Encoding Epistemic Strategies for General Game Playing

Shawn Manuel, David Rajaratnam, and Michael Thielscher

School of Computer Science and Engineering
University of New South Wales
AUSTRALIA

Abstract. We propose a general approach for encoding epistemic strategies for playing incomplete information games. A game strategy involves selecting actions in order to maximise an outcome (e.g., winning the game). In an epistemic strategy the selection of actions is based on reasoning about the knowledge of other players. We show how epistemic strategies can be encoded by supplementing a GDL-II game description with a set of epistemic rules to produce a GDL-III game that an appropriate reasoner can use to play the original GDL-II game. We prove the formal correctness of this approach and provide a practical evaluation to show its efficacy for playing the co-operative multi-player game of Hanabi. It was found that the encoded epistemic rules were able to provide players with a strategy that allowed them to play Hanabi near optimally.

1 Introduction

General Game Playing (GGP) is a sub-field within AI aimed at creating systems that can learn to play a variety of strategy games when given only the game rules at runtime [11]. Unlike specialised systems, a general game player cannot rely on game specific algorithms that have been designed in advance. Instead it requires a form of general intelligence that enables the player to autonomously adapt to new games. This is exemplified by the annual international GGP competition [13].

A feature of GGP is the Game Description Language (GDL) used to specify complete information games [11], and subsequently extended to deal with *imperfect information* games (GDL-II) [18]. However, while GDL-II can be used to specify the rules of imperfect information games, it lacks the expressive power to describe the *strategy* a player should follow to actually play such a game. In particular, a multi-agent, imperfect information game typically requires players to reason about the knowledge, or *epistemic state*, of other players in order to play effectively [1]. Such a player is said to be following an *epistemic strategy*.

While epistemic strategies cannot be encoded in GDL-II, a more recent language extension, GDL-III, does allow for the specification of epistemic goals and rules (GDL-III; for GDL with *imperfect information* and *introspection*) [22]. While intended as a language for describing epistemic games, it has also been used to model epistemic puzzles, such as the 'Muddy Children' puzzle, where the goal of each child is to know whether or not she has mud on her forehead [21].

In this paper we introduce a further application of GDL-III; as a language for representing epistemic strategies. In particular, we provide a framework for encoding epistemic strategies for GDL-II games within the GDL-III language. Any GDL-III reasoner (i.e., a logical reasoner that can track the state of a GDL-III game) can then be co-opted into being an effective GDL-II game player.

To motivate the use of epistemic strategies, and to study the potential efficacy of our approach, we consider the game of Hanabi[1]. Hanabi has been the subject of recent interest [17,9,7,4,20,24,10], and has been proposed as a new frontier for AI research in a similar league to games such as poker and Go [5]. Hanabi is well-suited for our purposes, as the game rules require no epistemic properties (i.e. it is GDL-II representable), yet it is a multi-player, imperfect information game that requires players to reason about the knowledge of other players in order to play effectively. Existing AI players are specialised [17,9,10,20] with epistemic strategies that can be abstracted from the underlying search algorithms. We investigate the application of two of these strategies encoded in GDL-III.

The rest of the paper proceeds as follows: Section 2 provides background to GGP and Hanabi, Section 3 formalises the encoding of epistemic strategies in GDL-III, Section 4 outlines the modeling of strategies for Hanabi, and Section 5 provides an experimental evaluation of these strategies.

2 Background

The Game Description Language (GDL) [12], and its extension GDL-II for imperfect-information games [19], is a formal language for specifying the rules of strategy games to a general game-playing system. GDL uses a prefix-variant of the syntax of logic programs along with the following special keywords:

(**role** ?r)	?r is a player
(**init** ?f)	feature ?f holds in the initial position
(**true** ?f)	feature ?f holds in the current position
(**legal** ?r ?m)	?r has move M in the current position
(**does** ?r ?m)	player ?r does move M
(**next** ?f)	feature ?f holds in the next position
terminal	the current position is terminal
(**goal** ?r ?v)	player ?r gets payoff ?v
(**sees** ?r ?p)	player ?r observes ?p in the next position
random	the random player (aka. Nature)

GDL-II can be used to describe a variety of commonly played imperfect-information games (see http://ggpserver.general-game-playing.de).

