Learning in Continuous State-Space MDPs for Network Inventory Management

Hansheng Jiang[1*] **Shunan Jiang**[2] **Zuo-Jun Max Shen**[3]

[1]University of Toronto [2]UC Berkeley [3]University of Hong Kong

Motivation

Urban mobility platforms (e.g. vehicle sharing) must **dynamically reposition** a fleet across a network to meet **unknown, spatially-correlated demand**.

This is an **infinite-horizon, average-cost MDP** with a **multi-dimensional continuous state space**.

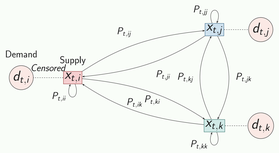
Key challenges:

Continuous, high-dimensional state/action space

Censored feedback — true demand unobserved beyond available inventory

Correlated, state-dependent transitions

No value function convexity — standard inventory control tools fail



Model: Network Inventory MDP

State: Inventory distribution $x_t = (x_{t,1}, \dots, x_{t,n}) \in \Delta_{n-1}$

Action: Target repositioning level $y_t \in \Delta_{n-1}$

Transition:

$$x_{t+1} = (y_t - d_t)^+ + P_t^\top \min(y_t, d_t)$$

Single-period cost:

$$C(x_t, y_t, d_t, P_t) = \underbrace{M(y_t - x_t)}_{\text{repositioning}} + \underbrace{L(y_t, d_t, P_t)}_{\text{lost sales}}$$

$M(\cdot)$: min-cost network flow for repositioning

$L(\cdot)$: penalty for unmet demand at each location

d_t, P_t i.i.d., distribution **unknown** to learner

Objective: Minimize long-run average cost

$$v^\pi(x) = \limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_\pi\big[C(x_t, y_t, d_t, P_t) \mid x_1 = x\big]$$

Benchmark & Regret

Benchmark: Best base-stock policy $S^* \in \operatorname{argmin}_{S\in\Delta_{n-1}} \lambda^S$

The loss λ^S is the long-run average cost under policy S, independent of starting state.

Censored feedback: True lost sales cost is unobservable. We use a *modified cost*:

$$\widetilde{C}_t = M(y_t - x_t) - \sum_{i,j} l_{ij} P_{t,ij} \min(d_{t,i}, y_{t,i})$$

which is fully **observable** and does not change optimal policy ranking.

Regret:

$$\mathrm{Regret}(T) = \mathbb{E} \sum_{t=1}^{T} \widetilde{C}_t^{\mathrm{ALG}} - \min_{S\in\Delta_{n-1}} \sum_{t=1}^{T} \mathbb{E}\, \widetilde{C}_t^{S}$$

Key Insight: Lipschitz Continuity

Lemma (Lipschitz Property). The long-run average cost λ^S is η-Lipschitz in the base-stock level:

$$|\lambda^S - \lambda^{S'}| \le \eta \|S - S'\|_1, \quad \eta = \max_{ij} l_{ij} + 6 \max_{ij} c_{ij}$$

Unlike prior work, we do **not** need convexity of the cost function.

This allows us to **discretize the policy space** via δ-covering of the simplex Δ_{n-1}.

We then cast the problem as a **Lipschitz bandit** problem.

Covering number:

$$N(\delta,\ \Delta_{n-1},\ \|\cdot\|_1) = O\left(\delta^{-(n-1)}\right)$$

Custom Concentration Inequalities

Standard i.i.d. bounds fail because MDP costs are **correlated**. We derive:

Lemma. For any base-stock policy S, starting state x_1, and N consecutive periods, w.p. $1-\epsilon$:

$$\left| \frac{1}{N} \sum_{t=1}^{N} \widetilde{C}_t^S - \lambda^S \right| \le \frac{O(1)}{N} + O(1)\sqrt{\frac{\log(1/\epsilon)}{N}}$$

Decompose via **loss–bias relation**

Construct **martingale difference sequence** from bias terms

Apply **Azuma–Hoeffding** inequality

Algorithm: LipBR

Lipschitz Bandits-based Repositioning — an epoch-based UCB algorithm over a discretized policy space.

