2048
Austin Carnahan

1) MinMaxABMO: Add AB Pruning and Move Ordering

Hypothesis: Adding AB Pruning to the minmax agent should improve its ability to search deeper into the game tree and score higher.

Experiment: Play 10 games. Get a solid baseline for performance.

Result:

Games played: 10
Average score: 13029.2
Top score: 29892
Average moves per game: 687.2
Total evaluation time: 688.0s (11.5min)

Average depth: 7.55
Branching factor: 3.37

Max tile distribution:
  512: 3 games (30.0%)
  1024: 6 games (60.0%)
  2048: 1 games (10.0%)

Hypothesis: Adding move ordering should allow the agent to evaluate the best possible branches first before time runs out.

Experiment: Run simulations with a variety of move ordering strategies.

Strategy	Avg Score	Max Score	Avg Moves	1024 Tile %	512 Tile %	Other Max Tiles
Corner + Score Diff	11,845.2	16,856	650.7	60.0%	40.0%	—
Empty Tiles + Score Diff	12,013.2	16,232	644.8	80.0%	20.0%	—
Empty Tiles + Score	11,377.2	15,872	626.5	60.0%	40.0%	—
ULDR Priority	10,415.7	14,192	579.1	58.3%	33.3%	128 (8.3%)
Just Score	9,951.7	15,856	566.2	41.7%	50.0%	256 (8.3%)
Just Empty Tiles	11,328.4	15,764	625.8	60.0%	40.0%	—
No Move Ordering	10,980.8	15,932	607.3	60.0%	30.0%	256 (10.0%)

Results: There isn't a huge difference if the performance of most of these strategies across ~10 games. But sorting by the empty tiles and score difference seems to do the best! This is underperforming the first test with no move ordering – but I think it was a fluke. This result for No move ordering in this experiment shows it consistently underperforms.

2) Improved Heuristic

Hypothesis: Improving our heuristic method beyond just score should allow the agent to play better.

Experiment: In order to get a better sense of different heuristics, run individual heuristics and see how they perform before combining them into a linear evaluation function. Just score is already done from the last experiment. I'll be running 10 games of each: 1. Empty tiles, 2. Simple corner anchoring, 3. More complex "monotonicity" measurement

Results: As expected score performs best here. Empty tiles outperforms everything else and the simple “corner anchoring” metric outperforms the more complex “monotonicity” measurement.

Strategy	Avg Score	Max Score	Avg Moves	1024 Tile %	512 Tile %	Other Max Tiles
Empty Tiles	6,800.4	11,808	397.0	30.0%	50.0%	256 (10.0%), 128 (10.0%)
Simple Corner Anchoring	5,783.6	10,216	347.8	40.0%	10.0%	128 (30.0%), 64 (20.0%)
Score	12,013.2	16,232	644.8	80.0%	20.0%	—
Monotonicity	4,667.6	7,220	309.9	0.0%	50.0%	256 (40.0%), 128 (10.0%)

Hypothesis: I think I can boost the scores of these corner anchoring, monotonicity, and empty tiles heuristics with a little tweaking. **Corner anchoring**: Look at a full row and column, plus 1 diagonal, add directional preference (top left). **Monotonicity**: Reduce penalty and add directional preference (top left). **Empty Tiles:** Weight the empty tiles on the edges and corners as more valuable

Experiment: 10 more games each

Results: Monotonicity V2 and the weighted empty tiles performed significantly worse as standalone heuristics. The Corner Anchoring V2 shows a huge boost!

Strategy	Avg Score	Max Score	Avg Moves	1024 Tile %	512 Tile %	Other Max Tiles
Corner Anchoring V2	8,954.0	18,840	505.3	50.0%	20.0%	256 (20.0%), 128 (10.0%)
Monotonicity V2	5,098.8	8,836	339.6	0.0%	50.0%	256 (50.0%)
Weighted Empty Tiles	4,566.4	6,796	300.3	0.0%	60.0%	256 (40.0%)

Hypothesis: The best performing standalone heuristics were: score, empty tiles, and the corner anchoring V2. Combining these into a linear evaluation function should allow the agent to consider all of these factors and perform better. Also going to test the introduction of a “board fullness” measure to boost the importance of empty tiles when running out of space. Also going to test a version that includes “merge-ability” as a feature.

