AI Tetris - A CS3243 Project

Lau Kar Rui, A0155946U
Poh Jie,
Matilda,
Chee Wee,

25th March 2018

1 Introduction

2 Utility Function

We defined the utility function as a linear function F with each heuristic $h_i \in Heuristics$ having an assigned weight w_i, where $i = 1...n$, with n being the number of heuristics. Each h_i derives a real value from a state s. The function is then defined as:

$$F(s) = \sum_{i=1}^{n} w_i h_i(s)$$

The heuristics h_i used will be explained in depth in Section 3.

3 Heuristics Used

3.1 Rows Cleared

3.2 Maximum Height Increase

3.3 Average Height Increase

3.4 Number of Holes

3.5 Well depth

3.6 Game Loss

3.7 Levelness of Top

4 Training Function

5 Implementation

5.1 StateCopy

In order to correctly apply the heuristics, a new `StateCopy` class was created, extending the original `State` class, serving as a clean starting state to apply our heuristics on in order to derive the heuristic value.

Extra variables like `currentRowsCleared` and `previousTop` is also added to the `StateCopy` class in order to obtain the information needed by various heuristics such as the RowsCleared heuristic and the AverageHeight heuristic.

Using `StateCopy` also allows us to play moves without affecting the original state of the game.

5.2 Heuristic

Talk about the heuristic class here

5.3 Learning Algo? Learner?

6 Scaling The Algorithm For Big Data

7 Results

8 Conclusion

References

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.

[2] Albert Einstein. *Zur Elektrodynamik bewegter Körper*. (German) [*On the electrodynamics of moving bodies*]. Annalen der Physik, 322(10):891921, 1905.

[3] Knuth: Computers and Typesetting,
`http://www-cs-faculty.stanford.edu/~uno/abcde.html`