Negotiation Strategy using Reinforcement Learning for SCML OneShot Track

Takumu Shimizu
Tokyo University of Agriculture and Technology

RLAgent

Concept

- Negotiates with the opponents independently
- Applied Reinforcement Learning
- Defines the Markov Decision Process (MDP)

MDP for OneShot Track

- **State** consists of the following factors:
  - The current number of rounds
    $r \in \{0,1,\dots,R\}$
    - R is the negotiation deadline
  - The current needs q_r^{needs}
  - The opponent's offer $\omega_r'^a$

Possible values of items in the opponent's offer $\omega_r'^a$

Item	Value
quantity q'	0 ~ 10
time t'	0 ~ 200
Unit price p'	High or Low

- **Action** consists of the following factors:
  - The accept signal η_r^a
  - The counter offer ω_r^a

Possible values of items in the counter offer ω_r^a

Item	Value
quantity q'	0 ~ 10
Unit price p'	High or Low

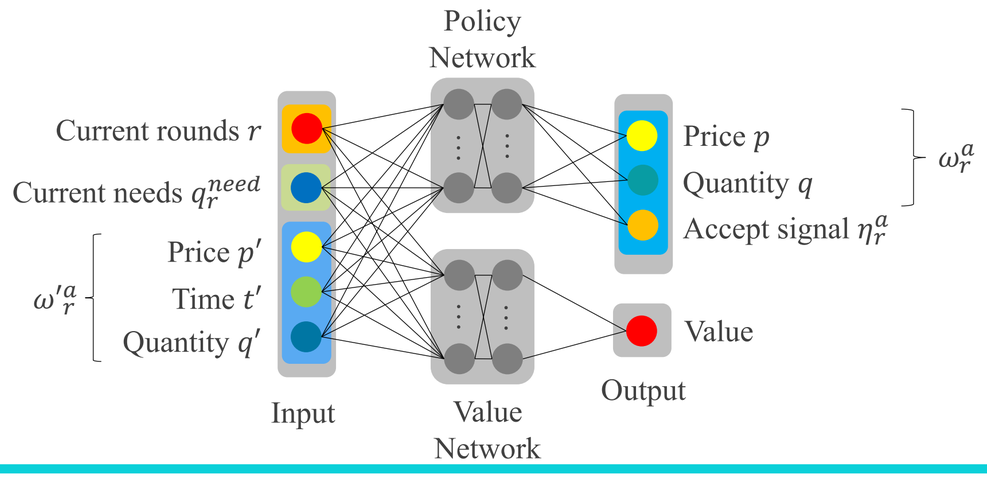
- **Reward** is the profit of the day
  - Calculated by the utility function (OneShotUfun)
    - Using the contract of the day and the exogenous contract
  - RLAgent get the profit as the reward in the last round of the day
    - Otherwise, the reward is 0

RLAgent Negotiation Strategy

1. Receives the opponent's offer
   - In the first round, it receives the supposed offer
2. Enters the offers into the model as the state and gets an action
3. Sends the response to the opponent
   - Depends on the accept signal η_r^a
     - **True**: an acceptance response
     - **False**: a counter offer ω_r^a
   - When the needs $q_r^{needs} \leq 0$, RLAgent ends the negotiation

Model Overview

- State and Action are expressed by MultiDiscrete
  - Converts each item to one-hot representation



RLSyncAgent

Concept

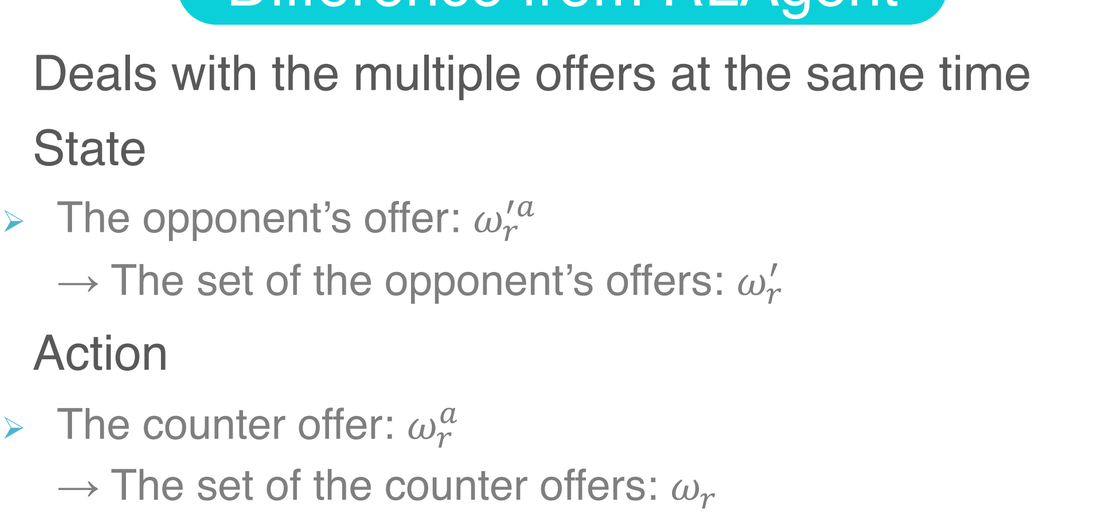
- Negotiates with the opponents concurrently
- Applied Reinforcement Learning
- Defines the Markov Decision Process (MDP)

Difference from RLAgent

- Deals with the multiple offers at the same time
- State
  - The opponent's offer: $\omega_r'^a$
    → The set of the opponent's offers: ω_r'
- Action
  - The counter offer: ω_r^a
    → The set of the counter offers: ω_r
- Model
  - The number of nodes is added with changes in the state and the action

Model Overview

- The nodes of input and output layer are added



Evaluation

- RLAgent gets lower scores than the sample agents
  - RLAgent cannot consider other negotiations
- RLSyncAgent gets significantly lower on all scores
  - The challenge is how to make the offer
    - It is difficult to adjust the total quantity due to predictions of accepted offers
- We submitted RLAgent to the competition

Table 1: The test results of RLAgent and RLSyncAgent

Agent	score	min	Q1	median	Q3	max
RLAgent	0.927	0.708	0.864	0.947	0.991	1.051
RLSyncAgent	0.712	0.173	0.461	0.809	0.910	1.056
SimpleAgent	1.035	0.595	1.004	1.080	1.127	1.204
AdaptiveAgent	0.978	0.620	0.883	0.989	1.083	1.206
LearningAgent	0.982	0.618	0.881	0.981	1.110	1.212