**Intro to AI Final Project: Pirate Ship Battle**

Javier Cabanellas

Department of Computer Sc. and Engg.

University of South Florida, Tampa, Florida, USA

***Abstract—This report examines the implementation and evaluation of various artificial intelligence (AI) strategies within "Pirate Ship Battle," a digital adaptation of the classic game Battleship. Four distinct AIs—Random AI, Greedy AI, Monte Carlo AI, and Hybrid AI—were developed and tested across multiple simulations to assess their strategic effectiveness and efficiency. The experimental results revealed significant differences in performance, with the Hybrid AI demonstrating superior adaptability and effectiveness. Additionally, an unexpected discovery regarding the first-move advantage in symmetric AI matches provided insights into the impact of initial game conditions on final outcomes. This study not only highlights the capabilities and challenges of developing AI for strategic games but also suggests directions for future research and development.***

**1. Introduction**

"Pirate Ship Battle" serves as a platform to explore the application of artificial intelligence in recreating and enhancing the traditional Battleship game experience. This project aimed to develop and compare different AI strategies to identify which method best adapts to and competes in a simulated maritime combat environment. The AI strategies ranged from simple random decision-making processes to complex probabilistic and hybrid approaches, reflecting varying levels of strategic depth. The primary focus was to evaluate each AI's ability to efficiently sink opponent ships while adapting to dynamic game states, thereby enhancing the gameplay experience. This report details the design, implementation, and testing phases of the AI strategies, culminating in a comprehensive performance analysis.

**2. Description of System**

In the development of "Pirate Ship Battle," a computerized version of the classic game Battleship, four distinct artificial intelligence (AI) strategies were implemented to provide dynamic and challenging gameplay. Each AI employs unique algorithms to enhance the game's complexity and engagement. Here's a detailed overview of each AI strategy:

**2.1 Random AI**

The Random AI serves as a fundamental strategy where decisions are made entirely by chance, without any regard to the game state beyond the legality of moves. It selects from any available unknown squares randomly, ensuring unpredictability but often lacking efficiency. This AI is particularly useful as a fallback method when other, more sophisticated strategies do not have sufficient data to proceed.

Implementation Details:

* Searches through all cells marked as "UNKNOWN" on the grid.
* Randomly selects one of these cells to make a move.

**2.2 Greedy AI**

Greedy AI adopts a more strategic approach by capitalizing on successful hits. Once it registers a hit, it prioritizes adjacent cells in an attempt to quickly sink the discovered part of the ship. This method is termed 'greedy' because it focuses on immediate rewards (sinking ships) rather than the broader state of the game.

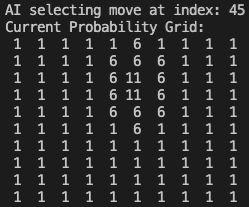
Implementation Details:

* Identifies cells with hits that are not part of completely sunk ships.
* Searches for adjacent unknown cells to target next, aiming to follow the line of the ship.
* If no adjacent targets are found, it defaults to the Random AI to avoid stalling.

**2.3 Monte Carlo AI**

Monte Carlo AI uses probabilistic simulation to enhance decision-making. It updates a probability grid based on the current state of the game, increasing the likelihood of targeting areas around known hits. This AI is effective in managing uncertainty and adapting its strategy as the game progresses. Figure 1 shows an example of the probability grid when a ship is hit.

**Figure 1: Monte Carlo Probability Grid**



Implementation Details:

* Maintains and updates a probability grid where each cell's value indicates the likelihood of an opponent's ship being present.
* After each move, probabilities are adjusted, particularly increasing those around recent hits to simulate potential ship positions.
* The AI selects the move with the highest probability from the updated grid.

**2.4 Hybrid AI**

The Hybrid AI incorporates multiple strategies to form a robust opponent. It combines the direct targeting of the Greedy AI with pattern recognition and probabilistic reasoning, making it adaptable to various game situations.

Implementation Details:

* Uses hits to guide searches for additional ship parts, expanding both directly adjacent and nearby cells.
* Implements a checkerboard pattern strategy to efficiently cover the grid when fewer cues are available.
* Integrates random selection to maintain unpredictability when other strategies yield no clear advantage.

Each AI strategy in "Pirate Ship Battle" offers a different level of complexity and strategic depth, making the game engaging for players of all skill levels. The Random AI provides basic unpredictability, the Greedy AI offers a focused approach to sinking ships, the Monte Carlo AI brings advanced probabilistic calculations into play, and the Hybrid AI combines these strategies to adapt dynamically to the unfolding game. This diverse array of AI opponents not only enhances the gameplay experience but also simulates various levels of human-like strategic thinking.

**3. Experimental Results**

To rigorously evaluate the performance of various artificial intelligence strategies implemented in "Pirate Ship Battle," a series of simulations was conducted, each consisting of 1500 games. These experiments provided critical insights into the effectiveness of each AI approach, highlighting their strengths and limitations in the context of strategic gameplay. The outcomes of these simulations are depicted in accompanying charts that illustrate the distribution of shots per game, serving as a quantitative measure of each AI's efficiency.

