Dynamic Inventory Repositioning in On-Demand Rental Networks

Daniel Jiang[4]
Joint work with Saif Benjaafar[2], Xiang Li[3], Xiaobo Li[1]

Industrial Systems Engineering and Management, National University of Singapore[1]
Industrial & Systems Engineering, University of Minnesota[2]
Target Corporation[3]
Industrial Engineering, University of Pittsburgh[4]

2020 INFORMS Annual Meeting

Outline

Bicycle Sharing

Bicycle Repositioning

Car Rental

Car Repositioning

Shipping Containers

Outline

Problem Description

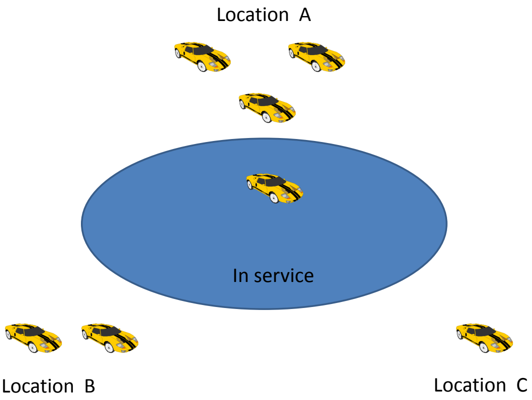
- The rental firm has a stock of N units of a single product distributed in several locations.
- The rental firm faces stochastic demands in different locations.
- The unsatisfied demand is lost.
- Inventory at locations cannot be replenished using an external source.
- The firm can reposition the inventory before demand realization.
- The objective is to minimize the total discounted lost sales cost and repositioning cost.
- Rented units can be returned to different locations than its origin.
- Rented units can be "in-service" or "ongoing" and are not assumed to be returned after one period.

The Sequence of Events

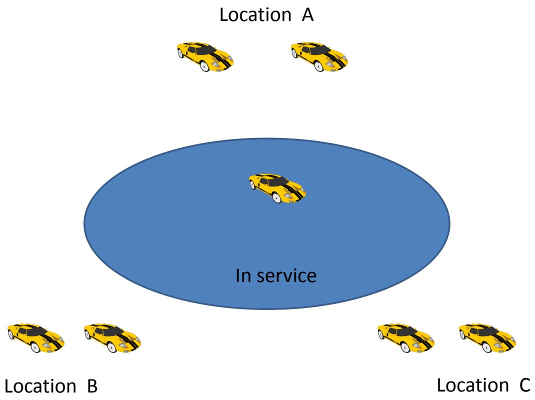
In each period, the sequence of events is as follows:

1. The current inventory level $\boldsymbol{x}_t = (x_{t,1}, \ldots, x_{t,n})$ and the current ongoing rentals $\boldsymbol{\gamma}_t = (\gamma_{t,1}, \ldots, \gamma_{t,n})$ are reviewed.
2. A decision on inventory repositioning is made, with $\boldsymbol{y}_t = (y_{t,1}, \ldots, y_{t,n})$ being the new inventory.
3. The repositioning cost $C(\boldsymbol{y} - \boldsymbol{x})$ is incurred.
4. The random demand $\boldsymbol{d}_t$ at all locations is realized.
5. The rented units enter service; demand that cannot be satisfied at location i incurs a lost sale cost l_i.
6. A random fraction of the in-service units returns to locations.