Input: δ-cover $\mathcal{A}_\delta = \{S_1, \dots, S_K\}$, scale H
for each epoch $i = 1, 2, \dots$ **do**
  Pick arm $k^{(i)} \in \operatorname{argmax}_j \mathrm{UB}_j$
  for $j = 1, \dots, N_{k^{(i)}}$ steps **do**
    Reposition to $S_{k^{(i)}}$
    Observe censored demand
    Compute pseudo modified cost $\widetilde{C}'_t$
  end for
  Update UCB: $\mathrm{UB}_k = \bar{\mu}_k + H\sqrt{\frac{\log T}{\tau_k}}$
  Update memory point $m_k \leftarrow x_t$
  Double epoch length $N_k \leftarrow 2N_k$
end for

Key Design Elements

1) Policy Space Discretization. Each arm S_k in the δ-cover represents a candidate base-stock policy. The covering satisfies:

$$\Delta_{n-1} \subseteq \bigcup_{k=1}^{K} B(S_k,\ \delta,\ \|\cdot\|_1)$$

By the Lipschitz property, discretization error is controlled by $O(\delta)$.

2) Memory Points & Epoch Structure.

Concentration bounds require **consecutive** observations under the same policy.

When switching back to arm k, we use a **memory point** m_k (the state when k was last used) to simulate consecutive execution.

Pseudo modified cost:

$$\widetilde{C}'_t = \begin{cases} \widetilde{L}(S_k, d_t, P_t) + M(S_k - m_k) & \text{if switch} \\ \widetilde{C}_t & \text{otherwise} \end{cases}$$

Epoch lengths **double** to tighten confidence bounds.

Stitching cost $\le M(m_k - x_t)$ per switch.

Main Results

Theorem (Regret Upper Bound).

With $\delta = (\log T/T)^{1/(n+1)}$, the LipBR algorithm achieves:

$$\mathrm{Regret}(T) \le C_1\, T^{\frac{n}{n+1}} (\log T)^{\frac{1}{n+1}}$$

where C_1 depends on cost parameters, not on T.

Theorem (Matching Lower Bound).

For any online algorithm under censored feedback, $\exists$ an instance s.t.:

$$\inf_{\mathrm{ALG}} \sup_{\mathrm{inst.}} \mathbb{E}[\mathrm{Regret}(T)] \ge c\, T^{\frac{n}{n+1}}$$

$\Rightarrow$ LipBR is **minimax optimal** up to log factors.

Recovers $\widetilde{O}(\sqrt{T})$ for single location ($n = 1$)

Captures the **curse of dimensionality**: regret grows with n

First sublinear regret for multi-location inventory with censored demand and no convexity

Numerical Experiments

n	Policy	T=1000	T=2000	T=3000
2	LipBR	**6.322**	**6.363**	**6.409**
	NoRepo	7.230	7.289	7.306
	Uniform	6.473	6.487	6.506
3	LipBR	**10.533**	**10.593**	**10.537**
	NoRepo	12.041	11.976	11.925
	Uniform	11.500	11.439	11.398
4	LipBR	**15.240**	**15.413**	**15.255**
	NoRepo	16.787	16.883	16.723
	Uniform	16.355	16.475	16.334

LipBR achieves ~**11%** reduction vs. NoRepo, ~**5%** vs. Uniform

Gains grow with network size n

Summary of Contributions

1. **Lipschitz framework:** Replace convexity with Lipschitz continuity to enable learning in continuous-state, non-convex MDPs
2. **Matching lower bound:** $\Omega(T^{n/(n+1)})$ regret lower bound reveals an inherent **curse of dimensionality** in network size n
3. **Minimax-optimal algorithm:** LipBR achieves $\widetilde{O}(T^{n/(n+1)})$ regret, tight up to log factors

Code: `github.com/hanshengjiang/Lipschitz-NetworkInventory`

Correspondence: Hansheng Jiang hansheng.jiang@utoronto.ca
The 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026 Proceedings of Machine Learning Research