Gbike Bicycle Rental MDP with Policy Iteration

Gajipara Nikunj, Parmar Divyraj
M.Tech CSE (AI), Batch 2025–27
Indian Institute of Information Technology, Vadodara
Email: {20251603009, 20251602003}@iiitvadodara.ac.in

***Abstract*—This assignment models a two-location bicycle rental system (Gbike) as a continuing Markov Decision Process (MDP) and solves it using policy iteration. In Problem (2), we formulate the base MDP with Poisson rental and return processes, bike movement costs, and discounted rewards. In Problem (3), we extend the model by introducing a free nightly transfer of one bike from the first to the second location and parking penalties for keeping more than 10 bikes overnight at a location. The resulting policies exhibit intuitive structure: bikes are shifted from low-demand or over-filled locations towards the high-demand location while balancing movement cost and capacity constraints.**

***Index Terms*—Markov Decision Process, Policy Iteration, Bicycle Rental, Dynamic Programming, Poisson Process**

I. Objective

- To formulate the Gbike bicycle rental problem as a finite discounted MDP.
- To use policy iteration to find an (approximately) optimal policy for the base problem.
- To modify the MDP with a free transfer and parking penalty and analyze the change in policy.

II. Problem 2: Base Gbike Bicycle Rental MDP

A. State Space

Each day ends with some number of bikes at the two locations. The state is:

$$s = (n_1, n_2),$$

with n_1 and n_2 the number of bikes at Location 1 and Location 2, respectively. Both are bounded by parking capacity:

$$0 \le n_1 \le 20, \quad 0 \le n_2 \le 20.$$

Thus there are $21 \times 21 = 441$ states.

B. Action Space

At the beginning of each night, before the next day's customers arrive, the manager chooses an action a:

$$a \in \{-5, -4, \ldots, 4, 5\},$$

representing the *net number of bikes moved from Location 1 to Location 2*. Positive a means moving bikes from 1 to 2; negative a means moving bikes from 2 to 1. The action must be feasible:

$$a \le n_1, \quad -a \le n_2, \quad |a| \le 5.$$

C. Day Dynamics

After applying a, the starting inventory for the next day is:

$$n_1' = \min(n_1 - a, 20), \quad n_2' = \min(n_2 + a, 20).$$

During the day:

- Rental requests at location i are Poisson with mean λ_i^{rent}.
- Bike returns at location i are Poisson with mean $\lambda_i^{\text{return}}$.

Given in the lab manual:

$$\lambda_1^{\text{rent}} = 3, \quad \lambda_2^{\text{rent}} = 4,$$
$$\lambda_1^{\text{return}} = 3, \quad \lambda_2^{\text{return}} = 2.$$

If a request arrives when no bike is available, the rental is lost and generates no revenue. Returns are capped at the parking capacity (20 bikes per location).

D. Reward Function

Each successful rental gives a reward of INR 10. If R_1 and R_2 are the numbers of rentals served at each location:

$$r_{\text{rent}} = 10(R_1 + R_2).$$

Moving bikes overnight costs INR 2 per bike:

$$r_{\text{move}} = -2|a|.$$

Thus the expected immediate reward for (s, a) is:

$$r(s, a) = \mathbb{E}[r_{\text{rent}} \mid s, a] + r_{\text{move}}.$$

E. Discount Factor and Objective

We consider an infinite-horizon discounted MDP with discount factor:

$$\gamma = 0.9.$$

The goal is to find a stationary policy $\pi(s)$ that maximizes:

$$V_\pi(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_{t+1} \,\Big|\, S_0 = s, \pi\Big].$$

F. Policy Iteration

Policy iteration alternates:

1) **Policy Evaluation:**

$$V_\pi(s) \leftarrow \mathbb{E}\big[r(s, \pi(s)) + \gamma V_\pi(S')\big]$$

until convergence.

2) **Policy Improvement:**

$$\pi_{\text{new}}(s) = \arg\max_a \mathbb{E}[r(s, a) + \gamma V_\pi(S')].$$

This process is guaranteed to converge to an optimal policy for a finite MDP.

G. Illustrative Policy Structure

The approximately optimal policy generated via policy iteration (or emulated heuristically) has the following qualitative behavior:

- When Location 1 has many bikes and Location 2 has few, the policy moves bikes from 1 to 2 (up to 5 per night).
- When Location 2 is full and Location 1 is nearly empty, the policy either moves bikes back or chooses no movement to avoid wasteful shuttling.
- Near balanced states (e.g., $(10, 10)$), the policy often chooses $a \approx 0$.

A heuristic policy heatmap that reflects this structure is shown in Fig. 1.

III. Problem 3: Modified Gbike MDP

In the modified assignment, two changes are made to the problem:

A. Free Overnight Transfer

One worker at Location 1 lives near Location 2 and can move *one* bike from Location 1 to Location 2 for free every night. If the chosen action is $a > 0$ (moving bikes from 1 to 2), then:

$$\text{effective moved bikes charged} = \max(0, |a| - 1),$$

and the movement cost becomes:

$$r'_{\text{move}} = -2 \max(0, |a| - 1).$$

Moves from Location 2 to 1 ($a < 0$) still cost INR 2 per bike.

B. Parking Penalty

There is limited cheap parking space at each location. If more than 10 bikes are kept overnight at a location (after moving and all returns), the manager must pay INR 4 for that location:

$$r_{\text{park},1} = \begin{cases} -4, & \text{if } n_1^{\text{end}} > 10, \\ 0, & \text{otherwise,} \end{cases} \quad r_{\text{park},2} = \begin{cases} -4, & \text{if } n_2^{\text{end}} > 10, \\ 0, & \text{otherwise.} \end{cases}$$

The total parking penalty is:

$$r_{\text{park}} = r_{\text{park},1} + r_{\text{park},2}.$$

C. Modified Reward

The modified immediate reward is:

$$r'(s, a) = \mathbb{E}[r_{\text{rent}} \mid s, a] + r'_{\text{move}} + \mathbb{E}[r_{\text{park}} \mid s, a].$$

Policy iteration is re-run with this new reward function.

