2019

Clues About Bluffing in Clue: Is Conventional Wisdom Wise?

David Hansen

Kyle D. Hansen

Clues About Bluffing in Clue : Is Conventional Wisdom Wise?

David M. Hansen *Affiliate Member, IEEE*[1], Kyle D. Hansen[2]
[1]College of Engineering, George Fox University, Newberg, OR, USA
[2]Westmont College, Santa Barabara, CA, USA

We have used the board game Clue as a pedagogical tool in our course on Artificial Intelligence to teach formal logic through the development of logic-based computational game-playing agents. The development of game-playing agents allows us to experimentally test many game-play strategies and we have encountered some surprising results that refine "conventional wisdom" for playing Clue. In this paper we consider the effect of the oft-used strategy wherein a player uses their own cards when making suggestions (i.e., "bluffing") early in the game to mislead other players or to focus on acquiring a particular kind of knowledge. We begin with an intuitive argument against this strategy together with a quantitative probabilistic analysis of this strategy's cost to a player that both suggest "bluffing" should be detrimental to winning the game. We then present our counter-intuitive simulation results from playing computational agents that "bluff" against those that do not that show "bluffing" to be beneficial. We conclude with a nuanced assessment of the cost and benefit of "bluffing" in Clue that shows the strategy, when used correctly, to be beneficial and, when used incorrectly, to be detrimental.

Index Terms—**Benchmarking, Board games, Competitions, Multi-player games, Strategy games**

I. Introduction

For many years we have used Clue®[1] as a pedagogical tool[2] in our Artificial Intelligence course at George Fox University as a motivating platform for the development of "intelligent" computational agents [1].

Clue is a strategy game of reasoning with limited knowledge to deduce an initially hidden solution — the "who, what where" of a murder. The game's domain consists of 6 persons ("suspects"), 6 weapons, and 9 rooms, each represented by a card in the physical game. Game setup involves drawing one random card from each category (suspect, weapon, room) and hiding them as the solution. The remaining 18 cards are shuffled and dealt to the players.[3]

The game is played on a physical board that depicts the rooms distributed in space and, on each turn, a player rolls a die to move about the board with the intent of entering a room where they can make a guess about what suspect, weapon, and room are in the solution. These guesses as referred to as "suggestions" and the one restriction on a suggestion is that it must include the room where the player is located. When a player makes a suggestion, opponents are polled in turn-order as to whether or not they are able to "refute" the suggestion by privately revealing to the suggester alone which card in the suggestion the opponent holds. Once the suggestion has been refuted, no further opponents are polled and the turn passes to the next player. Once a player has deduced the hidden set of cards in the solution they may make a final game-winning "accusation" on their turn.[4][5]

In our AI course, students develop "agents" (i.e., computer programs) that are capable of playing Clue using a game-server we have developed that enables students to compete against one another, and faculty, to test the logic and strategies employed by their agents. Over time, some interesting results have emerged with respect to the strategies that are most effective at winning the game. Somewhat surprisingly, one strategy many human players and student agents routinely employ — using cards in a suggestion that they hold in their hand — is not well-studied yet assumed to be beneficial.[6]

We begin with a discussion of common logical approaches to playing the game of Clue. We then present an intuitive and probabilistic analysis of the effects of using one's own cards in suggestions followed by counter-intuitive experimental evidence of the effect of using this strategy.

II. Related Work

Clue has sometimes been used as a pedagogical device for teaching formal logic [2], [3]. Others have used Clue to investigate how to develop higher-level systems for machine learning and automated theorem proving [4]–[6]. While these approaches formalize the logic for playing a legal and even "intelligent" game of Clue, they do not address game-play strategy in general, nor the use of one's own cards in suggestions in particular.

However, non-academic web sites with discussions of strategies for playing Clue endorse using one's own cards in suggestions with a variety of rationales that generally include 1) confusing opponents or 2), gathering information on a particular kind of card [7]–[10].

With respect to the strategy of using cards one holds, Thomas Ferguson addresses what he denotes as "bluffing" in the context of "Two-Person Zero-Sum Games" in Part II of

[1]Clue is known outside the United States as Cluedo®.

