FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Shaping Co-Transport Behavior: Deep Reinforcement Learning for Mixed Passenger and Parcel Mobility

Gonçalo Jorge Soares Ferreira

Master in Informatics and Computing Engineering

Supervisor: Prof. Rosaldo José Fernandes Rossetti

Co-Supervisor: Prof. Tânia Daniela Fontes

Co-Supervisor: Prof. António Ribeiro da Costa

June 30, 2026

Shaping Co-Transport Behavior: Deep Reinforcement Learning for Mixed Passenger and Parcel Mobility

Gonçalo Jorge Soares Ferreira

Master in Informatics and Computing Engineering

Approved in oral examination by the committee:

President: Ana Paula Rocha
Referee: Thiago Silva
Referee: Rosaldo Rossetti

June 30, 2026

Resumo

O transporte urbano de passageiros e a entrega de encomendas de last-mile são tipicamente assegurados por frotas separadas que operam em paralelo, cada uma contribuindo de forma independente para o congestionamento e para as emissões. O transporte co-modal propõe que um único veículo sirva ambos os tipos de procura, aproveitando a capacidade dos veículos de passageiros para transportar encomendas entre cacifos automáticos. Embora a Aprendizagem por Reforço Profunda (DRL) seja uma candidata natural para aprender tais políticas combinadas de despacho, esta abordagem introduz um desafio crítico: o agente optimiza o sinal de recompensa que lhe é dado, e não o comportamento pretendido pelo projetista.

Esta dissertação investiga como a estrutura da função de recompensa molda o comportamento que um agente de DRL aprende num ambiente de grelha urbana estocástico e congestionado, com procura contínua de passageiros e encomendas. Através de um estudo sistemático de modelação de recompensa numa formulação determinística, são identificados três regimes comportamentais qualitativamente distintos — manter o passageiro, acumular encomendas e co-modalidade equilibrada — e demonstra-se que o regime com o maior retorno de treino é operacionalmente não equilibrado. Este desacoplamento entre recompensa e comportamento é depois confirmado no ambiente estocástico, onde uma penalização por passo condicionada à carga produz co-transporte genuíno: a política mista atinge 29,1% de condução em vazio (contra 55,1% para frotas dedicadas), entrega mais 32,4% por veículo e emite menos 12,7% por entrega.

Uma ablação controlada estabelece o mecanismo causal: remover o gradiente condicionado à carga elimina o comportamento de co-transporte sem alterar as taxas de entrega, provando que a estrutura em níveis da penalização é a causa do encaminhamento integrado, e não apenas uma correlação. A política aprendida iguala ou supera uma heurística greedy forte que procura a tarefa mais próxima em todos os cenários testados (todas as comparações estatisticamente significativas a $p < 0{,}01$ para cenários de propósito único; empate estatístico no cenário misto). Experiências com a métrica Patience implementada revelam um limite da abordagem: sob prazos concorrentes apertados, o agente especializa-se em vez de dominar o agendamento conjunto, motivando trabalho futuro em arquiteturas mais expressivas e aprendizagem curricular.

Abstract

Urban passenger transport and last-mile parcel delivery are typically served by separate vehicle fleets that operate in parallel, each contributing independently to congestion and emissions. Co-modal transport proposes that a single vehicle serve both demand types, exploiting the idle capacity of passenger vehicles to carry parcels between automated lockers. While deep reinforcement learning (DRL) is a natural candidate for learning such combined dispatch policies, it introduces a critical challenge: an agent optimizes the reward signal it is given, not the behavior the designer intends.

This dissertation investigates how the structure of the reward function shapes the behavior a single DRL agent learns in a stochastic, congested urban grid environment with continuous passenger and parcel demand. Through a systematic reward-shaping study in a deterministic formulation, three qualitatively distinct behavioral regimes are identified — passenger-shielding, parcel-hoarding, and balanced co-modal — and it is shown that the regime with the highest training return is operationally degenerate, not balanced. This reward–behavior decoupling is then confirmed in the stochastic setting, where a load-conditioned (tiered) step-penalty design produces genuine co-transport: the mixed policy achieves 29.1% empty driving (versus 55.1% for dedicated fleets), delivers 32.4% more per vehicle, and emits 12.7% less per delivery.

A controlled ablation establishes the causal mechanism: removing the load-conditioned gradient eliminates the co-transport behavior while leaving delivery rates unchanged, proving that the tiered penalty structure is the cause of the integrated routing, not merely a correlate. The learned policy matches or outperforms a strong greedy nearest-task heuristic across all scenarios tested (all comparisons statistically significant at $p < 0.01$ for single-purpose scenarios; statistical tie in the mixed scenario). Experiments with the Patience metric reveal a boundary of the approach: under tight competing deadlines, the agent specializes rather than mastering joint scheduling, motivating future work on richer architectures and curriculum learning.

UN Sustainable Development Goals

This section reflects on how the work presented in this dissertation aligns with the United Nations Sustainable Development Goals (SDGs).

Table 1: Alignment of this dissertation with the UN Sustainable Development Goals.

SDG	Target	Contribution	Performance Indicators and Metrics
9	9.1	Applies advanced AI methods to develop adaptive dispatch policies for urban transport, capable of learning efficient routing strategies directly from interaction with a simulated environment.	Convergence rate toward co-transport behavior. Competitiveness of learned policy against hand-designed heuristics.
11	11.2	Supports the development of efficient, inclusive urban transport systems by integrating passenger and freight flows into a single shared fleet, reducing the number of vehicles required and empty running.	Reduction in empty-driving ratio. Increase in per-vehicle throughput. Fleet size reduction for equivalent service.
12	12.2	Promotes responsible use of transport resources by maximizing vehicle utilization and filling idle capacity with parcel deliveries instead of deploying additional dedicated vehicles.	Improvement in vehicle utilization rate. Reduction in distance traveled per delivery.
13	13.2	Enables the reduction of transport-related emissions by minimizing wasted kilometers and demonstrating that co-modal operation produces fewer emissions per unit of service delivered.	Reduction in emissions (VKT) per delivery. Total distance savings compared with dedicated fleets.

Acknowledgments

Writing this thesis has been a challenging and rewarding journey, and I would not have reached this point without the support of many fantastic people.

First, I would like to express my gratitude to my supervisors, Prof. Rosaldo Rossetti, Prof. Tânia Fontes and Prof. António Costa, for their insightful guidance and mentorship throughout this process. I am grateful for the time and energy you invested in my work.

On a personal note, I want to thank my family, especially my parents and brother, for their encouragement and understanding during this long process. Thank you for always believing in me.

I also owe a special thank you to my wonderful girlfriend, Leonor, for her constant encouragement and for providing the support and stability that made this work possible.

To all my friends, namely the ones I have known since childhood, whom I have the honor to call family, thank you for every blessing and for always being present for me. To Tabuleiros, thank you for making this university journey the best I could have asked for. To those I see every day, and to those I share weekends, holidays, and every other moment with, thank you for the laughter, the distractions, and the constant support that kept me moving forward. I am incredibly lucky to have you all in my life.

Finally, thank you to everyone else who supported me in any way. I am deeply appreciative of all of you.

Gonçalo Ferreira

"A few months ago, I woke up and I suddenly got it.
I understood suddenly how thought was just an illusory thing, and how thought is responsible for, if not all, most of the suffering we experience.
And then I suddenly felt like I was looking at these thoughts from another perspective, and I wondered: who is it that's aware that I'm thinking? Suddenly I was thrown into this expansive, amazing feeling of freedom from myself, from my problems. I saw that I was bigger than what I do. I was bigger than my body. I was everything and everyone.
I was no longer a fragment of the universe; I was the universe."

Jim Carrey

Contents

List of Figures

List of Tables

Abbreviations and Symbols

CI	Confidence Interval
CVRP	Capacitated Vehicle Routing Problem
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
GAE	Generalized Advantage Estimation
MDP	Markov Decision Process
MLP	Multilayer Perceptron
PPO	Proximal Policy Optimization
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
RL	Reinforcement Learning
SARP	Share-a-Ride Problem
SDG	Sustainable Development Goal
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem
VKT	Vehicle Kilometers Traveled

Chapter 1

Introduction

1.1 Context and Motivation

Contemporary cities face a twofold pressure on their streets. On one side, the demand for personal mobility continues to grow, sustained by ride-hailing and taxi-like services that have become a structural component of urban transport (25). On the other side, the rapid expansion of electronic commerce has turned last-mile parcel delivery into one of the fastest-growing and most costly segments of urban logistics (21). These two flows of demand, people and goods, are typically served by separate fleets that operate in parallel, each contributing independently to congestion, energy consumption, and greenhouse-gas emissions.

A defining inefficiency of passenger transport is the prevalence of empty running. Ride-hailing and taxi vehicles spend a substantial fraction of their operating time without a passenger on board, cruising between trips or waiting for the next request (4). This idle capacity results in additional distance traveled, fuel burned, and emissions produced without any productive output. At the same time, parcel-delivery operators add further vehicles to the same road network to move goods that are often traveling along comparable routes. The coexistence of under-utilized passenger vehicles and a separate goods fleet suggests a clear opportunity, specifically the spare capacity of vehicles that are already circulating could absorb part of the parcel-delivery workload.

This idea, broadly termed co-modal or integrated passenger-and-freight transport, proposes that a single vehicle serve both demands, collecting and delivering parcels during the natural gaps of its passenger service. In the setting studied in this dissertation, parcels move between automated lockers distributed across the city, and a passenger vehicle that already operates in the area transports them from origin to destination locker while continuing to serve people. By filling otherwise empty time and distance, such an arrangement has the potential to reduce the total number of vehicles required, lower the share of empty running, and consequently cut emissions, while remaining economically attractive to the operator.

Realizing this potential, however, raises a difficult control problem. At every moment the vehicle must decide where to go and which demand to prioritize. Whether to collect a waiting passenger, pick up or deliver a parcel, or reposition itself in anticipation of future demand.

These decisions must be taken under uncertainty, since requests for both people and parcels arrive stochastically over time, and under the additional friction of urban congestion. Classical rule-based dispatching struggles to balance the competing objectives that arise when one vehicle serves heterogeneous demand, motivating the use of learning-based control.

Deep Reinforcement Learning (DRL) is a natural candidate for this problem, as it allows an agent to learn a control policy directly from interaction with a simulated environment, without requiring an explicit model of demand or traffic. Yet applying DRL to co-modal transport exposes a subtle but critical difficulty that is central to this dissertation: an agent optimizes the reward signal it is given, not the behavior the designer intends. A reward function that appears reasonable can be maximized through degenerate strategies, so that a high numerical reward coexists with behavior that is operationally undesirable. In a co-modal setting, where the reward must simultaneously encourage passenger service, parcel delivery, and efficient use of the vehicle, this decoupling between reward and behavior becomes both more likely and more consequential. Understanding how the structure of the reward function shapes the behavior that is actually learned is therefore the central concern of this work. A preliminary study preceding this dissertation (5) already provided empirical evidence of this decoupling in a controlled deterministic setting: the configurations that achieved the highest training return did so by adopting degenerate shielding or hoarding strategies, while the configuration that behaved correctly earned among the lowest returns—so that ranking policies by return systematically favors degenerate behavior.

1.2 Problem Definition

The problem addressed in this dissertation sits at the intersection of reinforcement learning and urban transport optimization. Specifically, it asks how to design a reward function for a single deep reinforcement-learning agent so that the policy it learns produces efficient co-transport of passengers and parcels in a congested urban grid, rather than one of the several degenerate strategies to which poorly structured rewards give rise.

Urban co-modality refers to the integration of passenger transport and freight delivery into a single shared fleet (9). Rather than operating separate vehicle pools for people and goods, a co-modal system allows a vehicle currently serving passenger trips to also collect and deliver parcels along its route, exploiting the spare capacity that would otherwise be wasted as empty running.

The concept is motivated by two empirical observations. First, ride-hailing vehicles spend a significant fraction of their operating time without a passenger on board, cruising or repositioning between trips (4). Second, last-mile parcel delivery is a rapidly growing segment of urban logistics whose vehicles often follow routes that overlap with passenger demand (21). Combining the two demand streams into a single vehicle has the potential to reduce the total number of vehicles on the road, lower the share of empty running, and consequently decrease congestion and emissions.

