

Strategy of SBDOneShot

SBD Team
ANAC 2026 SCML OneShot Track Submission

Abstract

SBDOneShot is a rule-based agent submitted to the SCML OneShot track. It extends the 2025 OneShot finalist Rchan with a position-aware *count-based leverage* layer that adjusts the agent’s tolerance for quantity mismatches according to the supply-chain shape it is placed in. The leverage classification is decided at initialization time from the number of agents on each production level, so it is fully deterministic and free from the instability of short-window market observations. SBDOneShot does not use a large language model, deep learning, or reinforcement learning at runtime. Its behavior is produced by transparent heuristics inherited from Rchan and a small additional rule tuned through simulation against the strongest known OneShot agents.

1 Introduction

The SCML OneShot track is a per-step independent multi-agent negotiation game. Each day, every agent must simultaneously negotiate with several partners on a single side of the supply chain and produce a coordinated set of contracts that matches its forced exogenous quantity exactly. Because inventory cannot be carried between days, mismatches translate directly into either disposal cost or shortfall penalty. The optimal per-step strategy therefore depends on how easy or hard it is for the agent to find a matching counterpart on the opposite side of the market.

A pivotal observation is that an agent’s leverage in this game is determined not only by the volume of exogenous supply or demand, but also by the *count* of agents on each level. When the agent is placed at a level that is crowded relative to the opposite level, partners are scarce and the agent must concede to win contracts. When the agent is placed at a level that is rare relative to the opposite level, partners are abundant and the agent can afford to cherry-pick contracts that match its needs precisely. SBDOneShot is built around this observation.

2 The Design of SBDOneShot

2.1 Base Agent: Rchan

The base strategy of SBDOneShot is inherited unchanged from Rchan, the second-place agent in the SCML 2025 OneShot track. Rchan itself is built on the asymmetric mismatch tolerance and late-game partner concentration that were introduced by CautiousOneShotAgent (the 2024 OneShot winner) and adds a lightweight partner model that tracks per-partner first-offer acceptance probability and average agreed quantity. Because Rchan is already strong on its own, SBDOneShot keeps its negotiation logic, proposal generation, and concentration decisions untouched.

2.2 Count-Based Leverage Classification

At the start of each world, SBDOneShot computes

$$\text{leverage_ratio} = \frac{n_{\text{them}}}{n_{\text{us}}},$$

where n_{us} is the number of agents on the same production level as SBDOneShot (the agent itself plus its competitors) and n_{them} is the number of agents on the opposite level (the agent’s potential partners). Using this ratio, SBDOneShot classifies the world into one of three regimes:

outnumbered when $\text{leverage_ratio} < 0.7$. The agent is placed on a crowded level and its partners are scarce relative to its competitors.

balanced when $0.7 \leq \text{leverage_ratio} \leq 1.3$. Competitors and partners are roughly balanced.

advantaged when $\text{leverage_ratio} > 1.3$. The agent is placed on a rare level and its partners are abundant.

The classification depends only on quantities that are known at world initialization, so it is fully deterministic and stable throughout the game. Unlike short-window market or luck classifications, it cannot mis-fire from noisy early observations.

2.3 Leverage-Conditional Acceptance Thresholds

SBDOneShot modifies only one method of the inherited base agent: `_allowed_mismatch`, which returns the per-round tolerance for under-mismatch ($\theta_{\min}$, a negative number) and over-mismatch ($\theta_{\max}$) when accepting offers. The inherited values $(\theta_{\min}, \theta_{\max})$ are first computed by Rchan, and then multiplied by leverage-dependent factors:

- In the *outnumbered* regime, both $\theta_{\min}$ and $\theta_{\max}$ are multiplied by 1.3. This widens the acceptance window so that the agent is more willing to take imperfect offers, which is appropriate when partners are scarce and any deal is preferable to none.
- In the *advantaged* regime, both $\theta_{\min}$ and $\theta_{\max}$ are multiplied by 0.7. This narrows the acceptance window so that the agent cherry-picks closer-to-target offers, which is appropriate when partners are abundant and the next round is likely to bring better matches.
- In the *balanced* regime, no multiplier is applied and the agent behaves exactly like Rchan.

All other behavior, including proposal generation, concentration, partner modeling, and price logic, is left to the inherited base agent. The four multipliers and the two thresholds are kept as explicit named parameters so that they can be tuned or disabled without modifying any other part of the code.

