Composing Efficient, Robust Tests for Policy Selection

Dustin Morrill, Thomas J. Walsh, Daniel Hernandez, Peter R. Wurman, Peter Stone

Sony AI
https://ai.sony/joinus/
Dustin.Morrill@sony.com

A real AI deployment problem: ***Hundreds of candidate deployment policies, dozens of test cases, but you can only run a few test cases…***

Robust Test Construction

Test case t ∈ $\mathcal{T}$ measures a particular skill

Policies* π

* At test construction time, only a small set of **tuning policies** set is available.

Test – <τ, σ> Collection of test cases τ and weights σ

Test score – A weighted average of the test case results.

Problem: Construct an efficient test <τ, σ_τ> by selecting a small number of test cases $\tau \subset \mathcal{T}$ and test case weights $\hat{\sigma} \in \triangle^{|\tau|}$ to approximate a full test, $\langle \mathcal{T}, \sigma \in \triangle^{|\mathcal{T}|} \rangle$ despite **uncertainty** over future policies to evaluate and the future test case weights.

Test construction example

Uncertainty over future policies (e.g. a higher proportion of aggressive agents)

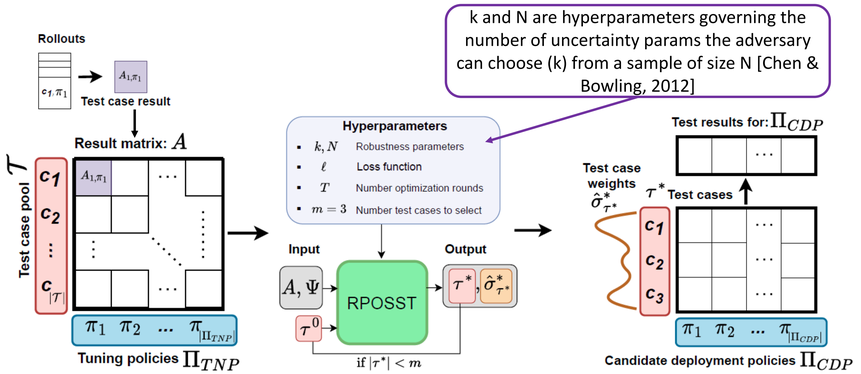
Uncertainty over target distribution (e.g. different emphasis on off-course behavior)

Target (σ)	policy / case	#1	#2	#3	#4	#5	…	…
?	C1	0	+1	+1	+1	+1	?	?
?	C2	-1	0	+1	+1	+1	?	?
?	C3	-1	-1	0	+1	+1	?	?
?	C4	-1	-1	-1	0	+1	?	?
?	C5	-1	-1	-1	-1	0	?	?

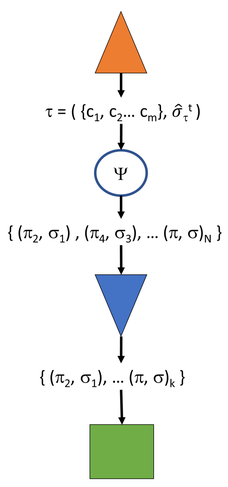
Possible Test	max error	size
Uniform	0	5
Strongest (C5)	1.4	1
Middle (C3)	0.6	1
[0, 0.5, 0, 0.5, 0]	0.2	2
[0.27, 0, 0.47, 0, 0.27]	0.07	3

RPOSST Selections

Constructing a Robust Test with RPOSST



RPOSST$_{SEQ}$ Game



Protagonist chooses a test <τ^t, $\hat{\sigma}_\tau^t$> by learning weights $\hat{\sigma}_\tau^t$ with *regret matching*$^+$ [Tammelin et al., 2015] across rounds 1,…,t and selecting test cases τ^t with Successive Rejects [Audibert et al., 2010]