Example Hanabi is a fully cooperative, incomplete information game where a team of two to five players work together to play cards from a deck in order to complete up to five stacks of sequentially numbered cards. Crucially, each player

[1] https://en.wikipedia.org/wiki/Hanabi_(card_game)

only sees the cards of the other players' and must therefore rely on those other players to inform them about cards in their own hand in order to play correctly.

The game uses a special deck of 50 cards consisting of five colours where each colour has 10 ranked cards from 1 to 5. Each player begins with a randomly dealt hand of four or five cards where a player's hand is held with the cards facing away such that only the other players can see their colour and rank. Play proceeds with players taking turns to select one of three types of actions:

– **Play**: Select any one of the player's own cards to reveal; and add it to a stack of the same colour if its number is the next in sequence for that stack.
– **Discard**: Select any one of the player's own cards to discard from the game.
– **Hint**: Select another player and declare the positions of all cards in their hand that share the same colour or rank.

The game also features two types of tokens, *information* and *life* tokens. Information tokens restrict the number of hint actions that can be made and can only be regained when a discard action is taken. Life tokens limit the number of unsuccessful play actions where all players lose if the last life token is lost.

An example set of rules for a Hanabi version with just 2 players, colours and ranks and a hand size with 1 card position is shown below[2]:

```
(role random) (role player1) (role player2)

(cardCol ukn) (cardCol red) (cardCol grn)
(cardNum ukn) (cardNum 1) (cardNum 2)
(position 1) (succ 0 1) (succ 1 2)

(<= (card ?colour ?number)
    (cardCol ?colour) (cardNum ?number))

(<= (legal ?r (play ?pos))
    (true (control ?r))
    (true (hand ?r ?pos (card ?col ?num))))

(<= (legal ?r noop)
    (role ?r) (not (true (control ?r))))

(<= (sees ?r1 (does ?r2
              (play ?pos (card ?col ?num))))
    (role ?r1) (does ?r2 (play ?pos))
    (true (hand ?r2 ?pos (card ?col ?num))))

(<= (next (stacksize ?col ?x))
    (does ?r (play ?pos (card ?col ?num)))
    (correct_play ?r ?pos))

(<= (correct_play ?r ?pos)
    (true (hand ?r ?pos (card ?col ?num)))
```

[2] For the full Hanabi GDL-II encoding see: https://git.io/fhbVz

```
(true (stacksize ?col ?prev))
(succ ?prev ?num))
```

Semantics A game description Σ that obeys GDL syntactic restrictions [16] determines a state transition system as follows. A move `m` is *legal* for role `r` in state $s = \{$`f1...fn`$\}$ if `(legal r m)` follows from Σ and the facts $s^{\texttt{true}} = \{$`(true f1)...(true fn)`$\}$. Given a state s and a joint move M (i.e. a legal move `m` for every player `r`), the *updated state* $u(M, s)$ consists of all `f` for which `(next f)` follows from Σ and the facts $s^{\texttt{true}} = \{$`(true f1)...(true fn)`$\}$ and $M^{\texttt{does}} = \{$`(does r1 m1)...(does rk mk)`$\}$. The *observations* for player `r` after joint move M in state s are given by the derivable instances of `(sees r p)` in the same way. For example, consider $s = \{$`(hand player1 1 (card red 2))`, `(control player1)`, `(stacksize red 1)`$\}$. Then the move `(play 1)` is legal for `player1` and `noop` is legal for both `player2` and **`random`**. The state resulting from this legal move is $\{$`(stacksize red 2)`$\}$, where both players will observe `(does player1 (play 1 (card red 2)))`.

Definition 1 ([19]). *The* semantics *of a valid GDL-II game description G is given by*

- $R = \{r\colon G \models$ `(role r)`$\}$
- $s_0 = \{f\colon G \models$ `(init f)`$\}$
- $t = \{S\colon G \cup s^{\texttt{true}} \models$ **`terminal`**$\}$
- $l = \{(r, m, S)\colon G \cup s^{\texttt{true}} \models$ `(legal r m)`$\}$
- $u(M, S) = \{f\colon G \cup M^{\texttt{does}} \cup s^{\texttt{true}} \models$ `(next f)`$\}$
- $\mathcal{I} = \{(r, M, S, p)\colon G \cup M^{\texttt{does}} \cup s^{\texttt{true}} \models$ `(sees r p)`$\}$
- $g = \{(r, v, S)\colon G \cup s^{\texttt{true}} \models$ `(goal r v)`$\}$

Legal play sequences are sequences of joint moves, beginning in the initial state, in which all players always select a legal move. Legal play sequences δ and δ' are *indistinguishable* by player r (i.e., are in the same information set), written $\delta \sim_r \delta'$, iff r's moves and observations are identical in δ and δ'.

