Solving 2048 with RL

Yevhen (Jake) Horban

Table of Contents

Environment

- 4x4 grid starts with two randomly placed tiles
- p(2-tile) = 9 / 10, p(4-tile) = 1 / 10
- A = {L, R, D, U} to slide and merge tiles
- Valid action if the board changes
- If valid spawns a new tile in an empty space
- Reward is the sum of all merged tiles values
- $v = 2^i$, $i \in \{0, 1, \ldots, 17\}$
- $|S| = 18^{16} \cong 1.2*10^{20}$ = 2.8 times 3x3x3 Rubik's cube (some are unreachable)

Methods

Multistage TD learning

- Well modeled as an MDP

$$s_0 \cdots \rightarrow s_t \underset{r_t}{\overset{a_t}{\rightarrow}} s'_t \rightarrow s_{t+1} \underset{r_{t+1}}{\overset{a_{t+1}}{\rightarrow}} s'_{t+1} \rightarrow \cdots s_T.$$

Optimistic Initialization

$$\pi(s_t) = \text{argmax}_{a_t} \left(r_t + \sum_{\forall s_{t+1}} \mathcal{P}(s'_t,\ s_{t+1})\, V(s_{t+1}) \right)$$

TC Learning

$$\theta_i \leftarrow \theta_i + \alpha \beta_i \delta_t.$$

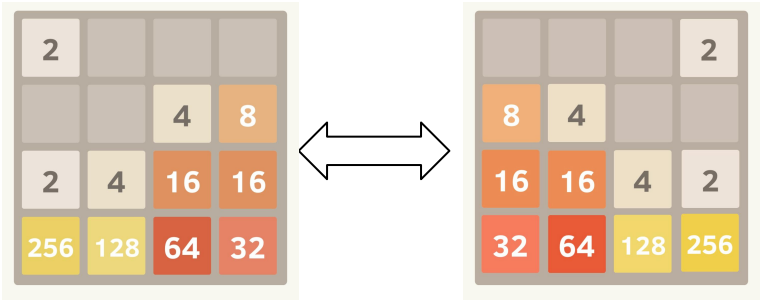
$$\beta_i = \begin{cases} |E_i| / A_i, & \text{if } A_i \neq 0 \\ 1, & \text{otherwise.} \end{cases}$$

$$E_i \leftarrow E_i + \delta_t \ \text{ and } \ A_i \leftarrow A_i + |\delta_t|.$$

Methods

Expectimax Search

- monotonicity of a board
- number of empty tiles
- number of mergeable tiles



Transposition table

- Zobrist hashing

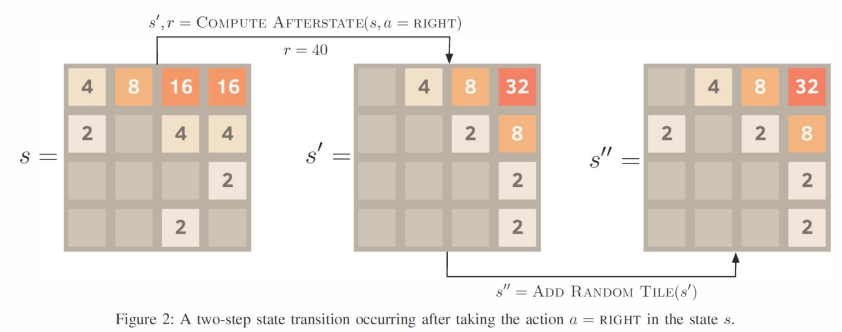


Figure 2: A two-step state transition occurring after taking the action a = RIGHT in the state s.

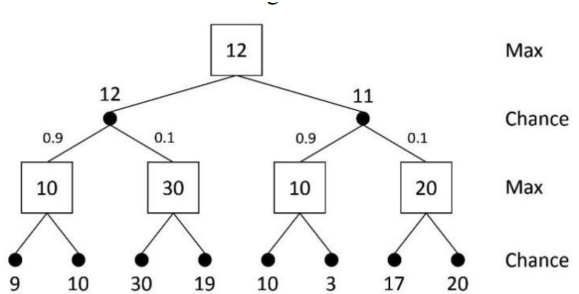


Fig. 2. An expectimax search tree.