$\tau = (\{c_1, c_2 \ldots c_m\}, \hat{\sigma}_\tau^t)$

Ψ

Nature draws N samples

$\{(\pi_2, \sigma_1), (\pi_4, \sigma_3), \ldots (\pi, \sigma)_N\}$

Antagonist chooses k param tuples as best response to τ and σ_τ^t

$\{(\pi_2, \sigma_1), \ldots (\pi, \sigma)_k\}$

Protagonist payout is $-1/k \sum_i^k \ell(\hat{\sigma}_\tau^t; \pi_2, \sigma_1)$
Antagonist receives the negation

Deterministic **CVaR(η) RPOSST$_{SEQ}$** used in the experiments fixes the ratio of η = k/N (e.g. 1%). The adversary selects pairs until the cumulative probability reaches η.

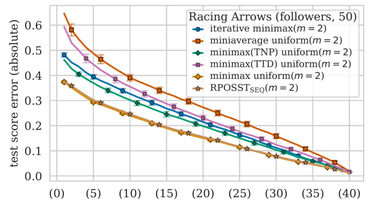
Theorem 4.1. *After $T' \sim \text{Unif}(\{1, \ldots, T_1\})$, $T_1 > 0$, rounds of its optimization game, Algorithm 1 selects an m-tuple of test cases, τ^* and weights $\hat{\sigma}_{\tau^*}^{T'} \in \triangle^m$ that, with probability $(1-p)(1-q)(1-\alpha)$, $p, q, \alpha > 0$, are $\frac{\varepsilon}{q}$-optimal for Equation (2), where $\varepsilon = \mathcal{O}\left(\sqrt{\frac{1}{T_1}m} + \sqrt{\frac{1}{T_1}\log(1/p)}\right)$ and $\alpha = \mathcal{O}\left(e^{-T_2}\right)$.*

Corollary 4.2. *Assume that $\Psi \in \triangle^d$ for some finite $d \geq 1$. After T rounds of the CVaR(η) RPOSST$_{SEQ}$ optimization game (using regret matching$^+$), τ^* and $\sigma_{\tau^*}^T$ are ε-optimal for Equation (2) under the η-fractile CVaR robustness measure, where $\varepsilon = \mathcal{O}\left(\sqrt{\frac{1}{T}m}\right)$.*

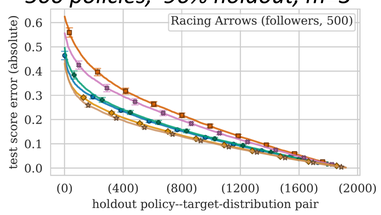
Empirical Results

Racing Arrows (one-shot simultaneous move game)

50 policies, 20% holdout, m=2



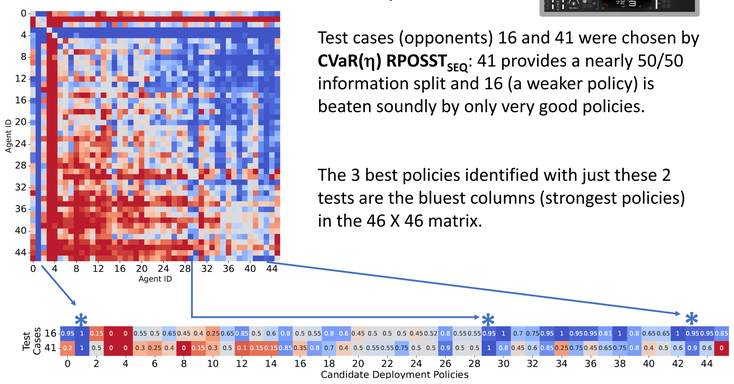
500 policies, 96% holdout, m=3



Poker (no-limit)

Gran Turismo 7 1v1

Inside the Gran Turismo 7 1v1 experiment



Test cases (opponents) 16 and 41 were chosen by **CVaR(η) RPOSST$_{SEQ}$**: 41 provides a nearly 50/50 information split and 16 (a weaker policy) is beaten soundly by only very good policies.

The 3 best policies identified with just these 2 tests are the bluest columns (strongest policies) in the 46 X 46 matrix.