GDL-III Hanabi can be sufficiently described as a GDL-II game with incomplete information where the value of the colour and number of cards in each position in a player's hand is hidden until they play or discard it. However, playing optimally requires players to maintain a knowledge base for each player of the known facts about each card in their hand to determine which cards are correct plays and what information other players need to identify correct plays. This provides an opportunity to use an epistemic strategy encoded in a recent extension of GDL to reason about a player's knowledge when selecting an action. GDL-III [22] introduces the following keywords in order to support the axiomatisation of game rules that depend on the knowledge of players:

`(knows ?r ?p)`	player `?r` knows `?p` in the current state
`(knows ?p)`	`?p` is common knowledge in the current state

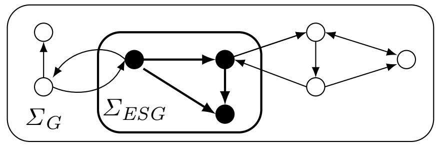

Fig. 1. Epistemic strategy game Σ_{ESG} is a sub-game of an existing GDL-II game Σ_G.

The semantics for GDL-III is more involved since the transition system is now also shaped by what players know: Let (s, K) be an arbitrary *knowledge state*, where s is a state and K a set of ground **`knows`**-instances. Move `m` for role `r` is *legal* in (s, K) if `(`**`legal`**` r m)` can be derived from $\Sigma \cup s^{\texttt{true}} \cup K$. The *resulting state* when joint move M is executed in (s, K) consists of all `f` such that `(`**`next`**` f)` can be derived from $\Sigma \cup s^{\texttt{true}} \cup K \cup M^{\texttt{does}}$. Player `r` *observes* `p` when M is executed in (s, K) iff `(`**`sees`**` r p)` can be derived from $G \cup s^{\texttt{true}} \cup K \cup M^{\texttt{does}}$. By definition, the initial state is common knowledge among the agents. A *legal play sequence* is a sequence of joint moves, beginning in the initial knowledge state, in which all players always select a legal action. Legal play sequences δ and δ' are *indistinguishable* by r, written $\delta \sim_r \delta'$, iff r's moves and observations are the same in δ and δ'. Player r *knows* a property ϕ after a legal play sequence δ iff ϕ is true in all δ' that r cannot distinguish from δ. Finally, ϕ is *common knowledge* after δ if it holds after all δ' in the transitive closure $\sim^C$ of $\bigcup_{r \in R} \sim_r$.

3 A Framework for Epistemic Strategies

In this section we formally introduce the framework for representing epistemic strategies as GDL-III games. We provide a general method to transform an arbitrary GDL-II game into a GDL-III game through the addition of legal move definitions based on the knowledge of players. We then prove a number of properties; starting with the correctness of the framework, through to establishing that the framework defines a computationally interesting GDL-III fragment.

3.1 Defining the Transformation

The general approach of enriching a given "source" game with a set of epistemic strategy rules is illustrated in Figure 1. A game can be viewed as a state transition system, with the joint moves defining the transition between states. Adding a set of epistemic strategy rules limits the available moves to only those that follow the strategy; effectively pruning the state space and defining a sub-game.

We define the GDL axiomatisation of this process in two stages. We first define a transformation function for arbitrary GDL-II games and then define how this transformation is applied to create a GDL-III game.

Definition 2. *Let Σ_G be a GDL axiomatisation of a GDL-II game. The set of rules $\tau(\Sigma_G)$ is obtained from Σ_G as follows:*

- *if* `ep_legal` *or* `src_legal` *are predicates defined in* Σ_G*, replace them with unique names not occurring in* Σ_G*,*
- *replace every occurrence of* `(legal r a)` *with* `(src_legal r a)`*, for arbitrary* `r` *and* `a`*; and*
- *add the following rules:*

```
(<= (legal ?r ?a) (ep_legal ?r ?a))
(<= (ep_legal ?r) (ep_legal ?r ?a))
(<= (legal ?r ?a)
    (not (ep_legal ?r)) (src_legal ?r ?a))
```

Definition 2 provides a purely syntactic transformation of a GDL-II game. It should also be noted that the resulting transformation is itself a legal GDL-II game. Furthermore, because `(ep_legal ?r ?a)` is used but not defined in the game $\tau(\Sigma_G)$, the non-monotonicity of default negation in the third additional rule ensures logical equivalent to the original game.

