

2048 AI – Analysis

Running the 2048 AI through various depths and propagation types gave interesting results, the solver would frequently achieve a score of above 5000 points regardless of any max depth between 1...6. A max depth of 0 corresponds to the ai choosing a random move each time and would only achieve scores of around 1000 points. This score difference is shown in Figures 1 and 2 where the mean score from 1...6 depth ranges from 6000 to >20,000 for both propagation types and a depth of 0 only achieving 1000 and 900 for avg and max respectively. All the values in the tables are averaged over 10 runs and the raw data can be provided on request.

The best depth for propagation is depth 6 as can easily be seen in Figure 3, this is true for propagation of either max or avg. Both propagation types achieve very similar scores for depth ≥ 4 and execute the algorithm in very similar run times. For a depth of 4 or less there doesn't seem to be much of a difference in which propagation type is chosen as they both effectively give the same results. However, for a depth of 5 and 6 the two different propagation types differ. The max propagation seems to become more unstable while avg propagation follows its upward trend and can achieve more increasing results than max propagation. This is especially seen in depth 5 where max achieves a mean_score of 12509 and avg achieves a score of 21232.

The max propagation does mostly have a lesser standard deviation than avg, with the exception of depth 2 where max has a std_dev of 7324.7 and avg of 3392.8, since this is only shown in depth 2 it can be treated as a trivial outlier. The mostly lesser standard deviation of mean_score in max propagation shows that max propagation does have more consistent scores within each depth however is not able to achieve as consistently high scores as avg propagation.

Although there is a tradeoff to avg achieving higher scores, the avg propagation type starts to take longer to run for depths of 5 and 6 taking 5.09 and 22.47 seconds compared to 3.55 and 17.52 seconds for max propagation. This time difference is only a few seconds and doesn't make too much of a difference in this environment but may be more significant if a larger depth is used or the algorithm is being run multiple times.

In general, the larger the depth used the better the algorithm is able to predict which move is more beneficial in the long run. Depths of 1...4 run very quickly taking 1.433 seconds at most, and can consistently achieve their max scores. Depths of 5 and 6 give greater scores than lower depths but take exponentially longer. The two types of propagations give identical results for depths of 1...4 but for depths of 5...6 it can be concluded that avg propagation gives better results with the tradeoff of taking a few seconds longer to run.

Figure 1: Table of Values for avg Propagation

DEPTH	0	1	2	3	4	5	6
Max_score	1032.6	6244.8	13212.4	11476	17606.4	21232	27616.8
Std_dev	588.9	2139.6	3392.8	4793.7	9056.8	8308.5	9207.7
Max_tile	105.6	512	921.6	793.6	1280	1484.8	1740.8
Std_dev	64.09	209.02	215.88	306.49	553.02	612.97	494.6
Run_time (s)	0.026	0.1	0.265	0.413	1.412	5.09	22.468

Figure 2: Table of Values for max Propagation

DEPTH	0	1	2	3	4	5	6
Max_score	903.2	6112	10706.4	12515.6	17714.8	12509.2	20883.6
Std_dev	385.45	1967.86	7324.7	3494.65	8243.5	4049.06	7015.98
Max_tile	92.8	512	793.6	921.6	1228.8	870.4	1433.6
Std_dev	38.31	209.02	518.36	215.88	600.98	247.32	528.79
Run_time (s)	0.023	0.102	0.209	0.452	1.433	3.551	17.518

Figure 3: Plot of Mean Max Score vs Depth for max and avg Propagations