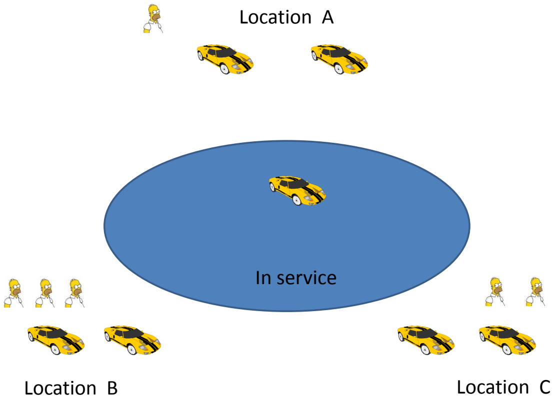
Problem Dynamics



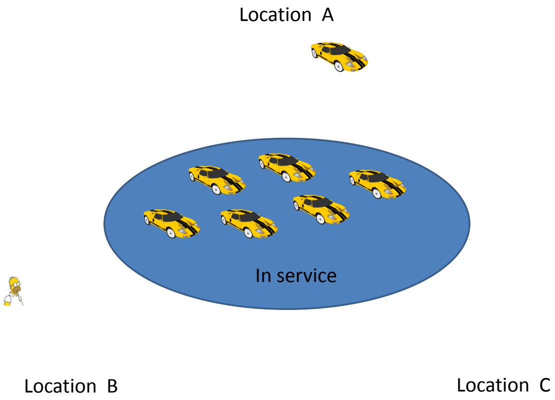
Problem Dynamics



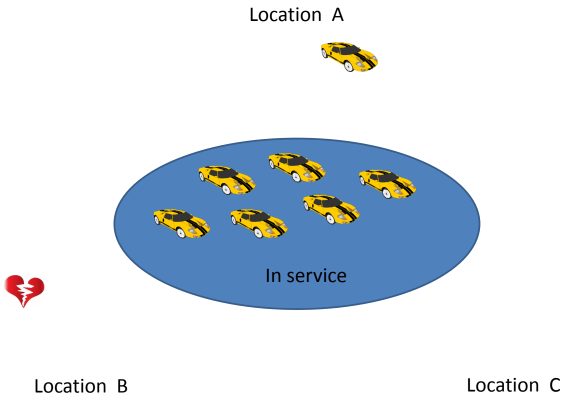
Problem Dynamics



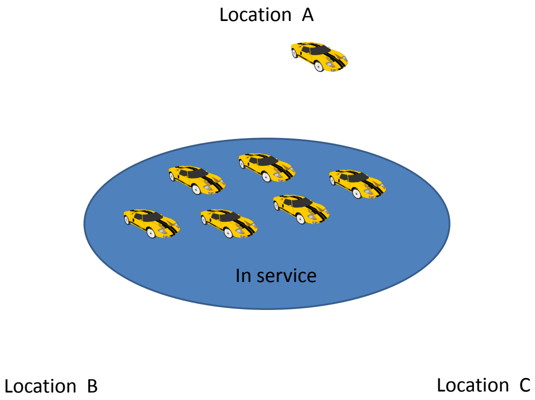
Problem Dynamics



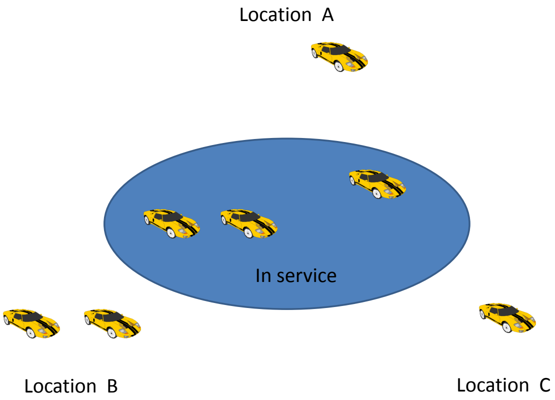
Problem Dynamics



Problem Dynamics



Problem Dynamics



Ongoing Rentals Assumption

Let $p_{t,ij}$ be the fraction of inventory rented from i that is returned at j after one period. Note that $\sum_j p_{t,ij} < 1$ means rentals can potentially be multiple periods.

Assumption 1

We assume the following conditions on π_t and the repositioning costs c_{ij}.

1. *For every period t, there exists a random variable $p_t \in [p_{\min}, 1]$ such that*

$$\sum_{j=1}^{n} p_{t,ij} = \sum_{j=1}^{n} p_{t,kj} = p_t, \text{ for all } i, k = 1, 2, \ldots, n.$$

   An alternative statement is that $p_{t,ij} = p_t\, \tilde{q}_{t,ij}$ for some $\tilde{q}_{t,ij}$ where $\sum_{j=1}^{n} \tilde{q}_{t,ij} = 1, \forall\, i$.

2. *The repositioning costs satisfy $\rho c_{\max} - c_{\min} \leq p_{\min}\,(\beta - c_{\min})$.*

Additional Assumptions

- $\boldsymbol{x}_t$, $\boldsymbol{y}_t$ and $\boldsymbol{d}_t$ are continuous.
- The cost of moving one unit from location i to location j is c_{ij}.
- $c_{ik} \leq c_{ij} + c_{jk}$ for all i, j, k (Triangle inequality)
- For simplicity, we assume $l_i = l$ for all locations i.
- $l \geq c_{ji}$ for all i, j.

