

TimeWeightedAgent

Yuto Ishigami, Yuta Kawasaki
Nagoya Institute of Technology

June 24, 2026

1 Introduction

TimeWeightedAgent is a synchronous negotiation agent that improves upon AlmostEqualAgent, one of the winning agents in the SCML OneShot 2025 competition. In SCML OneShot 2026, shortfall penalties are, on average, five to ten times larger than disposal costs [1], making it crucial to secure sufficient quantities to mitigate penalty risks.

TimeWeightedAgent continuously evaluates opponents based on their transaction history and allocates target quantities based on these scores. Past observations indicate that negotiation opponents often change their strategies as the simulation progresses. Consequently, relying on past transaction history without considering when the transactions occurred prevents the agent from adapting to these strategic shifts. To address this, TimeWeightedAgent introduces a time-weighted scoring mechanism that emphasizes recent transactions, allowing it to adapt continuously to changes in opponents' behavior.

Furthermore, the agent realizes an adaptive strategy within each daily negotiation session by incorporating stepwise probabilistic pricing, which adjusts price offers according to elapsed time, and a dynamic acceptance threshold, which adjusts the compromise level based on remaining time. Meanwhile, for acceptance decisions, the agent comprehensively evaluates the power set of received offers and prioritizes quantity matching, thereby robustly minimizing its own penalties.

2 TimeWeightedAgent

2.1 Offer Strategy

As its offer strategy, TimeWeightedAgent calculates weights for each partner based on past transaction history to allocate proposed quantities, and uses stepwise probabilistic pricing based on time progression. We distinguish two time scales throughout this report. Let D be the total number of simulation days and d the current day index. We define the normalized day-level progress as $\tau = d/D$ ($0 \leq \tau < 1$). This variable is used only to weight transaction history

across simulation days. In contrast, $\rho_k \in [0, 1]$ denotes the normalized progress of the ongoing negotiation with partner k , as reported by the negotiation mechanism. Thus, ρ_k represents progress within a negotiation conducted on a single day, rather than progress across simulation days.

The conventional allocation strategy used by AlmostEqualAgent switches strategies at fixed points in the simulation, making it difficult to adapt continuously to recent changes in partners' behavior. TimeWeightedAgent addresses this issue by determining offer quantities and prices through the following steps.

1. Calculation of Reliability Score Based on Past Transaction History

For partner i , let the transaction history be

$$\mathcal{H}_i = \left\{ \left(\tau_{i,j}, Q_{i,j}^{\text{agreed}} \right) \right\}_{j=1}^{N_i}. \quad (1)$$

where N_i is the number of contracts concluded with partner i , $Q_{i,j}^{\text{agreed}}$ is the quantity agreed upon in the j -th contract, and $\tau_{i,j}$ is the normalized simulation-day progress at which that contract was concluded. Using this history, we calculate the time-weighted transaction score W_i with $\lambda = 2.0$:

$$W_i = \sum_{j=1}^{N_i} Q_{i,j}^{\text{agreed}} \exp(\lambda \tau_{i,j}). \quad (2)$$

A contract concluded later in the simulation therefore contributes more to W_i than an equally sized contract j concluded earlier.

2. Target Quantity Considering Safety Margin

On the buyer's side, if the required quantity is strictly set as the target, a shortfall may occur due to some negotiation failures. Therefore, TimeWeightedAgent targets a quantity multiplied by a safety factor exclusively for the buyer's side.

$$Q_{\text{adjusted}} = \lfloor 1.20 Q_{\text{need}} \rfloor \quad (3)$$

Here, Q_{need} is the quantity actually required by the agent at that point in time.

3. Allocation Based on Reliability Score

We prioritize partners with higher reliability scores W_i and sequentially allocate quantities until Q_{adjusted} is reached, while satisfying the quantity constraints of each negotiation.

4. Stepwise Probabilistic Pricing

To encourage concessions and improve the chance of securing the required quantity, the agent selects offer prices probabilistically according

to within-negotiation progress. Let $\mathcal{A}$ denote the set of ongoing negotiations. During counteroffers, we use their average normalized progress $\bar{\rho}$, defined as:

$$\bar{\rho} = \frac{1}{|\mathcal{A}|} \sum_{k \in \mathcal{A}} \rho_k, \quad (4)$$

so that the pricing decision is not determined by the progress of only one negotiation. We define P_{ideal} as the endpoint of the price range that is most favorable to TimeWeightedAgent: the maximum price when selling and the minimum price when buying. Conversely, P_{concede} is the opposite endpoint, which is most favorable to the partner. Using $\bar{\rho} = 0.50$ as a threshold, the probability $Pr(P_{\text{ideal}})$ of offering P_{ideal} changes stepwise as follows:

$$Pr(P_{\text{ideal}}) = \begin{cases} 0.50 & (\text{if } \bar{\rho} < 0.50). \\ 0.25 & (\text{otherwise}). \end{cases} \quad (5)$$

If P_{ideal} is not selected, the agent offers P_{concede} , the endpoint of the price range that is most favorable to the partner. This strategy enables the agent to pursue profit more actively in the early stages while reducing aggressive demands later and prioritizing agreement and quantity fulfillment.