This is not merely an engineering challenge of tuning parameters. Because reward maximization is decoupled from operational behavior (Section 1.1), the problem demands not only a

well-structured reward but also a set of behavioral metrics that detect integration, expose failure modes, and allow causality to be established between reward structure and learned behavior.

A secondary question, contingent on the first, asks whether the co-modal policy that emerges from a well-designed reward delivers measurable system-level benefits, in throughput, efficiency, emissions, and profitability, compared with the alternative of operating dedicated single-purpose fleets. This question connects the reinforcement-learning contribution to the broader transport-planning motivation.

1.3 Research Questions

The research questions guiding this work are:

RQ1 — Reward design (mechanism): In a co-modal transport setting, how does the design of the reward function govern the behavior a single DRL agent learns and to what extent can a load-conditioned (tiered) reward structure produce efficient co-transport, given that reward maximization does not guarantee the intended behavior?

RQ2 — Viability and competitiveness: Can a single-agent DRL policy learn to serve combined passenger and parcel demand in a stochastic, congested grid, and how does it compare against random and greedy heuristic baselines?

RQ3 — System-level benefit: Does a single mixed-purpose vehicle improve per-vehicle throughput, utilization, empty driving, emissions, and profitability relative to dedicated single-purpose fleets?

RQ4 — Human-centric constraints: How do time-based service constraints (passenger patience, parcel carry-deadlines) shape the learned policy's prioritization and service quality?

1.4 Aim and Goals

The aim of this dissertation is to investigate how the idle capacity of passenger vehicles can be exploited to enable efficient co-modal transport of passengers and parcels, and to quantify the resulting system-level benefits compared with dedicated single-purpose fleets. To this end, this dissertation investigates how the structure of a reward function influences the behavior learned by a single Deep Reinforcement Learning (DRL) agent in a co-modal transport setting, enabling the agent to learn operational policies that effectively coordinate passenger and parcel transport.

The aim of this dissertation is to investigate how the structure of a reward function governs the behavior a single deep reinforcement-learning agent actually learns in a co-modal transport setting, and to quantify the system-level benefits of such co-modal operation against dedicated single-purpose fleets.

To achieve this aim and answer the Research Questions, the following goals are pursued:

1. Design and implement a stochastic, congested urban grid environment that models continuous passenger and parcel demand, building upon an initial deterministic formulation used for reward-shaping analysis.

2. Identify the reward structure that produces efficient co-transport behavior, and demonstrate, through a controlled ablation, that the load-conditioned, tiered penalty design is the cause of the integrated routing, rather than an incidental effect of penalizing movement at all.

3. Compare the learned mixed-purpose policy against both a random baseline and a strong greedy nearest-task heuristic, establishing viability across multiple scenarios (people-only, parcels-only, mixed).

4. Evaluate the system-level benefit of co-modal operation by comparing a single mixed vehicle against dedicated single-purpose fleets on per-vehicle throughput, empty driving, emissions per delivery, and profitability.

5. Probe the limits of the approach by introducing realistic time-based service constraints (passenger patience, parcel carry-deadlines) and analyzing how they affect the learned policy's behavior.

1.5 Document Structure

The remainder of this document is organized as follows. Chapter 2 provides the theoretical background on reinforcement learning, policy-gradient methods, and urban co-modality. Chapter 3 presents a systematic literature review following the PRISMA protocol, identifying the gap that this work addresses. Chapter 4 describes the methodology in full: the MDP formalization, the environment design (deterministic and stochastic), the reward function, the learning algorithm, the baseline policies, and the evaluation methodology including the economic and statistical procedures. Chapter 5 reports and discusses the experimental results, organized by research question. Chapter 6 concludes with the main contributions, answers to the research questions, limitations, and directions for future work.

Chapter 2

Background

This chapter introduces the theoretical and domain-specific foundations on which the work rests. Section 2.1 presents the reinforcement-learning framework and its central concepts. Section 2.2 describes policy-gradient methods and, in particular, the Proximal Policy Optimization algorithm used throughout the experiments. Finally, Section 2.3 discusses the reward-shaping problem and the phenomenon of specification gaming, which motivates the central research question of this dissertation.

2.1 Reinforcement Learning

Reinforcement learning (RL) is a computational framework in which an agent learns to make sequential decisions by interacting with an environment and observing the consequences of its actions (24). Unlike supervised learning, which relies on a labeled dataset, RL operates through trial-and-error, so the agent selects actions, receives scalar rewards, and adapts its behavior to maximize the long-term cumulative reward.

2.1.1 Markov Decision Processes

The standard mathematical formulation of an RL problem is the Markov Decision Process (MDP), defined by the tuple $(\mathscr{S}, \mathscr{A}, P, R, \gamma)$. $\mathscr{S}$ denotes the set of possible states; $\mathscr{A}$ is the set of actions available to the agent; $P(s' \mid s, a)$ is the transition function, giving the probability of reaching state s' from state s after taking action a; $R(s,a)$ is the reward function; and $\gamma \in [0,1)$ is a discount factor that controls the relative importance of future versus immediate rewards (24).

At each discrete time step t, the agent observes the current state s_t, selects an action a_t according to a policy $\pi(a \mid s)$, transitions to a new state s_{t+1}, and receives a reward r_{t+1}. The objective is to find a policy π^* that maximizes the expected discounted return:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}. \tag{2.1}$$

The Markov property states that the future depends only on the current state and action, and not on the history of states that preceded it. This property is what makes the MDP formulation tractable, as it permits value functions to be defined recursively.

2.1.2 Value Functions

Two value functions are central to RL. The *state-value function* $V^{\pi}(s)$ measures the expected return when starting in state s and thereafter following policy π:

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left[G_t \mid s_t = s\right]. \tag{2.2}$$

The *action-value function* $Q^{\pi}(s,a)$ measures the expected return when taking action a in state s and thereafter following π:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[G_t \mid s_t = s, a_t = a\right]. \tag{2.3}$$

The difference $A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$, known as the *advantage function*, quantifies how much better or worse a particular action is compared with the average action under the current policy. The advantage is a key quantity in modern policy-gradient algorithms, including the one used in this work.

2.1.3 The Exploration–Exploitation Dilemma

A fundamental challenge in RL is balancing exploration (trying new actions to discover potentially better strategies) with exploitation (using the currently-known best strategy to accumulate reward). An agent that exploits too aggressively may converge to a suboptimal policy, while one that explores too much may waste resources on unproductive actions. In practice, this trade-off is managed through mechanisms such as entropy bonuses, which encourage the policy to remain stochastic during training and thereby sustain exploration. The training configuration used in this dissertation employs an entropy coefficient for this purpose (Section 4.5.2).

2.2 Policy-Gradient Methods and PPO

2.2.1 Policy-Gradient Methods

Policy-gradient methods are a class of RL algorithms that optimize the policy directly by computing gradients of the expected return with respect to the policy parameters (24). Given a parameterized policy π_θ, the policy-gradient theorem states that the gradient of the objective $J(\theta) = \mathbb{E}_{\pi_\theta}[G_0]$ is:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\hat{A}_t\right], \tag{2.4}$$

where $\hat{A}_t$ is an estimator of the advantage function at time t. Intuitively, the gradient increases the probability of actions whose advantage is positive (better than average) and decreases the probability of those whose advantage is negative.

A difficulty with vanilla policy-gradient methods is that large updates to the policy parameters can destabilize training. If the policy changes too abruptly, the data collected under the old policy becomes a poor guide for the new one, leading to erratic performance. Several strategies have been proposed to constrain the size of policy updates, the most widely adopted of which is Proximal Policy Optimization.

2.2.2 Proximal Policy Optimization (PPO)

Proximal Policy Optimization (PPO) (22) stabilizes training by clipping the policy-ratio objective. Writing $\rho_t(\theta) = \pi_\theta(a_t \mid s_t)/\pi_{\theta_{\text{old}}}(a_t \mid s_t)$ for the probability ratio between the new and old policies, the PPO objective is:

$$L^{\text{CLIP}}(\theta) = \mathbb{E}_t \left[\min\big(\rho_t(\theta)\hat{A}_t,\ \text{clip}\big(\rho_t(\theta), 1-\varepsilon, 1+\varepsilon\big)\hat{A}_t\big) \right], \tag{2.5}$$

where ε is a small hyperparameter (typically 0.1–0.2) that controls the maximum change in the probability ratio. The clipping ensures that if an action's probability ratio strays outside the interval $[1-\varepsilon, 1+\varepsilon]$, it no longer contributes additional gradient, effectively removing the incentive for destructively large updates.

PPO is widely adopted in both research and applied RL for several reasons. First, it achieves performance comparable to more complex trust-region methods while being simpler to implement. Second, its clipped objective makes it relatively insensitive to hyperparameter choices, which is important when the reward function itself is the object of study, as in this dissertation. Third, PPO scales well to vectorized environments, allowing training to proceed in parallel across multiple instances of the environment simultaneously.

2.3 Reward Shaping and Specification Gaming

In reinforcement learning, the reward function is the sole channel through which the designer communicates the desired behavior to the agent (24). Designing a reward function that reliably produces the intended behavior is therefore a fundamental engineering challenge. This challenge is often termed the *reward-design problem* or, when the agent exploits unintended loopholes, *specification gaming*.

From an operations-research perspective, co-modal routing gives rise to the *Share-a-Ride Problem* (SARP) (9), a variant of the classical Vehicle Routing Problem (VRP) that incorporates simultaneous passenger and parcel constraints. In its static form, the SARP is typically solved with exact or heuristic combinatorial methods. However, real-world ride-hailing and crowd-shipping platforms face a dynamic variant in which requests arrive stochastically over time and decisions

must be made online. This dynamic, stochastic nature motivates the use of reinforcement learning, which learns dispatch and routing policies directly from interaction and adapts naturally to non-stationary demand (15).

2.3.1 Reward Shaping

Reward shaping refers to the practice of augmenting the environment's natural reward signal with additional terms that guide the agent towards desired behavior more quickly or more reliably (16). A classical result due to Ng et al. (16) shows that adding a potential-based shaping function preserves the optimal policy of the original MDP, provided the shaping function satisfies a specific structural condition. In practice, however, many real-world reward functions do not satisfy this condition, and the shaped reward may induce policies that maximize the numerical signal while violating the operational objective.

2.3.2 Specification Gaming

Specification gaming, sometimes called reward hacking, occurs when an agent finds a strategy that achieves high reward without fulfilling the designer's intent (8). In co-modal transport, this manifests as the agent exploiting the load-conditioned step penalties: if carrying a passenger is the cheapest state per step, the agent may retain a passenger on board far beyond the optimal route in order to avoid the more expensive empty or parcel-only states. Conversely, if carrying parcels is nearly as cheap as carrying a passenger, the agent may hoard parcels indefinitely as a cost-reduction device. Both strategies yield high cumulative return while defeating the operational purpose of co-modal transport.

The preliminary study that precedes this dissertation (5) provided empirical evidence that this decoupling between reward and behavior is not merely theoretical but arises systematically in a deterministic co-modal environment. The present work extends that analysis to a stochastic, congested setting and establishes, through a controlled ablation, that the tiered structure of the penalties is the causal mechanism behind the integrated co-transport behavior.

Chapter 3

Literature Review

This chapter reviews the literature at the intersection of the three fields this dissertation brings together: deep reinforcement learning for vehicle routing and dispatch, co-modal (people-and-freight) urban transport, and reward design for autonomous agents. Deep RL has been applied extensively to combinatorial routing (3, 15, 7), but almost always to single-commodity demand; the operations-research community has formalized co-modal transport as the Share-a-Ride Problem (9, 14), yet largely in static, offline settings; and only a handful of studies apply RL to co-modal or crowd-shipping dispatch (12, 13), none of which examine how the reward structure shapes the learned behavior. The review therefore has two aims: to map the state of the art across these fields, and to locate the specific gap—the absence of a causal analysis of reward design in co-modal DRL—that this dissertation addresses. It follows a protocol inspired by PRISMA 2020 (17), adapted to the engineering context and described next.