However, the default negation of the third rule can also be used to provide a mechanism for incorporating an epistemic strategy, which we outline now. For the following definition we rely on the notion of the *dependency graph* of a logic program and the related concepts of a logic program being *stratified* [3] and *safe* (or *allowed*) [15]. It should be noted that the GDL specification requires that all game descriptions are stratified and safe [16].

Definition 3 (Epistemic Enrichment). *For a GDL-II game* Σ_G *and a transformation function* τ *satisfying Definition 2, let* Σ_{ESG} *be the GDL-III game:*

$$\Sigma_{ESG} = \tau(\Sigma_G) \cup \Sigma_{strategy} \text{ ,where}$$

$\Sigma_{strategy}$ *is a set of GDL-III rules satisfying the following requirements:*

- $\Sigma_{strategy}$ *must be both stratified and safe,*
- *rule heads cannot include the GDL keyword predicates (e.g.,* **`next`***,* **`true`***),*
- *rule heads cannot include auxiliary predicates that have been defined in* Σ_G*,*
- *any rule containing* `(ep_legal r a)` *in the head, for arbitrary* `r` *and* `a`*, must also include* `(src_legal r a)` *as a positive dependency in the dependency graph for* $\Sigma_{strategy}$*.*

Definition 3 turns a GDL-II game into a GDL-III game by enriching the original game with a set of rules where the **`knows`** predicate can occur in the body of the rule. This new GDL-III game can be used by a GDL-III reasoner to play the original game; where it chooses actions if there is an epistemic strategy for the given state or simply plays a legal move otherwise. The effectiveness of the resulting player will be dependent on the extent to which the epistemic rules are able to encode a winning strategy.

3.2 Properties

In this section we establish some basic properties of our framework. In particularly, we show that, despite the difference in semantics between GDL-II and GDL-III, there is a clear link between the GDL-III game constructed through the method defined in Definition 3 and the original GDL-II game.

Proposition 1. *For any syntactically correct GDL-II game Σ_G and set of epistemic strategy rules $\Sigma_{strategy}$, the resulting epistemically enriched game Σ_{ESG} is a syntactically correct GDL-III game.*

Proof. Σ_G satisfies GDL keyword restrictions, is safe and stratified (see [16]). These properties are preserved by $\tau(\Sigma_G)$. The restriction on $\Sigma_{strategy}$ also satisfies these properties such that $\tau(\Sigma_G) \cup \Sigma_{strategy}$ is also stratified and safe. □

In order to establish further properties we use the notion of *legal play sequences*, where a sequence of moves $M_1, \ldots, M_n$ corresponds to legal moves from a starting state s_0, such that $s_i = u(M_i, s_{i-1})$, for state update function u. In particular, we establish that the game Σ_{ESG} is a restriction over the game Σ_G.

Proposition 2. *Given any GDL-II game Σ_G and set of epistemic strategy rules $\Sigma_{strategy}$, then for the resulting GDL-III game Σ_{ESG}, and legal play sequence of Σ_{ESG}, $M_1, \ldots, M_n$ with corresponding states $s_0, s_1, \ldots, s_n$ it holds that:*

- *the sequence $M_1, \ldots, M_n$ is a legal play sequence of Σ_G with corresponding states $s'_0, s'_1, \ldots, s'_n$.*
- *if s_n is a terminal state of Σ_{ESG} then s'_n is a terminal state of Σ_G.*
- *for each player r the goal value $g(r, s_n)$ of Σ_{ESG} in state s_n will be the same as the goal value $g(r, s'_n)$ in game Σ_G and corresponding state s'_n.*

Proof. By induction on the states in the play sequence. The initial state of a game in GDL-II/III corresponds to the fluents defined by **`init`**. Since $\tau(\Sigma_G)$ and $\Sigma_{strategy}$ do not introduce any new fluents or modify **`init`** therefore the objective fluents of the initial state of Σ_G will be identical to Σ_{ESG}; furthermore $\tau(\Sigma_G)$ does not change **`goal`** values or how **`terminal`** is determined.