Outline

DP Formulation

Let $\boldsymbol{x}$ be the initial inventory for each locations (excluding in-service products).

$$v_t(\boldsymbol{x}_t, \gamma_t) = \min_{\boldsymbol{y}_t \in \Delta_{n-1}(\boldsymbol{e}^T \boldsymbol{x}_t)} r_t(\boldsymbol{x}_t, \gamma_t, \boldsymbol{y}_t) + \rho \int v_{t+1}(\boldsymbol{x}_{t+1}, \gamma_{t+1}) \, \mathrm{d}\mu_t \quad (1)$$

where:

- $\Delta(I) \triangleq \{\boldsymbol{x} | \boldsymbol{x} \geq 0, \boldsymbol{e}^T \boldsymbol{x} = I\}$, $\boldsymbol{e}^T \boldsymbol{x} \leq N$
- $r_t(\boldsymbol{x}_t, \gamma_t, \boldsymbol{y}_t) = C(\boldsymbol{y}_t - \boldsymbol{x}_t) + l_t(\boldsymbol{y}_t)$
- $C(\boldsymbol{y}_t - \boldsymbol{x}_t)$ is the cost to reposition the inventory from $\boldsymbol{x}_t$ to $\boldsymbol{y}_t$.
- $l_t(\boldsymbol{y}_t) = \int L_t(\boldsymbol{y}_t, \boldsymbol{d}_t) \, \mathrm{d}\mu_t = \beta \int \sum_i (d_{t,i} - y_{t,i})^+ \, \mathrm{d}\mu_t$.
- $x_{t+1,i} = (y_{t,i} - d_{t,i})^+ + \sum_{j=1}^n (\gamma_{t,j} + \min(y_{t,j}, d_{t,j})) \, p_{t,ji}$

Outline

The Repositioning Cost

- Denote the space of feasible inventory repositioning by

$$\mathrm{dom}(K) = \left\{ (\boldsymbol{x}, \boldsymbol{y}) | \boldsymbol{x} \geq 0, \boldsymbol{y} \geq 0, \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \right\}.$$

- For each $(\boldsymbol{x}, \boldsymbol{y}) \in \mathrm{dom}(K)$, the repositioning cost is determined by solving the following optimization problem.

$$\begin{aligned} C(\boldsymbol{y} - \boldsymbol{x}) = \min_{\boldsymbol{z} = (z_{i,j}; i \neq j)} \quad & \boldsymbol{c}\boldsymbol{z} \\ \text{s.t.} \quad & \sum_{i,j: i \neq j} z_{i,j}(\boldsymbol{e}_j - \boldsymbol{e}_i) = \boldsymbol{y} - \boldsymbol{x} \\ & \boldsymbol{z} \geq 0, \end{aligned}$$

where $z_{i,j}$ is the number of cars to be relocated from location i to location j.

Properties of Repositioning Cost

- $C(\boldsymbol{y} - \boldsymbol{x})$ depends only on $\boldsymbol{y} - \boldsymbol{x}$.
- Due to triangle inequality, we have:

Proposition 1

There exists an optimal solution $\boldsymbol{z}$ *such that*

$$\sum_i z_{i,j} = (y_j - x_j)^+ \text{ and } \sum_k z_{j,k} = (y_j - x_j)^- \quad \forall \quad j.$$

Outline

The Generic One-Period Problem

- We are interested in solving a problem of the form:

$$v(\boldsymbol{x}, \gamma) = \min_{\boldsymbol{y} \in \Delta_{n-1}(\boldsymbol{e}^T \boldsymbol{x})} C(\boldsymbol{y} - \boldsymbol{x}) + u(\boldsymbol{y}, \gamma) \text{ for } (\boldsymbol{x}, \gamma) \in \Delta. \quad (2)$$

- Let:

$$\Omega_u(\gamma) = \{\boldsymbol{x} : u(\boldsymbol{x}, \gamma) \leq C(\boldsymbol{y} - \boldsymbol{x}) + u(\boldsymbol{y}, \gamma) \ \forall \ \boldsymbol{y}\}, \forall \gamma \in S \quad (3)$$

- $\Omega_u(\gamma)$ a region where no repositioning is needed if $\boldsymbol{x}$ is in this region.
- We called this region the *no-repositioning region*.

The Optimal Policy for the Generic One-Period Problem

Theorem 1

The no-repositioning set $\Omega_u(\gamma)$ is nonempty, connected and compact for all $\gamma \in S$. An optimal policy π^* to (2) satisfies

$$\begin{aligned} \pi^*(\boldsymbol{x},\gamma) &= \boldsymbol{x} && \text{if} \quad \boldsymbol{x} \in \Omega_u(\gamma); \\ \pi^*(\boldsymbol{x},\gamma) &\in \mathcal{B}(\Omega_u(\gamma)) && \text{if} \quad \boldsymbol{x} \notin \Omega_u(\gamma). \end{aligned} \tag{4}$$

Therefore, the optimal policy is:

- If $\boldsymbol{x} \in \Omega_u(\gamma)$, no relocation is needed.
- If $\boldsymbol{x} \notin \Omega_u(\gamma)$, $\boldsymbol{y}_t^*(\boldsymbol{x},\gamma)$ lies in the boundary of $\Omega_u(\gamma)$ and is determined by the convex optimization.