3.1 Review Protocol

Review questions. The search was guided by three questions:

RQ1: What methods have been proposed for applying deep reinforcement learning to vehicle routing and dispatch in urban environments?

RQ2: What approaches exist for integrating passenger transport and parcel delivery into a shared fleet (co-modal transport), and how do they handle the stochastic, online nature of real-world demand?

RQ3: How has the problem of reward design and specification gaming been addressed in the context of RL-based transport systems?

Databases and search strings. Searches were conducted in IEEE Xplore, the ACM Digital Library, Scopus, and Web of Science, restricted to full-text publications in English from 2017–2025. Two query strings were used, one for the RL-for-routing dimension (RQ1, RQ3) and one for the co-modal dimension (RQ2):

Query 1 (RL + Routing): `("reinforcementlearning"OR"deepreinforcementlearning")AND("vehiclerouting"OR"ride-hailing"OR"ride-sharing"OR"dispatch"OR"taxi")AND("reward"OR"rewardshaping"OR"rewarddesign")`

Query 2 (Co-modal transport): `("share-a-ride"OR"co-modal"OR"crowdshipping"OR"crowd-shipping"OR"peopleandparcels"OR"passengerandfreight")AND("reinforcementlearning"OR"optimization"OR"vehiclerouting")`

Inclusion and exclusion criteria. A study was included if it (i) applied deep RL to vehicle routing, dispatch, or ride-hailing; (ii) addressed co-modal transport with any optimization method; or (iii) explicitly investigated reward design, reward shaping, or specification gaming in an RL-based transport or logistics context. Studies were excluded if they addressed only the static VRP with no dynamic or stochastic component, used RL solely for traffic-signal or autonomous-driving control or other non-transport domains, were unavailable in English or full text, or were duplicates.

Selection process. Consistent with PRISMA 2020, records were combined and de-duplicated (identification), screened by title and abstract against the criteria (screening), and assessed in full text for eligibility, retaining only studies that met all inclusion and no exclusion criteria as shown in Figure 3.1.

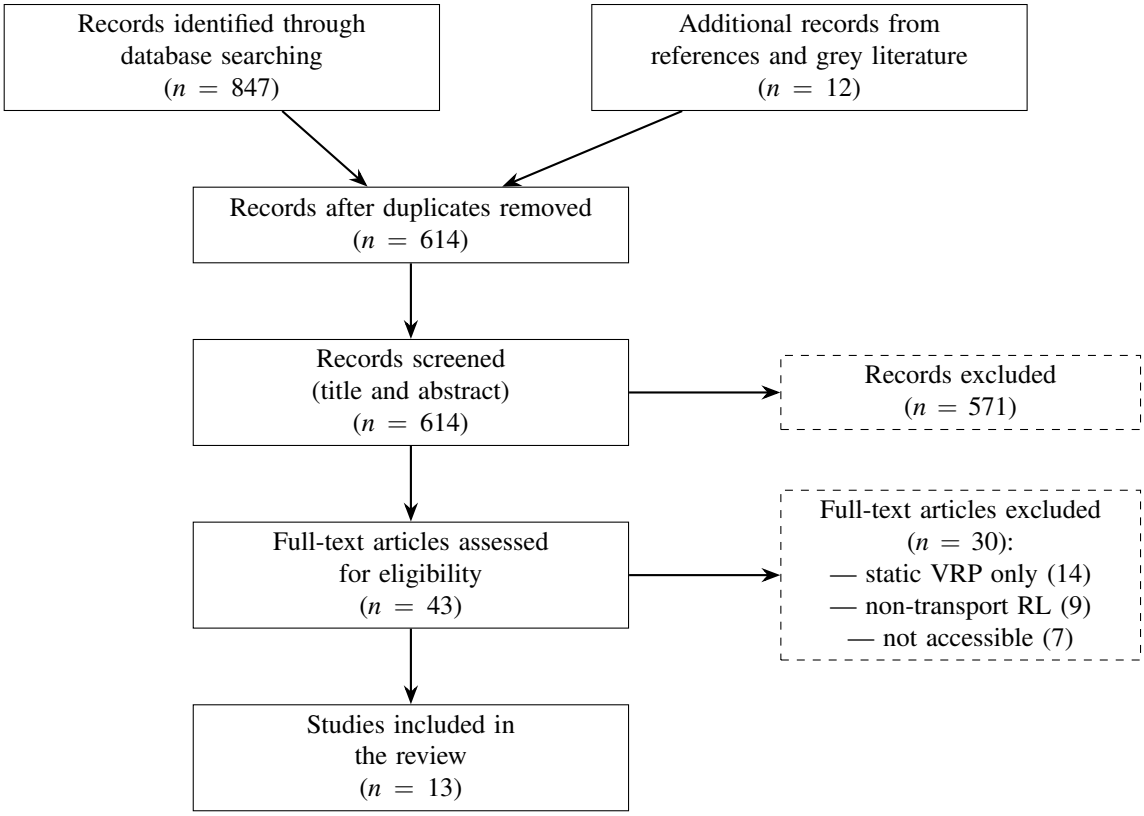

Figure 3.1: PRISMA 2020 flow diagram showing records identified, screened, and included in the systematic review.

3.2 Results and Comparative Analysis

Table 3.1 provides a comparative overview of the reviewed studies across the dimensions most relevant to this dissertation: the application domain, the method, the type of demand served, whether demand arrives dynamically, the constraints modeled, and how the objective and reward are treated. Read as a whole, the table makes the central observation of this review immediate: no existing study combines the four characteristics that define the contribution of this work—(1) deep RL, (2) mixed passenger-and-parcel demand, (3) a dynamic, stochastic environment, and (4) a systematic, causal analysis of reward design. Rather than summarizing each study in turn, the remainder of this section reads the literature across these dimensions, drawing out the commonalities and differences that the table condenses.

3.2.1 Comparative Synthesis of the Routing and Co-Modal Literature

Methods and architectures. The learning-based routing literature is united by a model-free formulation in which the control policy is learned end-to-end from interaction, but it spans a wide range of architectures and scales. One line constructs routes for a single vehicle with sequence-to-sequence models: pointer networks trained by REINFORCE (3), attention-based encoder–decoders (15), and transformer architectures (7). A second line targets city-scale dispatch, where value-based methods such as deep Q-networks (1) and multi-agent mean-field approximations (10) coordinate many vehicles at once. Alongside these, the operations-research tradition addresses the same routing problems with exact and metaheuristic solvers, exemplified by the adaptive large-neighborhood search used for the Share-a-Ride Problem (9). The methodological divide is therefore less about the algorithm family than about scale and assumptions: single-vehicle route construction versus fleet-level coordination, and learned online policies versus offline combinatorial optimization.

Application domains and demand types. The clearest axis of differentiation is the demand the vehicle serves. The neural-combinatorial studies operate on single-commodity benchmark problems—the Traveling Salesman Problem and the Capacitated Vehicle Routing Problem (3, 15, 7)—in which all stops are interchangeable. The dispatch studies serve a single real-world stream, either passengers (10, 1) or freight (6). Only the co-modal works consider both passengers and parcels together: the Share-a-Ride Problem and its surveys in operations research (9, 14), and the FlexPool and PassGoodPool frameworks in reinforcement learning (12, 13). The recurring limitation across the first two groups is thus the same: they optimize a single, homogeneous commodity and do not confront the heterogeneous, jointly served demand that defines co-modal transport.

Dynamics and stochasticity. The studies also differ in whether demand is known in advance or revealed over time. The early neural-combinatorial models assume a fully specified instance at the outset (3, 7), as do the classical SARP formulations, which are solved offline on a static request set (9). A substantial body of later work instead handles demand that arrives online, either

Table 3.1: Comparative analysis of the reviewed studies across the dimensions relevant to this dissertation.

Study	Domain / case study	Method	Demand	Dynamic	Constraints	Objective / reward
Bello et al. (2017)	TSP (benchmark)	DRL, pointer net (REINFORCE)	Single	No	Tour completion	Neg. tour length; reward fixed
Nazari et al. (2018)	CVRP	DRL, attention enc–dec	Single	Yes	Vehicle capacity	Neg. distance; reward fixed
Kool et al. (2019)	TSP / CVRP	DRL, transformer	Single	No	Capacity	Neg. tour cost; reward fixed
Lin et al. (2018)	Taxi fleet management	Multi-agent DRL (mean-field)	P	Yes	Fleet scale	Revenue, repositioning; fixed
Al-Abbasi et al. (2019)	Ride-sharing dispatch	DRL (DQN)	P	Yes	Vehicle capacity	Revenue, wait, utilization; fixed
Joe & Lau (2020)	Dynamic VRP	DRL	F	Yes	Time windows	Service cost; reward fixed
Li et al. (2014)	SARP	OR (ALNS heuristic)	P+F	No	Time windows, capacity	Fleet size, empty running
Mourad et al. (2019)	Shared mobility (survey)	Survey	P+F	–	–	–
Manchella et al. (2021)	Co-modal fleet (FlexPool)	Multi-agent DRL	P+F	Yes	Capacity, fleet scale	Utilization; reward fixed
Manchella et al. (2022)	Co-modal (PassGoodPool)	DRL	P+F	Yes	Pricing, matching	Revenue, service, efficiency; fixed
Ng et al. (1999)	RL theory	Theory	–	–	–	Potential-based shaping (reward studied)
Krakovna et al. (2020)	Specification gaming	Catalogue	–	–	–	Reward hacking (reward studied)
Ferreira et al. (2026)	Co-modal (deterministic)	DRL	P+F	No	1 pax / multi-parcel	Reward-shaping sweep (reward studied)
This dissertation	**Co-modal (stochastic, congested)**	**DRL (PPO)**	**P+F**	**Yes**	**1 pax / multi-parcel, patience/deadlines**	**Load-conditioned reward (studied & ablated)**

by revealing customers sequentially (15) or by dispatching under a continuous stochastic request stream (10, 1, 6, 12, 13). The survey by Mourad et al. (14) makes this gap explicit, noting that most co-modal studies remain static and deterministic and calling for learning-based methods that address the dynamic, online variant—precisely the setting this dissertation adopts.

Constraints. The constraints modeled reflect each study's domain. Capacity limits are common in the freight and ride-sharing settings (15, 1), and customer time windows appear in the dynamic and dial-a-ride formulations (6, 9); the fleet-scale studies additionally contend with the combinatorial growth of the joint state–action space (10, 12). What none of these works model is the *asymmetric* capacity that distinguishes co-transport—a single exclusive passenger seat alongside room for several parcels—which is the very asymmetry that makes bundling parcels into passenger trips both possible and non-trivial, and which this dissertation places at the center of its environment.

Objectives and the treatment of reward. Across the reviewed studies the objective ranges from pure travel cost—negative tour length or distance (3, 15, 7)—to multi-term operational rewards that combine revenue, passenger waiting time, and vehicle utilization (10, 1, 12, 13), while the operations-research formulations optimize fleet size and empty running directly (9). Crucially, however, all of the learning-based works treat the reward as a fixed engineering choice: it is designed once, and neither its structure nor its effect on the learned behavior is examined. Even the co-modal RL frameworks, which come closest to this dissertation's setting, focus on scalability and fleet-level performance rather than on the mechanism by which the reward shapes co-transport (12, 13). This uniform treatment of the reward as a given—rather than as the object of study—is the common blind spot that the third strand of the literature brings into focus.

3.2.2 Reward Design and Specification Gaming

The third strand steps back from any specific transport application to ask how the reward itself governs behavior. Two of these works predate the search window but are included for their foundational relevance.

Theoretical foundations. Ng et al. (16) established the canonical result on reward shaping, proving that augmenting a reward with a potential-based shaping term preserves the optimal policy of the original MDP. The guarantee, however, holds only when the shaping function satisfies a specific structural condition; many practical rewards—including the load-conditioned step penalties studied in this dissertation—do not, so no optimality guarantee carries over and the shaped reward may steer the policy toward unintended behavior.