[2]Calling it *Glomus* — a rough Latin translation of the word "clue" being a skein of yarn of the sort used by Ariadne to escape the maze in the myth of the Minotaur.

[3]Games can be played by 2-6 players and cards in games of 4 or 5 players will not be dealt evenly.

[4]There is restriction that the player occupy the room used in their accusation

[5]A false accusation ends the game for a mistaken player.

[6]Students in our course who are familiar with the game routinely discuss and use this strategy.

his textbook on Game Theory [11]. Ferguson describes a Clue-like game "Guess It!" to demonstrate the need for "bluffing" by a player to prevent an opponent from winning a game by presuming the player is always "honest":

> From a deck with $m + n + 1$ distinct cards, m cards are dealt to Player I, n cards are dealt to Player II, and the remaining card, called the "target card", is placed face down on the table. Players know their own cards but not those of their opponent. The objective is to guess correctly the target card. Players alternate moves, with Player I starting. At each move, a player may either
>
> (1) guess at the target card, in which case the game ends, with the winner being the player who guessed if the guess is correct, and his opponent if the guess is incorrect, or
>
> (2) ask if the other player holds a certain card. If the other player has the card, that card must be shown and is removed from play.
>
> With a deck of say 11 cards and each player receiving 5 cards, this is a nice playable game that illustrates need for bluffing in a clear way. If a player asks about a card that is in his own hand, he knows what the answer will be. We call such a play a *bluff*. If a player asks about a card not in his hand, we say he is *honest*. If a player is always honest and the card he asks about is the target card, the other player will know that the requested card is the target card and so will win. Thus a player must bluff occasionally. Bluffing may also lure the opponent into a wrong guess at the target card [11, p II-63].

While useful, this limits consideration of "bluffing" and "honesty" to the context of Two-Person Zero-Sum Games such as "Guess-It!"; these strategies are not addressed in *multi*-player games such as Clue. Moreover, Ferguson does not address the strategic question of when or how often one should bluff to optimize one's chance of winning, perhaps due to the difficulty of integrating "higher-order information (what are the other players thinking)" [12].

III. Game-play

The design of the board game suggests a game-play strategy of using direct proof through the "process of elimination" by learning which players hold what cards — the remaining card therefore being part of the solution. Players must also strategically move about the board in order to acquire information about the room where the "murder" took place.

A. Proving "Whodunit"

There are two ways to prove "whodunit." The first proof technique is through the process of elimination and is the mechanism most casual board game players use owing to the inclusion of a paper notepad designed for this purpose. As the game progresses, players learn what cards other players have in their hands and eliminate them from consideration. Thus a card is known to be part of the solution when there is *exactly one card left* that has not been determined to be in some player's hand. We might express this deductive rule concisely using predicate logic

$$
\begin{aligned}
&isInSolution(Card)\\
&\qquad \leftarrow \exists! Card \forall Player(\neg has(Player, Card))
\end{aligned}
$$

A player can be known to have a card in their hand if

- I am the player and have the card in my hand, or
- a player refuted my suggestion by showing the card to me, or
- a player refuted a suggestion and we know that the player can not have in their hand the other two cards used in the suggestion

Most casual board game players rely on the first two mechanisms and omit the third that depends on determining what cards other players *can not have* in their hands. Others formally [2], [3] and informally [10], [13], [14] observe that, with some additional note-taking, the "process of deduction" can also be employed by knowing what cards players do *not* have in their hand — a card all players can *not* have therefore being part of the solution. Leveraging a player's inability to refute a suggestion leads to the second proof technique.

The second proof technique is through the "process of deduction." This technique relies on deducing and inferring what cards are in the solution by observing what cards players *can not have* in their hands and drawing an indirect conclusion about what card is part of the solution. Thus a card is known to be part of the solution when it is known that *every* player *can not have* it in their hand

$$
\begin{aligned}
&isInSolution(Card)\\
&\qquad \leftarrow \forall Player(canNotHave(Player, Card))
\end{aligned}
$$

A player can be known to *not* have a card if

- the player was unable to refute a suggestion that included the card, or
- some other player is known to have the card, or
- the card is known to be part of the solution

An observant reader may notice the "indirect recursive" nature of these two proof techniques as proving what a player *has* may depend on proving what another player *can not have* which may depend on proving what another player *has*... The "mutually recursive" nature of these two proof techniques poses a programming challenge for developing computational agents [5]. Straightforward programming solutions can be implemented but are outside the scope of this paper and left as an exercise for the reader.