Characterizing the No-Repositioning Region

Proposition 2

$\boldsymbol{x} \in \Omega_u(\gamma)$ if and only if

$$-u'(\boldsymbol{x}, \gamma; \boldsymbol{z}, \boldsymbol{0}) \leq C(\boldsymbol{z}) \tag{5}$$

for any feasible direction $(\boldsymbol{z}, \boldsymbol{0})$ at $(\boldsymbol{x}, \gamma)$.

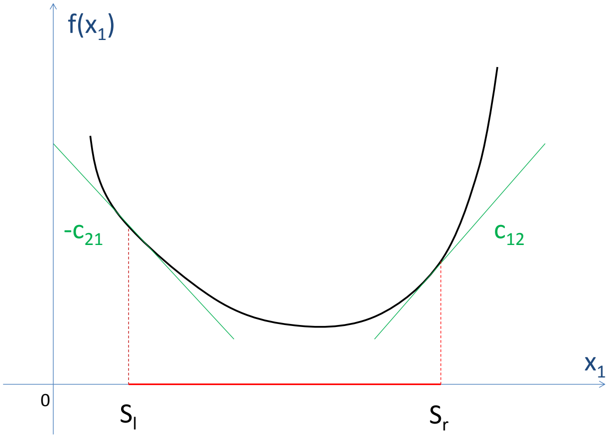
Proposition 3

Suppose $u(\cdot, \gamma)$ is differentiable at $\boldsymbol{x} \in \Delta_{n-1}(I)$. Then, $\boldsymbol{x} \in \Omega_u(\gamma)$ if and only if

$$\frac{\partial u(\boldsymbol{x}, \gamma)}{\partial x_i} - \frac{\partial u(\boldsymbol{x}, \gamma)}{\partial x_j} \leq c_{ij} \tag{6}$$

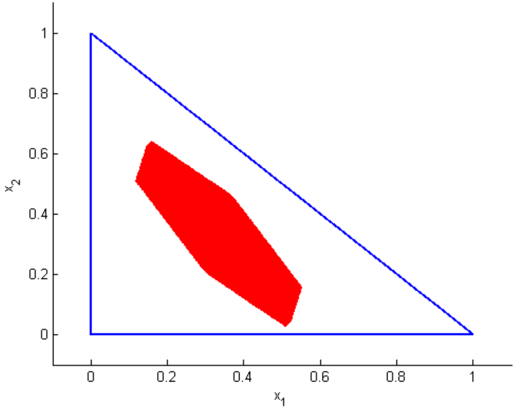
for all i, j.

Optimal Policy for Two Locations



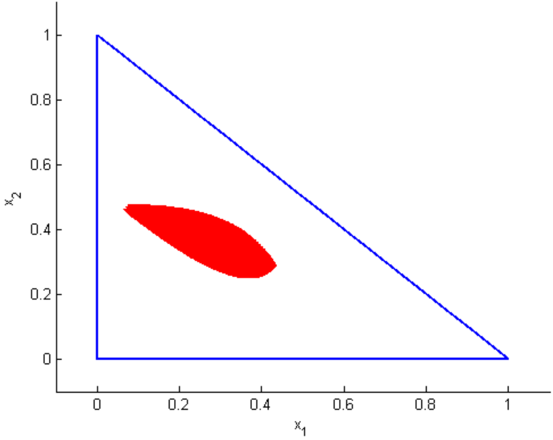
A Quadratic Example ($p_{\min} = 1$)

- $u(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{c})^T A(\boldsymbol{x} - \boldsymbol{c})$, $A = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 6 & 4 \\ 1 & 4 & 9 \end{pmatrix}$ $b = [1/3, 1/3, 1/3]^T$
- $c_{ij} = 2$.



