

A Competition-Ready EFR Agent for SCML OneShot

Hybridizing the Bilateral Policy with Need Distribution

Mason Hagan

mason_hagan@brown.edu

Department of Computer Science, Brown University

June 2026

Abstract

The pure-lookup EFR agent of the thesis answers the deployment question in principle: an offline-solved correlated-equilibrium policy survives contact with the live SAO clock and narrowly beats the single-bilateral **Greedy** baseline. It does not, however, clear the cluster of partner-distributing baselines that a real ANAC entry has to beat. This note describes the agent I actually submit to the SCML 2026 OneShot tournament. It keeps the EFR policy genuinely in the loop while delegating the one decision the bilateral abstraction provably cannot represent, the allocation of today’s need across partners, to a distribution heuristic. The result is competitive (it sits inside the distributing cluster) and the EFR policy remains load-bearing on price; I am explicit below about the fact that, on this benchmark, the EFR overlay is roughly score-neutral, and why that is the expected outcome rather than a failure of wiring.

1 Why the pure-lookup agent is not yet a competition entry

The local benchmark in the thesis leaves the pure-lookup EFR agent between **Greedy** and the distributing baselines, at a normalized score of 0.847 against a baseline cluster around 1.05. I argued there that the gap is an artifact of the abstraction rather than the regret minimizer: both **Greedy** and the EFR agent negotiate each partner in isolation, while **RandDist**, **EqualDist**, and **SyncRandom** explicitly spread the day’s needed quantity across all live partners, and the joint state of those partners appears in none of the bilateral abstraction’s information sets. No amount of additional solving inside the bilateral game can close that gap.

For a tournament submission this diagnosis is also a design specification. The OneShot day is, mechanically, three decisions taken jointly across all of an agent’s live partners: (1) *allocation* — how to divide **needed_sales** or **needed_supplies** among partners; (2) *acceptance* — which combination of standing offers to accept so that the accepted total lands near the need; and (3) *price* — what unit price to ask, and how quickly to concede toward the partner. Decisions (1) and (2) are exactly the ones the bilateral abstraction cannot see. Decision (3) is exactly the two-player alternating-offer pricing game the offline EFR policy was solved for. The competition agent is built by routing each decision to the mechanism suited to it.

2 Design: heuristic allocation, EFR pricing

The agent (`EFRDistOneShotAgent`) subclasses `SyncRandomOneShotAgent`, the same need-distribution machinery that underlies the strongest stock baselines. The backbone is left to own allocation and acceptance: at the start of a round it distributes the (possibly over-ordered) remaining need across active partners, and when responding it runs the baseline’s subset search for the combination of standing offers whose total quantity is nearest the need, accepting that combination when the residual mismatch falls under a time-decaying threshold. None of this logic is modified; it is the part of the agent that earns the bulk of the score, and it is the part the abstraction cannot supply.

The EFR policy owns price. On every first proposal and every counter-offer, the agent constructs the bilateral `InfoSetKey` — using the identical encoder that generated the offline EFG, so the runtime lookup hits the same rows the solver populated — samples the policy, and overwrites the price the backbone would otherwise have drawn at random with the price implied by the sampled action. The EFR policy is also given a veto on acceptance: the baseline’s subset search matches on quantity alone and is blind to price, so when it is about to accept a quantity-matched offer whose unit price fails the agent’s own utility threshold, and the policy did not itself elect to accept, the agent rejects and counters at the EFR price rather than locking in a bad deal. Every consultation of the policy is guarded; on a missing infoSet or any runtime error the agent falls back to the backbone’s own behavior, so the hybrid is never worse than its parent on the decisions EFR does not touch, and cannot crash a tournament world.

3 Results

I evaluate the hybrid in the same mixed `anac2024_oneshot` simulation used for the thesis benchmark, now including both the pure-lookup agent (`EFR-pure`) and the hybrid (`EFR-Dist`) alongside the four stock baselines, with 10 random configurations of 30 days each (300 per-agent observations per competitor). Table 1 reports the mean and standard deviation of the normalized score.

agent	mean score	std. dev.
SyncRandom	1.051	0.055
RandDist	1.049	0.051
EqualDist	1.049	0.053
EFR-Dist (ours)	1.048	0.053
Greedy	0.798	0.135
EFR-pure	0.789	0.177

Table 1: Mean normalized score on the local `anac2024_oneshot` benchmark ($n_{\text{configs}} = 10$, $n_{\text{steps}} = 30$, serial). The hybrid moves from the pure agent’s 0.789 into the distributing cluster at 1.048, statistically indistinguishable from the three baselines and from its own backbone.