B. Game-play Heuristics

A reasonable game-play heuristic is to suggest cards that have the highest probability of being part of the solution. Based on the proof techniques described above, this means suggesting a card that 1) is not known to be in any player's hand and 2), we know most other players *can not have* in their hand. Suppose we have six players and the traditional number of six weapons. If we know that a particular weapon is not in the hands of three players then there remain four possible locations for that card; it must be in one of the remaining three players' hands or the solution.[7] This weapon would make a more logical choice to explore than another weapon we know nothing about that may be present in any other player's hand as well as the solution.

Heuristics for choosing what room to suggest are complicated by the fact that the player must be present in the room used in their suggestion. All things being equal, the general heuristic used for suggesting the most probable suspect and weapon still applies. However, as the game progresses, it becomes increasingly rare that a player has a choice between two viable rooms in the same turn.[8] While a discussion of the heuristics for choosing where to move is beyond the scope of this paper, the computational agent we have developed to compete with our students uses over 25 heuristic rules to choose what room to suggest. We make no claim that the rules we use are optimal, only that they seem reasonable, are effective in playing against other agents, and serve to demonstrate that Clue has more complexity than it may appear.

There are also useful game-play heuristics one can employ when choosing how to refute another player's suggestion. For example, if one has both the suspect and the room that was suggested, one might choose to show the suspect since room choices are constrained and a player must leave and re-enter the room before suggesting it again. On the other hand, if one holds only one suspect card but many room cards, one may expect that the opponent has more suspect cards and fewer room cards in their hand and perhaps it would be better to show them a room card. As the game progresses, it is reasonable to reveal cards that one has previously shown other players since the fact that one holds such a card has likely been deduced by others using the rules of deduction described in Section III-A.

These game-play heuristics are by no means meant to be exhaustive nor authoritative and we refer interested readers to work by others discussed in Section II. As we have explored Clue in the development of computational agents, our students find that there is far more complexity and nuance to the game than it appears and even subtle game-play heuristics can have a measurable effect on an agent's performance.

One strategy we have yet to discuss is making a suggestion that includes cards one holds in their own hand. There are three common scenarios where this is considered to be advantageous. One recommended scenario is to suggest one's own cards in order to focus on a particular kind of card (e.g., use a suspect and room from one's hand to try to gain information specifically about the weapon). Many players (e.g., [14] and students of our AI course) report using this strategy when playing the board game. A related scenario is to suggest one's own cards early in the game to lead other players astray. A third scenario arises later in the game when a player has determined that some card, e.g. a particular weapon, is part of the solution. It is then reasonable for that player to suggest a weapon card they hold so that 1) their suggestion gathers information about the remaining unknown cards in the solution while 2), not leading others to the deduce the actual card in the solution. These uses of one's own cards corresponds to Ferguson's notion of "bluffing" and we will adopt the term "bluffing" to generally mean suggesting one or more cards from one's own hand for these purposes.

IV. Assessing The Conventional Wisdom of Bluffing

Our computational Clue-playing agents were initially designed to mimic the way we played the board game by periodically bluffing early in the game. But results emerged during the development and testing of our computational agents that led us to reconsider this strategy and assess, formally and experimentally, whether this "conventional wisdom" was wise.

A. An Intuitive Argument Against Bluffing

Although the design of the board game suggests using the "process of elimination" to win the game, the "deductive" proof technique described in Section III-A leads to a solution more quickly. To understand why this is so, consider what information players learn during a typical turn. The player who makes a suggestion that is refuted will definitively learn what other player holds a particular card. For many casual board game players who rely on the "process of elimination", that is the only information gained during a turn. However, using "deduction", this player has also learned that every other player can not have that card. Furthermore, every player that is unable to refute a suggestion yields up to 3 times as much information to all other players who definitively learn that player can not have the cards suggested. Thus turns can yield a good deal of information that can be used to quickly "deduce" the solution. Given that players acquire more information about who does *not* have a card and can arrive at the solution more quickly using deduction, we now give an intuitive explanation for the cost of bluffing.