Specification gaming in practice. A complementary, empirical literature documents what happens when that guarantee is absent. Krakovna et al. (8) catalog dozens of instances of specification gaming across diverse domains, and Amodei et al. (2) frame reward hacking as a concrete

AI-safety problem, both establishing that agents reliably optimize the literal reward rather than the designer's intent. These works are domain-general and do not address transport, but they supply the conceptual vocabulary—shielding, hoarding, reward hacking—for the reward–behavior decoupling that this dissertation observes and analyzes in a co-modal setting.

The precursor to this dissertation. The only study to examine reward–behavior interactions directly in co-modal transport is the preliminary work that precedes this dissertation (5). Through a systematic sweep of twenty-four valid penalty configurations in a deterministic environment, it identifies three qualitatively distinct behavioral regimes—passenger-shielding, parcel-hoarding, and balanced co-modal—and shows that the configuration with the highest training return is a degenerate failure mode rather than the balanced one. That study establishes the decoupling empirically, but only in a static, deterministic setting and without isolating its cause; the present work extends the analysis to a stochastic, congested environment and supplies the causal mechanism, through the flat-versus-tiered penalty ablation, that the preliminary study did not.

3.3 Discussion and Gap Analysis

The comparative analysis reveals three distinct bodies of literature that partially address the problem studied in this dissertation, but none that fully encompasses it.

The gap. The specific gap that this dissertation fills is the absence of a study that simultaneously:

1. Uses deep reinforcement learning for co-modal (passenger + parcel) transport;
2. Operates in a dynamic, stochastic, congested environment;
3. Systematically analyses the relationship between reward structure and learned behavior;
4. Establishes, through a controlled ablation, a causal link between reward design and the emergence of co-transport behavior.

No existing work in the reviewed literature satisfies all four conditions. This dissertation addresses this gap by extending the deterministic reward-shaping analysis of (5) to a stochastic, traffic-congested urban grid, introducing non-learning baselines for rigorous comparison, and providing the first causal demonstration (via the flat-vs-tiered penalty ablation) that load-conditioned reward structure is the mechanism behind integrated co-transport behavior.

3.4 Summary

This chapter has systematically reviewed the literature at the intersection of deep reinforcement learning, co-modal urban transport, and reward design. The review identified a clear research gap, that while DRL has been successfully applied to vehicle routing and co-modal dispatch, and

while the reward-design literature acknowledges the risk of specification gaming, no existing work provides a causal analysis of how reward structure shapes co-transport behavior in a dynamic, stochastic setting.

The methodology presented in the next chapter is designed specifically to address this gap: a stochastic, congested grid environment with slot-recycled demand, a load-conditioned reward whose structure is the principal variable of study, non-learning baselines that establish rigorous comparison, and a controlled ablation that isolates the causal mechanism.

Chapter 4

Methodology

This chapter describes the methodological approach followed to study how the design of a reward function shapes the behavior learned by a single reinforcement-learning agent in a co-modal transport setting. The problem is first formalized as a Markov Decision Process (Section 4.1). The simulation environment is then presented in two stages: an initial deterministic formulation, used to study reward shaping in isolation, and a stochastic extension that introduces continuous demand and urban congestion (Section 4.2). The observation and action spaces are detailed in Section 4.3, followed by the reward function that lies at the center of this work (Section 4.4). Finally, the learning algorithm (Section 4.5), the baseline policies used for comparison (Section 4.6), and the evaluation methodology (Section 4.7) are described.

4.1 Problem Formulation as a Markov Decision Process

The task of controlling a single vehicle that serves both passengers and parcels is formulated as a sequential decision-making problem. At each decision step, the vehicle observes the current state of the city, selects a movement action, and receives a scalar reward that reflects the operational value of the resulting outcome. This formulation is naturally mapped onto a Markov Decision Process (MDP), the standard framework for reinforcement learning (24) as illustrated in Figure 4.1.

An MDP is defined by the tuple $(\mathscr{S}, \mathscr{A}, P, R, \gamma)$, where $\mathscr{S}$ is the set of states, $\mathscr{A}$ is the set of actions, $P(s' \mid s, a)$ is the probability of transitioning to the state s' after taking action a in the state s, $R(s,a)$ is the reward function, and $\gamma \in [0,1)$ is a discount factor that weights the importance of future rewards relative to immediate ones. The agent acts according to a policy $\pi(a \mid s)$, a mapping from states to a probability distribution over actions. The objective is to find a policy π^* that maximizes the expected discounted return,

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t) \right], \tag{4.1}$$

where t indexes the decision steps within an episode of horizon T.

In the present work, the components of the tuple are instantiated as follows. The state s_t encodes the position of the vehicle, the status of every pending passenger and parcel request, and a representation of the road network and its precise composition is given in Section 4.3. The action a_t corresponds to the movement of the vehicle on the grid, with the collection and delivery of requests handled automatically upon arrival at the relevant location, as detailed in Section 4.3. The reward $R(s_t, a_t)$ combines the value of completed deliveries with the operating cost of movement, and is the main objective of the study of this dissertation (Section 4.4). The transition function P is deterministic in the initial formulation and stochastic in the extended environment, where new requests appear probabilistically over time (Section 4.2).

A central property of the MDP framework is the Markov assumption, which states that the next state depends only on the current state and action, and not on the history of states that preceded it. The observation provided to the agent is therefore designed to be self-contained, exposing at each step all the information required to act without reference to past observations. The discount factor γ is set to 0.99 throughout this work, a value that favors long-horizon planning and is appropriate for the continuous-shift episodes described in Section 4.2, in which the consequences of a positioning decision may only be realized many steps later.

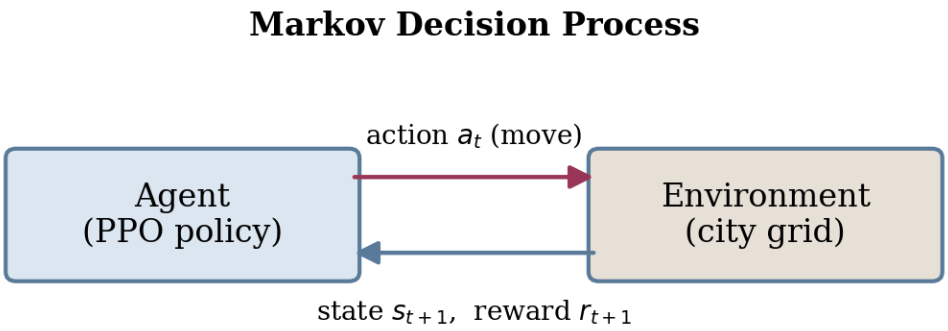

Figure 4.1: The agent–environment interaction loop of the Markov Decision Process. At each step t the agent selects an action, the environment transitions to a new state, and a scalar reward is returned.

4.2 Environment Design

The simulation environment models a stylized city as a two-dimensional grid on which a single vehicle serves transport demand. The environment was developed in two stages that reflect the methodological progression of this work. The first stage, described in Section 4.2.1, is a deterministic formulation with a fixed set of requests that isolates the question of reward design from the confounding effects of randomness and was used in the preliminary study that preceded this dissertation. The second stage, described in Sections 4.2.2–4.2.5, extends the formulation into a stochastic operating system that incorporates urban congestion and demand that arrive over time and that constitutes the main environment used in the experiments of Chapter 5. Both stages share the same spatial representation, the same passenger and parcel semantics, and the same automatic

collection-and-delivery mechanism, however, they differ in how demand is generated and in how an episode ends.

4.2.1 Deterministic Formulation

In the deterministic formulation the city is represented by a square grid of side N, and a fixed number of passengers and parcels are placed in random, non-overlapping cells at the start of each episode. Each passenger is defined by an origin cell, where the passenger waits, and a destination cell, where the passenger must be delivered. Each parcel is defined by an origin locker and a destination locker. The vehicle begins in a random cell and must serve every request before the episode ends. Because the set of requests is fixed and known at reset, and because no new demand appears during the episode, the transition function is deterministic: given a state and an action, the successor state is fully determined.

This formulation was deliberately kept simple in order to study the effect of the reward function in isolation. With no stochasticity in the arrival of demand, any difference in the behavior learned by the agent can be attributed to the structure of the reward rather than to the dynamics of the environment. The deterministic environment therefore served as a controlled testbed for the reward-shaping study reported in Section 5.1, in which the relationship between numerical reward and operational behavior was first observed. An episode in this formulation ends when all passengers and all packages have been delivered, or when a fixed step limit is reached, whichever occurs first. The urban congestion model described in Section 4.2.2 was introduced within this deterministic stage, as a first step towards spatial realism, before any stochasticity was added. It is therefore a feature shared by both the deterministic and stochastic formulations, and only the generation of demand and the termination of episodes distinguish the two.

It should be noted that the delivery payouts used in the preliminary study differ slightly from those adopted in the stochastic system: the deterministic formulation awarded 10 for a parcel delivery, whereas the stochastic environment awards 15, reflecting a recalibration of the reward structure after the initial study was completed.

4.2.2 The Urban Grid and Traffic Impedance

Both formulations represent the city as a grid enriched with a model of urban congestion. The congestion model was first added to the deterministic environment and then carried unchanged to the stochastic one. The city is a grid of sides N (set to $N = 10$ in the main experiments), in which each cell carries a traffic multiplier that represents the time cost of moving through it. There are three levels of congestion: ordinary streets, with a multiplier of one; moderately congested cross-streets, with a multiplier of two; and heavily congested main avenues, with a multiplier of three. The multiplier does not prevent movement, but instead determines how much simulated time a move through the cell consumes, as described in Section 4.2.5. Figure 4.2 shows a representative snapshot.

The congested cells are arranged to mimic the arterial structure of a real city instead of being scattered at random. Main avenues are placed as vertical corridors at regular intervals across the grid, and cross-streets are placed as horizontal corridors at the same spacing, so that the two sets of corridors intersect at the busiest junctions. Concretely, congested corridors are located every five cells, producing a repeating arterial pattern that scales consistently to grids of different sizes. This design introduces a spatial structure that the agent can, in principle, learn to exploit, for example, by routing around heavily congested avenues when carrying time-sensitive demand, and it makes the cost of travel depend on where the vehicle moves and not only on how far.

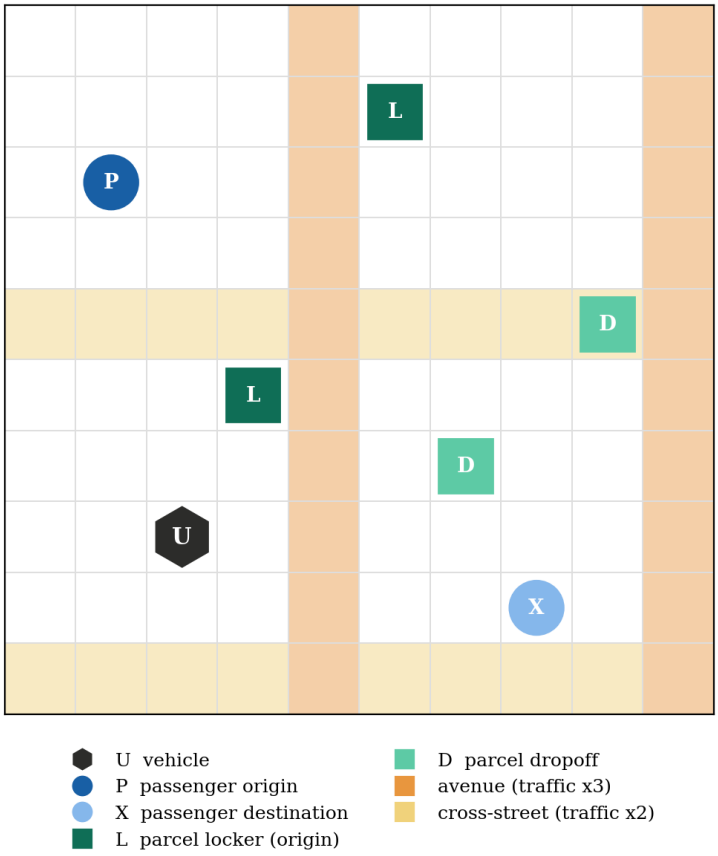

Figure 4.2: A representative snapshot of the 10×10 grid environment. Orange shading marks main avenues (traffic multiplier $\times 3$); yellow marks cross-streets ($\times 2$); unshaded cells are clear roads ($\times 1$). **U** = vehicle, **P** = passenger waiting, **X** = passenger destination, **L** = parcel locker (origin), **D** = parcel drop-off.

