RL in Elevator Dispatch

Course Project Initial Proposal
Reinforcement Learning and Decision Making Under Uncertainty,
University of Neuchatel, Spring 2025

William Dan*
University of Bern
william.dan@students.unibe.ch

24 April 2025

1 Introduction

Elevator dispatch is an NP-hard, stochastic scheduling task where even recent commercial heuristics still leave average passenger waits above 30s at peak traffic. Cutting those few extra seconds scales to thousands of saved person-hours, lower energy and higher tenant-satisfaction scores.

Brief explanation of what I do: The project will model elevator group control as an MDP, implement a Gymnasium simulator, and benchmark different RL models against the best heuristic optimization baselines. The state space and action space can be very small if we constraint the number of floors and elevators, also can be very large if we increase the number. We choose very common configuration in the skyscraper to align with the reality case.

2 Related Work

In elevator dispatch, there are already some mature heuristic algorithms. However, there are still some space in performance for improvement.[1]

3 Problem Statement

We model the problem into MDP, using the model from the previous work [Wei+20]. The state space is depicted in table 2[2] and the action space is depicted in table 3[3]. In the table 4, we show that the state space in this model is very large [4]. So, we use deep RL to solve this problem.

4 Expected Result

Based on the model [Wei+20], we try to add some new metrics, new logics to make it more aligned with the reality. Also, in the paper, they use CNN and LSTM as the neural network and propose A3C to solve the problem. We would like to benchmark different types of deep RL models to see the difference among them.

*University of Bern, 3012 CH-Bern, Switzerland.

Table 1. Classical algorithms of elevator dispatch

Algorithm	Description	Pros	Cons
FIFO (First-In-First-Out)	Processes requests in the order they are received.	• Simple to implement. • No starvation.	• Inefficient for scattered requests. • Longer wait times due to lack of prioritization.
SSTF (Shortest Seek Time First)	Selects the closest pending request based on current position.	• Minimizes seek time. • Reduces average wait time.	• Can cause starvation for distant requests. • Direction may change frequently.
SCAN	Moves in one direction, serves all requests, reverses at the end.	• Balanced load distribution. • Efficient for dense requests.	• Longer wait for opposite-direction requests. • Unnecessary stops at extreme floors.
LOOK	Similar to SCAN but reverses direction when no requests ahead.	• Avoids unnecessary end-floor stops. • Better efficiency than SCAN.	• Complex to implement. • Still delays reverse-direction requests.
RR (Round Robin)	Cycles through requests in fixed intervals or order.	• Ensures fairness. • Predictable stops.	• High energy use. • Inefficient with uneven demand.

References

[Wei+20] Q. Wei, L. Wang, Y. Liu, and M. M. Polycarpou. "Optimal Elevator Group Control via Deep Asynchronous Actor-Critic Learning". In: *IEEE Trans. Neural Networks Learn. Syst.* 31.12 (2020), pp. 5245–5256. DOI: 10.1109/TNNLS.2020.2965208.

Table 2. MDP state representation for an N-floor, M-car elevator group

Symbol	Shape	Value / Type	Description
$\bar{B}$	$N \times M$	$\mathbb{R}_{\geq 0}$	*Hall-call matrix* after direction replication and truncation; $\bar{b}_{i,j}$ is cumulative (or binary) wait of callers at floor i in the travel direction of car j.
A	$N \times M$	$\{0, 1\}$	*Car-call matrix*; $a_{i,j} = 1$ iff a passenger in car j wishes to alight at floor i.
p	M	$\{1, \ldots, N\}$	Discrete current floor index of each car.
d	M	$\{-1, 0, 1\}$	Travel direction of each car (-1 down, 0 idle, $+1$ up).

The four tensors are stacked as $(\bar{B}, A, P, D)$, making a 3-D array of shape $(N, M, 4)$ which is then
flattened to a feature vector of length $4NM$ for the neural policy.

Table 3. MDP action space

Symbol	Domain / Encoding	Cardinality	Interpretation
$a = (i, j)$	$i \in \{1, \ldots, N\},\ j \in \{1, \ldots, M\}$	$\lvert\mathcal{A}\rvert = N \times M$	"Dispatch car j to stop next at floor i" (subject to non-reversal and capacity constraints).

Table 4. Size explosion of the elevator MDP (binary buttons)

Parameter	Formula	8 floors, 3 cars	20 floors, 4 cars
Hall calls	2^{2N}	2^{16}	2^{40}
Car calls	2^{NM}	2^{24}	2^{80}
Car positions	N^M	8^3	20^4
Car directions	3^M	3^3	3^4
State space $\lvert\mathcal{S}\rvert$	$2^{2N+NM} N^M 3^M$	1.5×10^{16}	1.7×10^{43}
Action space $\lvert\mathcal{A}\rvert$	$N\,M$	24	80
Q-table size $\lvert\mathcal{S}\rvert \cdot \lvert\mathcal{A}\rvert$	—	3.6×10^{17}	1.4×10^{45}