BLUFF WITH AI



SJSU SAN JOSÉ STATE UNIVERSITY

Advisor Dr. Christopher Pollett

Committee Members Dr. Philip Heller Dr. Robert Chun





Agenda

- Project Goal
- Problem Statement
- Related Work
- Game Rules and Terminology
- Game Flow
- Agents
- Sampling Plan
- Experiments
 - Solving Bluff with a Tit for Tat strategy
 - Conclusion and Future Work
- References



- Bluff is a multi-player card game in which each player tries to empty their hand first
- Build four different agents to play Bluff and find out how they perform over thousands of games
- Create two AI computer players having offensive strategy and two others with defensive strategy
- Evaluate performance on various scenarios based on experiments such as Self play and Evolutionarily stable strategy
 - Develop variants of the agents 'mutants' to see how they perform against better players



Problem Statement

Bluff is a game of:

Imperfect information

e.g.: players are unaware of opponent's hand and its hard to predict whether the opponent is bluffing or not

Partial observability

at any time, there is some information hidden from a player and certain information that is known only to the player (private information)

Stochastic outcomes

hand is dealt completely at random. Produces more uncertainty and a higher degree of variance in results

Non-cooperation

players will not cooperate among each other to target other players and win at the game



Related Work

- Bluff is a game of deception and is generally called 'Cheat' in Britain, 'I doubt it' in the USA and Bluff in Asia
- <u>Edmond Hoyle</u> who was a writer best known for his works on the rules and play of <u>card</u> <u>games</u> called the game "I doubt it"
- No established research literature
- Various online game site



when opponent has limited cards



Game Rules and Terminology

- **Deck**: A set of 52 playing cards
- Hand: The cards assigned to one player
- **Rank**: The type of card, e.g. Ace, Two, Three, etc.
- **Turn**: The time a player is allowed to play his cards
- Round: A set of turns by all the players
- **Trial**: An entire game (until a winner is found)
- Challenger: The player who calls "Bluff" on the opponent
- **Discard pile**: The set of face down cards in the middle, to
 - which each player adds the cards removed from his hand





Game Flow

- We implemented Bluff from scratch in Java
- Driver class is the main class from where the game begins
- CardManagement class shuffles the deck and assigns the hand of each player
- ComputerPlayers class is the super class of all AI players.



- play() method in each of the AI then handles the logic of the game depending on the strategy of player
- The callBluff() method then asks all the remaining players whether they want to challenge the current player

BluffVerifier class, the cards just played by the current player are verified against the rank of the card to be played



- The game of Bluff has two main elements:
 - _ Which cards to play in the current turn Offense
 - ____ When to call Bluff on your opponents **Defense**
- The 4 agents we use in our game are:
 - ____ No-Bluff AI (NBAI)
 - ____ Smart AI (SAI)
 - ____ Reinforcement Learning AI (RLAI)
 - ____ Insecure AI (IAI)
- No-brainer decision to call Bluff on an opponent if he plays more than four cards.
- Call Bluff when an opponent plays a card of the rank for which we have more than one in our hand
- Additional defense mechanism is to call Bluff on the opponent if he has less than three cards in hand

SJSU SAN JOSÉ STATE UNIVERSITY

No-Bluff AI (NBAI)

- Plays game truthfully
- Offensive player

game

- Do not call Bluff on opponent
- Play the first card in hand if he does not have the card to play
- Useful to understand the
 - importance of bluffing in the





Smart AI (SAI)





Reinforcement Learning AI (RLAI)

- Offensive player
- Uses Reinforcement learning
- Agent learns which action to take based on reward mechanism
- Agent not told which action to take but must discover which action yields most reward by trying
- 2 stages: Training and Testing
 Result of training is updated to State-Action Matrix and Reward Matrix





Reinforcement Learning AI (RLAI)

For each training cycle:

Assign state as the current rank to be played.

Select one among all possible actions for the current state.

Using this possible action, observe the result.

Update State-Action matrix and Reward matrix.