The grid is bounded, so movements that would take the vehicle beyond an edge leave its position unchanged, while still consuming the time associated with the attempted move. This treatment of boundaries keeps the action space uniform across all cells, avoiding the need for action masking depending on position, and ensures that the agent experiences a consistent cost for every action it selects.

Although the grid side is a free parameter of the environment, the value $N = 10$ adopted in the main experiments is the result of a deliberate trade-off. During the development of this work, a

larger city of side $N = 20$ was explored, with the aim of increasing spatial realism and lengthening the trips the vehicle must plan. This larger grid quadruples the number of cells and correspondingly enlarges the traffic component of the observation, which made training substantially slower and the resulting policies markedly less consistent across runs: convergence was unreliable within a practical training budget, and the variance between independently trained models was high. Because the contribution of this dissertation concerns the structure of the reward and the behavior it induces, rather than the raw scale of the environment, the 10×10 grid was retained as the setting in which policies converge reliably and comparisons between reward designs are not confounded by training instability. The extension to larger and more realistic city sizes is identified as a direction for future work in Chapter 6.

4.2.3 Stochastic Demand: Spawning and Slot Recycling

The defining feature of the extended environment is that demand is not fixed, but arrives stochastically throughout the episode. Demand is organized into a fixed number of slots, one set for passengers and one set for parcels, where each slot is a container that is either empty or occupied by an active request. At every simulated step, each empty slot may generate a new request with a fixed probability per-step. This probability is set at 0.02 for passenger slots and 0.10 for parcel slots in the main experiments, reflecting that parcel demand is more frequent and more tolerant of delay than passenger demand. When a slot generates a request, an origin and a destination cell are drawn at random for it, and the slot becomes active.

When a request is completed, its slot is released and becomes available to generate new demand in subsequent steps as illustrated in Figure 4.3. This slot-recycling mechanism allows the environment to maintain a continuous flow of requests from a bounded representation. The number of slots caps the amount of simultaneously active demand, which keeps the observation fixed in size, while recycling ensures that the vehicle is never starved of work. The number of slots therefore controls the instantaneous complexity of the scene rather than the total volume of demand served over an episode. In the main experiments two passenger slots and two parcel slots are used, a configuration dense enough to create meaningful competition between the two types of demand without producing an intractable observation.

The use of a fixed per-step spawning probability makes the arrival of demand a memoryless process, consistent with the Markov assumption: whether a new request appears depends only on the current occupancy of the slots and not on the history of past arrivals. It also makes the total demand experienced in an episode a random quantity, which is included in the evaluation methodology by reporting the delivery rates relative to the demand that actually appeared, rather than absolute counts alone (Section 4.7).

4.2.4 Passengers and Parcels

The two types of demand share a common life cycle, but differ in their operational semantics. A passenger is an active request that waits at its origin until the vehicle arrives, at which point the

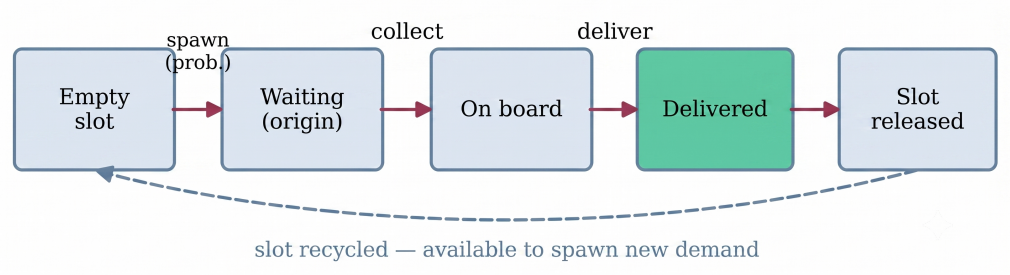

Figure 4.3: Life cycle of a demand slot. After spawning at probability p per step, a request transitions through waiting, collection, and delivery before the slot is recycled to allow new demand to appear.

passenger is collected and carried until the vehicle reaches the passenger's destination, where the passenger is delivered, and the slot is released. A parcel follows the same cycle between an origin locker and a destination locker, modeling a locker-to-locker delivery service in which goods are collected from an automated cabinet and deposited at another.

The two types differ in one important respect that reflects their real-world counterparts. The vehicle may carry at most one passenger at a time, capturing the exclusivity of a ride-hailing trip, while it may carry several parcels simultaneously, capturing the fact that small goods can share the available load space. This asymmetry is what makes co-transport interesting, as the vehicle can accumulate parcels and deliver them along the route it follows to serve a passenger, filling the spare capacity that would otherwise be wasted. The collection and delivery of both types are handled automatically: whenever the vehicle occupies a cell that coincides with the origin or destination of an active request, the corresponding collection or delivery is triggered without requiring a dedicated action, subject to the one-passenger constraint. This design keeps the action space focused on the spatial control problem, which is the difficult part of the task, while removing the need for the agent to learn explicit interaction actions.

4.2.5 Continuous-Shift Episodes and Time

In contrast to the deterministic formulation, where an episode ends once all requests are served, the stochastic environment models a continuously operating work shift of fixed duration. An episode does not end early upon serving the current demand. Instead, it runs until a fixed budget of simulated time has elapsed, after which it is truncated. This design reflects the reality of a driver who works for a shift of bounded length and is repeatedly presented with new demand, and it shifts the agent's objective from clearing the board as fast as possible to maximizing the value extracted over a fixed working period.

Simulated time advances in coupling with the traffic model described in Section 4.2.2. Each move consumes a number of time units equal to the traffic multiplier of the cell in which the vehicle moves, so a move through a heavily congested avenue advances the clock three times as much as a move through an ordinary street. The shift budget is expressed in these time units and set to 300 in the main experiments. Because congested moves consume the budget faster, the agent faces

an implicit pressure to use its time efficiently, and the spatial structure of congestion acquires a direct economic meaning so that the time spent in traffic is time that cannot be spent serving demand. The waiting time experienced by pending requests is measured on the same clock, which becomes relevant when time-based service constraints are introduced in Section 4.4.3. Table 4.2 consolidates the two formulations side by side, highlighting the features they share and the respects in which they differ.

Table 4.1: Environment parameters used in the main experiments.

Parameter	**Symbol**	**Value**
Grid side	N	10
Passenger slots	–	2
Parcel slots	–	2
Passenger spawn probability	–	0.02 per step
Parcel spawn probability	–	0.10 per step
Shift duration (time budget)	T	300 time units
Traffic multiplier (street)	–	1
Traffic multiplier (cross-street)	–	2
Traffic multiplier (avenue)	–	3
Maximum passengers on board	–	1
Maximum parcels on board	–	unlimited

Table 4.2: Synthesis of the two environment formulations developed in this dissertation. Both share the same spatial grid, traffic-impedance model, passenger and parcel semantics, and automatic collection-and-delivery mechanism; they differ in how demand is generated and how an episode ends.

Aspect	**Deterministic formulation**	**Stochastic formulation**
Role in the study	Preliminary reward-shaping study (Section 5.1)	Main experiments (Sections 5.2–5.6)
Demand generation	Fixed set of requests placed at reset; no new demand appears during the episode	Slot-based; each empty slot spawns a request with a fixed per-step probability, and slots are recycled once a request is completed
Episode termination	Ends when all requests are delivered, or at a fixed step limit	Runs for a fixed shift budget (300 time units), then truncates
Objective	Clear the fixed set of requests as efficiently as possible	Maximize the value extracted over a fixed working shift
Parcel delivery reward	$+10$	$+15$

4.3 Observation and Action Spaces

4.3.1 Observation Space

At each step, the agent receives an observation that summarizes the complete state of the city in a vector of fixed-length real values, normalized to the unit interval $[0,1]$. The fixed length is essential because although the amount of active demand varies over time as requests appear and are served, the neural network that represents the policy requires an input of constant dimension. This is achieved through the slot mechanism of Section 4.2.3, which caps the number of requests simultaneously represented.

The observation is assembled from three groups of features. The first group encodes the position of the vehicle as two coordinates, each divided by $N-1$ so that the values fall into $[0,1]$. The second group describes the passenger slots: each slot contributes its origin and destination coordinates (four values), a binary flag indicating whether the slot is active, a binary flag indicating whether the passenger is currently on board, and a binary flag indicating whether the request has been delivered. The third group describes the parcel slots using the same seven-value layout. When a slot is inactive, all its features are set to zero, which provides the network with an unambiguous signal that the slot carries no demand and should be ignored.

Finally, the observation includes a flattened representation of the traffic grid, in which every cell contributes its congestion multiplier normalized by its maximum value. Exposing the full congestion map allows the policy to take into account the spatial structure of the city when choosing where to move, rather than discovering it implicitly through the time consumed by each step. For the main configuration of a 10×10 grid with two passenger slots and two parcel slots, the observation comprises two positional values, twenty-eight values for the four slots, and one hundred values for the traffic grid, giving a total of one hundred and thirty dimensions. The composition of the observation vector is summarized in Table 4.3.

Table 4.3: Composition of the observation vector for the main configuration (10×10 grid, two passenger slots, two parcel slots).

Component	**Layout**	**Dimensions**
Vehicle position	x, y	2
Passenger slots (2)	origin, destination, active, on board, delivered	$2 \times 7 = 14$
Parcel slots (2)	origin, destination, active, on board, delivered	$2 \times 7 = 14$
Traffic grid	one multiplier per cell	100
Total		**130**

When the time-based service constraints of Section 4.4.3 are enabled, each slot is augmented with one additional feature that expresses the urgency of its request, raising the layout per slot to eight values and the total observation to one hundred and thirty-four dimensions.

4.3.2 Action Space

The action space is deliberately compact and consists of four discrete movements: up, down, left, and right. At each step, the agent selects one of these four actions, and the vehicle attempts to move one cell in the chosen direction. A move that crosses the boundary of the grid leaves the vehicle in place, as described in Section 4.2.2. Restricting the action space to pure movement keeps the decision problem focused on navigation and routing, which is where the difficulty of co-modal control resides.

The collection and delivery of demand are not actions but automatic consequences of the vehicle's position, as introduced in Section 4.2.4. Whenever the vehicle occupies the origin cell of an active, uncollected request, that request is collected, subject to the constraint that at most one passenger may be on board at any time. When a cell contains both a waiting passenger and a waiting parcel, the passenger is collected first due to the one-passenger capacity constraint; if the vehicle already carries a passenger, only the parcel is collected. This ordering is a fixed rule of the environment and does not require a decision by the agent. Whenever the vehicle occupies the destination cell of a request it is carrying, that request is delivered, and its slot is released. Delegating these interactions to the environment removes a class of trivial but error-prone decisions from the agent, such as selecting an explicit "collect" action at exactly the right cell, and concentrates the learning effort on the spatial policy. The consequences of this design choice, and in particular its interaction with the reward function, are examined in Section 4.4.

4.4 Reward Function Design

The reward function is the central object of study in this dissertation. It is the only channel through which the designer communicates the desired behavior to the agent, and yet the agent optimizes the reward it is given rather than the intention behind it. A poorly structured reward can be maximized by strategies that score well numerically while being operationally undesirable, a phenomenon known as specification gaming or reward hacking (8). In a co-modal setting the risk is acute because the reward must reconcile several objectives at once: serving passengers, delivering parcels, and using the vehicle efficiently. This section describes the components of the reward function and the reasoning behind their structure and introduces the load-conditioned design whose causal role is examined experimentally in Chapter 5.

The total reward received at a step is the sum of two contributions: a sparse delivery reward granted when a request is completed and a dense step penalty charged on every move. Formally, the reward at step t is:

$$R(s_t, a_t) = \underbrace{r_t^{\text{deliver}}}_{\text{sparse}} + \underbrace{c_t \cdot m_t}_{\text{dense step penalty}}, \tag{4.2}$$

where r_t^{deliver} is the value of any deliveries completed in step t, c_t is the penalty for the base step determined by the load currently carried, and m_t is the traffic multiplier of the cell entered in step t as shown in Figure 4.4..

