Policy-Driven World Model Adaptation for Robust Offline Model-based Reinforcement Learning

Jiayu Chen[1,2*] Le Xu[3*] Aravind Venugopal[4] Jeff Schneider[4]

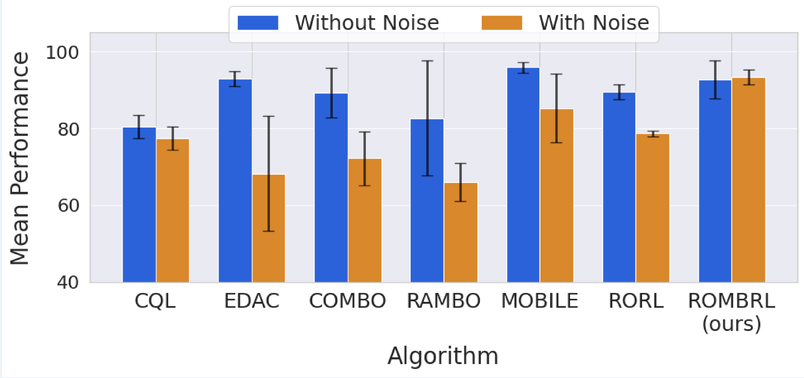
[1]The University of Hong Kong [2]INFIFORCE Intelligent Technology Co., Ltd. [3]Tsinghua University [4]Carnegie Mellon University (* equal contribution)

The Problem

Offline model-based RL learns a **world model** from a static dataset, then optimizes a policy inside it. Two flaws:

- **Objective mismatch:** the model is trained by likelihood, *not* for effective policy learning (two separate stages).
- **Fragile at deployment:** tiny dynamics noise collapses SOTA policies.

Motivation: SOTA policies break under 5% noise



Add zero-mean Gaussian noise (std = 5% of the state change) at deployment. Every baseline drops sharply — **ROMBRL barely moves**.

Our Idea: a robust max min objective

Train the policy to maximize its **worst-case** return over an uncertainty set of world models:

$$\max_{\theta} J(\theta, \phi') \text{ s.t. } \phi' \in \arg\min_{\phi \in \Phi} J(\theta, \phi)$$

- Policy π_θ **maximizes** return; world model P_ϕ is updated **adversarially** to minimize it — the *opposite* of online MBRL (conservatism is key offline).
- $\Phi = \{\phi : \mathbb{E}_{\mathcal{D}_\mu} D_{\mathrm{KL}}(P_{\tilde{\phi}} \| P_\phi) \le \epsilon\}$: plausible-but-worst dynamics around the MLE model.

Theory & Efficient Implementation

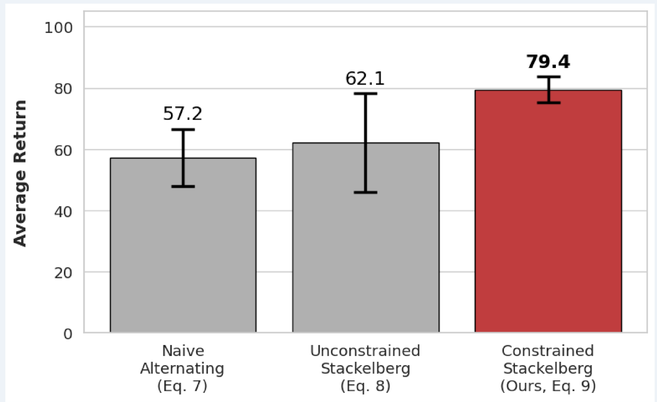
- **Suboptimality guarantee:** with prob. $1-\delta$,
  $J(\theta^*, \phi^*) - J(\hat{\theta}, \phi^*) \le \frac{\sqrt{C}}{(1-\gamma)^2}\sqrt{4\epsilon + \tilde{\mathcal{O}}(N^{-1/2})}$.
- Explicit ϵ for tabular & diagonal-Gaussian (deep NN) world models.
- **Fisher-information** approximation of 2nd-order terms + **Woodbury identity** ⇒ cost *linear* in #params.
- **Gradient-mask** mechanism for data-efficient off-policy training.

