

SHIMIJI^{SHI}JIMI: An Agent Submitted to the ANAC 2026 SCML Standard League

Rinon Asanuma
Tokyo University of Agriculture and Technology
`asanuma@katfujii.lab.tuat.ac.jp`

June 21, 2026

Abstract

This report details the comprehensive architectural design and strategic implementation of SHIMIJI^{SHI}JIMI, an autonomous negotiating agent developed for the Supply Chain Management League (SCML) Standard track at ANAC 2026. Operating within a volatile multi-tier supply chain, the agent addresses the core challenges of concurrent negotiation and inventory risk. By executing a strict time-based partner partitioning mechanism (the 7:2:1 ratio strategy) alongside an efficient quantity-sum based acceptance algorithm, SHIMIJI^{SHI}JIMI optimizes factory utilization, minimizes localized market starvation, and maximizes profitability under operational constraints without explicitly enumerating every partner subset.

1 Introduction

The Supply Chain Management League (SCML) Standard track presents a complex, dynamic environment where agents must simultaneously act as consumers (buying raw materials) and suppliers (selling processed products). The primary difficulty lies in managing multi-issue concurrent negotiations with multiple partners under time constraints, where failing to secure resources leads to factory idling, while over-purchasing leads to costly storage overheads.

To tackle these overlapping challenges, we propose SHIMIJI^{SHI}JIMI, a synchronous negotiation agent built upon the `StdSyncAgent` framework. The core philosophy of SHIMIJI^{SHI}JIMI is proactive resource scheduling coupled with rigorous balance-sheet management. Rather than aggressively competing only for immediate, same-day contracts, the agent shifts a structured portion of its negotiation capacity toward future steps, establishing a multi-layered buffer against supply chain disruptions.

2 The Design of ShimijimiShijimi

2.1 Demand and Supply Estimation

At each negotiation step, SHIMIJI^{SHI}JIMI evaluates its internal utility function by calculating the exact material needs for a target step t . The calculation integrates factory lines (N_{lines}), current inventory inputs, and already executed contracts. A specialized productivity modifier ($\alpha = 0.7$) scales the expected output to prevent over-optimistic planning:

$$\text{Production}_{\text{day}} = N_{\text{lines}} \times \alpha \tag{1}$$

For supply negotiations, the net deficit is derived by subtracting the current input inventory and secured supplies from this target production volume. For consumer (sales) negotiations, the agent ensures that total scheduled sales do not exceed factory capacities while dynamically attempting to liquidate accumulated inventory up to the physical constraints of the production lines.

2.2 Negotiation Choices and Pricing Strategy

The agent maintains distinct pricing profiles based on the urgency of the target contract. For immediate, current-step negotiations ($t = \text{current_step}$), the agent introduces negotiation margins to maximize profitability:

- **Suppliers:** Proposes an initial purchase price inflated by a `supplier_price_ratio` (1.2) relative to the issue’s minimum price boundary, bounded by the maximum issue price.
- **Consumers:** Proposes a sales price deflated by a `consumer_price_ratio` (0.7) relative to the issue’s maximum price boundary, bounded by the minimum issue price.

Conversely, for future-step negotiations, the agent bypasses the concession margin and immediately offers its `best_price` (the absolute boundary of the issue space) to prioritize contract certainty over marginal profit.

2.3 Concurrent Negotiation and Partner Partitioning

To prevent simultaneous, conflicting agreements with too many partners, SHIMIJI^{SHI}JIMI enforces a strict temporal partitioning strategy known as the 7:2:1 protocol. At the beginning of each step, the agent gathers all active, valid negotiating partners, shuffles them randomly to eliminate structural bias, and slices them into three distinct functional groups:

1. **Current Tier (70% of partners):** Assigned to negotiate exclusively for the immediate current step to satisfy pressing production requirements.
2. **Short-Term Future Tier (20% of partners):** Assigned to negotiate for contracts scheduled at $t = \text{current_step} + 1$.
3. **Mid-Term Future Tier (10% of partners):** Assigned to negotiate for contracts scheduled at $t = \text{current_step} + 2$.

For future tiers, the targeted quantity is scaled down by a `future_offer_ratio` (0.4) to prevent over-committing to highly uncertain future states. The allocated needs are then distributed as evenly as possible among the chosen partners using a deterministic remainder-distribution function.