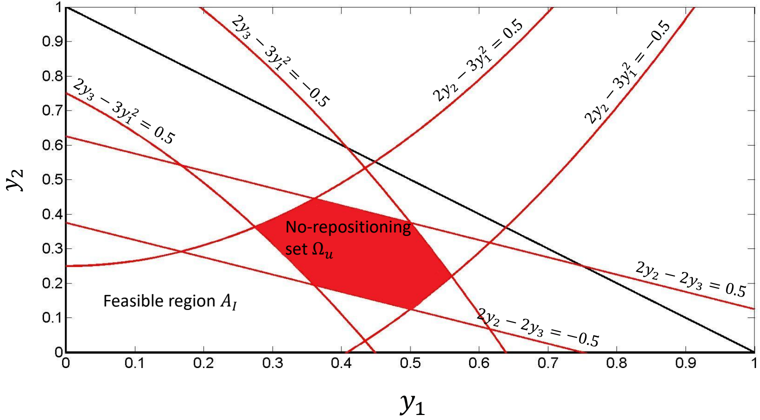
A Cubic Example ($p_{\min} = 1$)

- $u(\boldsymbol{x}) = \sum_{i=1}^{3} x_i^3$, $c_{ij} = 0.1$, $N = 1$.



Another Quadratic Example (Possibly Nonconvex)

- $\gamma = \mathbf{0}$ and $u(\boldsymbol{y}) = y_1^3 + y_2^2 + y_3^2$ and $c_{ij} = 0.5$.



Outline

1. Introduction
2. Model Description
3. Main Results
   - The Inventory Repositioning Problem
   - The Generic One-Period Problem
   - The Multi-Period/Infinite-Horizon Problem
   - Repositioning-ADP

The Multi-Period Problem with $v_{T+1} = 0$

Theorem 2

Suppose Assumption 1 holds. For any given $t = 1, \ldots, T$, the function $u_t(\cdot)$ is convex and continuous in Δ. The no-repositioning set $\Omega_{u_t}(\gamma)$ is nonempty, connected and compact for all $\gamma \in S$, and can be characterized as we did in the single period problem. An optimal policy $\boldsymbol{\pi}^* = (\pi_1^*, \ldots, \pi_T^*)$ to the multi-period problem satisfies

$$\begin{aligned} \pi_t^*(\boldsymbol{x}_t, \gamma_t) = \boldsymbol{x}_t \quad & \text{if} \quad \boldsymbol{x}_t \in \Omega_{u_t}(\gamma_t); \\ \pi_t^*(\boldsymbol{x}_t, \gamma_t) \in \mathcal{B}(\Omega_{u_t}(\gamma_t)) \quad & \text{if} \quad \boldsymbol{x}_t \notin \Omega_{u_t}(\gamma_t). \end{aligned} \tag{7}$$

Moreover, for any $t = 1, 2, \ldots, T$, we have

1. $u_t'(\boldsymbol{y}_t, \gamma_t; -\boldsymbol{\eta}, \boldsymbol{\eta}) \leq \beta \sum_{i=1}^n \eta_i$ *for all* $(\boldsymbol{x}, \gamma) \in \Delta$ *and any feasible direction* $(-\boldsymbol{\eta}, \boldsymbol{\eta})$ *with* $\boldsymbol{\eta} \geq \mathbf{0}$;
2. $u_t'(\boldsymbol{y}_t, \gamma_t; \mathbf{0}, \boldsymbol{v}) \leq (\rho c_{\max}/2) \sum_{i=1}^n |v_i|$ *for all* $(\boldsymbol{x}, \gamma) \in \Delta$ *and any feasible direction* $(\mathbf{0}, \boldsymbol{v})$ *with* $\boldsymbol{e}^T \boldsymbol{v} = 0$.

The Infinite-Horizon Problem

Now, let us consider

$$v(\boldsymbol{x},\gamma) = \min_{\boldsymbol{\pi}} \; \mathbb{E}_{\boldsymbol{x}}^{\boldsymbol{\pi}} \left\{ \sum_{t=1}^{\infty} \rho^{t-1} r(X_t, \Gamma_t, \pi(X_t, \Gamma_t)) \right\}.$$

Theorem 3

Suppose Assumption 1 holds. The function $u(\cdot,\gamma)$ is convex and continuous in Δ. The no-repositioning set Ω_u is nonempty, connected and compact for all $\gamma \in S$, and can be characterized as we did before. An optimal policy $\boldsymbol{\pi}^* = (\pi^*, \pi^*, \ldots)$ to the stationary problem with infinitely many periods satisfies

$$\begin{aligned} \pi^*(\boldsymbol{x},\gamma) &= \boldsymbol{x} && \text{if} \quad \boldsymbol{x} \in \Omega_u(\gamma); \\ \pi^*(\boldsymbol{x},\gamma) &\in \mathcal{B}\left(\Omega_u(\gamma)\right) && \text{if} \quad \boldsymbol{x} \notin \Omega_u(\gamma). \end{aligned} \tag{8}$$

Outline

Infinite-Horizon ADP Algorithm

1. Suppose we currently have $u_J(\boldsymbol{y},\gamma) = \max_{k=1,\dots,N_J} g_k(\boldsymbol{y},\gamma)$ where

$$g_k(\boldsymbol{y},\gamma) = (\boldsymbol{y} - \boldsymbol{y}_k)^T \boldsymbol{a}_k + (\gamma - \gamma_k)^T \boldsymbol{b}_k + c_k,$$

   and N_J is the total number of cuts in the approximation after iteration J.