Experiment: 10 games for each of the three setups.

Results: Still really underperforming what I had hoped for. But being conscious of how full the board is and boosting the importance of empty moves seems to pay off:

Strategy	Avg Score	Max Score	Avg Moves	1024 Tile %	512 Tile %	Other Max Tiles
Balanced Approach	11,276.0	15,756	625.7	60.0%	40.0%	—
Merge Factor	7,122.0	13,688	422.2	10.0%	90.0%	—
Dynamic Empty Tiles	13,352.8	22,928	708.8	70.0%	10.0%	2048 (10.0%), 256 (10.0%)

Hypothesis: Creating a more refined linear function that has weights/coefficients for normalized parameters will make tweaking the strategy easier. It also makes it trainable for future experiments.

Experiment: 10 Games each for 3 different strategies: Corner heavy, Score Focused, and Emptiness Dominant

Results: Despite trying several setups for the normalized parameters – Its not working well. I think the normalized values just aren’t providing the right signal and I cant seem to get the scaling right. I’m going to go back to the non-normalized approach.

Average score: 7210.8
Top score: 16436
Average moves per game: 438.6
Total evaluation time: 427.9s (7.1min)

3) Improved Heuristic V2

Hypothesis: Keeping the evaluation heuristic based in game points is the most straightforward and interpretable approach. But I still want a standardized way to do it so that it makes sense and can be easily refined. This version awards “bonus points” on top of game score in its evaluation. The bonus points are in terms of percentage of game score so they scale with different stages of the game (eg. 5% bonus for corner aligned state)

Experiment: Run 4 configurations of the heuristic for 10 games each. Balanced, Aggressive, Survivalist, and Corners

Results: Alright! Looking better. Our balanced agent performed best followed by the survivalist. Both doing as good or better as our last best agent.

Agent	Avg. Score	Top Score	Avg. Moves	2048 Tile Rate	1024 Tile Rate	Max Tile Distribution
Balanced	14,611.6	25,104	763.2	10.0%	80.0%	256×1, 1024×8, 2048×1
Survivalist	13,579.2	22,800	704.3	10.0%	80.0%	1×1, 1024×8, 2048×1
Aggressive	11,276.8	15,732	618.2	0.0%	70.0%	256×1, 512×2, 1024×7
Corner-Heavy	10,188.8	15,592	559.8	0.0%	60.0%	256×2, 512×2, 1024×6

4) Better Corner and edge strategy

Hypothesis: After watching the agent play – it was clear that it’s not really following a good technique of keeping the high tile anchored in the corner and having the tiles decrease along the edges from there. I re-did the corner evaluation entirely, and also added a weighted tile gradient to help move pieces up and to the left.

```
tile_weights = [
            4.0, 3.0, 2.0, 1.0,
            3.0, 1.0, 1.5, 1.0,
            2.0, 1.5, 1.0, 0.5,
            1.0, 0.5, 0.25, 0.1
        ]
```

Testing: I ran tons of different simulations until I could consistently see the agent following the up-and-to-the-left strategy. It works! But scores remain low. So I tested a bunch of different weight configurations.

Results:

Agent Variant	Avg Score	Top Score	Avg Moves	1024+ Rate	2048s	Notes
A2 Structural	10,120.8	16,068	577.3	40%	0%	Anchoring strong, limited scaling
A3 Aggressive	9,326.0	22,748	528.9	30%	10%	Volatile: 1 high win, weak avg
A4 Survivalist	12,197.6	15,920	664.9	70%	0%	Most consistent structure
A5 Minimalist	11,448.8	30,060	619.3	40%	10%	Best top score, higher variance

Agent Variant	Avg Score	Top Score	Avg Moves	1024+ Rate	2048s	Notes
A8 Chain-Hard	11,088.0	16,140	609.1	60%	0%	Solid baseline, decent structure
A6 Gradient-Driven	9,983.6	15,644	569.2	50%	0%	Early game ok, but lacks late game pull
A7 Score-Weighted	11,291.6	17,156	629.3	60%	0%	Strong merging, moderate structure
A1 Balanced	12,051.6	24,300	651.0	50%	10%	Highest top score in this batch