2.4 Quantity-Sum Based Acceptance Strategy

When processing incoming multi-partner offers (`counter_all`), the agent divides responses into current-day and future-day offers. Future-day offers are accepted greedily if they do not cause the aggregated volume for that future step to exceed the calculated deficit.

For current-day offers, SHIMIJI^{SHI}JIMI avoids simple greedy heuristics, which often lead to unnecessary over-supply. However, rather than explicitly generating the complete power-set of all valid partners, the implemented strategy performs a dynamic-programming style subset-sum search over reachable total quantities. Let q_i be the quantity offered by the i -th valid partner and let D be the target deficit. The agent maintains layered dictionaries:

$$L_r[Q] = I, \quad (2)$$

where r denotes the number of selected partners, Q denotes a reachable total offered quantity, and I stores the corresponding tuple of partner indices. Initially, $L_0[0]$ is set to the empty tuple. For each valid partner i , the algorithm updates the layers in reverse order:

$$L_{r+1}[Q + q_i] \leftarrow I \cup \{i\}. \quad (3)$$

If the same total quantity can be produced by multiple subsets, the agent keeps a deterministic representative with the earliest index tuple. After all partners have been processed, the layers are compressed into a mapping from total quantity to a representative partner combination.

The final acceptance decision is made over total quantities rather than over explicitly enumerated subsets. The agent first searches for the smallest reachable over-supply total:

$$Q^+ = \min\{Q \mid Q \geq D\}, \quad (4)$$

and accepts it only when $Q^+ - D \leq \tau$, where the threshold is dynamically adjusted as $\tau = N_{\text{lines}} \times 0.1$. If no acceptable over-supply combination exists, the agent instead chooses the largest reachable under-supply total:

$$Q^- = \max\{Q \mid 0 < Q < D\}. \quad (5)$$

The partners represented by the selected total quantity are issued an `ACCEPT_OFFER` response. Unselected partners are rejected and countered with fresh offers aligned with the 7:2:1 distribution rules. As a last-resort fallback, if no under-supply combination is available, the implementation returns all current partners, ensuring that the response generation remains well-defined even in edge cases.

3 Risk Management

Supply chain volatility is mitigated through two primary mechanisms: internal accounting and error-resilient exception handling.

- **Preemptive Contract Bookkeeping:** The agent maintains live internal dictionaries (`_secured_sales` and `_secured_supplies`). Upon receiving an `on_negotiation_success` callback, the agent immediately logs the volume and step of the agreement. This ensures subsequent calculations within the same step operate on perfectly updated information.
- **Issue Boundary Safety Wrappers:** During intense simultaneous negotiations, negotiation mechanism infos (`nmi`) can become volatile or occasionally throw runtime exceptions if a partner disconnects mid-negotiation. SHIMIJI_{MI}SHIJI_{MI} implements a defensive wrapper (`_safe_get_issues`) that catches runtime anomalies and gracefully falls back to global market issues, preventing catastrophic agent failure.

4 Evaluation

Preliminary behavioral simulations indicate that the 7:2:1 partner distribution protocol provides significant structural stability. By securing roughly 40% of future demand one to two steps in advance, the agent avoids the drastic price spikes and component starvation common during sudden market contractions. Furthermore, the quantity-sum based combination matching reduces the "sawtooth" inventory patterns typical of greedy agents while avoiding the direct memory cost of materializing every partner subset.

5 Lessons and Suggestions

The current implementation improves scalability by replacing explicit power-set materialization with a layered reachable-quantity representation. This is closer to a subset-sum dynamic programming method: its practical cost depends on the number of distinct reachable total quantities rather than directly on 2^N partner subsets. Consequently, when many offers share similar quantities, the algorithm can merge many equivalent totals and substantially reduce memory pressure.

Nevertheless, the method is not cost-free. In dense cases where almost every subset produces a distinct quantity total, the number of reachable totals can still grow substantially. Future improvements should therefore consider bounding the dictionary size, pruning totals far above the acceptable threshold, or introducing approximate matching when the negotiation time budget becomes tight.

Conclusions

SHIMIJI_{MI}SHIJI_{MI} offers a highly reliable framework for the ANAC 2026 SCML Standard League. By structurally separating short-term operational fulfillment from mid-term strategic buffering, and optimizing acceptance through reachable-quantity based combinational matching, the agent successfully maintains profitable margins while minimizing inventory risks and reducing the overhead associated with brute-force subset enumeration.