Let us first consider the extreme case where a player makes an irrefutable suggestion using *3* cards they hold in their hand (a strategy sometimes suggested by players [9], [14]). In this instance, the player making the suggestion acquires *no* information about what other players hold in their hands to aid in a "proof by elimination." The player also acquires *no* information to aid in a "deductive" proof since they *already* know that the other players do not have the cards they used in the suggestion as noted by *Gregor* and *user1873* in [8]. Worse yet, this strategy yields the *maximum* information that can be gained by opponents who will learn that every other player does not have the 3 cards suggested; the cards can be in only one of two places: 1) the hand of the player making the suggestion or 2), the "case file" that holds the solution. And, while this strategy may lead opponents to waste time by focusing on cards they deem "highly probable" that are not part of the solution, the amount of information this strategy gives away to opponents far outweighs any confusion incurred; confusion that is quickly resolved through subsequent turns by other players that will eliminate these cards from consideration.

[7] In [15] Neller and Luo provide a more robust assessment of probability in Clue by sampling Models of where cards could be based on what is known.

[8] Our game-server mentioned in Section 2 employs a simplified connected digraph of rooms, eliminating the "hallways" between them; players may still need to travel through a succession of rooms to reach a viable room.

But what about less aggressive strategies that bluff with only one card?

B. Quantitative Analysis of Bluffing

In what follows, we assume a game of 6 players with 21 cards total: 6 weapons, 6 suspects, and 9 rooms. Thus, each player (and the solution) are given 3 cards. Moreover, we consider bluffing on the very first turn of the game when the player has only knowledge of the cards in their hand, enabling a straightforward and consistent analysis.

Suppose player p_0 makes an "honest" (i.e., without bluffing) suggestion $s = (C_1, C_2, C_3)$, where each of these cards is of a different type. In the average scenario, these cards will be spaced out equally among the 18 cards not in the hand of player p_0. Thus we expect the remaining 15 cards not in s nor the hand of p_0 to be spaced evenly around C_1, C_2, and C_3, so that there are an expected number of 3.75 cards on each side of each card in s. This fact is illustrated in Figure 1 where each $\square$ represents a set of 3.75 cards.

$$handOf(p_0) \quad \square \quad C_1 \quad \square \quad C_2 \quad \square \quad C_3 \quad \square$$

Fig. 1. Expected Honest Distribution

Thus we expect to be refuted on card number 4.75, which is to say, by a hypothetical player $p_{1.58}$. As a discrete problem, the ceiling function helps us make sense of this to say that we expect an honest suggestion to be refuted by player p_2.

Suppose now that p_0 bluffs with a random card B in their hand, so that $g = (C_1, C_2, B)$. We say that a card C is *eligible* if the type of C is not the same as that of B (e.g., if p_0 bluffs with a room, a card is eligible if and only if it is a weapon or a suspect). Without considering the type of card B, we can calculate that 2/3 of all cards are expected to remain eligible. Thus there are an expected $\frac{2}{3} \cdot 18 = 12$ eligible cards outside of the hand of p_0.

Using the same approach as above, we space out the 2 "honest" cards of the suggestion (i.e., C_1 and C_2) with the remaining 10 eligible cards leaving $10/3 \approx 3.33$ cards on each side of C_1 and on each side of C_2 as shown in Figure 2.

$$handOf(p_0) \quad \square \quad C_1 \quad \square \quad C_2 \quad \square$$

Fig. 2. Expected Bluff Distribution

Therefore, we expect to be refuted at card 4.33. Because each player's hand consists of an expected 2 eligible cards, this corresponds to a hypothetical player $p_{2.17}$. Again, the ceiling function shows us that this will tend toward player p_3 refuting the suggestion.

But what does it matter how many opponents are queried before a suggestion is refuted? As in Section IV-A, let us consider what information is gained by players, on average, during a turn.

Without bluffing we expect, on average, that one opponent will be unable to refute a suggestion before it is refuted by the second (i.e., $\lceil 1.58 \rceil$). Thus, on average

- four players, including the player making the suggestion, learn that one player does not have 3 cards,
- the first queried opponent learns nothing,
- the opponent that refutes the suggestion learns that one other player does not have 2 cards (it already knew the other opponent did not have the card they used to refute),
- and the player making the suggestion additionally learns the location of 1 card, telling them that the card is not in the hand of any of the other four players or the solution.

