NEWS, INFORMATION, TOURNAMENTS, AND REPORTS

2048-BOT TOURNAMENT IN TAIWAN

Kun-Hao Yeh[1], Chao-Chin Liang[1], Kuang-che Wu[2], and I-Chen Wu[1]

1. The Tournament

The computer 2048 tournament was sponsored by Archilife Research Foundation. It was called the Taiwan 2048-bot contest, which took place in Taipei, Taiwan, from May 19th to June 6th, 2014. In this tournament, 76 teams participated. Table 1 lists the top-3 winners, KCWU (Wu, 2014), CGI-2048 (CGI-Lab, 2014), OWENLIN, and their final standings. They obtained points 40, 31 and 29 respectively.

Ranking	Program	Author(s)	Points
1	KCWU	Kuang-che Wu	40
2	CGI-2048	Chao-Chin Liang, Kun-Hao Yeh, I-Chen Wu	31
3	OWENLIN	Owen Lin	29

Table 1: The top-3 winners in Taiwan 2048-bot tournament.

2. 2048 Rules

The game 2048 (Cirulli, 2014), originated from Threes! (Vollmer, 2014), is a single-player game on a 4x4 board, where each cell is either empty or placed with a tile labeled with a value which is a power of two. Let v-tile denote the tile with a value v. Initially, two tiles, 2- or 4-tiles, are placed on the board at random. In each turn, the player makes a move by choosing one of the four directions, up, down, left and right. Upon choosing a direction, all the tiles move in that direction as far as they can until they meet the border or there is already a different tile next to it. When sliding a tile, say v-tile, if the tile next to it is also a v-tile, then the two tiles will be merged into a larger tile, $2v$-tile. At the same time, the player gains $2v$ more points in the score. A move is legal if at least one tile can be moved. After the player has made a move, the game generates a new 2-tile with a probability of 9/10 or a 4-tile with a probability of 1/10 on an empty cell chosen at random.

(a) (b) (c)

Figure 1: Examples of 2048 boards.

We consider an example, in which an initial board is shown in Figure 1 (a). After making a move to the left, the two 2-tiles merge into a 4-tile and becomes the one shown in Figure 1 (b). For this merge, the player gains 4 points. Then, a new 2-tile is randomly generated as shown in Figure 1 (c). The player can repeatedly make moves in this way.

A game ends when the player cannot make any legal move. The final score is the points accumulated during the game. The objective of the game is to accumulate as many points as possible. It is claimed that the player wins the game when a 2048-tile is created, but still allows players to continue playing.

3. Rules in the Tournament

In the Taiwan 2048-bot tournament (2014), each program plays 100 games with the average speed above 100 moves/second, and calculates the maximum score, the average score, the maximum tile and the win rate as indicators. For example, the first three programs obtain the indicators as shown in **Table 2**.

[1] Dept. of Computer Science, National Chiao Tung University, Hsinchu, Taiwan, and Email: {khyeh, ccliang, icwu}@aigames.nctu.edu.tw.
[2] Google Engineer, and Email: kcwu@csie.org

Program	Maximum Score	Average Score	Max Tile	2048 rate	4096 rate	8192 rate	16384 rate	32768 rate
KCWU	625260	277965	32768	100%	100%	96%	67%	2%
CGI-2048	367956	251794	16384	100%	100%	94%	59%	0%
OWENLIN	371908	171754	16384	99%	99%	83%	18%	0%

Table 2: Indicators for top-3 programs

In the Taiwan 2048-bot tournament (2014), these indicators of Table 2 are used for comparison in the following way. The maximum score is the first indicator. The top-10 programs in this indicator get points from 10 to 1. For example, KCWU obtained 10 points in this indicator as shown in **Table 2**, OWENLIN 9 points, and CGI-2048 8 points. The average score is the second indicator. The top-10 programs obtain points similarly. The maximum tile is the third indicator. The 1st program obtains 10 points, the 2nd to 4th 5 points, and the 5th to 10th 2 points. From **Table 2**, KCWU, CGI-2048, and OWENLIN obtained 10, 5, and 5 points respectively. The last indicator is the win rate. Originally, the win rate is defined as the ratios of the games that have reached a 2048-tile in the total of 100 games. If two programs have the same win rates, they compare their next-stage win rates, e.g., the win rate is redefined to the ratios of the games that have reached a 4096-tile in the total of 100 games. When comparing KCWU with CGI-2048 in this indicator from **Table 2**, they had the same 2048 rate and 4096 rate. Hence, the win rate that was used as their final comparison is the 8192 rate. In this way, KCWU, CGI-2048, OWENLIN obtained 10, 9, and 8 points respectively.

4. The Top Three Programs

In the tournament, the programs KCWU, CGI-2048, and OWENLIN won respectively the first place, second place and third place by obtaining points 40, 31 and 29 as shown in Table 1. The authors of KCWU and OWENLIN are senior programmers from Google Taiwan, and the authors of CGI-2048 are from a research group at National Chiao Tung University. All three teams used expectimax search. The former two teams exploited several heuristic techniques to carefully evaluate how well a position is and to optimize the performance. For example, they used monotonicity, empty tiles, distinct tiles, mergeable tiles, etc.

CGI-2048 was designed and implemented based on a new temporal difference (TD) learning method, called, *Multi-Stage TD Learning*, proposed by Wu *et al*. (2014). The method was improved from a TD learning technique with N-Tuple networks, proposed by Szubert and Jaskowski (2014). With the help of TD learning, CGI-2048 performed surprisingly well and fast. It ran about 500 moves/second.

F.l.t.r. Kun-Hao Yeh (CGI-2048, the second from left), Kuang-che Wu (KCWU, the third), and Chao-Chin Liang (CGI-2048, the fourth) in the Taiwan 2048-bot tournament.

Acknowledgement
The authors would like to thank the Taiwan 2048-bot tournament and the Archilife Research Foundation for offering and sponsoring the Taiwan 2048-bot tournament.

References

Cirulli, G. (2014). Game 2048, available at http://gabrielecirulli.github.io/2048/.

Vollmer, A. (2014). Game Threes, available at http://asherv.com/threes/.

Taiwan 2048-bot Tournament 2014, available at http://2048-botcontest.twbbs.org/.

Wu, K.C. (2014), 2048-c, available at https://github.com/kcwu/2048-c/.

CGI-Lab (2014), CGI-2048, available at https://github.com/CGI-LAB/Taiwan_Bot_Tournament_2048.

Szubert, M. and Jaskowaski, W. (2014). *Temporal Difference Learning of N-tuple Networks for the Game 2048,* IEEE CIG 2014 Conference.

Wu, I.C., Yeh, K.H., Liang, C.C., Chang, C.C. and Chiang, H. (2014). *Multi-Stage Temporal Difference Learning for 2048*, TAAI 2014 Conference.