Consider the first move M_1 and an arbitrary role r of Σ_{ESG}; $M_1(r) = a$ is a legal action for r in s_0 for game Σ_{ESG}. Hence either `(ep_legal r a)` or `(src_legal r a)` is true in s_0 (Definition 2). But if `(ep_legal r a)` is true then `(src_legal r a)` must also be true (by the dependency restriction in Definition 3). But `(src_legal r a)` is simply the rewrite of `(`**`legal`**` r a)` from the original game, hence `(`**`legal`**` r a)` must also be a legal move for role r in s'_0 for game Σ_G. Hence M_1 is also a legal move in Σ_G and so the transition from s'_0 by M_1 will also be a state s'_1 of Σ_G. Furthermore if s'_1 is terminal in Σ_{ESG} it will also be a terminal state of Σ_G with identical goal values for each player.

The same argument holds for the induction step; where assuming $M_1, \ldots, M_i$ $(i < n)$ is also a legal sequence of Σ_G with states $s'_0, \ldots, s'_i$, then s'_{i+1} is also a state of Σ_G with the correct termination and goal values. □

Now, Proposition 2 establishes that Σ_{ESG} represents a restriction over Σ_G and allows this restriction to be determined by a player's knowledge or the common knowledge of all players. However, in general GDL-III is a strictly more expressive language than GDL-II, since, determining if a game terminates in GDL-III is in general undecidable, even when subject to the usual syntactic restrictions that ensure the finiteness of the state space in GDL-II [22]. Consequently, the syntactic restrictions of Definition 3 result in the identification of an interesting fragment of GDL-III.

Proposition 3. *Given any GDL-II game Σ_G and set of epistemic strategy rules $\Sigma_{strategy}$, then for the resulting epistemically enriched game Σ_{ESG} determining if Σ_{ESG} terminates is decidable.*

Proof. A direct consequence of Proposition 2. Any legal play sequence of Σ_{ESG}, $M_1, \ldots, M_n$ with corresponding states $s_0, s_1, \ldots, s_n$ where s_n is a terminal state, is also a legal play sequence of Σ_G and corresponding state s'_n is also a terminal state of Σ_G. But Σ_G is a GDL-II game so determining termination is decidable, hence determining termination of Σ_{ESG} is also decidable. □

The key to the decidability of Σ_{ESG} is that the syntactic restrictions ensures that the truth of fluents in a state or the termination of a state is independent of the knowledge of players. This is not true of GDL-III in general.

Hence not only does the proposed framework allow for the encoding of epistemic strategies for playing specific games, but it does so in a manner that preserves the decidability of the original game. This means that the encoding of epistemic strategies in GDL-III is both of theoretical and potentially of practical interest. In the following section we apply this theory to encoding strategies for the game of Hanabi and show the efficacy of the approach.

4 Encoding Epistemic Strategies in Hanabi

This section outlines the GDL-III encoding of two epistemic rule-based strategies that can be used to extend the original GDL-II Hanabi source game to allow players to reason about game knowledge when selecting a move.

The two strategies considered are the *information strategy* and the *implicit strategy*. The information strategy takes a conservative approach to playing Hanabi. Hint moves are given to inform other players of card numbers or colours that are unknown to them, and cards are only played if its holder knows that it is playable based on knowing both its colour and number. This represents an individualistic strategy since it is agnostic to the strategies of other players.

In contrast, the implicit strategy represents a more optimistic approach to playing Hanabi. It selects moves based on players' knowledge, but with the additional implicit assumption that all players are following the same, or at least a pre-agreed, strategy. Hence, this strategy resembles that of an experienced group of players with an agreed convention on how to play the game.

4.1 The Information Strategy

This strategy models the use of hint moves to convey information of card properties that are not known to other players. It is adapted from the *Outer-State* strategy presented in Osawa's original paper on solving Hanabi [17], where players inform each other of properties that have not yet been stated. A player uses this strategy by selecting the first rule whose antecedent is satisfied from a list of epistemic rules. We provide a description of these rules, but also show a sample of the GDL encodings to illustrate the correspondence between the rule explanations and their GDL-III instantiations[3]:

1. If a known playable card is in our hand, play that card.
```
(<=(ep_legal ?r (play ?pos))
   (src_legal ?r (play ?pos))
   (knows ?r (correct_play ?r ?pos)))
```
2. If a known dead card is in our hand and there are clue tokens to be gained, discard that card.
```
(<=(ep_legal ?r (discard ?pos))
   (src_legal ?r (discard ?pos))
   (not (has_legal_play ?r))
   (knows ?r (has_dead_card ?r ?pos)))
```
3. If no known playable cards is in any hand, discard the card with the highest known number.
```
(<=(ep_legal ?r (discard ?pos))
   (src_legal ?r (discard ?pos))
   (not (has_legal_play ?r))
   (not (has_legal_discard ?r))
   (not (has_legal_hint_num ?r))
   (not (has_unknown_card ?r))
   (knows_highest_card ?r ?pos ?num))
```
4. If there are clue tokens, and a player has a playable card, hint its number if not known.
```
(<=(ep_legal ?r (hint ?r1 ?num))
   (src_legal ?r (hint ?r1 ?num))
   (not (has_legal_play ?r))
   (not (has_legal_discard ?r))
   (true (hand ?r1 ?pos (card ?col ?num)))
   (correct_play ?r1 ?pos)
   (not (knows ?r1 (hand ?r1 ?pos (cardNum ?num)))))
```
5. If there are clue tokens, and a player has a playable card, hint its colour if not known.
6. If there are clue tokens, hint a random card's number that is not known.
7. If there are clue tokens, hint a random card's colour.
8. If no clue tokens available, discard the highest known number card in hand.

4.2 The Implicit Strategy

This strategy aims to encode additional facts in certain moves which other players following the same strategy can infer when they observe those moves. This differs from the information strategy to use the hint moves to imply playability of a card based on the property hinted for that card. In the implicit strategy, the number property of a card is hinted if the card is playable otherwise the colour is hinted instead. The resulting strategy rules are as follows[4]:

[3] For the complete GDL-III information strategy encoding see: https://git.io/fhbVo
[4] For the complete GDL-III implicit strategy encoding see: https://git.io/fhbVK

1. If a known playable card is in our hand, play that card.
2. If a known dead card is in our hand and clue tokens to be gained, discard that card.
3. If no known playable cards are in any hand, then discard the card with the highest known number.
4. If there are clue tokens, and a player has a playable card, hint its number.
5. If there are clue tokens, and no known playable cards in any hands, hint a random card's colour.
6. If no clue tokens available, discard a random card.

5 Evaluation

Four experiments were conducted to evaluate the increased performance of the GDL-II game Hanabi extended with a GDL-III epistemic strategy. The first experiment investigated the practical lower bound of playing Hanabi without any strategy, where players select (legal) actions randomly. The second and third experiments evaluated the performance of the information and the implicit strategy respectively. Finally, the fourth experiment provided a crude upper bound by modelling the case of playing Hanabi where the cards in every players' hand is common knowledge, with the only unknowns being the cards in the deck.

Each experiment was run for 50 games each with six configurations of number of players and cards per player and a play clock of 10 seconds. Table 1 below shows the details of each configuration of number of players, **nPl**, with hand size **nHand** and number card counts for each colour. The **Colours** and **Numbers** column indicate the colour and number values used to build the deck.

Table 1. Outline of Hanabi Game Configurations

Id	nPl	Colours	Numbers	nHand	MaxScore
1	2	R,G	1,2	1	4
2	2	R,G	1,2,3	2	6
3	2	R,G	1,1,2,3	2	6
4	3	R,G,B	1,2	1	6
5	3	R,G,B	1,2,3	2	9
6	4	R,G,B,Y	1,2	1	8

The experiments were run on a 2.5GHz MacOS laptop with 16GB memory. A GDL-III knowledge reasoner was implemented in ASP according to the formula in [21] to calculate knowledge at each timestep. This reasoner was adapted for time-restrained GGP matches to incorporate a timeout equal to the play clock that only returns an approximation of the knowledge state. As a result, games

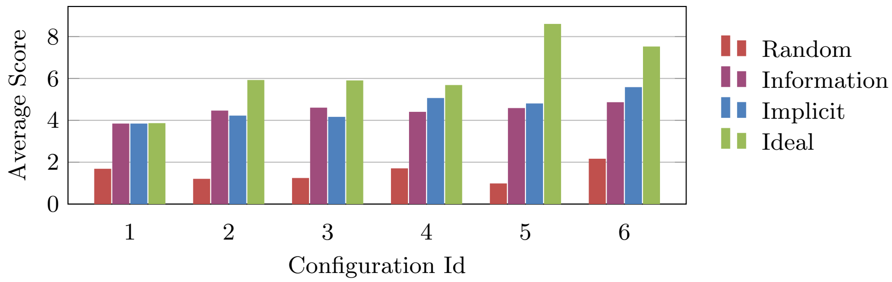

Fig. 2. Comparison of Endgame Results

were played within a reasonable amount of time ranging from 5 seconds for games with a decksize of 4 to 4 minutes at the maximum decksize of 9.