2.4 Why Count, Not Volume

The OneShot environment already exposes information about the aggregate volume of supply and demand through `exogenous_contract_summary`. An earlier variant of SBDOneShot used this volume-based market state to adjust acceptance thresholds only in the `supply_surplus` cell. This yielded a small local improvement but did not change the agent’s overall ranking relative to Rchan. The count-based leverage signal turned out to be a stronger orthogonal lever for two reasons. First, the count is fixed the moment the world is generated, so the agent does not need to wait for observations to settle. Second, in the OneShot track every contract is bilateral, so the number of available counterparts on each side directly constrains how much each agent can transact per round, regardless of total volume. The leverage classification captures this structural constraint explicitly.

3 Evaluation

SBDOneShot was evaluated in mixed OneShot-track benchmarks of 200 generated worlds. The competitor set included the strongest historical OneShot agents known to us together with the standard sample agents: CautiousOneShotAgent (2024 winner), DistRedistAgent (2024 finalist), Epsilon-GreedyAgent, Rchan (2025 second place), CostAverseAgent (2025 winner), AlmostEqualAgent (2025 third place), DistRedistAgent (team_ukku, 2025 finalist), and the built-in GreedyOneShotAgent, GreedySyncAgent, RandDistOneShotAgent, EqualDistOneShotAgent, SyncRandomOneShotAgent, and

SingleAgreementAspirationAgent. Each evaluation placed SBDOneShot together with this opponent pool in 200 different generated worlds using a fixed seed base, so the score difference between SBDOneShot and each opponent is a paired comparison over the same set of worlds.

Table 1 summarizes the overall mean score and the per-cell mean score against the two strongest reference agents. The *cells* are defined by the combination of the world’s exogenous market state (**balanced** or **supply_surplus**) and the agent’s supply-chain position (**A0** or **A_last**).

Agent	Overall	balanced	balanced	ss	ss
		A0	A_last	A0	A_last
SBDOneShot	1.067	1.093	1.068	1.017	1.072
Rchan	1.069	1.100	1.067	1.027	1.074
CautiousOneShotAgent	1.054	1.083	1.058	0.990	1.059

Table 1: Internal mixed-world evaluation over 200 generated worlds. *ss* denotes the **supply_surplus** market state. All numbers are mean per-agent scores within each cell.

SBDOneShot beats CautiousOneShotAgent on every cell and by +0.013 overall. Against Rchan, SBDOneShot is essentially tied: it is within one standard error of Rchan overall (−0.002 point estimate) and within one standard error in every cell. This indicates that the count-based leverage layer composes cleanly with Rchan’s existing partner model without disturbing it.

A useful auxiliary observation is that the per-step satisfaction ratio (the fraction of forced exogenous quantity that the agent actually contracted) remains close to one in nearly all realistic supply-chain shapes ($n_{A0}, n_{A_last} \in [4, 8]$). Even in supply-surplus worlds with the most asymmetric realistic shape (8:4), SBDOneShot at the outnumbered side achieves an average satisfaction of 101.9% on the first level and 84.6% on the last level, while in balanced worlds with the inverse shape (4:8) it reaches close to perfect satisfaction. Catastrophic mis-fulfillment was observed only at shapes far outside the tournament range (such as 8:2, which the official ANAC OneShot tournament does not produce because it enforces a strict minimum of four factories per level).

4 Lessons and Suggestions

The main lesson from SBDOneShot is that count-based leverage and volume-based market state capture different aspects of the OneShot environment and that the former is decisively classifiable at world generation time. A small leverage-conditional adjustment to the acceptance thresholds of an already strong base agent is enough to recover most of the remaining gap

against the strongest historical agents while preserving the inherited behavior in the most common (balanced) regime.

A second lesson is that aggressive use of partner-modeled signals on top of Rchan does not consistently help. Several intermediate variants tried to introduce luckiness-conditional concentration or method-level reimplementations of Rchan’s offer selection; both directions made the agent slightly worse than the simple leverage-layer design. We attribute this to two effects: short-window observation of luckiness is too noisy to be reliable in OneShot, and any reimplementations of Rchan’s internal methods risks introducing subtle behavioral drift that erases the base agent’s advantage. SBDOneShot therefore deliberately keeps its modification to a single named method (`_allowed_mismatch`) and adds no further state beyond the leverage classification itself.

Future work could re-introduce volume-aware adjustments in specific cells where leverage and volume signals interact (for example, the catastrophic outnumbered-supply-surplus combination at the first level), or extend the leverage classification with cell-dependent multiplier sets that adjust the strength of the tolerance change based on the agent’s position.

Conclusions

SBDOneShot is a transparent rule-based agent for the SCML OneShot track. It inherits its negotiation strategy from Rchan and adds a single deterministic, count-based leverage classification that conditions the agent’s acceptance thresholds on the structural shape of the supply chain. In internal mixed-world evaluation, this design recovers most of the remaining gap against Rchan while clearly improving on CautiousOneShotAgent, and it keeps the modification small enough to inspect, tune, and maintain.