Method: a Stackelberg game

We model policy & model as a **leader–follower Stackelberg game** (**first use in offline MBRL**) — capturing that the worst-case model is an *implicit function* of the policy, which alternating updates (RAMBO) ignore.

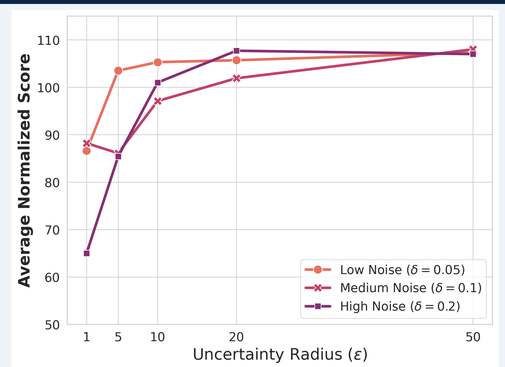
Solved with **constrained Stackelberg learning dynamics** + primal-dual updates on λ; converges to a local Stackelberg equilibrium under $\eta_\phi \gg \eta_\lambda \gg \eta_\theta$.

Ablation: why the *constrained* game matters



Anticipating **model params** alone helps little; anticipating the **uncertainty-set boundary** (λ) drives the gain.

Sensitivity: uncertainty radius ϵ



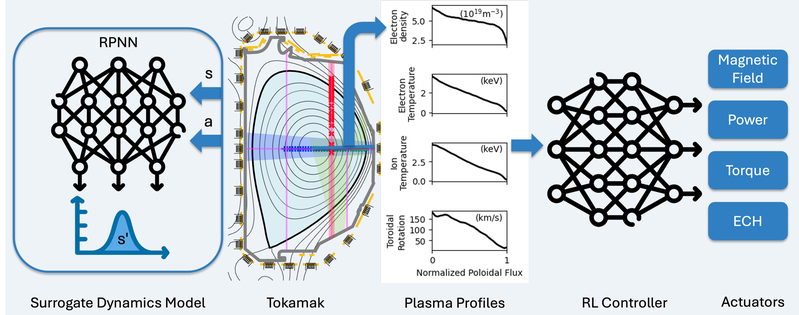
Robust across a **wide range of** ϵ; match ϵ to the disturbance magnitude for best results.

Result: near-zero performance drop

Method	Clean	Noisy	Drop↓
ROMBRL (ours)	92.8	**93.4**	**-0.6%**
MOBILE	**95.9**	85.3	11.1%
RORL	89.5	78.7	12.1%
COMBO	89.3	72.2	19.1%
RAMBO	82.7	66.0	20.2%
EDAC	93.0	68.3	26.6%

On 12 noisy D4RL MuJoCo tasks ROMBRL is **1st on 7, 2nd on 4** (avg 77.7 vs. 70.7 next best; all Cohen's $d \ge 2$).

Real-world case: Tokamak fusion control



Highly **stochastic** plasma-control tasks (DIII-D data). ROMBRL is **1st on 2 of 3** targets with the best average return and **lowest variance**; RAMBO/MOBILE/BAMCTS collapse.

Avg. return -47.1 vs. -68.3 (CQL) $\gg$ -155 to -168 (RAMBO/MOBILE/BAMCTS).

Robust to diverse perturbations (RWRL)

	ROMBRL	MOBILE	RAMBO	RORL
Avg. score	**51.9**	45.7	46.2	39.3
Perf. drop↓	**25.4%**	31.6%	30.0%	37.6%

Best on all 5 perturbation types (sensor dropout/stuck, mass, friction, damping) — robustness generalizes **beyond Gaussian noise**.

Takeaways

- One **unified max min objective** adapts world model *with* the policy.
- **Stackelberg dynamics** give convergence + strong robustness.
- SOTA on **15 noisy tasks**, with theory & linear-cost implementation.

Code: github.com/Agentic-Intelligence-Lab/ROMBRL

Contact: jiayuc@hku.hk