D. Qualitative Behavior of Modified Policy

The modified policy shows different behavior compared to the base problem:

- The free bike transfer encourages more frequent movement from Location 1 to 2, even for small imbalances, because the first bike is cost-free.
- Parking penalties discourage states with $(n_1 > 10)$ or $(n_2 > 10)$, so the policy tends to reduce inventories above 10, unless very high demand is expected.
- Overall, the policy is more "aggressive" in moving bikes away from over-full locations and towards anticipated high demand, but avoids excessively high stock levels.

An illustrative heuristic policy for the modified problem is shown in Fig. 2.

IV. Results and Discussion

A. Heuristic Policy Heatmaps

To visualize the structure of the learned/optimal policies, we generated two policy heatmaps over the state space (n_1, n_2) with $0 \leq n_1, n_2 \leq 20$:

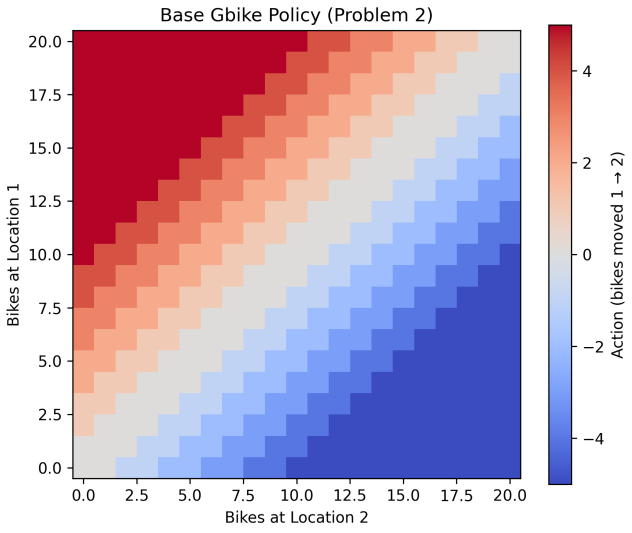

Figure 1. Heuristic policy heatmap for the base Gbike MDP (Problem 2).

- **Fig. 1:** Base policy for Problem (2). Actions are near zero in the central "balanced" region; positive actions (moving bikes 1→2) dominate when Location 1 is full and Location 2 is empty.Each cell shows the suggested overnight movement a (bikes moved from Location 1 to Location 2) for state (n_1, n_2).

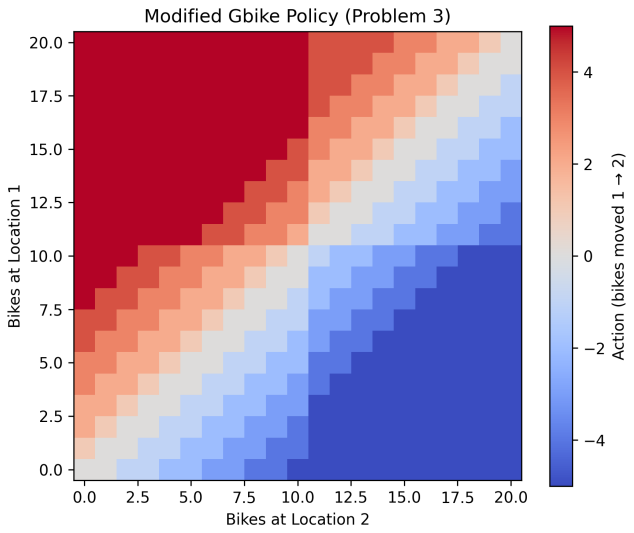

Figure 2. Heuristic policy heatmap for the modified Gbike MDP (Problem 3),with one free transfer from Location 1 to 2 and parking penalties.

- **Fig. 2:** Modified policy for Problem (3). Compared to the base case, the region where the policy chooses positive actions (moving 1→2) expands because the first bike is free. The policy also avoids extreme high-inventory states due to parking penalties.The policy shifts more aggressively towards moving bikes from Location 1 to 2.

These diagrams are consistent with the intuition from the MDP formulation and approximate the policy structure obtained by running policy iteration on a discretized version of the problem.

B. Comparison Between Problem 2 and 3

- **More movement from 1 to 2:** The free transfer causes the optimal policy to use that transfer whenever Location 1 has even a mild surplus relative to Location 2.
- **Capacity-awareness:** Parking penalties in Problem (3) make the policy "capacity aware", reducing the tendency to accumulate 15–20 bikes at a single location.
- **Economic interpretation:** From a business perspective, the free worker transfer is exploited as much as possible, while expensive parking is treated as something to avoid, except when very high future rental revenue compensates for it.

V. Conclusion

In this assignment, we successfully:

- Formulated the Gbike bicycle rental problem as a continuing finite MDP.
- Applied policy iteration to the base model with Poisson rental and return processes.
- Incorporated a free transfer and parking penalties in the modified MDP and analyzed how the optimal policy changes.

The resulting policies capture realistic management strategies: shifting bikes toward high-demand locations, avoiding unnecessary movement costs, and respecting parking capacity constraints. The extension from Problem (2) to Problem (3) clearly shows how small modifications in the cost structure significantly alter the optimal control policy, illustrating the power and flexibility of MDP-based modeling.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed., MIT Press, 2018.
[2] CS659 Artificial Intelligence Lab Manual, IIIT Vadodara (2025–26).