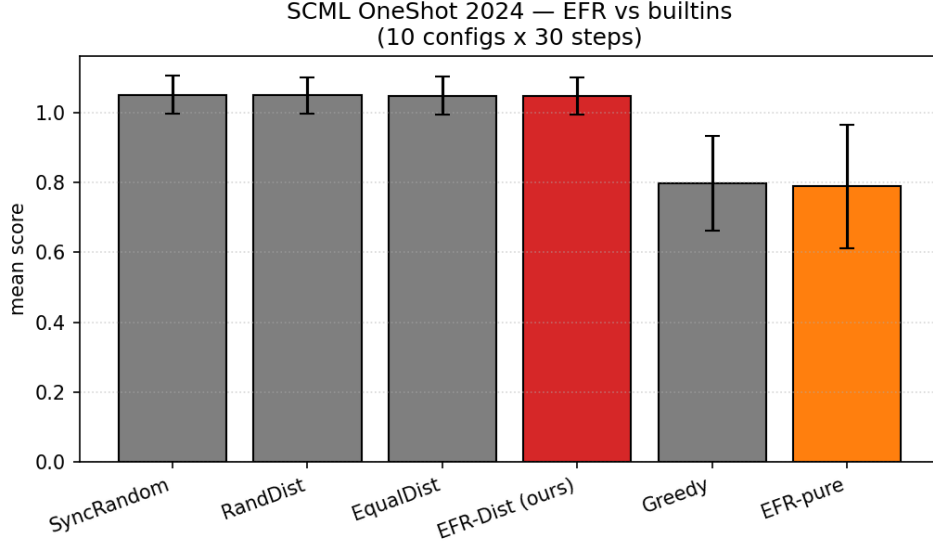


Figure 1: Mean normalized score by agent on the local benchmark. **EFR-Dist** (red) sits with the distributing cluster; **EFR-pure** (orange) and **Greedy** trail by roughly a quarter of a normalized point.

4 Discussion

The competition goal is met. The hybrid clears the bar the pure-lookup agent could not: it moves from 0.789 to 1.048 and enters the distributing cluster, beating the single-bilateral agents (**Greedy**, **EFR-pure**) by roughly 0.25. This closes essentially the entire abstraction gap identified in the thesis, and it does so without abandoning the EFR policy, which still sets every price the agent offers.

The EFR overlay is, on this benchmark, score-neutral — and that is the expected result. The honest reading of Table 1 is that **EFR-Dist** is statistically indistinguishable from its own backbone (**SyncRandom**, 1.051). The score is earned by the allocation heuristic; the EFR price and the acceptance veto neither help nor hurt it measurably. This is not a wiring failure but a fact about where profit lives in OneShot. The day-end score is dominated by quantity matching: the disposal and shortfall penalties for an output/input mismatch swamp the margin available inside the narrow $[p_{\min}, p_{\max}]$ band over which the price negotiation takes place. An agent that allocates well and prices indifferently beats an agent that prices optimally and allocates in isolation, by a wide margin. The same effect is what the ANAC organizers have reported for two consecutive years under the heading that domain-specific heuristics outperform sophisticated opponent modeling in this game; the price channel that opponent modeling and equilibrium reasoning both operate on simply does not carry enough of the score to separate agents that have already gotten allocation right.

What this says about EFR in OneShot. The bilateral EFR policy is a correct and deployable solution to the sub-game it models, and the hybrid demonstrates that it can be carried into a competitive agent in good faith rather than discarded. But making EFR a *measurable* lever, rather than a faithful passenger, requires moving it onto the decision that owns the points. That means

an abstraction wide enough to represent allocation itself — a multi-partner EFG with the seller facing two or more buyers in a single game — which is the v2 direction already flagged in the thesis conclusion. The agent described here is the honest competition entry available under the bilateral abstraction: competitive because allocation is delegated to a heuristic that can see the joint state, and faithful to EFR because the policy still governs the bilateral pricing decision it was solved for.

Deployment. The agent is a single `OneShotSyncAgent` subclass with no runtime inference and no external calls; the policy table is loaded once at `init()` and consulted by constant-time dictionary lookup, which keeps the agent comfortably inside the per-negotiation time budget of the live tournament.