Figure 2 below displays the comparison of the average scores for each experiment from the above table grouped by each of the six configurations. There is a clear increase in the performance of the players following an epistemic strategy to select moves based on their own knowledge and that of other players. For the random players without a strategy, multiple games were lost due to too many incorrect plays resulting in an average score almost always less than 2. On the other hand, the information and implicit strategies provided a similar increase in performance. The information strategy was seen to achieve a more consistent score, where legal play moves tend to follow from a sequence of two hint moves. While this strategy is less variable, it is also highly dependent on clue tokens for increasing its scalability to larger game configurations. The implicit strategy experiences more variance in achievable scores but is able to scale to games with more players due its ability to signal playability of a card with a single hint move. It is also worth noting that for configurations 5 and 6, the knowledge calculation for most time steps exceeded the timeout and resulted in players relying on an incomplete set of knowledge facts. Despite this, games using epistemic strategies were still able to outperform those with players playing randomly.

6 Conclusion

We developed a framework for modelling epistemic games in GGP allowing players to reason about their knowledge of the game state using epistemic strategies. We presented a formal approach to represent these strategies for GDL-II games as specialised GDL-III games. This approach was evaluated for the game of Hanabi where two epistemic strategies, the information and implicit strategies, were used to select player actions. From the evaluation, it was observed that the experimental upper bound for reasoning with complete knowledge was able to achieve a perfect score most of the time, although not guaranteed for all configurations. We then found that the information strategy achieved a more consistent score in contrast to the implicit strategy which achieved a higher maximum score. This was done by encoding an implicit recommendation to play a card if a player was hinted its number, which resulted in more effective use of limited hint actions. Yet this advantage was lost when multiple hints were given for a combination of both playable and unplayable cards.

In terms of related work, a number of logical frameworks exist for reasoning about the strategic abilities of players in games [14,23,8], mostly based on Alternating-time Temporal Logic [2]. However, these logics are based on modalities for the existence of strategies and do not provide means for specifying them [6]. An exception is a recent special-purpose modal logic for reasoning about strategies [25]. The main differences to our framework are that their strategies can only be conditioned on state properties and not on players' knowledge, and that using their logic would first require the development of a special-purpose automated theorem prover.

As future work, probabilistic reasoning could be incorporated within the implicit strategy to allow the same action to indicate either playable or non-playable cards with an associated probability, which could also be used to condition hint moves in some cases. Another avenue would consider optimal action selection using epistemic strategies combined with game independent strategies such as Monte-Carlo Tree Search to reduce the reliance on game specific strategy rules.

7 Acknowledgements

This research was supported under the Australian Research Council's (ARC) Linkage Projects funding scheme (project number LP 180100080).

References

1. Ågotnes, T., Harrenstein, P., van der Hoek, W., Wooldridge, M.: Boolean games with epistemic goals. In: Grossi, D., Roy, O., Huang, H. (eds.) Logic, Rationality, and Interaction — 4th Intl. Workshop, LORI, Hangzhou, China. Lecture Notes in Computer Science, vol. 8196, pp. 1–14. Springer (2013)
2. Alur, R., Henzinger, T.A., Kupferman, O.: Alternating-time temporal logic. Journal of the ACM **49**(5), 672–713 (2002)
3. Apt, K.R., Blair, H.A., Walker, A.: Towards a theory of declarative knowledge. In: Minker, J. (ed.) Foundations of Deductive Databases and Logic Programming, pp. 89–148. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1987)
4. Baffier, J., Chiu, M., Diez, Y., Korman, M., Mitsou, V., van Renssen, A., Roeloffzen, M., Uno, Y.: Hanabi is NP-complete, even for cheaters who look at their cards. In: FUN. LIPIcs, vol. 49, pp. 4:1–4:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2016)
5. Bard, N., Foerster, J.N., Chandar, S., Burch, N., Lanctot, M., Song, F., Parisotto, E., Dumoulin, V., Moitra, S., Hughes, E., Dunning, I., Mourad, S., Larochelle, H., Bellemare, M.G., Bowling, M.: The hanabi challenge: A new frontier for AI research. CoRR **abs/1902.00506** (2019), http://arxiv.org/abs/1902.00506
6. van Benthem, J.: In praise of strategies. In: Eijck, J.V., Verbrugge, R. (eds.) Games, Actions, and Social Software. ILLC scientific publications, Institute for Logic, Language and Computation (ILLC), University of Amsterdam (2008)