End For

For each testing cycle:

Assign state as the current rank to be played. Select the most rewarded action for the current state from the State-Action matrix Using this possible action, observe the result. Update Reward matrix.

End For



Insecure AI (IAI)

- Uses card counting to keep track of the cards in each players hand
- Calls Bluff on player with < 3 cards
- Towards the end, it is very rare that players have the actual card to play
- Forces them to cheat since there is no option to pass a turn
- IAI thus delays opponent's win





Sampling Plan

- Bluff is categorical win/lose
- Each game has an independent outcome
- Confidence interval denotes the number of samples required to compute a result with a certain confidence level such as 95% or 99%.
- To determine the least sample size (run size) for our experiment to result in 99% confidence and 99% reliability level, we use the following formula

Run size (n) = $\frac{\text{Ln}(1 - \text{confidence interval})}{\text{Ln}(\text{Reliablity})} = \frac{\ln(1 - 0.99)}{\ln(0.99)} = 299 \text{ Runs}$

• We run all our experiments for 300 trials.



Experiments and Observations

- Experiment 1: Self Play
 - To find which position would have advantage over other
- Experiment 2: NBAI vs. SAI
- Experiment 3: IAI vs. RLAI
 - To find which is the better strategy of the two
- Experiment 4: NBAI vs. SAI vs. RLAI vs. IAI
 - To find which is the better strategy of all
- Expt. 5: True Bluff calls vs. False Bluff calls
 - To find if the Insecure AI's strategy is a good one or not
- Expt. 6: Evolutionary Game Theory

- To find which agent is in evolutionarily stable state



Experiment 1: Self Play

Hypothesis:

No position would have advantage over other positions during self play.

Result:

Almost half the time, player in position 1 won the games even though the deck was shuffled and cards were assigned randomly without any bias.

Bias towards the player in position 1, since he leads the round

Conclusion:

For all the AIs, we note that player in position one has an advantage over others and so our hypothesis is wrong.

Win rate of Experiment 1 (300 trials/player)

	NBAI	SAI	RLAI	IAI
Position 1	44	36	47	34
Position 2	26	23	17	14
Position 3	16	19	16	24
Position 4	14	22	20	28

No. of wins in % for the 4 AIs in Self-play (300 trials/player)



** In each of the runs, the results were fairly consistent with a confidence interval of 99% and a reliability of 99%.



Experiment 2: NBAI vs. SAI

Hypothesis:

Smart AI would beat No-Bluff AI.

Result:

In a four player game with players 1 and 3 as the No-Bluff AI and players 2 and 4 as the Smart AI, Player1, the No-Bluff AI had the most number of wins

Conclusion:

This experiment shows that when No-Bluff AI is in position 1 he has an advantage over Smart AI, and won the game.

But when No-Bluff AI is not in first position, Smart AI could beat him.

No. of wins (%) for NBAI vs. SAI (300 trials)



Players

** In each of the runs, the results were fairly consistent with a confidence interval of 99% and a reliability of 99%.



Experiment 3: IAI vs. RLAI

Hypothesis:

Reinforcement Learning AI would beat the Insecure AI.

Result:

In a four player game with players 1 and 3 as the Insecure AI and players 2 and 4 as the Reinforcement Learning AI, Player1, the IAI had the most number of wins.

Conclusion:

This experiment also proves that the Player 1 has an advantage over other player, which can be proved by the RLAI winning over IAI when it was not in position 1.

No. of wins (%) for IAI vs. RLAI (300 trials)



** Confidence interval of 99% and a

reliability of 99%.

SJSU SAN JOSÉ STATE Experiment 4: NBAI vs. SAI vs. RLAI vs. IAI

Null Hypothesis (H_o): Learning AI would have the highest number of wins since Learning AI has the knowledge of previous outcomes, which other players lack.

Alternate Hypothesis (H₁): Learning AI would have equal or lower win rates when compared to other players.

Experimental setup: All possible combinations of the four AI players were tested for 300 trials, totaling 7200 games.