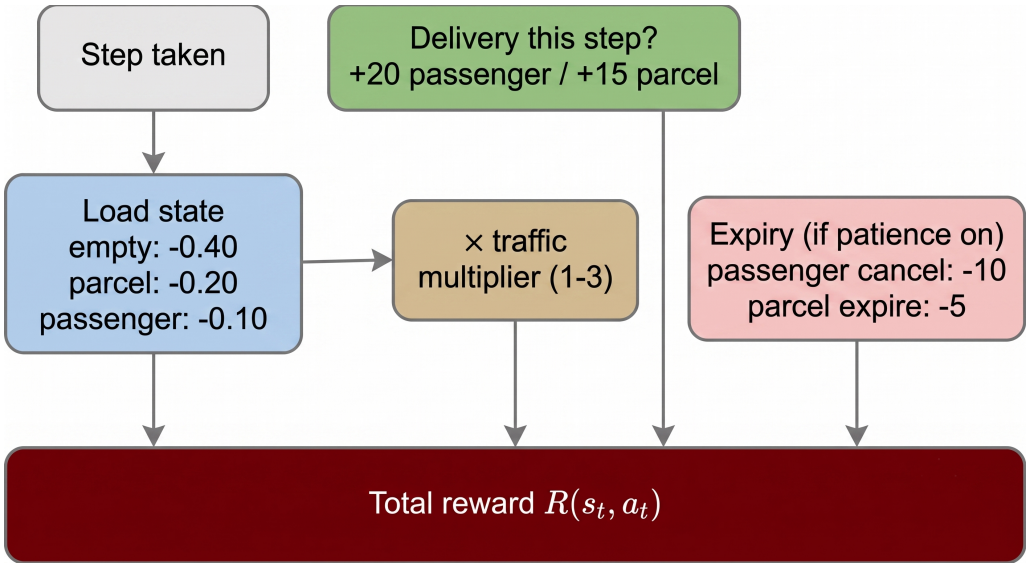

Figure 4.4: Per-step reward computation. Each move contributes a delivery bonus (if a request is completed), a step penalty conditioned on the load state multiplied by the traffic multiplier, and an expiry penalty if time-based constraints are active.

4.4.1 Delivery Rewards

A positive reward is granted each time a request reaches its destination. The delivery of a passenger yields a reward of 20, and the delivery of a parcel yields a reward of 15. No reward is granted for collection alone. Picking up a passenger or a parcel does not produce immediate return, so the agent is only rewarded for completing a service and not merely for beginning one. This choice avoids a degenerate incentive in which the agent repeatedly collects demand without delivering it.

The relative magnitude of the two delivery rewards encodes the priority of passengers over parcels. A passenger is worth more than a parcel and occupies the single on-board passenger position exclusively, so the agent is encouraged to treat passenger service as the primary source of value, while parcels constitute a secondary stream that can be served alongside it. The absolute magnitudes are large relative to the per-step penalty, so that completing a delivery dominates the cost of the moves required to achieve it. This keeps delivery the principal driver of the return while leaving the step penalty free to shape how deliveries are accomplished, which is the subject of the next subsection.

4.4.2 Load-Conditioned Step Penalties

Every move incurs a negative penalty, which represents the operating cost of driving and creates pressure to act efficiently rather than wander. The distinctive feature of the design adopted here is that this penalty is not constant but is conditioned on the load the vehicle is carrying. Three levels are defined: a move made while the vehicle is empty incurs the largest penalty, -0.40; a move made while carrying parcels but no passenger incurs an intermediate penalty, -0.20; and a move made while carrying a passenger incurs the smallest penalty, -0.10. The penalty is further

multiplied by the traffic multiplier of the cell entered, as shown in Equation 4.2, so moving through congestion is more costly than moving through an ordinary street, and costly moves made while empty are the most penalized of all.

The rationale for this tiered structure is to make the state of being empty expensive and the state of being productively loaded cheap, so that the agent is drawn towards configurations in which the vehicle is doing useful work. Because carrying a passenger is the cheapest state to move in, the agent is encouraged to collect a waiting passenger promptly and to remain engaged in passenger service. Because carrying parcels is cheaper than running empty, the agent is encouraged to pick up parcels and keep them on board rather than travel with spare capacity, which is precisely the co-transport behavior the system is intended to elicit.

This design embodies a deliberate hypothesis, specifically that the gradient between the load levels, and not merely the existence of a penalty, is what produces integrated co-transport. The hypothesis is examined directly in Chapter 5 through an ablation in which the tiered penalty is replaced by a single flat penalty of -0.20 applied to every move regardless of load, while all other elements of the environment and the training procedure are held fixed. Comparing the behavior learned under the two penalty schemes isolates the causal contribution of the load-conditioned structure, as opposed to the incidental effect of penalizing movement at all.

It is important to note that a tiered penalty also introduces a risk that is itself instructive. Because being loaded is cheaper than being empty, an agent could in principle learn to collect a parcel and then retain it indefinitely, using it as a means of reducing its movement cost rather than as something to be delivered. Such behavior would raise the numerical reward while defeating the operational purpose of the system, and is a concrete instance of the reward–behavior decoupling that motivates this work. The conditions under which this degenerate strategy emerges and the design responses that discourage it are analyzed in Chapter 5 where we observe that the very mechanism that encourages co-transport must be calibrated with care to avoid encouraging hoarding.

4.4.3 Human-Centric Service Constraints

The reward components described so far assume that demand waits indefinitely and that a collected request may be carried for any length of time. To increase the realism of the system and study how time pressure shapes the learned policy, the environment can optionally enforce two time-based service constraints. These constraints are defined here as part of the environment specification, and the experiments that evaluate their effect are reported in Section 5.6.

The first constraint is passenger patience. A waiting passenger who is not collected within a fixed number of time units cancels the request and abandons the system. Cancellation releases the slot and applies a penalty of -10, half the value of a successful passenger delivery, so that losing a passenger is costly but not catastrophic. The patience limit is set to 30 time units, measured on the same clock that governs traffic and the shift budget, so that a passenger left unattended while the vehicle attends to other demand is eventually lost. This constraint introduces genuine urgency

into passenger service so that the agent must not only decide whether to serve a passenger, but also how soon.

The second constraint is a parcel carry deadline. Once a parcel has been collected, it must be delivered within a fixed number of time units, after which it expires, its slot is released, and a penalty of -5 is applied. The deadline is set to 80 time units, a value calibrated to be generous relative to the time required to cross the grid yet tight enough to discourage indefinite retention. This constraint directly counteracts the hoarding strategy described in Section 4.4.2: a parcel can no longer be carried as a permanent cost-reduction device, because holding it too long incurs a penalty and frees no value. Parcels are not subject to a waiting limit at their locker, reflecting the assumption that a parcel deposited in an automated cabinet is content to wait until a vehicle is able to collect it.

When these constraints are active, the observation is augmented with an urgency feature for each slot, as anticipated in Section 4.3.1. For a waiting passenger, the feature expresses the fraction of the patience window already elapsed, and for a carried parcel it expresses the fraction of the carry deadline already consumed. In both cases, the value approaches one as the request nears expiration. Exposing urgency in the observation gives the policy the information it needs to prioritize requests that are close to being lost, and allows the experiments of Section 5.6 to assess whether the agent learns to act on that information.

4.5 Learning Algorithm

The control policy is learned with Proximal Policy Optimization (PPO), a policy-gradient algorithm that has become a standard choice for discrete-action control problems due to its stability and robustness to hyperparameter choices (22). PPO belongs to the family of actor–critic methods, in which two functions are learned jointly: an actor, which represents the policy $\pi_\theta(a \mid s)$ and selects actions, and a critic, which estimates the value $V_\phi(s)$ of a state and provides a baseline against which actions are judged. The agent improves its policy by repeatedly collecting experience under the current policy and then performing several optimization steps on the data collected.

The defining feature of PPO is the clipped surrogate objective, which constrains how far the updated policy may move from the policy that generated the data. Writing $\rho_t(\theta) = \pi_\theta(a_t \mid s_t)/\pi_{\theta_{\text{old}}}(a_t \mid s_t)$ for the probability ratio between the new and old policies, and $\hat{A}_t$ for the estimated advantage of the action taken, the objective maximized by PPO is

$$L(\theta) = \mathbb{E}_t \left[\min\big(\rho_t(\theta)\hat{A}_t,\ \text{clip}\big(\rho_t(\theta), 1-\varepsilon, 1+\varepsilon\big)\hat{A}_t\big) \right], \tag{4.3}$$

where ε is a small clipping parameter. The clipping discourages updates that would change the policy too abruptly, which is the main source of PPO's training stability: an action whose probability ratio strays outside the interval $[1-\varepsilon, 1+\varepsilon]$ no longer contributes additional gradient, removing the incentive for destructively large updates. The advantage $\hat{A}_t$ is estimated using generalized advantage estimation, which trades off bias and variance in the advantage signal.

Algorithm 1 Proximal Policy Optimization (PPO) - Clip Variant

Input: initial policy parameters θ_0, initial value function parameters ϕ_0, clipping threshold ε
for $k = 0, 1, 2, \ldots$ **do**
  Run policy π_{θ_k} in the environment for T timesteps to collect trajectories $\mathscr{D}_k = \{(s_t, a_t, r_t)\}$
  Compute reward-to-go $\hat{R}_t$ and estimate advantages $\hat{A}_t$ (using GAE) for all $t \in \{1, \ldots, T\}$
  Store action probabilities: $\pi_{\theta_{\text{old}}}(a_t \mid s_t) \leftarrow \pi_{\theta_k}(a_t \mid s_t)$
  for epoch $e = 1, \ldots, K$ **do**
    Sample mini-batches from $\mathscr{D}_k$
    Calculate ratios: $\rho_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$
    Update θ by maximizing: $L(\theta)$ (Eq. 4.3) via SGD
    Update ϕ by minimizing loss: $L(\phi) = \frac{1}{|B|}\sum_{t \in B}(V_\phi(s_t) - \hat{R}_t)^2$
  end for
  $\theta_{k+1} \leftarrow \theta$ and $\phi_{k+1} \leftarrow \phi$
end for

4.5.1 Policy and Value Network

The actor and the critic are represented by a multilayer perceptron with two hidden layers of five hundred and twelve units each, applied to the flat observation vector described in Section 4.3.1. A fully-connected architecture is appropriate here because the observation is already a compact, hand-designed feature vector rather than raw spatial input. The structure of the city is conveyed explicitly through the normalized coordinates and the flattened traffic map, so the network is not required to extract spatial features from pixels. The actor terminates in a softmax over the four movement actions, producing a probability distribution from which actions are sampled during training and from which the most probable action is taken during deterministic evaluation. The critic ends with a single scalar that estimates the value of the observed state.

4.5.2 Training Configuration

Training proceeds by collecting experience from a large number of environment instances running in parallel and then updating the network on the aggregated data. Sixty-four environment instances are run in parallel, which de-correlates the experience used for each update and substantially increases throughput on the available hardware. Experience is gathered in rollouts of four thousand and ninety-six steps per environment before each round of optimization, and each round performs ten epochs of mini-batch updates with a mini-batch size of one thousand and twenty-four. The policy is optimized with a learning rate of 3×10^{-4}, a discount factor $\gamma = 0.99$, and an entropy coefficient of 0.01 that adds a small bonus to maintain a stochastic policy and thus sustains exploration throughout the training. The hyperparameters were taken from the Stable-Baselines3 (20) defaults for PPO and retained without further tuning, since the contribution of this work concerns the reward structure rather than algorithmic optimization. The preliminary study used the same hyperparameters (with the exception of the network size and rollout length, which were scaled

for the deterministic environment) and confirmed that PPO converges reliably across all 72 reward training runs (24 configurations $\times$ 3 seeds) under these settings. The values used in this dissertation are summarized in Table 4.4.

