in the same game but with different labels. This is desirable as we would naturally expect agents to behave the same under isomorphic instances of the same game, but we will show that this empirically does not happen for higher levels of OBL agents. We investigate this by recoloring a chosen base game, and performing what we call the Intervention Analysis at various points in the game to see if it diverges from the base game as a result of the recoloring. We also perform this for several levels of OBL agents to see how it changes across higher-level agents.

Our final result is to apply OBL on a smaller version of Hanabi that we can feasibly train and fine-tune without industrial-scale of compute ability, which we call Mini-Hanabi . This is to investigate if ensuring convergence of policies would ensure that the policies found are invariant to coloring. We received mixed results, as it appears that the setting of Mini-Hanabi caused the agents to learn a self-preserving policy that was unexpected and had very low variance in the outcomes of the games. This translated to very little observed variance when we performed Intervention Analysis on it, which we do not believe to be representative of the actual symmetries of the policies learnt when we contrasted the results from Intervention Analysis to the correlations found between the actual actions that it took, which exhibited signs of arbitrary color-aware conventions.

2 Background

2.1 Hanabi

This section will explain the rules of Hanabi. It can be safely skipped if the reader is already familiar with Hanabi.

Hanabi is a game played between 2 to 5 players. The deck comprises of 50 cards. There are 10 cards of 5 different colors: red, yellow, green, blue, white. There are also 5 ranks, numbering from 1 to 5. For each color, there are 3 cards of rank 1, 2 cards of rank 2 through 4, and only 1 card of rank 5. When there are 2 or 3 players, each player starts with 5 random cards, which will always be the case for the gameplay context considered in this paper. Otherwise, when there are 4 or 5 players, everyone starts with 4 cards.

On top of cards, all players also share the same pool of 8 information tokens and 3 lives at the start of the game.

Players are not allowed to see their own cards at any point in time until they are played; however, they can see the card of other players.

Play proceeds clockwise sequentially in a turn-based fashion among the players. There are three actions that a player can perform during their turn:

1. Hint a color or rank of another player
2. Discard a card from their hand
3. Play a card from their hand.

Players must perform exactly one of the actions during their turn, and cannot skip their turn.

We explain each of these actions in detail.

Action: Hinting A player can choose to perform the hinting action as long as the team has at least one information token. Hinting will result in the usage of one information token, which is removed from the team's count of information tokens.

When giving a hint, the player will choose another player that they would like to provide the hint to. There are two types of hints that can be given: color hints, and rank hints.

To give a color hint, the hinting player will tell the target player all the positions of the cards that they have of one particular color. An example would be telling her teammate that she has two yellow cards at positions 1 and 4. The positions of all cards of the specified color must be given.

To give a rank hint, similarly, the hinting player will tell the target player all the positions of the cards that they have of a particular rank, with the same constraint that all the positions must be given. For instance, a player could hint to her teammate that she has a rank 2 card in positions 2 and 3.

Action: Discarding a Card A player can choose to discard a card from their hand. This will remove the card from play permanently. The current player will also be allowed to see the face of the card after it has been discarded. Discarding a card will earn the team an additional information token, up to a maximum possible of 8 information tokens.

If there are still cards in the deck, a new card will be drawn after a discard action.

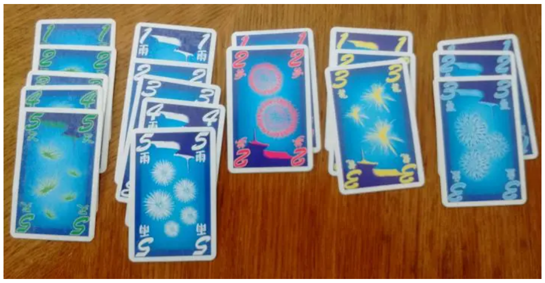


Figure 1: A possible game state in Hanabi, showing all played cards. The hands of the players are not shown.

Action: Playing a Card Finally, a player can choose to play a card from their hand. A chosen card is either playable, or unplayable.

In order to explain whether a card is playable, we need to first introduce what a firework is. A firework is a series of cards in sequentially increasing rank of a specific color that has been played. For instance, Figure 1 shows 5 different fireworks of all possible colors. The green and white fireworks on the first two columns are completed, since rank ranges from 1 to 5. The other fireworks of other colors are not completed.

A playable card is then one that can extend an uncompleted firework. For the game state shown in Figure 1, this would be either a red card of rank 3, or a yellow or blue card of rank 4.

If a playable card was chosen to be played, that card will be added to the firework. Otherwise, playing an unplayable card results in the loss of a life, and the card is discarded from play. In either case, the player will then draw a new card, such that they always have the same number of cards. The only exception is if the deck runs out of cards, in which case play proceeds for one more round before ending.

Note that players are not allowed to have any other form of communication other than performing hint actions.

Scoring The game ends when either all fireworks are completed, the players run out of cards, or the players lose all 3 lives. In the last case, players get a score of 0. Otherwise, the score is equal to the number of cards that have been played successfully. This means that a perfect score in a standard Hanabi game is 25, with a game state given by Figure 2.

