Proceedings of the Fifteenth AAAI Conference on Artificial
Intelligence and Interactive Digital Entertainment (AIIDE-19)

Learning Strategies for Real-Time Strategy Games with Genetic Programming

Julian R. H. Mariño
Departamento de Sistemas de Computação, ICMC
Universidade de São Paulo, Brazil

Abstract

Planning in real-time strategy (RTS) games is challenging due to their very large state and action spaces. Action abstractions have shown to be a promising approach for dealing with this challenge. Previous approaches induce action abstractions from a small set of hand-crafted strategies, which are used by algorithms to search only on the actions returned by the strategies. Previous works use a set of expert-designed strategies for inducing action abstractions. The main drawback of this approach is that it limits the agent behaviour to the knowledge encoded in the strategies. In this research, we focus on learning novel and effective strategies for RTS games, to induce action abstractions. In addition to being effective, we are interested in learning strategies that can be easily interpreted by humans, allowing a better understanding of the workings of the resulting agent.

Introduction

Real-time strategy (RTS) games are challenging for search-based planning systems. This is because RTS games normally have very large action spaces. Further, search algorithms are allowed only a few milliseconds for planning. Churchill and Buro (2013) introduced the concept of action abstraction, a promising approach for dealing with large action spaces in the context of real-time planning. Action abstractions allow search algorithms to explore only a subset of the legal actions available at a given state. Previous works used a small set of expert-designed strategies for inducing action abstractions by considering during search only the actions returned by the expert-designed strategies. The problem of using a small set of expert-designed strategies is that the resulting agent can be limited by the expert's knowledge. In Mariño et al. (2019), we mitigated this problem by generating a large pool of strategies $\mathcal{Z}$ from a small set of strategies designed by experts. We also presented an evolutionary algorithm that returns a subset of $\mathcal{Z}$, which was then used to induce an action abstraction. The main drawback of Mariño et al.'s approach is that the agent behaviour could still be limited by the experts' knowledge encoded in the initial set of strategies $\mathcal{Z}$.

In this research, we are interested in learning novel and effective strategies from scratch, i.e., without an initial set of expert-designed strategies. In order to address this problem, we aim to generate and evolve computer programs where each program defines a strategy for playing the game. Namely, we propose a Genetic Programming approach that is able to discover entirely novel strategies represented by computer programs. Our hypothesis is that abstractions induced by the strategies generated with our Genetic Programming approach could define more promising search spaces for heuristic search algorithms.

In addition to be effective, we are interested in generating strategies that can be interpretable by humans. Artificial Intelligence algorithms do not usually instruct people on how to solve the problem in question. This is because planning algorithms only tell you the best action to take without explaining the reasons for their chooses. As example, the algorithm AlphaGo described in Silver et al. (2016), during its match against Lee Sedol, made the famous movement 37, which was initially considered a wrong movement because it was not a common one according to professionals. However, movement 37 turned out to be key for AlphaGo's victory. We argue that easily interpretable strategies can help in the learning process of humans, who could train from the agent behaviours encoded in the strategies.

Empirical results on μRTS, a real-time strategy game developed for research purposes, show that our previous methods are effective and our current work is promising.

Previous Work: Evolving Expert-Designed Strategies

In a previous work, we introduced an evolutionary approach for generating action abstractions (Mariño et al. 2019). In that work, we introduced an approach to automate the generation of action abstractions for two-player extensive form games while still using the knowledge encoded in a small set of expert-designed strategies. Namely, our approach generates a large pool of strategies $\mathcal{Z}$ from a small set of options $\mathcal{I}$ designed by domain experts. This is achieved by defining the strategies in $\mathcal{I}$ as rule-based strategies. By changing the parameter values of those rule-based strategies we are able to generate a large pool of strategies $\mathcal{Z}$, and any non-empty subset of $\mathcal{Z}$ can be used to induce an action abstraction.

In this previous work, we cast the problem of generating an action abstraction from $\mathcal{Z}$ as a two-player simultaneousmove game, which we name the subset selection game (SSG). In the SSG each player selects a subset of $\mathcal{Z}$, A and B, which are then used to induce an action abstraction for each player. The payoff of the SSG is defined by the result of the original two-player extensive-form game played by two versions of a planning system: one using the action abstraction induced by A and another using the action abstraction induced by B.

For solving the SSG we introduced an evolutionary algorithm and evaluate it in the domain of μRTS. Empirical results on small μRTS matches showed that our evolutionary approach is able to converge to a Nash equilibrium profile for the SSG. Results in medium-sized and large μRTS matches showed that two search-based planning systems using action abstractions derived by our method are able to outperform all state-of-the-art planning systems tested in our experiments.

The results of this work were published in Mariño et al. (2019). A version of the search-based algorithm A3N (Moraes et al. 2018) using an action abstraction generated by our method won the 2018 μRTS competition in two tracks. The μRTS competition is organized to approach research questions underlying the development of algorithms able to planning in large action spaces.

Despite the promising results of Mariño et al. (2019), the agent behaviour could still be limited by the experts' knowledge encoded in the initial set of strategies in $\mathcal{I}$.

Current Work: Discovering New Strategies Via Genetic Programming

In order to avoid that the resulting agent behaviour depends on an initial set of expert-designed strategies, we address the generation of new strategies through a Genetic Programming approach. Instead of evolving predefined expert-designed strategies, we aim to generate and evolve multiple sets ∇ of computer programs, where each program in ∇ defines a strategy for playing the game. A set ∇ is considered "fit for survival" if a search algorithm using the action abstraction induced by ∇ is able to defeat the same search algorithm using an action abstraction induced by another set of programs. Evolutionary operators such as mutation and crossover are applied directly to the computer programs, enabling the discovery of novel strategies that could go beyond the expert's knowledge.

Another research question we will investigate is if the interpretable strategies generated by our approach can help in the learning process of humans. Since the strategies generated by genetic programming are encoded as code, our hypothesis is that the code itself can be easily interpreted and used by humans to learn directly from the strategy. There are two target audiences that can learn from these strategies. Players could learn from the strategies in order to improve their skills. Professional game AI programmers can be assisted in the creation of agents through the examples given by novel and effective strategies that our method discovers.

Conclusion

In a previous work, we demonstrated that evolutionary approaches can be effective to generate action abstractions. The main drawback of that approach is that it limits the agent's behaviour to the knowledge encoded in an initial set of expert-designed strategies. In our current research we are trying to use Genetic Programming approaches to evolve novel strategies that do not have to rely on an initial set of expert-designed strategies. Also, we hope that the strategies generated by our proposed Genetic Programming approach can be easily interpretable by humans, allowing a better understanding of the workings of the resulting agents.

References

Churchill, D., and Buro, M. 2013. Portfolio greedy search and simulation for large-scale combat in StarCraft. In *Proceedings of the Conference on Computational Intelligence in Games*, 1–8. IEEE.

Mariño, J. R. H.; Moraes, R. O.; Toledo, C.; and Lelis, L. H. S. 2019. Evolving action abstractionsfor real-time planning in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Moraes, R. O.; Mariño, J. R. H.; Lelis, L. H. S.; and Nascimento, M. A. 2018. Action abstractions for combinatorial multi-armed bandit tree search. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 74–80. AAAI.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of go with deep neural networks and tree search. *nature* 529(7587):484.