2. At iteration J, add cuts **(solve LPs)** $N_J + 1, \dots N_{J+1}$ by computing tangent hyperplanes to $\mathcal{L}u_J$ at randomly sampled states $\mathcal{S}_J$, where

$$(\mathcal{L}f)(\boldsymbol{y},\gamma) = l(\boldsymbol{y}) + \rho \int \min_{\boldsymbol{y}' \in \Delta_{n-1}(\boldsymbol{e}^T \boldsymbol{x}')} C(\boldsymbol{y}' - \boldsymbol{x}') + f(\boldsymbol{y}',\gamma')\, d\mu$$

   is the Bellman operator.

Infinite-Horizon ADP Algorithm

1. By using the no-repositioning set characterization, cuts computation can be sped up (thus, we are utilizing both policy and value structure in this algorithm).
2. This algorithm is related to the stochastic dual dynamic programming (SDDP) algorithm, which is an algorithm for finite-horizon problems (and is known to converge).
3. We prove a new convergence result for the infinite horizon setting using a different proof technique.

Infinite-Horizon ADP Algorithm

Assumption 2

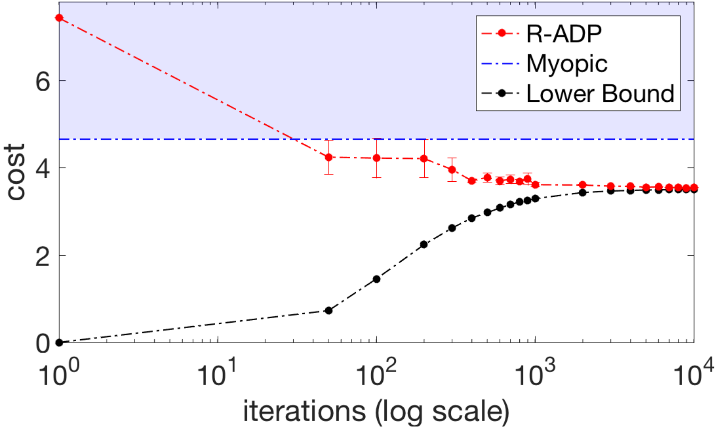
The sampling distribution produces sets $\mathcal{S}_J$ *that satisfy* $\sum_{J=1}^{\infty} \mathbf{P}\big(\mathcal{S}_J \cap A \neq \emptyset\big) = \infty$ *for any set* $A \subseteq \Delta$ *with positive volume.*

Theorem 4

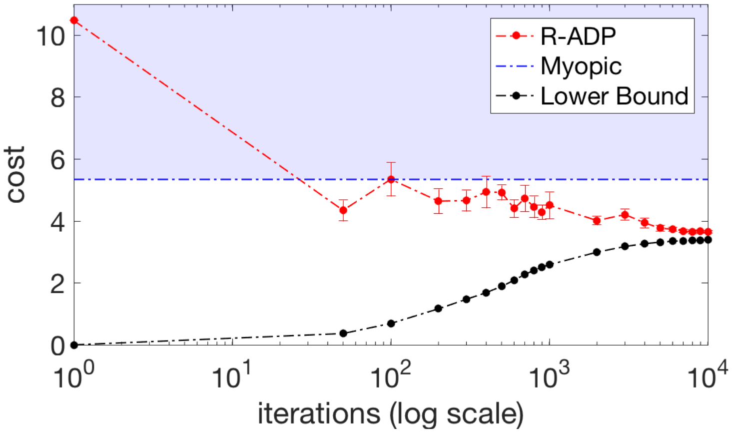
Suppose Assumption 1 and 2 hold. If $u_0(\cdot)$ *satisfies a technical condition and* $u_0(\cdot) \leq u(\cdot)$*, then*

- $\{u_J(\cdot)\}$ *converges uniformly and almost surely to the optimal value function* $u(\cdot)$*, i.e., it holds that* $\|u_J - u\|_\infty \to 0$ *almost surely.*