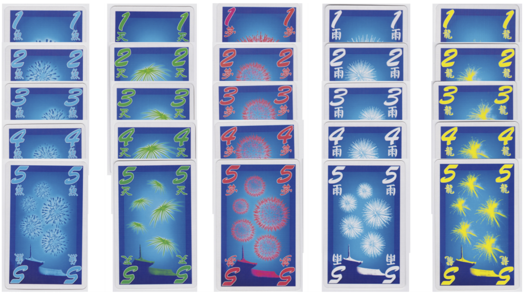


Figure 2: A completed game of Hanabi with a perfect score of 25. Only the played cards are shown.

2.2 Human Conventions in Hanabi

Hanabi is very much a game of conventions. This makes sense from an information-theoretic standpoint, as hints are expensive and we would like to convey the maximum amount of information for each provided hint. Having conventions therefore gives players reliable and strong priors when inferring why other players decided to perform a particular action. In fact, there are entire online forums[1] dedicated to curating and developing conventions in Hanabi. In a similar vein, it is very interesting to study how AI agents can find and develop such conventions.

We describe one of the most commonly used and uncontroversial conventions called "The Chop" so that readers new to Hanabi have a better understanding of its importance in gameplay, since we will frequently allude to conventions in the rest of the paper. In "The Chop" convention, when a player has nothing else to do, they should discard their oldest (i.e rightmost) unclued card (called the "chop" card) in their hand. If players all follow "The Chop", then everyone will discard in a predictable manner, and players do not have to spend hints to let others know that certain cards are unplayable. For instance, a red rank-1 card can be discarded when the red firework has already been extended to rank 3, and if no other players provide a hint about the card, it will eventually become the oldest card in hand after others have been played, and will be discarded without requiring the use of any hints.

2.3 Settings for Cooperative AI

We introduce and motivate common settings for evaluating cooperative AI agents.

Self-Play In the self-play context, agents are evaluated by playing against themselves. This is the simplest setting, but is unrealistic since agents can form arbitrary and unnatural conventions that they can exploit among themselves.

Many reinforcement learning techniques train agents via self-play, and therefore perform very well in this evaluation metric.

Cross-Play In the cross-play context, agents are put together in ad-hoc teams, and work with other agents that they have never seen before. As such, they must be able to coordinate and adapt together (Hu et al. 2020). Being able to design agents that can do so is one the central problems in cooperative AI.

For instance, consider the Lever Coordination Game shown in Figure 3. This is a multi-player cooperative game where on each round, each player can choose one of 10 levers to pull. On a given turn, all players receive the same reward of 1.0 if they all pull the same lever, and otherwise receive a reward of 0. An agent that relies on arbitrary conventions, such as always pulling lever 7, can succeed every turn in the self-play context, but would fail once it is paired with other unseen agents with different conventions, such as always pulling lever 3.

There is a lot of recent work done in investigating how an agent that was added to a team of other agents can effectively learn the social conventions of the existing team in order to play together successfully (Tucker, Zhou, and Shah 2020).

[1] H-Group Conventions : Strategies for Hanabi, a cooperative card game of logic and reasoning. `https://hanabi.github.io/`

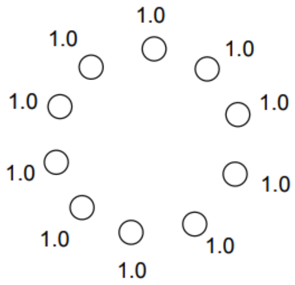


Figure 3: The Lever Coordination Game

2.4 Zero-Shot Coordination

A refinement of the cross-play setting is the Zero-Shot Coordination (ZSC) setting. In the ZSC setting, agents are not allowed to have any form of communication, including warmup-rounds, before test time. To mitigate this highly restrictive setting, the designers of the AI agents are allowed to agree on a high-level idea of the learning algorithm used to train their agents beforehand. This corresponds to agreeing on a high-level implementational plan in real-world decision making. However, they are not allowed to share labels for observations, actions, and states in the environment, which corresponds to low-level decision making that is performed independently (Hu et al. 2020).

2.5 Dec-POMDP

Hanabi can be formalized as a Decentralized Partially Observable Markov decision process (Dec-POMDP). A Decentralized Partially Observable Markov decision process (Dec-POMDP) is a framework used to model decision making in situations where multiple agents must make decisions while interacting with each other and operating in stochastic environments. Dec-POMDP extends the standard Partially Observable Markov decision process (POMDP), which is used to model decision making in single-agent settings.

In a Dec-POMDP, each agent must make decisions based on its own local observations, without having access to the full state of the environment. In the case of Hanabi, this means not knowing your own cards and other agent's private states.

It has been proven that solving Dec-POMDP problems is NEXP-hard in general.

2.6 Symmetry

A symmetry, intuitively defined, is a relabeling of the state and action space without affecting the trajectories of the game. Symmetry can be viewed as a form of equivalent equilibrium selection. Re-discussing the one-shot lever coordination game, all levers are symmetrical from the view of any given agent. Training an agent in self-play typically trains an agent to break the symmetry in an arbitrary fashion. Under an environment where the other player submits that particular agent, it is a best response to submit the identical copy to coordinate symmetry breaking – submitting any other learned agent will yield worse or equal payoffs.