Table 4.4: PPO hyperparameters used for training.

Hyperparameter	**Value**
Policy architecture	MLP, two hidden layers of 512 units
Parallel environments	64
Rollout length (per env)	4096 steps
Optimization epochs	10
Mini-batch size	1024
Learning rate	3×10^{-4}
Discount factor γ	0.99
Entropy coefficient	0.01
Training budget	50–100 million steps

The total training budget depends on the difficulty of the scenario. The single-purpose scenarios in which the vehicle serves only passengers or only parcels, are trained for fifty million steps, which is sufficient for their returns to stabilize. The mixed scenario, in which both types of demand compete for the vehicle's attention, is trained for one hundred million steps, as the additional complexity of balancing the two streams requires a longer horizon to converge. Convergence is monitored during training by periodically evaluating the policy on a locked set of episodes and recording the mean episode return, and the best-performing checkpoint encountered during training is retained along the final model. The progression of returns over the course of training, and the point at which each scenario stabilizes, are reported in Chapter 5.

A practical consequence of the continuous-shift design of Section 4.2.5 is worth noting. Because an episode never terminates early on success and always runs for the full shift budget, every episode contributes a comparable amount of experience, and the return is a direct measure of the value extracted over a fixed working period rather than of how quickly a fixed set of requests was cleared. This makes returns directly comparable across episodes and across policies, which simplifies both training monitoring and the evaluation methodology described in Section 4.7.

4.6 Baseline Policies

To judge whether the learned policy is genuinely competent, its performance must be placed in context with reference policies that do not require learning. Two baselines are used throughout this work. The first is a random policy, which establishes a lower bound on performance and confirms that the task is non-trivial. The second is a greedy heuristic, which represents the kind of sensible rule-based dispatching that a non-learning system might plausibly employ, and therefore constitutes the more demanding and more informative comparison. Both baselines act through exactly the same interface as the learned policy, observing the environment and selecting from the

same four movement actions, so that any difference in performance is attributable to the decision-making and not to a difference in capabilities. Table 4.5 summarizes the differences between the two baselines at a glance; each is then described in detail in the following subsections.

Table 4.5: Comparison of the two non-learning baseline policies.

Aspect	**Random policy**	**Greedy nearest-task heuristic**
Role in the study	Performance floor (lower bound); confirms the task is non-trivial	Strong rule-based reference and behavioral contrast
Decision rule	Selects one of the four moves uniformly at random	Moves one cell toward the highest-priority outstanding task
Use of observation	Ignores the observation entirely	Reads entity positions directly from the environment (privileged, noise-free)
Task prioritization	None	Fixed order: deliver on-board passenger, collect passenger, deliver on-board parcel, collect parcel
Tie-breaking	—	Nearest target by Manhattan distance
Planning horizon	None (memoryless)	Myopic: only the immediate next task, no look-ahead or repositioning
Urgency awareness	None	Patience-blind: ignores the urgency features
Action interface	Four discrete moves	Four discrete moves (identical interface)

4.6.1 Random Policy

The random policy selects one of the four movement actions uniformly at random at every step, completely ignoring the observation. It serves as a floor against which all other policies are measured. Because collection and delivery are automatic, a random walker will occasionally drift onto an origin or destination cell and complete a request by chance, so the random policy achieves a non-zero but very low level of service. Its purpose is to quantify how much of the performance of the other policies is due to deliberate decision-making rather than to the mechanics of the environment and to verify that the environment does not reward aimless movement.

4.6.2 Greedy Nearest-Task Heuristic

The greedy heuristic embodies a rule-based dispatching strategy. At every step it identifies the most appropriate outstanding task and moves one cell towards it, choosing the direction that most reduces the Manhattan distance to the target. Unlike the learned policy, which acts from the

normalized observation vector, the greedy heuristic reads the positions of the requests directly from the environment, giving it perfect, noise-free access to the state, which makes this a strong baseline.

Tasks are prioritized according to a fixed ordering that reflects the value structure of the problem. The highest priority is to deliver a passenger who is already on board, so that a passenger is never carried further than necessary. Next, collect a waiting passenger, since passengers are the most valuable and time-sensitive demand. After that is to deliver a parcel that is already on board and the lowest is to collect a waiting parcel from its locker. Within a given priority level, the ties are broken by choosing the nearest target in Manhattan distance. This ordering encodes the same passenger-first preference that the delivery rewards express and produces coherent, purposeful behavior.

A deliberate decision was made to keep the greedy heuristic patience-blind, so it does not consult the urgency features introduced in Section 4.4.3, and it follows the same fixed priority ordering whether or not time-based constraints are active. This choice keeps the baseline simple and interpretable, and it sharpens the comparison with the learned policy. If the learned policy is to demonstrate value beyond sensible routing, it must do so precisely in the situations the greedy heuristic ignores, such as prioritizing a passenger who is about to cancel or a parcel that is about to expire. The greedy heuristic therefore serves not only as a performance reference but as a behavioral contrast.

It should be emphasized that the greedy heuristic is myopic in the sense that it optimizes only the immediate next task and never reasons about the longer-term consequences of its choices. It does not plan to bundle a parcel collection into a passenger trip, nor does it reposition the vehicle in anticipation of future demand. Each decision is taken with respect to the current set of unfinished tasks alone. This myopia is the principal conceptual difference between the heuristic and the learned policy, and the experiments of Chapter 5 are designed to reveal where it matters and where it does not.

4.7 Evaluation Methodology

A policy is evaluated by running it for a fixed number of complete shifts and recording, for each shift, a comprehensive set of measurements. Unless otherwise stated, every policy is evaluated over one hundred independent episodes, each initialized with a distinct random seed so that the policies under comparison encounter the same sequence of demand realizations. The evaluation is conducted with the policy acting deterministically, selecting its most probable action rather than sampling, so that the reported behavior reflects the policy the agent has settled upon. The measurements are grouped into three families: domain metrics (describe operational behavior), economic and environmental metrics (translate behavior into monetary and emissions terms) and the statistical procedures used to compare policies. Each family is described in turn.

4.7.1 Experimental Scenarios

All experiments use the stochastic environment and the PPO configuration described earlier in this chapter, on the 10×10 grid with two passenger slots and two parcel slots. Three scenarios are distinguished according to the type of demand the vehicle is required to serve:

- **People** — the environment generates only passengers, and the vehicle operates as a conventional ride-hailing service.
- **Parcels** — the environment generates only parcels, and the vehicle operates as a pure locker-to-locker delivery service.
- **Mixed** — both types of demand are generated simultaneously and must be served by the same policy. This is the co-modal setting that is the subject of this dissertation; the two single-purpose scenarios serve as references representing dedicated fleets.

In each scenario, three policies are compared: the random policy, the greedy nearest-task heuristic (Section 4.6.2), and the PPO policy trained for that scenario. Every policy is evaluated over one hundred independent shifts under a common set of seeds, so that the policies encounter the same realizations of demand. These three scenarios are exercised under four distinct experimental conditions, which vary the environment formulation, the reward scheme, and the presence of time-based service constraints. Table 4.6 summarizes these conditions and indicates the section in which each is reported.

Table 4.6: Summary of the experimental configurations reported in this dissertation.

Experiment	**Environment**	**Reward scheme**	**Patience**	**Purpose (section)**
Reward sweep	Deterministic	24 tiered configurations	Off	Reward–behavior decoupling (Section 5.1)
Main (people / parcels / mixed)	Stochastic	Tiered ($-0.40/-0.20/-0.10$)	Off	Viability, fleet, mechanism (Sections 5.2–5.4)
Flat-penalty ablation	Stochastic (mixed)	Flat (-0.20)	Off	Causal role of the gradient (Section 5.5)
Patience experiments	Stochastic (mixed)	Tiered	On (various)	Human-centric constraints (Section 5.6)

4.7.2 Domain Metrics

The domain metrics characterize what the policy does in operational terms, independently of the reward it accumulates. Because demand arrives stochastically, the volume of requests that appears

in a shift is itself a random quantity, so service is reported both as absolute throughput and as a rate relative to the demand that actually appeared.

Throughput is the number of passengers and the number of parcels delivered per shift.

Delivery rate is the proportion of the demand that appeared during the shift that was successfully served, reported separately for passengers and parcels and computed only over episodes in which demand of the relevant type actually arrived, so that the rate is well defined. These two views are complementary, as throughput measures the absolute work done, while the rate measures the fraction of demand satisfied. Several additional metrics characterize the quality and efficiency of the service.

The *empty-driving ratio* is the fraction of travel time in which the vehicle carries neither a passenger nor a parcel. It is the principal indicator of wasted capacity and a direct measure of the inefficiency that co-transport is intended to reduce.

The *passenger waiting time* is the number of time units that a passenger waits between appearing and being collected.

The *detour factor* compares the time a request spends on board with the shortest possible time for its trip, expressed as a multiple of the direct distance. Specifically, the detour is computed as the ratio of the actual number of steps the request spends on board to the unweighted Manhattan distance between its origin and destination, expressed as a percentage of excess steps: a detour of 0% means the request was delivered along a shortest Manhattan path, while a detour of 100% means the vehicle spent twice as many steps as the direct route required. The computation uses the unweighted Manhattan distance (ignoring traffic multipliers) as the reference, because the traffic multipliers affect the time budget consumed but not the number of grid steps taken, and because the Manhattan distance provides a stable, interpretable baseline that does not depend on the particular route chosen. A detour factor of one indicates a direct trip, while larger values indicate that the request was carried along a longer route. The detour factor is of particular interest for parcels, because a parcel that is bundled into a passenger's journey will exhibit a large detour, making this metric a direct behavioral signature of co-transport.

When time-based constraints are active, two further metrics are reported: the *cancellation rate* of passengers who abandoned the system before being collected and the *expiry rate* of parcels that exceeded their carry deadline. Table 4.7 lists the full set of domain indicators considered.

4.7.3 Economic and Environmental Model

To express the operational behavior in terms that a fleet operator and a city planner would recognize, a simple economic and environmental model is applied to the recorded shifts. This model is used purely for reporting and does not enter the reward signal that the agent optimizes. It is applied after the fact to the evaluation episodes, so that the values it assumes can be revised without retraining.

Revenue is generated by completed deliveries. Each delivered passenger yields a fare composed of a fixed base component of 5.0 plus a distance component of 1.0 per unit of trip distance, reflecting the structure of real ride-hailing tariffs, while each delivered parcel yields a flat fee of

Table 4.7: Domain indicators used to characterize operational behavior.

Indicator	**Description**	**Reported**
Throughput	Passengers and parcels delivered per shift (absolute counts)	Always
Delivery rate	Fraction of the demand that appeared that was served, reported per type and combined	Always
Empty-driving ratio	Share of travel time with neither a passenger nor a parcel on board	Always
Passenger waiting time	Time units between a passenger appearing and being collected	Always
Detour factor	On-board steps relative to the shortest-path (Manhattan) distance	Always
Cancellation rate	Passengers that abandoned the system before being collected	Patience on
Expiry rate	Parcels that exceeded their carry deadline before delivery	Patience on

6.0, smaller than a typical passenger fare. Against this revenue, an operating cost of 0.3 per tile traveled is charged, representing fuel and wear, and the difference between revenue and operating cost defines the profit per shift. The fare and cost values were chosen so that a dedicated single-purpose service operates close to break-even, which avoids overstating the advantage of co-modal operation. The benefit reported in Chapter 5 arises from the structure of the behavior rather than from an arbitrarily favorable choice of tariff.

The environmental cost is approximated by the total distance traveled, measured in grid tiles, which serves as a proxy for fuel consumption and the associated greenhouse-gas emissions. The key environmental indicator is emissions per delivery, the total distance traveled in a shift divided by the number of deliveries completed, which captures the environmental efficiency of the service: a policy that delivers more while traveling less is both cheaper and cleaner. This per-delivery normalization is essential for a fair comparison, because it relates the environmental cost to the useful work performed rather than to the raw distance, and it underlies the comparison between mixed and dedicated operation in Chapter 5. A summary of the economic and environmental parameters is given in Table 4.8.

