KanbeAgent for ANAC SCML One Shot Track

Masato Kijima
Tokyo University of Agriculture and Technology
kijima@katfuji.lab.tuat.ac.jp

May 3, 2023

1 Negotiation Strategy

KanbeAgent inherits *AdaptiveAgent* and negotiates with opponents independently. We don't set utility functions and the decision of KanbeAgent is mainly based on the quantity.

The following shows the trading information for the decision of our agent.

$$
\begin{aligned}
i &: \text{simulation step} \\
s &: \text{negotiation step} \\
n^{ap} &: \text{the number of all negotiation partners} \\
n^{c}_{s} &: \text{the number of all agreements by step s} \\
n^{fo}_{s} &: \text{the number of all finished partners by step s} \\
q^{need} &: \text{the quantity of items needed to achieve the exogenous contract} \\
q^{max}, q^{min} &: \text{the maximum and minimum quantity of the negotiation issues} \\
p^{max} &: \text{maximum unit price of the negotiation} \\
p^{min} &: \text{minimum unit price of the negotiation}
\end{aligned}
$$

1.1 Setting the Quantity Range

The feature of KanbeAgent is that it sets the range of quantity of trading. It is very risky to conclude an exclusivity agreement with one agent. This is because no agent wants to conclude negotiations on terms that are unfavorable to it. In the worst case, negotiations may be protracted and the required quantity may not be secured.

Therefore, KanbeAgent decided to secure quantity by negotiating with multiple agents and set minimum and maximum targets for the number of negotiation conclusion partners. In other words, KanbeAgent sets a limit on the number of quantities it handles and negotiates within this range. Specifically, KanbeAgent aims to conclude negotiations with 3/4 to 1/2 of its all negotiating partners. The maximum and minimum target number of contracting partners at step s n_s^{mxto}, n_s^{mnto} is expressed by the following formula.

$$
\begin{aligned}
n_s^{mxto} &= \begin{cases} \frac{3}{4}n^{ap} - n_s^c & n_s^{fo} < \frac{1}{4}n^{ap} \\ n_s^{remain\_ptr} & \text{otherwise} \end{cases} \\
n_s^{mnto} &= \begin{cases} \frac{1}{2}n^{ap} - n_s^c & n_s^{fo} < \frac{1}{2}n^{ap} \\ max(1, \frac{1}{4}(\frac{1}{2}n^{ap} - n_s^c + n_s^{mxto})) & \frac{1}{2}n^{ap} < n_s^c \\ n_s^{remain\_ptr} & \text{otherwise} \end{cases}
\end{aligned} \tag{1}
$$

Here, $n_s^{remain\_ptr} = n^{ap} - n_s^c - n_s^{fo}$ is the number of partners KanbeAgent can negotiate at step s.

Using these, the upper limitq_{trades}^{max} and the lower limitq_{trades}^{min} of the quantity KanbeAgent handles at step s are decided by the following formula.

$$
\begin{aligned}
q_{trades}^{max} &= min(q^{max}, \frac{q^{need}}{n_s^{mnto}}) \\
q_{trades}^{min} &= max(q^{min}, \frac{q^{need}}{n_s^{mxto}})
\end{aligned} \tag{2}
$$

1.2 Offer Strategy

At the negotiation step s, KanbeAgent determines the offer unit price $p_{s,a}^{offer}$ and the offer quantity $q_{s,a}^{offer}$ to the partner agent a based on some trading information. We divided the negotiation into three parts and determine the offer based on each strategy. The first part of the negotiation is $0 \leq s \leq 4$, the second part of it is $5 \leq s \leq TT$, and the final part of it is $TT \leq s$. Here, TT denotes the time to transition from the second to the final part, and $TT = 20 - q^{need}$.

1.2.1 Determine Price

First, KanbeAgent defines the best and the worst price p^{best}, p^{worst} as the following formula.

$$
p^{best} = \begin{cases} p^{max} & \text{if selling} \\ p^{min} & \text{if buying} \end{cases}, \; p^{worst} = \begin{cases} p^{min} & \text{if selling} \\ p^{max} & \text{if buying} \end{cases} \tag{3}
$$

In the first part of the negotiation, KanbeAgent doesn't have to concede because the acceptable standards of other agents are usually estimated to be high. Therefore, KanbeAgent always determines the offer price $p_{s,a}^{offer}$ to be equaled its best price p^{best}.