If we consider the information that a player (or the solution) does not have a card to be a "unit" of information, then the player making the suggestion acquires 7 units of information; three opponents each acquire 3 and the opponent who refutes acquires 2 for a total of 11.

When a player bluffs with a single card we expect, on average, that two opponents will be unable to refute the suggestion before refuted by the third (i.e., $\lceil 2.17 \rceil$). Thus, on average

- two opponents learn that two players do not have these 3 cards,
- two queried opponents each learn that one other player does not have the 3 cards,
- the opponent that refutes the suggestion learns that two other players do not have 2 cards, and
- the player making the suggestion learns that two players do not have 2 of the cards (since they themselves hold the third), along with the location of 1 card that tells them three remaining players and the solution do not have that card.

By bluffing, the player making the suggestion now acquires 8 units of information; two opponents each acquire 6, two queried opponents each 4, and the opponent who refutes acquires 4 for a total of 24.

Thus bluffing a single card, on average, provides all five opponents with more than twice the information they would otherwise have obtained while providing little additional information to the player making the suggestion. Therefore, contrary to conventional wisdom, we would expect bluffing even a single card to be detrimental to the player that employs this strategy.

C. Experimental Evidence

Our game-server allows us to make incremental changes in the strategies used by our computational agents and assess the cost or benefit of a change over many thousands of games to detect subtle effects. We can play six-player games that are a mix of bluffing and honest players to measure the effect of bluffing in actual game-play.

1) Computational Agent Heuristics

We have chosen to experiment using a top-performing computational agent developed by the authors.

This agent plays using the following general heuristics:

- If unknown, suggest the most likely suspect and weapon.
- If a card is known to be in the solution, suggest a card we have or the card in the solution if we have none.
- Move to a viable[9] room with the most viable rooms adjacent to it; room movement rules are fairly complex,

[9] A "viable" room is a card that is not known to be in any player's hand and may still be part of the solution.

but the general heuristic is to move to a viable room we don't know any player has, with the most viable adjacent rooms; if this is not possible then a variety of rules are used to find the least bad non-viable room to move to.
- Refute a suggestion with a suspect or weapon card, if possible, or a room if necessary.

The heuristic for refuting may seem quantitatively counter-intuitive as there are only 6 suspects but 9 rooms so that divulging a suspect card statistically gives more information to the opponent about the solution than a room card. However, this value is counter-balanced by the rules of the game that require a player to be in the room they are suggesting. By refuting with a suspect or weapon card, an opponent is forced to leave and later reenter the room in order to suggest it again — a diversion that can be costly in the short games that are common when our computational agents compete. The special nature of room cards will become readily apparent in our evaluation of the experimental results that follow.

2) Competing Bluff vs. Honest

All experiments are conducted by playing 20,000 games among the players. Players are randomly placed in a room to start and turn order is randomly shuffled for each game.

Table I presents the results of our baseline experiment that competes six honest players against one another.

Player	Winning %	
Honest 1	16.8	
Honest 2	16.3	
Honest 3	16.9	
Honest 4	16.6	
Honest 5	16.7	
Honest 6	16.7	
Mean of	$\overline{16.7}$	$\pm.19$

TABLE I
SIX "HONEST" PLAYERS

Analyzing the log of the games in Table I we also find that
- each game lasts only an average of 3.1 rounds (i.e., each player makes ~3 suggestions)
- the average number of players that are unable to refute the initial suggestion of a game is 2.0 — showing good agreement with our quantitative analysis in Section IV-B.[10]

Table II presents the somewhat surprising results from introducing one agent that has been modified to *always* bluff on their first turn by randomly choosing from their hand a single suspect, weapon, or room.