Experiment: More tuning! Refined weights based upon best performers

Variant	Avg Score	Top Score	Avg Moves	1024+ Rate	2048s	Notes
B1: Anchored Minimalist	9,949.2	16,688	569.0	40%	0%	Very high variance; steep drop-off
B2: Balanced Boost	10,144.8	15,676	570.0	50%	0%	Slightly better, but still mid-tier
B3: Aggro Struct	12,908.8	25,228	694.2	70%	10%	Best top score & strong average
B4: Hybrid Max	11,992.4	15,840	663.0	70%	0%	Most stable structure & late-game

5) Better Search

Hypothesis: We can do some early pruning of moves that don't change the board state. Also Make sure we are returning the best evaluated leaf even if we don't finish evaluating a given depth level.

Continue to refine weights around the Aggro Struct agent.

Results:
Agents have increasingly high scores and achieve better 2048 tile rates – but sometimes collapse early in the game (not able to get the corner and monotonicity set) leading to lower average game scores.

Agent	Avg Score (3 Rounds x 20)	Top Scores (best round)	2048+ Total	1024+ Consistency
A	~13,024 (16.8k, 10.9k, 11.9k)	**31,444** (R3)	7	27/60 (45%)
B	**12,753** (10k, **13.0k**, **13.1k**)	28,424	5	**36/60 (60%)**
C	11,101 (12.8k, 12.8k, 10.1k)	24,936	3	30/60 (50%)
D	11,329 (12.7k, 12.8k, 10.3k)	26,644	5	33/60 (55%)

Hypothesis: If we add a gentle “nudge” to our heuristic early in the game, maybe these early failures won't happen as much and we can boost average scores. Testing 3 Variants and a baseline.

Results:

Agent	Base	Avg Score	Top Score	1024+ Rate	2048s	Notes
C1	A + early anchor	11,516.6	25,976	55.0%	2	Stabilized version of volatile A — but still mid-tier
C2	B + early anchor	10,680.2	26,176	40.0%	2	Underperformed despite the fix — possibly overcorrected
C3	B + early anchor + corner++	11,397.5	25,568	57.5%	2	Slight boost in structure retention, but not transformative
C4	B (unmodified)	12,579.2	32,400	57.5%	5	Best average, best ceiling, best 2048 rate

Hypothesis: We have a solid strategy. It’s very focused on corner position and edge monotonicity. Our move ordering doesn't really reflect this. Can we test different move ordering strategies that align better with the heuristic? Prioritize moves that don't lose corner position first, and then score, and then board openness.

Results: IT turns out move ordering by our board gradient outperforms other options. That was a wild hair to throw in but worked well.

Agent	Strategy	Avg Score	Top Score	1024+ Rate	2048s	Notes
M1	Anchor first → old hybrid	10,273.2	25,948	42.5%	1	Weakest; anchoring priority isn’t enough by itself
M2	Anchor first → raw score	11,356.7	27,028	60.0%	2	➖ Mid-tier, stronger but unstable (some early crashes)

M3	Gradient score only	**12,857.1**	**28,404**	**67.5%**	3	Clear winner: favors long-term structure + strong average
M4	Baseline hybrid (C4-style)	12,101.0	26,672	60.0%	3	Consistent, but slightly behind M32

Agent	Description	Avg Score	Top Score	2048+ Rate	1024+ Rate	Moves/Game	Notes
N1	Baseline (Structure Gradient)	**13,667.3**	**33,648**	**15.0%**	**60.0%**	723.1	Best overall performance
N2	↑ More Gradient Bonus (0.12)	13,090.8	31,944	10.0%	62.5%	702.0	Very solid, but slightly behind
N3	↑ Anchor (0.7), 0.9:0.1 chain	12,948.3	29,836	10.0%	57.5%	697.5	Good top scores, but less consistent
N4	↑ Open Tile Bonus (survivalist)	9,853.0	15,788	**0.0%**	40.0%	563.5	Underperformed badly