Zero-shot coordination, whereby an agent is forced to cooperate with other agents without prior communication, poses an unique challenge for learned strategies where symmetry is involved. An agent will achieve favorable payoffs when paired with another agent that breaks symmetry in a compatible manner, and in other cases, unfavorable outcomes.

We note that formally, a game's symmetries are the set of bijections from the game state, observation, and reward space onto itself, while leaving the Dec-POMDP unchanged. Note that transforming the policy consistently with respect to the symmetry produces a symmetrical policy which does not alter the game trajectory or the reward. In other words, applying a consistent relabeling to a given trajectory would yield the same reward, and conditional on the learning algorithm, treated as identical.

2.7 Off-Belief Learning

We now describe Off-Belief Learning (OBL) introduced by (Hu et al. 2021), which is the ZSC algorithm that we are analyzing in this paper.

From a high level, OBL works by starting from a base policy π_0, and then iteratively training successive policies π_1 from π_0, π_2 from π_1, and so on. The idea behind how each policy trained from the previous one is very similar to the established Cognitive Hierarchies (CH) reasoning model (Chong, Ho, and Camerer 2016). In CH, the hierarchy starts with level 0, where players at level 0 play without any beliefs and regard to what other people do (i.e choosing actions uniformly randomly). Players at level 1 play as if all other players are at level 0, and in general, players at level i will play their best response assuming all other players will perform reasoning at level $i - 1$.

Similarly, OBL works by training each successive policy π_i via self-play, and minimizing its regret by assuming that the action distributions of other players come from them playing at level π_{i-1}. This update is summarized by the OBL operator.

The OBL Operator For simplicity, we discuss the case of how to obtain π_1 from π_0. Doing so for higher levels is analogous.

Trajectories are collected via self-play. Using the same notation as in (Hu et al. 2021), let the ground-truth trajectory where all everything is fully observable to be defined $\tau = (s_1, a_1, \ldots, a_{t-1}, s_t)$, and let the action-observation history that agent i actually observes to be defined as $\tau^i = \left(\Omega^i\left(s_1\right), a_1, \ldots, a_{t-1}, \Omega^i\left(s_t\right)\right)$. Then given some partially observed trajectory τ^i, for any possible fully observed trajectory τ, the belief of agent i that τ is the true trajectory under the assumption that all other players are playing policy π_0 can be given by the belief distribution

$$\mathcal{B}_{\pi_0}\left(\tau \mid \tau^i\right) = P\left(\tau \mid \tau^i, \pi_0\right).$$

Instead of learning the standard value and action-value function V^{π_1} and Q^{π_1}, OBL learns their counterfactual variants $V^{\pi_0 \rightarrow \pi_1}\left(\tau^i\right)$ and $Q^{\pi_0 \rightarrow \pi_1}\left(a \mid \tau_t^i\right)$. This is defined in terms of the policy playing against itself, but pretending that actions taken by other players (i.e the other agent in self-play) is following policy π_0.

We can then formalize the counterfactual value and action-value functions as

$$V^{\pi_0 \rightarrow \pi_1}\left(\tau^i\right) = E_{\tau \sim \mathcal{B}_{\pi_0}(\tau^i)}\left[V^{\pi_1}(\tau)\right], \tag{1}$$

$$Q^{\pi_0 \rightarrow \pi_1}\left(a \mid \tau_t^i\right) = \sum_{\tau_t, \tau_{t+1}} \mathcal{B}_{\pi_0}\left(\tau_t \mid \tau_t^i\right)\left[R\left(s_t, a\right) + \mathcal{T}\left(\tau_{t+1} \mid \tau_t\right) V^{\pi_1}\left(\tau_{t+1}\right)\right]. \tag{2}$$

Then finally, the OBL operator to compute policy π_1 from π_0 is defined as

$$\pi_1\left(a \mid \tau^i\right) = \frac{\exp\left(Q^{\pi_0 \rightarrow \pi_1}\left(a \mid \tau^i\right)/T\right)}{\sum_{a'} \exp\left(Q^{\pi_0 \rightarrow \pi_1}\left(a' \mid \tau^i\right)/T\right)}, \tag{3}$$

where T is a temperature hyperparameter that is decreased over time.

We call agents trained with policy π_i OBLi, i.e OBL1 for π_1, and OBL5 for π_5.

3 Related Work

3.1 Other-Play

Other-Play (OP) was introduced in (Hu et al. 2020) to solve the problem of ensuring that policies learnt by agents in a ZSC context is equivariant with regards to isomorphisms of game states. OP can be applied to any game where there are symmetries in its DEC-POMDP formalization. OP achieves this as follows. To train some Agent 1 while playing with its counterpart Agent 2, the optimization problem posed for agent 1 during training will relabel the policies of Agent 2 in a symmetry-preserving way, therefore allowing Agent 1 to learn a policy that is symmetry-invariant. The downside to this approach is that it requires knowledge of all the symmetries of the game beforehand.

4 Methods

We describe our approach to convention analysis, introduce our novel technique of deciding whether a policy learnt is invariant to symmetries called Intervention Analysis, and describe the simplified Hanabi game Mini-Hanabi which we use to investigate whether the shortcomings of OBL found from our convention and Intervention Analysis could be addressed by ensuring that the policies have converged to a high degree. We used the pre-trained models of OBL agents provided by (Hu et al. 2021).