Result:

- The No-Bluff AI was the best performer followed closely by Smart AI
- The Smart AI has very good performance rate and is closely followed by the Learning AI
- The Learning AI could not beat other players as we expected it to
- The Insecure AI was the lowest performer

Conclusion:

Alternate Hypothesis (H₁) is true. Null hypothesis can be rejected. No-Bluff AI has the most wins of all players.

Total wins for each player	2591	2330	2247	32
Win%	36%	32%	31%	1%

** Confidence interval of 99% and a reliability of 99%.

SJSU SAN JOSÉ STATE Expt. 5: True Bluff calls vs. False Bluff calls

Null Hypothesis (H_o): Insecure AI would have the most number of False Bluff calls as Insecure AI tends to call Bluff every time if its sees an opponent with less than 3 cards in hand.

Alternate Hypothesis (H₁): Insecure AI would have the highest success rate in catching Bluff, since most players would not have the correct card to play towards the end.

Result:

- IAI has the winning strategy
- RLAI is better at catching Bluff than SAI
 Conclusion:

The Null Hypothesis (H_o) was rejected and the Alternate ^[] Hypothesis (H₁) was accepted as IAI had the best success rate at calling Bluff. True Bluff vs. False Bluff

	Number of Correct Bluff calls in					
	1200 Games					
	NBAI	NBAI SAI RLAI		IAI		
Total	0	2114	2988	9508		
True						
Bluff %	0.0%	62.8%	69.5%	75.0%		
	Num	Number of False Bluff calls in				
	1200 Games					
G	NBAI	SAI	RLAI	IAI		
Total	0	1251	1310	3163		
False	SUG IS	SU S	SU SU SU SU SU SU			
		07 004	00 50/	25.00/		

SJSU SAN JOSÉ STATE Evolutionary Game Theory

- EGT is the application of game theory to evolving populations in biology.
- It defines a framework of contests, strategies, and analytics into which Darwin's evolution can be modeled.
- Strategy success is determined by how well one strategy is, in presence of a competing strategy.
- The players aim to replicate themselves by culling the weakest player and thus defeating the competing strategy.
- Replicator dynamics model: strategy which does better than its opponents and replicates at the expense of strategies that do worse than the average.



Replicator Equation is defined as: $\dot{x}_i = x_i [f_i(x) - \phi(x)]$

Where, $\phi(x) = \sum_{j=1}^{n} x_j f_j(x)$

 x_i – Proportion of type i in the population

 $f_i(x)$ – is the fitness of type i

 $\phi(x)$ – is the average population fitness

SJSU SAN JOSÉ STATE Expt. 6a: Finding dominant strategy

Evolutionarily Stable Strategy (ESS): A given strategy is called an evolutionarily stable strategy if a population adopting this strategy cannot be defeated by a small group of invaders using a different strategy which was initially weak.

Aim:

To find the Evolutionarily Stable Strategy among the four agents.

Experiment:

- We run the four agents for one Evolution (300 trials) and observe the fitness of a player.
- Fitness was evaluated as a measure of the number of wins against other opponents.
- We repeated this experiment over several evolutions and results were observed.
- For each evolution, we calculated the fitness of each player using the replicator equation and eliminated the player with the weakest strategy (least fit) and replicated the agent with the strongest value to take its position.

SJSU SAN JOSÉ STATE Expt. 6a: Finding dominant strategy

Calculations:

 $f_j(x)$ – is the fitness of type j and is calculated as the number of wins.