In the second part of the negotiation, it is necessary to determine the appropriate price to agree certainly. However, KanbeAgent would also like to focus on concluding negotiations at a better price. Therefore, we make KanbeAgent determines the offer price based on the number of agents it can negotiate $n_s^{remain\_ptr}$. KanbeAgent determines the offer price $p_{s,a}^{offer}$ to be equaled its best price p^{best} at first. And, it determines the offer price to be equaled p^{worst} if $\frac{1}{4}n^{ap} \leq n_s^{fo}$ or $\frac{3}{4}n^{ap} \leq n_s^c$. However, if the price of the last offer from the partner agent a p_{s-1}^a is equal to p^{best}, KanbeAgent determines the offer price to be equaled p^{best}.

In the final part of the negotiation, it is necessary to conclude negotiations as soon as possible. Therefore, KanbeAgent usually determines the offer price $p_{s,a}^{offer}$ to be equaled its worst price p^{worst}. However, if the price of the last offer from the partner agent a p_{s-1}^a is equal to p^{best}, KanbeAgent determines the offer price $p_{s,a}^{offer}$ to be equaled p^{best}.

The following formula is a summarized offer price of KanbeAgent.

$$\begin{aligned} p_{s,a}^{offer} &= \begin{cases} p^{best} & \text{TF} = 0 \text{ or } p_{s-1}^a = p^{best} \\ p^{worst} & \text{TF} = 1 \end{cases} \\ TF &= \begin{cases} 1 & \frac{1}{4}n^{ap} \leq n_s^{fo} \text{ or } \frac{3}{4}n^{ap} \leq n_s^c \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4}$$

Here, TF is the variable that indicates the timing of the transition from p^{best} to p^{worst}.

1.2.2 Determine Quantity

At negotiation step s, KanbeAgent determines the offer quantity to the agent a $q_{s,a}^{offer}$ based on its quantity range.

In the first part of the negotiation, KanbeAgent doesn't have to concede because the acceptable standards of other agents are usually estimated to be high. Therefore, KanbeAgent determines the offer quantity *quantity* as high as possible by the following formula. Here, $q_{best\_price}^a$ is the best number of quantity that agent a has accepted when the price was p^{best}.

$$quantity = min(q^{need}, max(\frac{1}{2}q^{max}, q_{best\_price}^a)) \tag{5}$$

However, it is necessary to match the number of the offer quantity *quantity* with the number of the agent a's last offer quantity q_{s-1}^a to avoid being rejected by the other party. Therefore, when the last offer price from the opponent is p^{best}, KanbeAgent matches *quantity* with q_{s-1}^a. The offering quantity $q_{s,a}^{offer}$ is shown by the following formula.

$$q_{s,a}^{offer} = \begin{cases} quantity & p_{s-1}^a = p^{worst} \\ max(min(quantity, q_{s-1}^a), q_{trades}^{min}) & \text{otherwise} \end{cases} \tag{6}$$

In the second part of the negotiation, KanbeAgent decreases the offer quantity from q_{trades}^{max} to q_{trades}^{min} to make it easy for the opponent to accept. However, when KanbeAgent determines the unit price offer to change from p^{best} to p^{worst}, KanbeAgent starts to offer the quantity $q^{q_{trades}^{max}}$ again. KanbeAgent also matches the quantity with the last opponent's offer quantity to end the negotiation earlier.

In the final part of the negotiation, KanbeAgent should end the negotiation as soon as possible because the number of partners, KanbeAgent can negotiate, is decreased. Therefore, KanbeAgent determines the offer quantity to be equaled q^{max}_{trades} at first. KanbeAgent determines it to be equaled q^{min}_{trades} at the end of the negotiation step. KanbeAgent also matches the quantity with the last opponent's offer quantity to end the negotiation earlier.

The following formula is a summarized offer quantity of KanbeAgent.