4.1 Infrastructure

OBL (Hu et al. 2021) released their source code, which built upon DeepMind's Hanabi Learning Environment. They also released their trained OBL1 through OBL5 models that they used for their submission. We leveraged this infrastructure to perform our convention analysis and Intervention Analysis.

We also made further augmentations to allow the OBL and, by extension, Hanabi infrastructure to run and learn from games parameterized by the number of colors, number of ranks, and the number of cards in each player's hand. The diff-log for reference is at `https://github.com/facebookresearch/off-belief-learning/compare/main...17zhangw:off-belief-learning:main`.

4.2 Convention Analysis

To perform convention analysis, we made our agents play 10,000 random games in a self-play setup. We then analyzed the played games by computing the number of times that each player performed some action at time t for each action that the other player performed at time $t-1$. For graphical representation, we normalize by each action taken at $t-1$.

4.3 Intervention Analysis

We consider a generalized version of the Hanabi game, so the algorithm works for both Hanabi and Mini-Hanabi . Let R be the number of ranks, and C be the number of colors (in a normal game $R = C = 5$). Then we can represent any card s as $s \in [R] \times [C]$. Denote the total number of the cards in the deck as N (in a normal game $N = 50$).

In Intervention Analysis, we generate $m = 100$ base games for a given OBL agent, with the eventual goal of choosing one that is representative of a game played by the agent. This is decided in terms of the score, number of lives remaining, and information tokens remaining at the end of the game. For each base game which is associated with a deck that is dealt, we generate $K = 1,000$ different permutations of the available colors, and apply this permutation to base deck to obtain a new recolored deck. Note that the new deck is semantically the same as the old deck, as recolorings of a Hanabi game forms a symmetry. For intervention fractions of $h \in [0.25, 0.5, 0.75]$, we force agents to take the same actions as they did in each of the recolored games up until h fraction of the game length, and then allow them to play as per normal. We then record the outcome of the game.

If the policies learnt are truly invariant to recolorings, then we should expect all the recolored games to perform just as well as the base game. However, if it is not equivalent across recolorings, then the players' actions in the recolored game will begin to diverge, which could result in a different final score, number of lives remaining, and information tokens remaining.

In practice, the base game that is chosen for our analysis is one that does relatively well but does not receive full score, which allows for both upsides and downsides in the score of the recolored games.

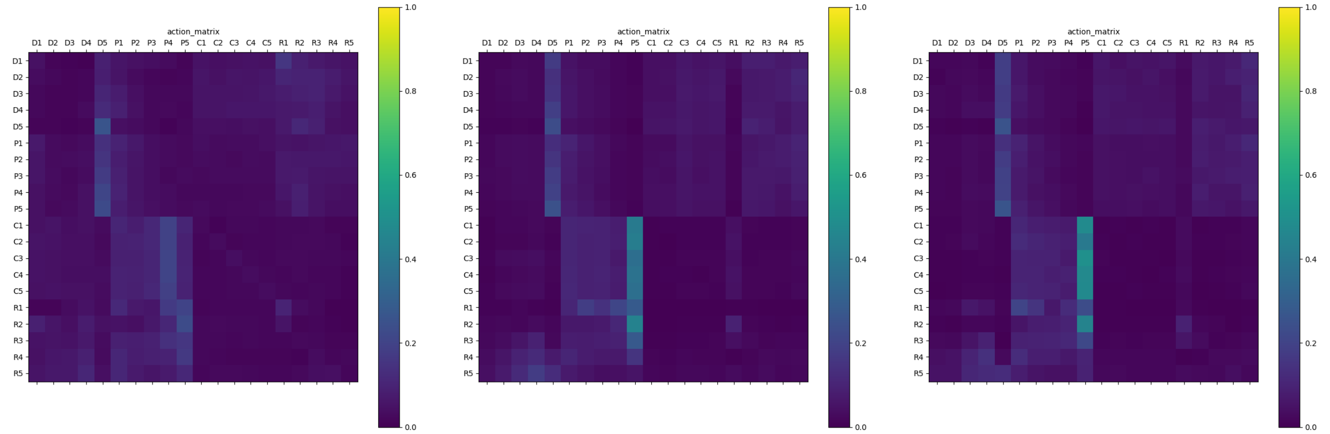
We also noticed when running Intervention Analysis that the agents already wanted to diverge from the forced policies in the recolored game even before the intervention point is over, which already provides evidence that the policies are not equivalent across symmetries.

As a sanity check, we also verified that Intervention Analysis always gives that the agent plays the exact same game when the recoloring is the identity coloring, i.e it replays the same game. This confirms that there is no inherent stochasticity that is not replicable across different runs in the neural networks used.

The formal specification of our Intervention Analysis technique is given in Algorithm 1.