- x_i Proportion of type i in the population
- In the first Evolution, the total wins of each players are shown in table $\varphi(x) = \sum_{j=1}^{n} x_j f_j(x) = \text{Sum (Proportion of j * Fitness of j)}$ = (0.25*82) + (0.25*79) + (0.25*95) + (0.25*44) = 18.75
- The Replicator Equation for No-Bluff AI is calculated as follows:

 $f_i(x) - \phi(x)$ = Total wins – Average population fitness

= 82 - 18.75 = 63.25

 $x_i[f_i(x) - \phi(x)] = 0.25 * 63.25$ = 15.8125

*				
Players	NBAI	SAI	RLAI	AAI
Total wins in 300 games	82	79	95	44
$f_i(x) - \phi(x)$	63.25	60.25	76.25	25.25
$x_i[f_i(x) - \phi(x)]$	15.8125	15.0625	19.0625	6.3125

SJSU SAN JOSÉ STATE Expt. 6a: Finding dominant strategy

Observation:

- In the very 1st Evolutionary run, IAI was eliminated with an offspring of RLAI.
- In the 2nd evolutionary run, RLAI was culled by SAI.
- By the 5th evolutionary run, the whole population is using the SAI strategy and has reached the stable state.



Population Growth of Player Types Using Evolutionary Game Theory

Ruses Ruse Ruses Ruse Ruses Ruses

SJSU MAN JOSÉ STATE Expt. 6a: Finding dominant strategy

Conclusion:

- SAI has overcome all other competing strategies and successfully multiplied its own strategy into the entire population.SAI may possibly be the ESS given that it has successfully established its population.
- To verify ESS a subsequent experiment (Experiment 7b) has to be conducted with a small group of invaders.

SJSU SAN JOSÉ STATE Expt. 6b: Test for finding the ES Strategy

Evolutionarily Stable Strategy (ESS): A given strategy is called an evolutionarily stable strategy if a population adopting this strategy cannot be defeated by a small group of invaders using a different strategy which was initially weak.

Aim:

To test the stability of Evolutionarily Stable Strategy with invaders.

Experiment:

In this experiment, we ran six agents (four SAI and two mutated IAI) for one Evolution (set of 300 games) and observed the fitness of a player against other players' fitness.

We repeated this experiment over several evolutions and results were observed.

SJSU SAN JOSÉ STATE Expt. 6b: Test for finding the ES Strategy

Observation:

- Insecure AI was modified to call bluff on opponents with less than 2 cards and then introduced as the fifth and sixth players (mutants) to invade the SAI ESS state.
- Over three generations, the Mutant-Insecure AI population was eliminated by the SAI strategy.
- Therefore the SAI strategy is the Evolutionarily Stable Strategy and this state is called Evolutionarily Stable State.



Test For ESS Stablity - Introducing Mutant Players in a Six Player Game.

SJSU SAN JOSÉ STATE Expt. 6: Evolutionary Game Theory

Conclusion:

- A small invading population using a strategy T would have lesser fitness than the evolutionarily stable strategy S and would be overcome by majority population, provided the disturbance by invading strategy T is not too large.
- More formally, we will phrase the basic definitions as follows:
 - The fitness of a player is based on the expected payoffs from the interactions with other players.
 - Strategy T invades a strategy S at level x, where x is a small positive number and denotes the population that uses T and (1 – x) denotes the population using S
 - Finally, strategy S is said to be evolutionarily stable if a strategy T invades S at any level x < y, where y is a positive number, and the fitness of strategy S is strictly greater than the fitness of a strategy T.

SJSU SAN JOSE STATE Solving Bluff with Tit for Tat

- Nash equilibrium is a set of strategies, where each player's strategy is optimal and no player has incentive to change his or her strategy given what other players are doing.
- Bluff is bounded by finite number of players with finite strategy space and therefore there exists at least one Nash Equilibrium.
- (2, 2) The state is Nash equilibrium because no player has incentive to change his or her strategy given what the other players are doing.
- (-3,3) If player X bluffs and gets caught the penalty is maximum. Player Y has most payoffs if player X is caught bluffing.
- (2,-2) & (2, 2) Player X has identical payoff for being honest. On the other hand Player Y has one strategy with Penalty of 2 and another with reward of 2.