Analyzing the logs from the games in Table II show that
- each game lasts an average of 3.3 rounds; the increase coming from the longer games that begin with a bluffing suggestion
- an honest player's suggestion passes 2.0 opponents before it is refuted

Player	Winning %	
Bluffer 1	17.9	
Honest 2-6	$\overline{16.4}$	$\pm.16$

TABLE II
FIVE "HONEST" PLAYERS AND ONE BLUFFER

- the bluffing player's initial *bluffing* suggestion is refuted after passing 2.7 opponents — a value slightly less than 3 since our computed expected value of player $p_{2.17}$ refuting the suggestion from Section IV-B is close enough to 2 that the second player is occasionally still able to refute the suggestion. Players hold multiple cards and, despite the strategy of laying out a sequence of viable cards used in Section IV-B to derive the expected value, there is no notion of capturing how many of their cards a player "consults"; we can only observe the ordinal number of the player that refutes the suggestion

However, the improved performance of a bluffing player is counter to our hypothesis that, on average, bluffing provides opponents with *more* information that should improve the performance of the *honest* opponents!

In order to appreciate why the results in Table II defy our expectations, we return to an observation we made at the end of Section IV-C1 — that room cards are a different category than suspect and weapon cards. Specifically, when the player bluffs with a suspect or weapon card there is a good chance that they will be shown a room card. Because gathering information about rooms is made difficult by the restriction that one can must occupy the room one suggests, room knowledge is more difficult to acquire and therefor more valuable than suspect or weapon knowledge. In fact, as noted at the beginning of Section IV-C1, our agent avoids refuting with room cards (though they are unlikely to have a choice from the 3 cards held in a six-player game).

To refine our hypothesis, we re-ran our simulation twice: 1) with a player that bluffs a suspect or weapon card and 2), with a player that only bluffs a room card. Table III shows that

Player	Winning %	
Bluffer 1	17.8	
Honest 2-6	$\overline{16.4}$	$\pm.17$

TABLE III
FIVE "HONEST" PLAYERS AND ONE SUSPECT OR WEAPON BLUFFER

bluffing a suspect or weapon provides a strategic advantage.

Player	Winning %	
Bluffer 1	15.6	
Honest 2-6	$\overline{16.9}$	$\pm.26$

TABLE IV
FIVE "HONEST" PLAYERS AND ONE ROOM BLUFFER

Table IV demonstrates that our initial hypothesis — bluffing is detrimental to a player — is correct with one significant qualification: Knowledge gained about room cards is far more valuable than suspect and weapon cards and more than offsets

[10]Note that we confine ourselves to analyzing only the initial suggestion of each game as all subsequent suggestions are 1) no longer random since the agents have acquired some knowledge from previous turns and thus 2), the probabilities we assumed in Section IV-B no longer hold.

the cost of bluffing as demonstrated by Table III. When bluffing with a suspect or weapon card, the player is fairly likely to learn that the room is not in the solution. Conversely, when bluffing with a room card, the player learns *nothing* about rooms while providing opponents with information about what players do *not* have the room used in the suggestion.

Since room knowledge would seem to be so valuable, a strategy of bluffing with both the suspect *and* the weapon to insure a room card is shown would seem useful. However, Table V demonstrates that bluffing two cards appears to have no additional benefit. This lack of improvement is understandable as it now takes over 3 players to refute the suggestion, providing most opponents with additional information about who does not have the suspect, weapon, and room while the player making the suggestion learns *only* the location of a room card.

Player	Winning %	
Bluffer 1	17.7	
Honest 2-6	16.4	$\pm .28$

TABLE V
FIVE "HONEST" PLAYERS AND ONE SUSPECT *and* WEAPON BLUFFER

V. CONCLUSIONS

We began with an intuitive and analytical analysis of bluffing in Clue that suggested this commonly used strategy was detrimental to winning the game as it provides additional information to opponents that offsets the presumed benefit.

However, our game-play simulations demonstrated that bluffing appeared to be beneficial. This unexpected outcome lead to a more nuanced understanding that the cards used in bluffing determine its usefulness — that owing to the unique nature of room cards, bluffing to gain knowledge *about* rooms by bluffing with suspects and weapons is beneficial while bluffing *with* rooms is detrimental.