Algorithm 1: Intervention Analysis

```
Input:
    $\mathcal{A}_1, \mathcal{A}_2$, two independently trained OBL agents of the same level;
    $h$, the intervention fraction;
    $M$, the number of trials to perform;
    $K$, the number of recolorings per trial
Result: score_delta[], lives_remaining_delta[], tokens_remaining_delta[]
Zero-initialize three $M \times 1$ arrays: score_delta[], lives_remaining_delta[], tokens_remaining_delta[]
for $m \leftarrow 0$ to $M - 1$ do
    Generate a uniformly randomly shuffled deck of cards $S = (s_1, \ldots, s_N), s_i \in [R] \times [C]$;
    $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S} \leftarrow$ simulate a complete game between $\mathcal{A}_1, \mathcal{A}_2$ where cards are dealt following $S$;
    $(a_1, \ldots, a_t) \leftarrow$ actions taken in $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}$;
    score $\leftarrow$ final score in $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}$;
    lives_remaining $\leftarrow$ number of lives remaining in $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}$;
    tokens_remaining $\leftarrow$ number of information tokens unused in $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}$;
    for $k \leftarrow 0$ to $K - 1$ do
        Sample a uniformly random permutation of colors $\sigma : [C] \to [C]$;
        Define $\rho_\sigma : [R] \times [C] \to [R] \times [C]$ via $\rho_\sigma(r, c) = (r, \sigma(c))$;
        $S_\sigma \leftarrow (\rho_\sigma(s_1), \ldots, \rho_\sigma(s_N))$;
        $f \leftarrow \lfloor h \cdot t \rfloor$;
        Simulate a game until time $f$ where cards are dealt according to $S_\sigma$, and $\mathcal{A}_1, \mathcal{A}_2$ forced to take actions
         $a_1, \ldots, a_f$;
        $\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S_\sigma,(a_1,\ldots,a_f)} \leftarrow$ Continue from the current state of the game but allow $\mathcal{A}_1, \mathcal{A}_2$ to choose actions based on
         their own learnt policies and play until the end;
        score_delta[$m$] $\mathrel{+}= \frac{1}{K}\left(\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}[\text{score}] - \mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S_\sigma,(a_1,\ldots,a_f)}[\text{score}]\right)$;
        lives_remaining_delta[$m$] $\mathrel{+}= \frac{1}{K}\left(\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}[\text{lives\_remaining}] - \mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S_\sigma,(a_1,\ldots,a_f)}[\text{lives\_remaining}]\right)$;
        tokens_remaining_delta[$m$] $\mathrel{+}= \frac{1}{K}\left(\mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S}[\text{tokens\_remaining}] - \mathcal{G}_{\mathcal{A}_1,\mathcal{A}_2,S_\sigma,(a_1,\ldots,a_f)}[\text{tokens\_remaining}]\right)$;
    end
end
return score_delta[], lives_remaining_delta[], tokens_remaining_delta[]
```



(a) OBL1

(b) OBL3

(c) OBL5

Figure 4: OBL1, OBL3, and OBL5 action matrices. There are conventions learnt, but they are natural.

4.4 Mini-Hanabi

In addition to the standard Hanabi game with 5 ranks and 5 colors, we also trained OBL agents for an alternative Hanabi game called Mini-Hanabi with 3 ranks, 3 colors, and all other game parameters held the same.

The motivation for training agents for Mini-Hanabi is because we observed non-trivial deviations in the score when performing Intervention Analysis on the recolored games, and suspected that this could be due to the number of epochs used for training the agents not being large enough such that the V and Q estimates did not fully converge yet, resulting in a policy that was not invariant across policies. We therefore wanted to experiment on a model trained with more epochs to see if it closes the gap.

(Hu et al. 2021) mentions that they trained OBL on the original Hanabi game using 7 GPUs (5 for training 5 independent agents in parallel, 2 for belief model inference) on 40 CPU cores over 40 hours. Due to our compute limitations, we had to consider a smaller game, which is the motivation for Mini-Hanabi . We did not decrease the number of lives or information tokens from the original Hanabi game even though the state space is much smaller as we wanted the game to be easier, so that rewards are more plentiful, which makes it faster for the learning algorithm to converge.

We trained our agents with 12-core Intel(R) Xeon(R) W-1350, containing 32 GB of RAM with a single RTX 3080 GPU. For the most part, we utilize the same parameters used by (Hu et al. 2021) in their training process. Substantive differences are primarily motivated by resource constraints. In short, we used a single for both training and inference, reduced the overall epochs by a factor of 4 for each OBL model instance, and adjusted the rate at which game trajectories were generated and kept in the replay buffer to remain in memory. Mini-Hanabi OBL1 took 71 hours to train, with OBL2 and OBL3 both taking approximately a day each.