Methods

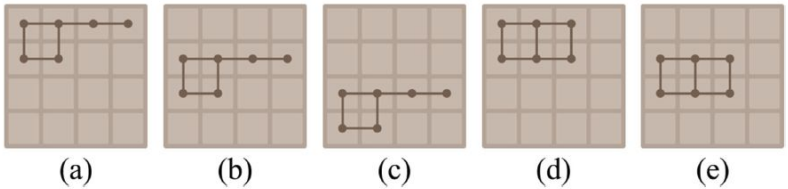


Fig. 2. (a), (b), (d), and (e) 4 × 6-tuple network proposed by Yeh *et al.* [3]. (a)–(e) 5 × 6-tuple network used by Jaśkowski [5].

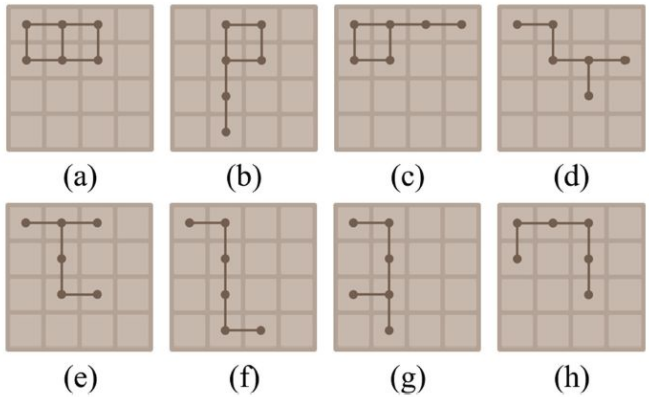


Fig. 3. (a)–(h) 8 × 6-tuple network proposed by Matsuzaki [8].

N-Tuple Networks

- output of a network is a linear summation of feature weights for all occurring features

$$V\ (s) = \sum_{i=1}^{m} \mathrm{LUT}_i\left[\phi_i\ (s)\right]$$

- n=4, 17 * 15^4 = 860,625 weights

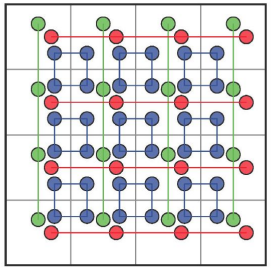


Figure 8: The n-tuple network consisting of all possible horizontal and vertical straight 4-tuples (red and green, respectively), and all possible 2 × 2 square tuples (blue).

Results - TD2048+

optimistic TD+TC (OTD+TC) learning for training and tile-downgrading (DG) expectimax search for testing by Hung Guei

Search	Score	8192 [%]	16384 [%]	32768 [%]	# Games
1-ply w/ DG	412,785	97.24%	85.39%	30.16%	1,000,000
2-ply w/ DG	513,301	99.17%	94.40%	48.92%	100,000
3-ply w/ DG	563,316	99.63%	96.88%	57.90%	10,000
4-ply w/ DG	586,720	99.60%	98.60%	62.00%	1,000
5-ply w/ DG	608,679	99.80%	97.80%	67.40%	100
6-ply w/ DG	625,377	99.80%	98.80%	72.00%	100

In addition, for sufficiently large tests, 65536-tiles are reached at a rate of 0.02%

Future Work

- Finish implementation
- Collect results
- Research additional features

Code: https://github.com/h0rban/2048-rl

Citations

Guei, Chen, L.-P., & Wu, I.-C. (2021). Optimistic Temporal Difference Learning for 2048. IEEE Transactions on Games, 1–1. https://doi.org/10.1109/TG.2021.3109887

Kohler, Migler, T., & Khosmood, F. (2019). Composition of basic heuristics for the game 2048. Proceedings of the 14th International Conference on the Foundations of Digital Games, 1–5. https://doi.org/10.1145/3337722.3341838

Yeh, Wu, I.-C., Hsueh, C.-H., Chang, C.-C., Liang, C.-C., & Chiang, H. (2016). *Multi-Stage Temporal Difference Learning for 2048-like Games*. https://doi.org/10.48550/arxiv.1606.07374

Szubert, & Jaskowski, W. (2014). Temporal difference learning of N-tuple networks for the game 2048. 2014 IEEE Conference on Computational Intelligence and Games, 1–8. https://doi.org/10.1109/CIG.2014.6932907