An Evolutionary Online Adaptation Method for Modern Computer Games Based on Imitation

Steffen Priesterjahn
Department of Computer Science
University of Paderborn
33098 Paderborn, Germany
spriesterjahn@upb.de

Alexander Weimer
Department of Computer Science
University of Paderborn
33098 Paderborn, Germany
aweimer@upb.de

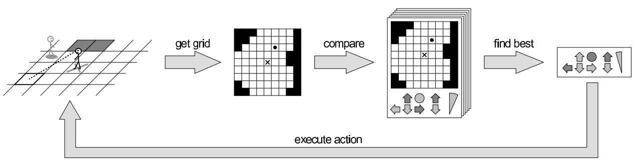
Categories and subject descriptors: I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—Intelligent agents, Multiagent systems; I.2.6 [Artificial Intelligence] Learning—Knowledge Acquisition

General Terms: Algorithms

Keywords: Computer Games, Distributed Learning, Evolutionary Computation, Imitation, Memetics

Creating AI agents for computer games is very special. The agents should not be as good as possible, but approximately as good as the current human players. They should impose a challenge on the human players but still be beatable. Most importantly they should not be easily identifiable as agents, but show human like behaviours and movements. We think that the application of imitation learning and other imitation techniques are very well equipped for handling such conditions. This paper presents an online evolutionary learning approach to create artificial players which are able to compete in combat situations for a three-dimensional action game. It is largely based on our previous work published in [1,2], which use offline approaches to learn successful behaviour in such situations using standard evolution.

The agents sense their direct environment by dividing it into a grid of quadratic regions. The agent is always in the centre of this grid and the grid is always rotated in relation to the agent. So, each grid field has always the same relative position to the agent. Traces are used to see, if a grid field is filled or empty. If an opponent occupies a grid field, the respective field is marked with another special value. The size of the grid is finite and covers only the vicinity of the agent. The agents classify this grid by comparing it to a set of grids in their memory. Each grid is part of a rule which also contains an according command which is then executed. More detail on this representation can be found in [1,2]. The following figure illustrates the described procedure.



While playing in parallel the agents evaluate their performance. They record the direct impact $v_0(r)$ of each executed rule r. After a fixed time frame the agents then do a policy evaluation, based on reinforcement learning theory, according to the following formulas.

$$
\begin{aligned}
v_0^{\text{new}}(r) &= v_0^{\text{old}}(r) + \text{applied Damage} - \text{received Damage} \\
v(r) &= v_0(r) + \gamma \sum_{r' \in R} p_{rr'} v(r')
\end{aligned}
$$

$p_{rr'}$ is the transition probability between r and r' and γ is a discount value chosen from $[0, 1[$. The resulting values $v(r)$ are used to determine the most important rules. Then the following algorithm is used to adapt the behaviour of the agents.

```
initialize from recorded behaviours
loop
    evaluate agents for one minute
    determine the best agent $a^*$
    do policy evaluation
    determine the $\sigma$ best rules $R^*$ of $a^*$
    for all agents except $a^*$ do
        replace rules with rules from $R^*$
    end for
    determine the worst agent $a^-$
    replace $a^-$ by a mutated copy of $a^*$
    for all agents do
        if agent is too bad or too good then
            randomly draw $\sigma$ imitation rules from last opponent
            replace rules with imitation rules
        end if
    end for
end loop
```

Rule replacement is done by comparing the value of with the most similar rule and keeping the better rule.

We tested our approach in various parameter setups using a small map and the standard Quake3-bot as the opponent. The following results were obtained: The agents imitate their opponents, often mirroring their movements. They are able to improve over their opponents, but do not dominate them. The percentage of winning agents raises over time reaching 35% after one hour and 50% after 2 hours in the best experiments. However, a sufficient number of agents is needed. The experiments using at least 16 agents worked, but not the ones using 8 agents. With the right setup no defunct agents were produced in any adaptation step.

In conclusion the presented approach shows very promising results which will be further researched. Our main focus lies in the acceleration of the learning rate and a further stabilisation of the learning process.

[1] S. Priesterjahn, O. Kramer, A. Weimer, and A. Goebels. Evolution of Reactive Rules in Multi Player Computer Games based on Imitation. In *Proceedings of the International Conference on Natural Computation (ICNC'06)*, volume 2, pages 744–755, 2005.

[2] S. Priesterjahn, O. Kramer, A. Weimer, and A. Goebels. Evolution of Human-Competitive Agents in Modern Computer Games. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC'06)*, pages 777–784, 2006.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
ACM 978-1-59593-697-4/07/0007.