5 Results and Analysis

5.1 Convention Analysis

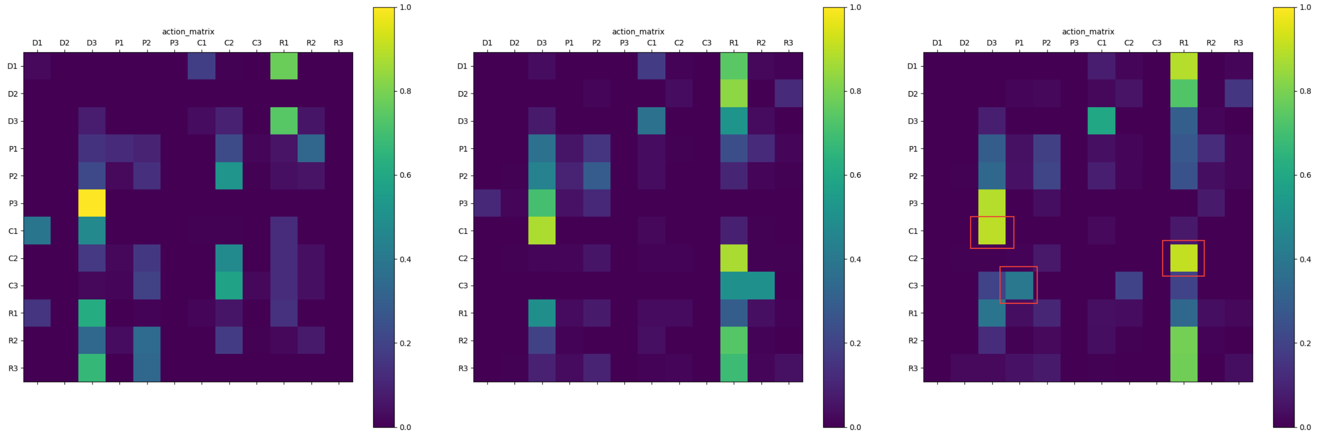
The action matrices of OBL1 and OBL5 is given in Figure 4. The action matrix shows the correlation between the actions taken by player i at time $t-1$ (vertical axis) against the actions taken by player $-i$ at time t (horizontal axis). We only consider 2-player Hanabi and therefore this is complete. The meaning of the labels are given below:

- DX: Discard the Xth card in your hand
- PX: Play the Xth card in your hand
- CX: Hint all cards with color X to the other player
- CX: Hint all cards with rank X to the other player

Note that X is positional for DX and PX, i.e the higher X is, the longer the card at position X has been in the hand.

We noticed that in OBL5, there is some correlation between hinting colors and rank of the previous player to the playing the oldest card of the next player. This can be seen from the bright vertical line in column P5 in Figure 4(b). Similarly, there is also correlation between the previous player discarding or playing any card, and the next player discarding their oldest card.

We deduced that this actually corresponds to OBL5 learning “The Chop” convention that was introduced in Section 2.2. This is because since the previous player did not hint anything about the next player’s oldest card, the next player could infer that



(a) OBL1

(b) OBL2

(c) OBL3

Figure 5: OBL1, OBL2, and OBL5 action matrices on Mini-Hanabi . The undesirable coloring-aware conventions learnt by OBL3 are marked in red boxes.

it is not playable and therefore discard it. Simiarly, should the chop card of the next player actually be playable, the previous player would have realized that it must provide a hint to let the next player infer it is playable.

This is both surprising and elegant, and it might show how "The Chop" is actually a natural convention, which explains its widespread acceptance and popularity among the Hanabi community.

On the other hand, OBL1 is mostly convention-free, and only exhibits weak correlations between the previous player discarding or playing any card, and the next player discarding its oldest card. There is also some very weak correlation between hinting a color, and the next player playing the fourth oldest card. It appears to be in the early stages of learning "The Chop", as the patterns are similar but very weak. The results also matches our expectations that OBL1 should exhibit weaker correlations than OBL5, since it is trained on a lower level of iterated reasoning.

5.2 Convention Analysis on Mini-Hanabi

The action matrices for OBL1, OBL2, and OBL5 on Mini-Hanabi are given in Figure 5. The labels follows the same conventions as for the plots in Section 5.1.

We noticed that a self-preservation convention emerged as the model was progressively trained from OBL1 to OBL3. In OBL2 and OBL3, we can see a very strong relationship between any action being played by the previous player, and the playing of $R1$ by the next player, as seen from the bright vertical bar in the $R3$ column. Then the next action that will be subsequently played (by reading from the $R1$ row) would be to either play $D3$ or $R1$. If $D3$ is played, there is a large probability of playing $R1$ yet again. We therefore see that this leads to a cycle of hinting a rank, and either discarding the oldest card or hinting a rank again.

Why does the agent do this? We believe this is because we made talk too cheap in Mini-Hanabi since it has 8 information tokens relative to having only 18 cards in total in the deck, contrasted to 50 cards in total in the standard Hanabi game. This allows it to learn to perform self-preservation by refusing to play any action that could lead to a loss of life, and instead cycles between giving hints and discarding cards (which replenishes hints). A further analysis of a Mini-Hanabi game played by the OBL1 agent in Section 5.4 confirms this hypothesis.

In addition, there are also some asymmetrical color conventions that were learned which are undesirable. For instance, as seen in Figure 5(c), the OBL3 agent has a very strong action correlation between $C1$ and $D3$, $C2$ and $R1$, and $C3$ and $P1$, which are all bounded in red boxes in the diagram. These correlations are undesirable as all other colors very rarely or do not play the corresponding actions at all, indicating that it has learned to play color-aware (and possibly secretive) conventions.

5.3 Intervention Analysis

Table 1 provides our results from Intervention Analysis on Hanabi using agents OBL1, OBL2, and OBL5. The main entry to focus on are the Score columns, since this is how the rewards of the game is calculated. Table 1 shows that the change in score decreases across all intervention points as the level of the agent increases. This suggests that beyond policies of higher level agents being better, the higher level agents are also more robust to recolorings.

