2048 AI Agent Development Report

1. Overview

This report details the development and strategy of a 2048 AI agent implemented in MyAgent.py. The agent is designed to play the 2048 game using a depth-limited Expectimax search combined with a comprehensive heuristic evaluation function. The goal is to outperform baseline agents and compete effectively on the class leaderboard.

2. Strategy

The agent utilizes **Expectimax search** to plan ahead, alternating between maximizing moves (player turns) and averaging over probabilistic tile spawns (computer turns). A **time-aware iterative deepening mechanism** increases the depth of search without exceeding the allotted move time (0.5 seconds per move on the server).

3. Heuristics

The agent's evaluation function combines multiple features to assess the quality of a game state. Each feature is weighted to reflect its importance:

- **Empty Tiles**: Encourages keeping the board open to avoid being trapped.
- **Smoothness**: Penalizes large differences between adjacent tiles, encouraging tile merges.
- **Monotonicity**: Rewards rows or columns that increase or decrease consistently.
- **Max Tile**: Gives a bonus for having large tiles on the board.
- **Corner Bonus**: Rewards positioning the max tile in the top-left corner.
- **Merge Potential**: Encourages move options that allow merges.
- **Island Penalty**: Penalizes isolated tiles which reduce merge opportunities.

4. Move Ordering

Move ordering prioritizes actions that are likely to result in beneficial outcomes. Moves that:

- increase the number of empty tiles,
- perform merges, or
- maintain the max tile in the corner

are ranked higher to speed up search and improve pruning efficiency.

5. Fallback Logic

In cases where the main search fails due to time limits or move pruning, the agent falls back to safer options, including:

- choosing the best merging move, or
- preserving the max tile in the corner.

6. Performance and Results

The agent performs **consistently above baseline agents** and incorporates several advanced evaluation techniques. The **iterative deepening Expectimax** and **tuned heuristics** enable strategic planning while maintaining fast execution within server constraints (PyPy, 0.5s/move).

Previous Performance (MinMax Baseline Agent)

- **Average Score**: ~4,000
- **Max Score**: ~10,000

Improved Performance

- **Average Score**: **15,777.52**
- **Maximum Score**: **50,236**
- **Minimum Score**: **15,054**
- **Games Played**: 50

This improvement reflects:

- Nearly **2x increase in average score**
- More consistent high-scoring games
- Better long-term planning and tile management

7. Conclusion

This AI agent combines solid search techniques with practical heuristics and fallback strategies to achieve strong performance in the 2048 game. Further improvements may involve:

- incorporating tile gradients,
- caching evaluations for deeper searches, and
- dynamic tuning of heuristic weights based on game progression.