7. Bouzy, B.: Playing Hanabi near-optimally. In: Adv. in Computer Games. pp. 51–62. Springer (2017)
8. Chatterjee, K., Henzinger, T.A., Piterman, N.: Strategy logic. Information and Computation **208**(6), 677–693 (2010)
9. Cox, C., De Silva, J., Deorsey, P., Kenter, F.H., Retter, T., Tobin, J.: How to make the perfect fireworks display: Two strategies for Hanabi. Mathematics Magazine **88**(5), 323–336 (2015)
10. Eger, M., Martens, C., Córdoba, M.A.: An intentional AI for Hanabi. In: IEEE Conf. on Computational Intelligence and Games (CIG), pp. 68–75. IEEE (2017)
11. Genesereth, M., Love, N., Pell, B.: General game playing: Overview of the AAAI competition. AI magazine **26**(2), 62 (2005)
12. Genesereth, M., Thielscher, M.: General game playing. Synthesis Lectures on Artificial Intelligence and Machine Learning **8**(2), 20,172–178 (2014)
13. Genesereth, M.R., Björnsson, Y.: The international general game playing competition. AI Magazine **34**(2), 107–111 (2013)
14. van der Hoek, W., Jamroga, W., Wooldridge, M.: A logic for strategic reasoning. In: Proc. of the Fourth Intl. Joint Conf. on Autonomous Agents and Multi Agent Systems (AAMAS-05). pp. 157–164. ACM (2005)
15. Lloyd, J., Topor, R.: A basis for deductive database systems {II}. The Journal of Logic Programming **3**(1), 55 – 67 (1986)
16. Love, N., Henrichs, T., Haley, D., Schkufza, E., Genesereth, M.: General game playing: Game description language specification. Tech. rep., Stanford Logic Group Computer Science Department Stanford University (2006)
17. Osawa, H.: Solving Hanabi: Estimating hands by opponent's actions in cooperative game with incomplete information. In: AAAI workshop: Computer Poker and Imperfect Information. pp. 37–43 (2015)
18. Schiffel, S., Thielscher, M.: Reasoning about general games described in GDL-II. In: AAAI. vol. 11, pp. 846–851 (2011)
19. Schiffel, S., Thielscher, M.: Representing and reasoning about the rules of general games with imperfect information. JAIR **49**, 171–206 (2014)
20. Spieksma, F.M.: Aspects of the cooperative card game Hanabi. In: Artificial Intelligence: 28th Benelux Conf. on Artificial Intelligence (BNAIC), Amsterdam, The Netherlands, Revised Selected Papers. vol. 765, p. 93. Springer (2017)
21. Thielscher, M.: A formal description language for epistemic games. Tech. Rep. UNSW-CSE-TR-201708, The University of New South Wales (2017)
22. Thielscher, M.: GDL-III: A description language for epistemic general game playing. In: Sierra, C. (ed.) IJCAI. pp. 1276–1282. AAAI Press, Melbourne (Aug 2017)
23. Walther, D., van der Hoek, W., Wooldridge, M.: Alternating-time temporal logic with explicit strategies. In: Proc. of the 11th Conf. on Theoretical Aspects of Rationality and Knowledge. pp. 269–278. ACM (2007)
24. Walton-Rivers, J., Williams, P.R., Bartle, R., Perez-Liebana, D., Lucas, S.M.: Evaluating and modelling Hanabi-playing agents. In: Evolutionary Computation (CEC), 2017 IEEE Congress on. pp. 1382–1389. IEEE (2017)
25. Zhang, D., Thielscher, M.: A logic for reasoning about game strategies. In: AAAI. vol. 15, pp. 1671–1677 (2015)