As the intervention point decreases from 75% to 25%, the difference in score also increases for all levels of agents. This indicates that there is divergence at all levels of agents, since it performed differently when it had to play fewer forced moves

Intervention Point	Score (Delta)			Lives Remaining (Delta)			Info Tokens Remaining (Delta)		
	OBL1	OBL2	OBL5	OBL1	OBL2	OBL5	OBL1	OBL2	OBL5
25%	-1.4040	-0.0370	0.0110	0.3750	-0.5960	-0.3560	-1.1010	-0.4600	0.2280
50%	-1.6560	0.0000	-0.0080	-0.0540	-0.6210	-0.1560	-0.6970	-0.4190	0.4150
75%	0.2650	0.0000	0.0000	-0.0080	-0.4180	-0.0190	-1.0870	0.2770	0.2150

Table 1: Results of Intervention Analysis of playing 1,000 recolored Hanabi games for various intervention points, given as the average of the score, lives remaining, and info tokens remaining subtracted from the corresponding values achieved in the base game. Higher positive values are better.

and could therefore play its own chosen policy for most of the time. While the difference in score appears minute for OBL5, we can see from the relatively large difference in lives remaining and information tokens remaining that the actions being performed during the recolored game has definitely diverged from the base game.

Table 2 summarizes the results of playing 1,000 Mini-Hanabi games for a representative base game. OBL1 has a 0 in all entries since it learns to perform self-preservation but does not exploit rewarding actions, such as playing playable cards. This behavior is further elaborated in Section 5.4.

Mini-Hanabi Agent	Score (Delta)	Lives Remaining (Delta)	Info Tokens Remaining (Delta)
OBL1	0.0000	0.0000	0.0000
OBL2	0.3940	0.0000	-3.0000
OBL3	0.0000	0.0000	1.6950

Table 2: Results of Intervention Analysis at the 50% point of playing 1,000 recolored Mini-Hanabi games, given as the average score, lives remaining, and info tokens remaining subtracted from the corresponding values achieved in the base game.

5.4 Mini-Hanabi Agent's Learned Behavior

```
P1: (Deal Y2)
P2: (Deal G1)
P1: (Deal R1)
P2: (Deal G2)
P1: (Deal R2)
P2: (Deal R1)
P1: (Reveal player +1 rank 2)
P2: (Play 1)
P2: (Deal Y1)
P1: (Reveal player +1 color Y)
P2: (Reveal player +1 color Y)
P1: (Reveal player +1 color Y)
P2: (Reveal player +1 color Y)
P1: (Reveal player +1 color Y)
P2: (Reveal player +1 color Y)
P1: (Play 1)
P1: (Deal G1)
P2: (Play 1)
P2: (Deal G3)
P1: (Reveal player +1 color Y)
P2: (Play 1)
P2: (Deal Y1)
P1: (Discard 2)
P1: (Deal R3)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal G2)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal R2)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal Y3)
```

```
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal G1)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal Y2)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal R1)
P2: (Reveal player +1 rank 1)
P1: (Discard 2)
P1: (Deal Y1)
P2: (Reveal player +1 rank 1)
P1: (Discard 0)
```

Listing 1: Game played by OBL1 on Mini-Hanabi

A representative game played by the OBL1 agent trained on Mini-Hanabi is given in Listing 1. It learns self-preservation by playing a strategy of cycling through actions of giving hints and discarding, which restores a hint. This allows it to avoid playing cards, which could lead to a loss of a life if the card happens to be unplayable. This game received a score of 3, and one of the moves played was illegal and the agent lost a life.

The fact that the OBL1 agent (as well as higher-level agents) tends to employ this strategy in Mini-Hanabi diminishes the value of the results of Intervention Analysis of Mini-Hanabi as given in Table 2. It explains why there is so little variance in the score and lives remaining, since a self-preserving strategy would never lose a life or die, which therefore makes it hard to establish whether the actions that were taken in the recolored games are equivalent policies up to recoloring from just the Intervention Analysis alone. Our convention analysis of Mini-Hanabi as given in Figure 5 provides evidence for this to be not the case.

6 Conclusion and Future Work

6.1 Symmetry Variance

We empirically demonstrate that OBL does not learn symmetry-invariant policies in practice. This is because the learnt policy diverges from the base game when playing in a recolored game, even though it is being evaluated in a self-play setting. We hypothesize that this could be due to a variety of factors.

The first reason is the inherent impossibility of performing the counterfactual Q and V estimates exactly. This could result in tiny divergences in the inherited policy that manifest visibly over long episode trajectories. Furthermore, since episode trajectories are generated from real games, there are cases where certain trajectories are not sampled or explored well. This is already commented upon by (Hu et al. 2021).

The second reason could be due to the use of the replay buffer during training. While it is a commonly used technique in reinforcement learning to ensure that the agent revisits promising trajectories instead of forgetting them. However, this could introduce bias in the distribution of states that are explored particularly when the rate at which promising trajectories are sampled and the rate at which the replay buffer is replenished with new trajectories are misaligned. This can be difficult to overcome, which can drastically affect the quality of the policy learned. Furthermore, since OBL (Hu et al. 2021) does not directly account for symmetries which is further exacerbated by the distribution of trajectories in the replay buffer.