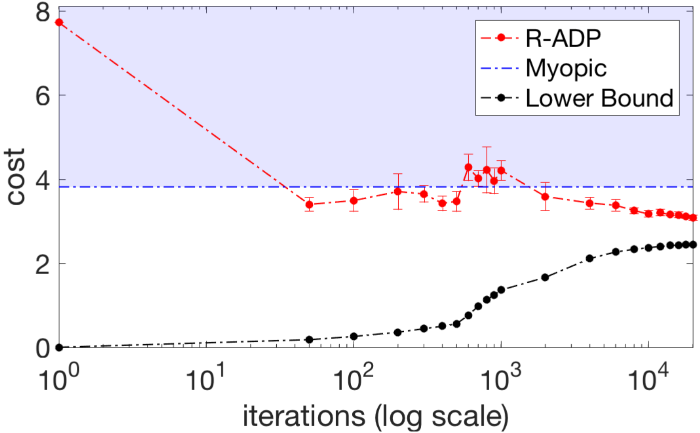
Results for $n = 3$ Locations / $d = 6$ Inventory States



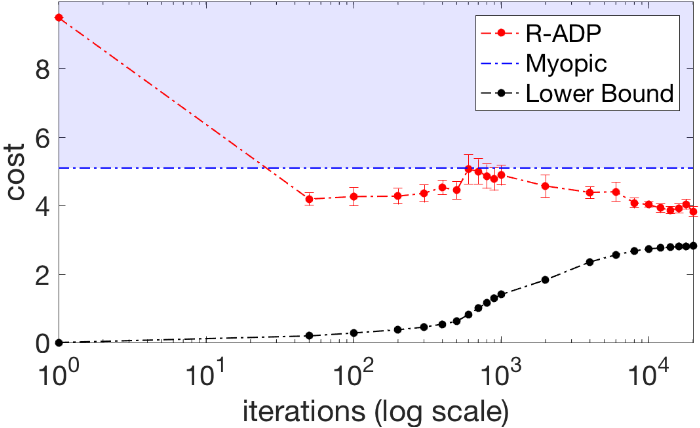
Results for $n = 5$ Locations / $d = 10$ Inventory States



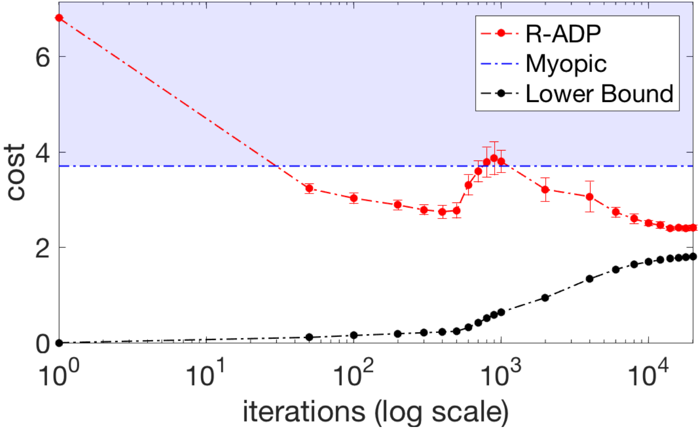
Results for $n = 7$ Locations / $d = 14$ Inventory States



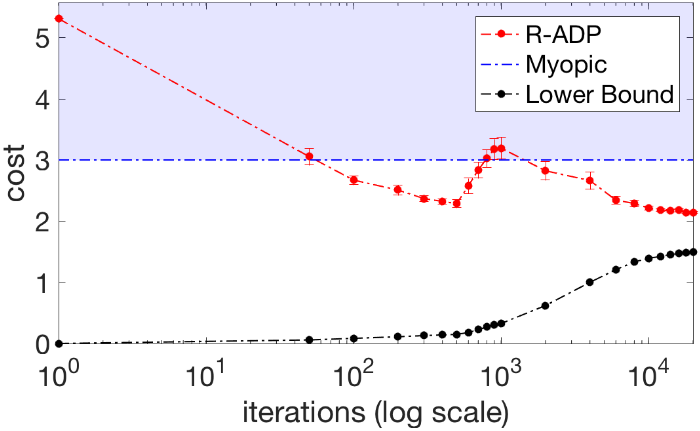
Results for $n = 8$ Locations / $d = 16$ Inventory States