**3.1 Simulation: Random AI vs. Random AI**

The first simulation pitted two Random AI opponents against each other. Results showed Player 1 winning 816 times and Player 2 winning 684 times, with the number of shots frequently reaching the upper limit of 200 per game. This outcome, as shown in Figure 2, underscores the inherent inefficiency and lack of strategic depth when both players make decisions purely based on chance, leading to a high variability in game length and outcome.

**Figure 2: Random AI vs. Random AI Distribution of Shots per Game**

A graph of a number of shots

Description automatically generated

**3.2 Simulation: Greedy AI vs. Random AI**

In the second simulation, the Greedy AI competed against the Random AI. The Greedy AI demonstrated a significant advantage, securing 1486 wins compared to only 14 by the Random AI. This marked improvement in performance was due to the Greedy AI’s ability to pursue a more targeted approach upon making a hit, drastically reducing the average number of shots per game. The chart from this simulation illustrates a noticeable shift toward fewer shots, highlighting the strategic efficiency of the Greedy AI over the random approach.

**Figure 3: Greedy AI vs. Random AI Distribution of Shots per Game**

A graph of a number of shots

Description automatically generated

**3.3 Simulation: Monte Carlo AI vs. Greedy AI**

The third simulation featured the Monte Carlo AI against the Greedy AI, where the former won 825 games against the latter’s 675. This simulation provided a close competition, showcasing the Monte Carlo AI's slightly better utilization of probabilistic data and strategic planning, which led to a more efficient game resolution. The results here indicate that Monte Carlo AI can outmaneuver the Greedy AI by adapting more dynamically to the unfolding game state.

**Figure 4: Monte Carlo AI vs. Greedy AI Distribution of Shots per Game**

A graph of a number of shots

Description automatically generated

**3.4 Simulation: Hybrid AI vs. Monte Carlo AI**

The final simulation tested the Hybrid AI against the Monte Carlo AI, with the Hybrid AI emerging significantly ahead with 1084 wins to 420. This simulation highlighted the superior adaptability of the Hybrid AI, which combines various strategic elements from the other AIs, allowing it to outperform the Monte Carlo AI across a broad range of scenarios. The corresponding chart shows a progression towards more efficient gameplay, with the number of shots per game approaching, but not reaching, the minimum theoretical number of shots needed to win.

**Figure 5: Hybrid AI vs. Monte Carlo AI Distribution of Shots per Game**

A graph of a number of shots

Description automatically generated

**3.5 Summary**

These experimental simulations clearly delineate the ascending scale of sophistication and effectiveness from Random AI to Hybrid AI. The visual data from the charts corroborates that more strategic AIs, particularly the Hybrid AI, significantly enhance gameplay by reducing the number of shots needed to conclude a game, thereby increasing efficiency. This comprehensive testing not only validates the effectiveness of advanced AI strategies in "Pirate Ship Battle" but also underscores the potential of combining different AI tactics to achieve superior performance in strategy-based games.

**4. Surprising Discoveries**

During the extensive testing of the AI strategies in "Pirate Ship Battle," an unexpected yet enlightening observation was made regarding the influence of the first-move advantage in games featuring symmetric AI opponents.

In a particular simulation where both Player 1 and Player 2 used the Hybrid AI—the most sophisticated of the implemented strategies—the results were intriguing. Despite both players employing the same advanced algorithm, Player 1, who had the advantage of the first move, won more games, securing 781 victories compared to Player 2's 719. The chart from this simulation demonstrated a skew towards fewer shots, indicating both AIs' effectiveness in quickly identifying and sinking ships. However, it was notable that neither AI consistently minimized the game to the theoretical minimum number of shots, which would be 17 for a perfectly efficient game.

**Figure 6: Monte Carlo AI vs. Greedy AI Distribution of Shots per Game**

**A graph of a number of shots

Description automatically generated**

The concept of a first-move advantage is well-documented in various strategic games, but its impact in a setting where both competitors use identical AI was particularly pronounced. This observation suggests that even small initial advantages can be amplified in games involving complex decision-making and strategic depth. The Hybrid AI, capable of dynamically adjusting its strategy based on the game state, exploited the first move to establish a slightly more advantageous position from the outset. This advantage likely compounded over the course of the game, allowing Player 1 to secure more wins overall.

**5. Conclusion**

The development and comparative analysis of AI strategies in "Pirate Ship Battle" provided profound insights into the mechanics of strategic game AI. The Hybrid AI, with its integration of multiple decision-making techniques, outperformed other strategies in terms of adaptability and overall efficiency. The experimental simulations underscored the significance of the first-move advantage, revealing inherent biases that could influence game outcomes even among equally matched AIs. This finding suggests potential areas for further refinement, including the balancing of game mechanics and the optimization of AI responses to disadvantageous situations. Future work will focus on addressing these challenges, enhancing the AI's strategic capabilities, and exploring new AI methodologies to ensure fair and engaging gameplay. This research not only advances the field of game AI but also contributes to the broader understanding of AI applications in entertainment and simulation technologies.