Finally, a third reason could be the existence of multi-modal equilibriums, where it has to choose the same equilibrium across all symmetries for it to be robust to it. OP (Hu et al. 2020) randomly recolors trajectories during game generation to force the training process to be robust to the color symmetries.

6.2 Conventions

We also demonstrate that OBL does learn conventions, but they are not arbitrary. In particular, in the standard Hanabi game it learns to play “The Chop”, which is also an established and widely-accepted convention played by humans, and in Mini-Hanabi where talk is cheaper, it learns to perform self-preservation. The former point is exciting as it shows that it is possible for AI agents to learn policies which are natural and compatible with human intuition, which is desirable when humans and AI need to cooperate together.

A philosophical question surrounding agents and conventions is when is a secret convention necessary and when is not? A secret convention is any action taken by a player that conveys an additional piece of information that cannot be derived otherwise (for instance, hint rank 1 means discard the 3rd card). We observe that in situations where the submitted agents are known ahead of time, it may often be dominant to submit an agent, identically trained or identical agent, that exploits any hidden convention to the fullest. However, in cases whereby the team of agents is not known in advance or we are required to submit independently trained agents, OBL or OBL-like algorithms are of interest in that they provide theoretical guarantees for policy unique-ness and robustness.

6.3 Future Work

An interesting direction for future work is to solve the problem of symmetry invariance, while still maintaining the proven training architecture that OBL uses. We suggest a model-free approach to this, as opposed to OP's model-based approach of requiring prior knowledge of the symmetries in the Dec-POMDP. This could be achieved by training an additional model that is able to learn symmetries of the game, and then have it map observations to a hidden latent state that is blind to symmetrical states of the game. Such a network can be trained using a similar idea to Cycle-Consistent Adversarial Networks (CycleGAN) (Zhu et al. 2017), where it takes as input sampled trajectories, and evaluates its known mappings by performing recolorings in a cycle-consistent manner such that it is eventually able to recover the original coloring. Whether a cycle-consistent recoloring is correct can be trained by whether its predictions of the next state for a given action in the latent state of the model matches that of what was actually observed.

Another possible method of resolving the issue with learning a uniform policy across symmetries of the game due to the fact that it does not visit all states of the game equally is by adding a curiosity-driven exploration reward to the game. This provides additional rewards for states that are either very rarely or never visited. To this end, there are many techniques in the reinforcement learning literature that can be adapted (Burda et al. 2018), (Savinov et al. 2019), (Ecoffet et al. 2019).

A further line of inquiry lies in being able to fine-tune, introduce, and/or learn conventions as part of a single game or a set of games. This is typically observed when two human players start to play together for the first time and establish a common set of conventions. Existing techniques, while they do learn a policy, are unable to learn to mutate the policy to develop mutual conventions with the other player during a single game or a fixed set of games.

References

[Bard et al. 2020] Bard, N.; Foerster, J. N.; Chandar, S.; Burch, N.; Lanctot, M.; Song, H. F.; Parisotto, E.; Dumoulin, V.; Moitra, S.; Hughes, E.; Dunning, I.; Mourad, S.; Larochelle, H.; Bellemare, M. G.; and Bowling, M. 2020. The hanabi challenge: A new frontier for ai research. *Artificial Intelligence* 280:103216.

[Burda et al. 2018] Burda, Y.; Edwards, H.; Pathak, D.; Storkey, A. J.; Darrell, T.; and Efros, A. A. 2018. Large-scale study of curiosity-driven learning. *CoRR* abs/1808.04355.

[Chong, Ho, and Camerer 2016] Chong, J.-K.; Ho, T.-H.; and Camerer, C. 2016. A generalized cognitive hierarchy model of games. *Games and Economic Behavior* 99:257–274.

[Conitzer and Oesterheld] Conitzer, V., and Oesterheld, C. Foundations of cooperative ai.

[Ecoffet et al. 2019] Ecoffet, A.; Huizinga, J.; Lehman, J.; Stanley, K. O.; and Clune, J. 2019. Go-explore: a new approach for hard-exploration problems. *CoRR* abs/1901.10995.

[Hu et al. 2020] Hu, H.; Lerer, A.; Peysakhovich, A.; and Foerster, J. N. 2020. ”other-play” for zero-shot coordination. *CoRR* abs/2003.02979.

[Hu et al. 2021] Hu, H.; Lerer, A.; Cui, B.; Wu, D.; Pineda, L.; Brown, N.; and Foerster, J. 2021. Off-belief learning.

[Savinov et al. 2019] Savinov, N.; Raichuk, A.; Vincent, D.; Marinier, R.; Pollefeys, M.; Lillicrap, T.; and Gelly, S. 2019. Episodic curiosity through reachability. In *International Conference on Learning Representations*.

[Tucker, Zhou, and Shah 2020] Tucker, M.; Zhou, Y.; and Shah, J. 2020. Adversarially guided self-play for adopting social conventions. *CoRR* abs/2001.05994.

[Zhu et al. 2017] Zhu, J.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR* abs/1703.10593.