While we have examined the cost and benefit of bluffing on the first turn, a number of open questions remain; among them are determining

- whether bluffing on subsequent turns is useful, and
- if bluffing only one non-room card on the first turn is optimal, can players benefit by *assuming* that opponents are generally honest (i.e., they do not have the cards they suggest) and especially never bluff with a room card

This last question relates to the rationale for bluffing in Ferguson's two-player game "Guess It!" — namely, to prevent the exploitation of predictable "honesty." Thus the question of whether to bluff in Clue is even more nuanced and less obvious when playing with more sophisticated game-playing agents.

Although our students' computational agents largely share the basic game-play heuristics described in Section IV-C1, small differences in strategy, including when and what to bluff, result in surprisingly wide differences in performance that lead to a consistent and predictable ranking when these agents compete over many thousands of games. This suggests that the seemingly simple game of Clue is more nuanced than it appears and we continue to explore Clue game-play strategies as we refine our computational agent.

REFERENCES

[1] D. M. Hansen, J. Bruce, D. Harrison, and Z. Hayes, "Give students a clue," *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, March 2007.

[2] E. Neufeld, "Clue as a testbed for automated theorem proving," in *AI '02: Proceedings of the 15th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence*, R. Cohen and B. Spencer, Eds. Springer-Verlag, 2002, pp. 69–78.

[3] T. W. Neller, A. Markov, and I. Russell, "Clue deduction: Professor plum teaches logic," *American Association for Artificial Intelligence*, 2006.

[4] H. van Ditmarsch, "Knowledge games," *Bulletin of Economic Research*, vol. 53, no. 4, pp. 249–273, 2001.

[5] V. I. Aartun, "Reasoning about knowledge and action in cluedo using prolog," Master's thesis, University of Bergen, 2016.

[6] C. Cai and S. Ferrari, "On the development of an intelligent computer player for clue: a case study on preposterior decision analysis," in *Proceedings of the 2006 American Control Conference*, June 2006, pp. 4350–4355.

[7] Steam Community, *Clue/Cluedo: The Classic Mystery Game*, https://steamcommunity.com/app/794800/discussions/0/1743346190288401675/.

[8] Board & Card Games Stack Exchange, *When to make suggestions that include cards in my own hand in clue?*, https://boardgames.stackexchange.com/questions/7635/when-to-make-suggestions-that-include-cards-in-my-own-hand-in-clue.

[9] Board Game Geek, *User Review*, https://www.boardgamegeek.com/thread/19573/user-review. [Online]. Available: https://www.boardgamegeek.com/thread/19573/user-review

[10] L. van de Braak, L. Boulogne, and R. Mellema, *Perfect Epistemic Logicians playing Cluedo*, http://rmellema.github.io/Cluedo/.

[11] T. S. Ferguson, *Game Theory*. World Scientific, 2018. [Online]. Available: https://books.google.com/books?id=1yM4tAEACAAJ

[12] E. Pacuit and O. Roy, "Epistemic foundations of game theory," in *The Stanford Encyclopedia of Philosophy*, summer 2018 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.

[13] Yanick, *Clue Board Game Strategy*, http://game-board.blogspot.com/2008/04/clue-board-game-strategy.html.

[14] Anonymous, *The Clue player that takes the best notes is the one with the best Clue board game strategy, period!*, http://gatelcusd.weebly.com/uploads/2/6/0/2/26020511/cluestrategies.pdf. [Online]. Available: http://gatelcusd.weebly.com/uploads/2/6/0/2/26020511/cluestrategies.pdf

[15] T. W. Neller and Z. Luo, "Mixed logical and probabilistic reasoning in the game of clue," *ICGA Journal*, vol. 40, no. 4, pp. 406–416, 2018. [Online]. Available: https://doi.org/10.3233/ICG-180063

[16] H. van Ditmarsch, "The description of game actions in cluedo," University of Otago, Tech. Rep., 2001.

David M. Hansen received his Ph.D. in Computer Science and Engineering from the Oregon Graduate Institute of Science & Technology.

Dr. Hansen is Associate Professor of Computer Science and Information Systems at George Fox University. Recent research includes computer science pedagogy, intelligent agents, applied machine learning, and partition-tolerant distributed systems.

Kyle D. Hansen received his BS in Mathematics and BA in Computer Science from Westmont College.

Mr. Hansen is pursuing his Ph.D. in Mathematics at the University of California, Santa Barbara.