Results for $n = 9$ Locations / $d = 18$ Inventory States



Results for $n = 10$ Locations / $d = 20$ Inventory States

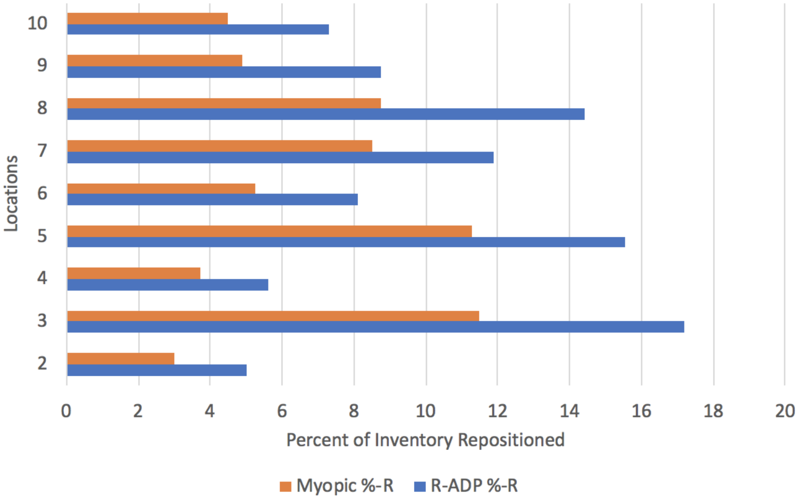


Results for $n = 2$ to $n = 10$ Locations

n	Sec./Iter.	R-ADP Cost	% Decr. Myo.	% Decr. No-R	% to LB	R-ADP %-R	Myo. %-R
2	0.06	1.20	56.16%	76.48%	99.18%	5.01%	3.01%
3	0.21	3.55	23.63%	52.17%	98.70%	17.20%	11.50%
4	0.27	1.69	22.25%	60.69%	95.84%	5.62%	3.72%
5	0.22	3.65	31.64%	65.14%	96.38%	15.54%	11.30%
6	0.29	2.45	37.70%	75.20%	94.08%	8.09%	5.26%
7	0.42	3.08	19.23%	60.12%	88.10%	11.87%	8.49%
8	0.44	3.83	24.91%	59.67%	85.03%	14.42%	8.77%
9	0.48	2.40	35.04%	64.65%	88.20%	8.77%	4.89%
10	0.54	2.14	28.58%	59.65%	83.31%	7.28%	4.49%

Table 1: Summary of Results for Repositioning-ADP Benchmarks

Policy Behavior



Scaling to Large-Scale Instances (20-100 Locations)

Common heuristic is to use a **deterministic lookahead** approximation.

1. k-RH-M: Replace random quantities with their means and solve a k-period problem (large-scale LP),
2. k-RH-S: Replace random quantities with a single sample and solve a k-period problem (large-scale LP),
3. Implemented as a repositioning policy in a rolling-horizon fashion.

Scaling to Large-Scale Instances (20-100 Locations)

We also propose **Clustered R-ADP**. Suppose a problem has n locations.

1. Cluster locations together (summing/averaging problem parameters) to create an m-location problem,
2. Solve the m-location problem using R-ADP.
3. Construct an n-location policy via an appropriate "splitting" heuristic (split cluster decisions to individual location; e.g., scale by demand).

Results for $n = 20$ to $n = 100$ Locations using 10 Clusters

n	CR-ADP	Myo.	No-R	3-RH-M	3-RH-S	5-RH-M	5-RH-S	7-RH-S	7-RH-S	10-RH-M	10-RH-S
20	**3.62**	4.51	9.30	4.51	5.77	4.43	5.21	4.38	5.05	4.31	5.97
30	**2.92**	3.24	6.34	3.70	3.83	3.66	4.19	3.62	3.98	3.59	4.58
40	**3.37**	4.01	7.77	4.60	4.86	4.53	4.92	4.04	4.55	4.05	4.35
50	**3.70**	4.09	8.35	4.33	4.77	4.19	4.66	4.14	4.48	-	-
60	**3.76**	4.12	8.90	4.36	4.88	4.24	4.72	-	-	-	-
70	**3.57**	4.11	8.70	4.31	4.77	4.18	4.67	-	-	-	-
80	**3.34**	3.81	8.17	3.96	4.51	3.87	4.37	-	-	-	-
90	**3.94**	4.25	8.74	4.33	4.87	-	-	-	-	-	-
100	**3.35**	3.70	7.44	3.89	4.39	-	-	-	-	-	-

Table 3: Summary of Results on Large-scale Instances

Summary of Contributions

1. We prove structural results for the dynamic repositioning model (to our knowledge, this model is the most general of its kind due our consideration of in-service rentals).
2. A provably convergent, infinite-horizon, cutting-plane ADP method that is also of broader interest, particularly as a contribution toward the SDDP literature.
3. A cluster-based extension of the ADP method for problems of up to 100 locations; outperforms common heuristics.

Thank you! Questions?