SCML 2021 - OneShot Track

Submitted by:
Inbal Avraham - Bar Ilan University
Josef Weizman - Bar Ilan University
Assaf Tirangel - Bar Ilan University

Introduction

The OneShot game is a simplification of the SCML standard game, designed to de-emphasize long-term production and planning, and instead emphasize negotiations.
We chose to improve existing strategies and solve some of their drawbacks.

Strategy

Our approach is a semi-greedy agent that attempts to maximize the profit. The agent is an enhancement of several agents from the documentation.
One of the existing suggested OneShot agents is “BetterAgent " which is a greedy agent that uses concessions exponent. The agent does not take into account the amounts on previous negotiations. The agent “AdaptiveAgent” considers previous negotiations and limits the concessions. This logic aims to improve the agent’s performance over time. However, this approach makes the agent hard-headen by not compromising the price it demands, once it gets a better price.

Our strategy uses AdaptiveAgent’s approach, but solves the main drawback, which is insisting on the best price it observes and thus losing negotiations. Our agent uses information from previous negotiations and tries to balance between the attempt to reach a good price from one side, and not reaching too many disagreements from the other side.

Instead of saving the best selling\buying prices, our agent maintains a vector of prices. In each negotiation, if the agent reaches a better price compared to the previous ones, it adds this price to the vector. With this approach, the agent doesn’t save information only about the best price, but a list of good prices compared to the previous stages.

At a decision point (sell or buy), the agent calculates the average of the price history. It determines whether to agree or decline the negotiation based on whether the current price is above\below the average price. With this approach, while the agent improves over time, it doesn’t reject each offer that doesn’t reach the best observed price. This addresses the main problem of AdaptiveAgent and significantly improves its performance.

Other modifications

As written above, our agent addresses the main drawback of AdaptiveAgent, which uses concession exponent for negotiations. We evaluated the performances of several exponent values: 0.15, 0.2, 1, 5. Our performance tests showed that choosing the value 0.2 gives the best results.

Conclusion

Our agent is based on the documented AdaptiveAgent. We solved the main issue of this agent by maintaining a list of previous prices, and limiting the current price by calculating the average of previous prices. We demonstrated that this behavior significantly improves the performance of the agent compared to the documented agents, as well as the competitors.

Link to the agent:
https://github.com/user148212/scml/blob/main/myagent.py