$$
\begin{aligned}
&\text{if: } s < 5\\
&q^{offer}_{s,a} = \begin{cases} quantity & p^a_{s-1} = p^{worst}\\ max(min(quantity, q^a_{s-1}), q^{min}_{trades}) & \text{otherwise}\end{cases}\\
&quantity = min(q^{need}, max(\frac{1}{2}q^{max}, q^{opp}_{best\_price}))\\
&\text{elif: } 5 \leq s < TT\\
&q^{offer}_{s,a} = \begin{cases} max(min(q^a_{s-1}, q^{need}), q^{min}_{trades}) & p^a_{s-1} = p^{best}\\ max(q^{min}_{trades}, min(quantity, q^a_{s-1})) & p^{offer}_{s,a} = p^{worst}\\ max(min(q^{max}_{trades}, q^a_{s-1}), q^{min}_{trades}) & p^a_{s-1} = p^{worst}\end{cases}\\
&quantity = \begin{cases} q^{offer}_{s-1,a} - 1 & q^{offer}_{s-1,a} > q^{min}_{trades}\\ q^{min}_{trades} & \text{otherwise}\end{cases}\\
&\text{elif: } TT \leq s < 18\\
&q^{offer}_{s,a} = \begin{cases} min(q^{need}, q^a_{s-1}) & q^{max}_{trades} < q^a_{s-1}\\ max(min(q^{max}_{trades}, q^a_{s-1}), q^{min}_{trades}) & \text{otherwise}\end{cases}\\
&\text{elif: } 18 \leq s\\
&q^{offer}_{s,a} = min(q^{min}_{trades}, q^a_{s-1})
\end{aligned}
\tag{7}
$$

1.3 Acceptance Strategy

KanbeAgent determines the response to the opponent a's offer p^q, q^a based on the unit price and the range of the quantity of KanbeAgent. We explain responses in each case. Here, p^a is expressed as the unit price of the offer and q^a is expressed as the quantity of it.

Firstly, if $q^{need} \leq 0$, KanbeAgent responses END_NEGOTIATION to all opponents. KanbeAgent reject all offers if $q^a > q^{need}$.

If p^a is equaled to p^{best}, KanbeAgent mainly determines the response whether q^a is bigger than q^{min}_{trades} or not. KanbeAgent mainly accept the offer when $q^{min}_{trades} \leq q^a$, and reject other offers. However, if the negotiation step $s \geq 18$ and $q^a \leq q^{need}$, it accepts all offers. KanbeAgent rejects all other offers.

If p^a is equaled to p^{worst}, KanbeAgent mainly determines the response based on the negotiation step s and the price of KanbeAgent's last or next offer $p^{offer}_{last}, p^{offer}_{next}$. When $s < TT$, KanbeAgent accepts the offer if $q^{min}_{trades} \leq q^a$ and p^{offer}_{last} or p^{offer}_{next} is equaled to p^{worst}. When $TT \leq s$, KanbeAgent accepts under the same conditions as if $p^a = p^{best}$. KanbeAgent rejects all other offers.

The following formula is summarized responses to the partner a of KanbeAgent $response^a$.

$$
\begin{aligned}
&\text{if: } q^{needed} \leq 0 \\
&\quad response^a = \text{END\_NEGOTIATION} \\
&\text{if: } p^a = p^{best} \\
&\quad response^a = \begin{cases} \text{ACCEPT} & \text{if } s \leq 17 : q^{min}_{trades} \leq q^a \leq q^{need} \\ & \text{elif } s \geq 18 : q^a \leq q^{need} \\ \text{REJECT} & \text{otherwise} \end{cases} \\
&\text{if: } p^a = p^{worst} \\
&\quad response^a = \begin{cases} \text{ACCEPT} & \text{if } s < TT : q^{min}_{trades} \leq q^a \leq q^{need} \text{ and} \\ & (p^{offer}_{last} \text{ or } p^{offer}_{next} = p^{worst}) \\ & \text{elif } TT \leq s < 18 : q^{min}_{trades} \leq q^a \leq q^{need} \\ & \text{elif } s \geq 18 : q^a \leq q^{need} \\ \text{REJECT} & \text{otherwise} \end{cases}
\end{aligned} \tag{8}
$$

Here, TT denotes the time to transition from the second to the final part, and $TT = 20 - q^{need}$.

2 Evaluation

We tested KanbeAgent in 100 simulations against *SyncAgent*, *AdaptiveAgent*, and *BetterAgent*. The conditions of the simulation are shown in Table 1.

Table 1: The condition of simulations

Condition	Value
$n\_configs$	5
$n\_runs\_per\_world$	20
n_steps	100

The results are shown in Table 2. The results show that all scores of KanbeAgent are equal to or better than other agents.

Table 2: The test results of KanbeAgent

Agent Type	mean	min	Q1	median	Q3	max
KanbeAgent	1.06	0.89	1.02	1.05	1.10	1.24
SyncAgent	1.00	0.75	0.98	1.01	1.05	1.18
AdaptiveAgent	0.96	0.44	0.86	0.98	1.05	1.24
BetterAgent	0.96	0.40	0.85	0.97	1.05	1.24