2.2 Acceptance Strategy

Rather than evaluating each offer independently, TimeWeightedAgent comprehensively evaluates the power set of the received offers, that is, every possible combination of offers. For each combination, the agent computes an overall score based on quantity and price and compares it with a dynamic threshold that decreases as the negotiation progresses. It accepts the highest-scoring combination when that score exceeds the threshold. To reduce the risk of severe shortfall penalties, the agent also employs a fixed- α strategy that gives substantially greater importance to quantity matching than to price.

1. Calculation of the Price and Quantity Scores

We calculate a price score P and a quantity-matching score Q and normalize each to the range $[0, 1]$. For a buyer, P_{score} is the negative of the sum of P_{total} and P_{penalty} ; for a seller, it is P_{total} minus P_{penalty} . Here, P_{total} is the total transaction value of an offer combination, P_{penalty} is the estimated shortfall or disposal penalty, Q_{all} is the total offered quantity, and Q_{need} is the required quantity. The quantity score is normalized so that smaller differences between Q_{all} and Q_{need} receive higher scores. This evaluation therefore accounts for both price and the penalties caused by shortages or excess quantities.

2. Scoring for each contract combination

To compute the overall score R_{score} for each candidate combination, we introduce a weight parameter α that controls the importance of quantity

matching:

$$R_{\text{score}} = \alpha Q_{\text{norm}} + (1 - \alpha) P_{\text{norm}} \quad (6)$$

Based on preliminary experiments, we set $\alpha = 0.85$. This strongly emphasizes quantity matching and reduces the risk of accepting price-attractive combinations that would result in costly quantity mismatches.

3. Acceptance decision with dynamic threshold

For the acceptance decision, we use the most advanced negotiation time ρ_{max} , defined as:

$$\rho_{\text{max}} = \max_{k \in \mathcal{A}} \rho_k \quad (7)$$

We lower the hurdle according to the lack of time remaining by calculating the dynamic threshold β_ρ :

$$\beta_\rho = \beta_0(1 - 0.5\rho_{\text{max}}) \quad (8)$$

We select the combination with the highest R_{score} from the power set, and accept it if it exceeds β_ρ . We set the initial threshold to $\beta_0 = 0.6$. Consequently, the acceptance threshold is 0.6 at the beginning of the negotiation ($\rho_{\text{max}} = 0$) and decreases to 0.3 at the end ($\rho_{\text{max}} = 1$). This makes the agent selective early in the negotiation while relaxing its acceptance criterion later to reduce the risk of quantity shortfalls caused by negotiation failure.

3 Evaluation

In the experiments, we compare TimeWeightedAgent with AlmostEqualAgent, Rchan, and CostAverseAgent. These were selected as comparison targets because they are top-ranking agents in SCML OneShot 2025. The number of world environment configurations (n_configs) was set to 30. The experiments were conducted with three simulation lengths (n_steps): 50, 100, and 150 days. The score is used as the evaluation metric. The scores are shown in Ta-

Table 1: Experimental Results - Scores

Agent	n_steps=50	n_steps=100	n_steps=150
TimeWeightedAgent	1.0535	1.0724	1.0661
AlmostEqualAgent	1.0471	1.0671	1.0630
Rchan	1.0407	1.0658	1.0613
CostAverseAgent	1.0364	1.0591	1.0445

ble 1. TimeWeightedAgent achieved the highest average score among the compared agents under all three simulation lengths. The score differences between TimeWeightedAgent and AlmostEqualAgent were 0.0064, 0.0053, and 0.0031 for n_steps values of 50, 100, and 150, respectively. Thus, the gap tended to decrease as the number of simulation steps increased. One possible explanation

is that, in shorter simulations, TimeWeightedAgent can obtain an advantage by continuously applying time-weighted allocation based on transaction history from the early stages. In contrast, AlmostEqualAgent concentrates its offer quantities on partners with larger transaction histories during the later stages of a simulation. Longer simulations provide more opportunities for this strategy to operate, which may explain the smaller gap between the two agents.

4 Conclusion

TimeWeightedAgent uses time-weighted quantity allocation and stepwise probabilistic pricing when making offers, and evaluates all combinations of received offers when making acceptance decisions. To address limitations observed in previous agents, we introduced separate mechanisms for risk reduction and profit maximization. As a result, TimeWeightedAgent achieved the highest average score among the compared agents.

References

- [1] Y. Mohammed, A. Greenwald, K. Fujita, M. Klein, S. Morinaga, and S. Nakadai, *Supply Chain Management League (OneShot): Automated Negotiating Agents Competition*, SCML Organizing Committee, March 18, 2026.