SJSU SAN JOSÉ STATE Tit for Tat strategy against AI players

- Tit for Tat vs. No Bluff AI:
 - Tit for Tat player will always cooperate with No Bluff AI and exhibit similar behavior of No Bluff AI.
- Tit for Tat vs. Smart AI:
 - Tit for Tat player will cooperate most of the time, until Smart AI defects when it estimates a bluff.
- Tit for Tat vs. Reinforcement Learning AI:
 - similar outcome is expected as of Tit for Tat player against Smart AI.
- Tit for Tat vs. Insecure AI:
 - Tit for Tat player will cooperate in the beginning until Insecure AI defects. However when Insecure AI
 detects less than 3 cards with Tit for Tat player it defects all the time, which might create a chain of bluff
 calls between Tit for Tat and Insecure AI.
- Tit for Tat vs. Tit for Tat
 - When matched against itself, the tit for tat strategy always cooperates and takes On-equilibrium path.



Conclusion and Future Work

- In this project, we created four different Als with different tactics.
- Multiple experiments were conducted and results observed.
 - Position 1 yielded advantage.
 - Smart AI was ESS.
 - No-Bluff AI was the second best strategy
 - Learning AI could not beat Smart AI
- In future, it would be interesting if an AI could use different strategies in different levels of the game (Adaptive Strategy).
- An agent which employs the DQN algorithm and use 2 different neural networks to make two different decisions:
 - which card to play and
 when to bluff.



- [1] D. Billings, "Algorithms and assessment in computer Poker," University of Alberta.
 2007 Available: <u>http://poker.cs.ualberta.ca/publications/billings.phd.pdf</u>
- [2] E. Hurwitz and T. Marwala, "Learning to bluff," 2007 IEEE International Conference on Systems, Man and Cybernetics, Montreal, Que., 2007, pp. 1188-1193. Available: <u>http://ieeexplore.ieee.org/document/4413589/</u>
- [3] S. Russell and P. Norvig, "Adversarial search," in *Artificial Intelligence: A Modern Approach*, 3rd ed. New Jersey: Pearson, 2010, Ch. 5, pp. 161-189.
- [4] E. Hurwitz and T. Marwala, "A Multi-agent approach to Bluffing," 2009 Salman Ahmed and Mohd Noh Karsiti (Ed.), InTech, DOI: 10.5772/6603. Available: <u>https://www.intechopen.com/books/multiagent_systems/a_multi-agent_approach_to_bluffing</u>
- [5] J.Colton, "How many samples do you need to be confident your product is good," 2017 Available: <u>http://blog.minitab.com/blog/the-statistical-mentor/how-many-samples-do-you-need-to-be-confident-your-product-is-good</u>

SJSU SAN JOSÉ STATE UNIVERSITY

References

- [6] J. Frost, "Regression Analysis: How do I interpret R-squared and assess the goodness of a fit," 2013. Available: <u>http://blog.minitab.com/blog/adventures-in-statistics-</u> <u>2/regression-analysis-how-do-i-interpret-r-squared-and-assess-the-goodness-of-fit</u>
- [7] Easley David and Kleinberg Jon. 2010. "Networks, Crowds, and Markets: Reasoning about a Highly Connected World," *Cambridge University Press*, New York, NY, USA. [Online]. Available: <u>https://www.cs.cornell.edu/home/kleinber/networksbook/networks-book-ch07.pdf</u>
- [8] E.V. Belmega, S. Lasaulce, H. Tembine, M. Debbah. "Game Theory and Learning for Wireless Networks: Fundamentals and Applications", Academic Press, Elsevier, pp.122-124, 2011. [Online]. Available: <u>https://www.elsevier.com/books/gametheory-and-learning-for-wireless-networks/lasaulce/978-0-12-384698-3</u>
- [9] V. Mnih , K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. <u>Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. <u>abs/1312.5602, 2013. [Online]. Available:</u></u>
- [10] <u>http://arxiv.org/abs/1312.5602</u>T. Matiisen, "Demystifying deep reinforcement learning, "University of Tartu, Estonia, 2015. [Online]. Available: <u>https://www.intelnervana.com/demystifying-deep-reinforcement-learning/</u>

SAN JOSÉ STATE UNIVERSITY *powering* silicon valley





Appendix

Susta Susta