Table 4.8: Parameters of the economic and environmental model. These values are applied post hoc for reporting and do not affect the reward signal used during training.

Parameter	**Value**
Passenger base fare	5.0
Passenger fare per unit distance	1.0
Parcel base fare	6.0
Operating cost per tile	0.3

4.7.4 Statistical Methodology

Because each policy is evaluated over a finite sample of stochastic episodes, differences in mean performance must be assessed for statistical significance rather than taken at face value. Three complementary procedures are used to this end, and they are applied to every pairwise comparison between policies reported in Chapter 5.

First, the uncertainty in each reported mean is quantified by a *bootstrap confidence interval*. The distribution of the mean is estimated by resampling the episode results with replacement ten thousand times, and the 2.5th and 97.5th percentiles of the resampled means define a ninety-five per cent confidence interval. This procedure makes no assumption about the shape of the underlying distribution, which is appropriate given the skew that can arise in episode returns.

Second, the difference between two policies is tested for significance with a *paired test*. Because the policies under comparison are evaluated on the same set of seeds, their results are naturally paired episode by episode, and a paired t-test on the per-episode differences is used to obtain a p-value. A non-parametric *Mann–Whitney U test* is reported alongside it as a robustness check that does not rely on the assumption of normally distributed differences. Agreement between the two tests provides confidence that the conclusion does not depend on distributional assumptions.

Third, because statistical significance does not by itself convey the magnitude of a difference, the *effect size* is reported using Cohen's d, the difference in means expressed in units of the pooled standard deviation. By convention, values around 0.2, 0.5, and 0.8 are interpreted as small, medium, and large effects respectively. Reporting the effect size alongside the p-value distinguishes differences that are merely detectable from those that are operationally substantial, a distinction that proves important in Chapter 5, where a statistically significant difference in one scenario coexists with a negligible effect size in another.

Chapter 5

Results and Discussion

This chapter presents and discusses the experimental results obtained with the methodology described in Chapter 4. The experiments are organized to answer the research questions in turn. Section 5.1 reports the preliminary reward-shaping study conducted in the deterministic environment, which first exposed the decoupling between numerical reward and operational behavior that motivates the work. Section 5.2 establishes the viability of the learned policy by comparing it with the random and greedy baselines in the three scenarios. Section 5.3 examines the system-level benefit of co-modal operation by comparing a single mixed-purpose vehicle against dedicated single-purpose fleets. Section 5.4 analyzes the co-transport behavior itself through the detour signature, and Section 5.5 isolates the causal role of the load-conditioned reward through the flat-penalty ablation. Section 5.6 studies the effect of human-centric time constraints.

5.1 Reward Shaping and the Reward–Behavior Decoupling

Before developing the stochastic system that is the focus of this dissertation, a preliminary study was conducted in the deterministic environment of Section 4.2.1 to investigate how the structure of the load-conditioned step penalties shapes the behavior the agent learns. The study and its findings have been reported separately (5). They are summarized here because they establish the central premise on which the rest of this chapter is built, namely that maximizing the training return is not the same as producing the intended operational behavior. This subsection restates the design and the principal results at the level of detail required for the dissertation to stand on its own.

5.1.1 The Reward Sweep

The study held the delivery payouts fixed, at 20 for a passenger and 10 for a parcel, and swept the three load-conditioned step penalties over a $3 \times 3 \times 3$ grid of values: the empty penalty over $\{-0.45, -0.40, -0.35\}$, the parcel-only penalty over $\{-0.25, -0.20, -0.15\}$, and the passenger penalty over $\{-0.15, -0.10, -0.05\}$. The strict ordering that empty driving must be more costly than carrying parcels, which in turn must be more costly than carrying a passenger, reduces the twenty-seven candidate combinations to the twenty-four that respect the hierarchy. Each valid

combination was trained from scratch with three independent random seeds, for a total of seventy-two training runs, and each resulting policy was evaluated over a large number of deterministic episodes. Reporting every configuration as a mean across its three seeds, with the inter-seed standard deviation as a measure of reliability, allowed the study to distinguish genuine behavioral regimes from the noise of individual runs.

The choice to sweep only the step penalties, while holding the delivery payouts constant, was deliberate because it isolates the effect of the relative cost of the load states, which is the lever hypothesized to govern co-transport behavior. The value ranges were themselves selected through prior exploration, so as to concentrate the sweep on the region in which delivery rates are near close but distinct routing behaviors nonetheless emerge. In this way, any behavioral difference observed across the grid can be attributed to the relative magnitude of the penalties rather than to a failure to learn the basic task.

5.1.2 Three Behavioral Regimes

All twenty-four valid penalty configurations of the sweep (Section 5.1.1)—the twenty-four combinations that respect the load ordering, each trained with three seeds for the seventy-two runs mentioned above—solved the basic routing task, achieving combined delivery rates between 93.0% and 97.1%, and the mean episodic return varied across only a narrow band of roughly eight points. Beneath this superficial uniformity, however, the domain metrics revealed that the configurations partition into three sharply distinct behavioral regimes, distinguished most clearly by the detour factors of the two cargo types as shown in Figure 5.1. Table 5.1 summarizes the three regimes.

Table 5.1: The three behavioral regimes induced by the load-conditioned step penalties, characterized by mean return $\bar{R}$, empty-driving ratio E, and the passenger and parcel detour factors δ_{pax} and δ_{par}. From (5).

Regime	n	$\bar{R}$	E	δ_{pax}	δ_{par}
Passenger-shielding	6	47.10	24.9%	198.0%	69.2%
Parcel-hoarding	11	43.43	31.4%	5.0%	181.8%
Balanced co-modal	7	43.26	40.2%	0.6%	82.3%

The *passenger-shielding* regime arises whenever the passenger penalty is set to its smallest magnitude and combined with a more punitive parcel penalty. Every step taken with a passenger on board is then much cheaper than any alternative, including driving empty, and the agent learns to exploit this. Once a passenger is collected, the policy executes long indirect routes to the destination, using the passenger as an economic shield against the higher penalties of the empty and parcel-only states. The most extreme configuration in this regime detours passengers by 225%, driving more than three times the necessary distance with the passenger on board, while still delivering almost all passengers.

The *parcel-hoarding* regime arises whenever the parcel penalty is set to its smallest numbers. The parcel-only state is then only marginally more costly than carrying a passenger, so the agent

collects parcels but defers their delivery, accumulating cheap steps while the parcels remain on board. Because the vehicle may carry several parcels at once but only one passenger, this exploit is available to parcels in a way it is not to passengers, as parcels can be stacked and hoarded. The exemplar configuration produces parcel detours above 200% with essentially direct passenger trips, the mirror image of the shielding regime.

The *balanced co-modal* regime arises when no single load state is disproportionately cheap, so that neither shield is available. These configurations deliver both passengers and parcels along routes close to optimal for passengers and moderately extended for parcels, which is precisely the integrated behavior the system is intended to produce. Parcels are bundled into the vehicle's natural movement and incur a reasonable detour, while passengers are served directly. Crucially, this is also the regime with the lowest mean return of all three.

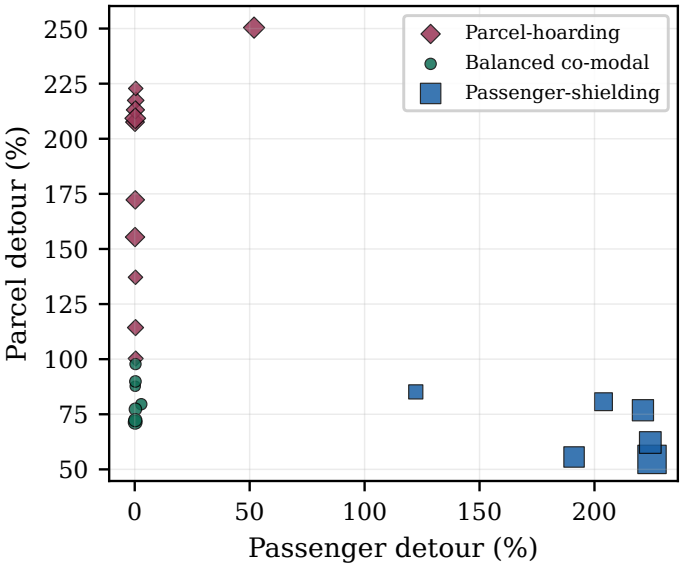

Figure 5.1: The three behavioral regimes identified by the deterministic reward sweep. Each marker corresponds to one of the twenty-four valid reward configurations, positioned by its mean passenger and parcel detour factor across the three training seeds, and colored by the regime to which it belongs. The balanced co-modal regime (green) is the only one that achieves genuine integration, while the other two are degenerate. From (5).

5.1.3 Reward and Behavior are Decoupled

The contrast in Table 5.1 carries the central message of the preliminary study. The regime that achieves the highest mean return, passenger-shielding, is one of the two operational failure modes, while the regime that behaves most correctly, balanced co-modal, earns the lowest return. Selecting a reward configuration by the standard reinforcement-learning criterion, the mean episodic return, would therefore systematically favor a degenerate policy over a well-behaved one. The study quantified this directly as across the configurations, the mean return is positively rank-correlated

with passenger detour and negatively rank-correlated with parcel detour, so that optimizing for return alone pushes the policy away from one failure mode and towards the other rather than towards balanced behavior.

This decoupling between cumulative return and logistical correctness is the finding on which the present dissertation is founded. It demonstrates, in a controlled deterministic setting, that a reward signal which appears reasonable can be maximized by behavior that defeats the operational purpose of the system, and that behavioral metrics such as the detour factor are therefore indispensable alongside the return when evaluating co-modal policies. It also yields a concrete design lesson that is carried forward into the stochastic system which is the strict ordering of the penalties. It is necessary but not sufficient, because it is the size of the gaps between adjacent load states, and not merely their order, that determines whether a state becomes cheap enough to be exploited. A state whose penalty is much smaller than the others invites the agent to linger in it, whether that state is carrying a passenger or holding a parcel.

These lessons directly informed the design of the stochastic environment evaluated in the remainder of this chapter. The reward configuration adopted there places the load-conditioned penalties in the balanced region identified by the sweep, with moderate gaps between adjacent levels, in order to encourage genuine co-transport rather than either shielding or hoarding. The remaining sections examine whether this design, once transferred to a far more demanding stochastic and congested setting, succeeds in producing the intended behavior, how the resulting policy compares with non-learning baselines, and what system-level benefits it delivers.

5.2 Viability of the Learned Policy

The first question to resolve is whether the learned policy is genuinely competent. Whether it serves demand effectively and whether it does so by deliberate decision-making rather than by exploiting the mechanics of the environment. This section answers that question by comparing the PPO policy against the random and greedy baselines of Sections 4.6.1–4.6.2 in each of the three scenarios. The comparison uses the mean episodic return as a headline figure and the domain metrics to characterize the behavior behind it, with statistical significance assessed as described in Section 4.7.4.

5.2.1 Return Against the Baselines

Table 5.2 reports the mean episodic return of the three policies in each scenario, together with the significance and effect size of the PPO advantage over the greedy heuristic. In every scenario the PPO policy substantially outperforms the random baseline, by margins of between one hundred and thirty-seven and two hundred and forty points, all significant at $p < 0.01$ with very large effect sizes. This confirms that the task is far from trivial and that the learned behavior is the product of deliberate control. A vehicle that moves at random achieves a strongly negative return in every scenario, because it accumulates step penalties while completing only the occasional delivery by chance.

Table 5.2: Mean episodic return ($\pm$ standard deviation) of the random, greedy, and PPO policies across the three scenarios, over one hundred shifts. The final columns report the significance and effect size of the PPO advantage over the greedy heuristic.

Scenario	**Random**	**Greedy**	**PPO**	**PPO vs. Greedy**
People	-91.15 ± 27.84	16.32 ± 41.59	45.94 ± 39.87	$+29.6$, $p < 0.01$, $d = 0.72$
Parcels	-89.82 ± 23.78	111.01 ± 23.97	132.73 ± 50.86	$+21.7$, $p < 0.01$, $d = 0.54$
Mixed	-68.32 ± 27.30	167.54 ± 28.76	171.52 ± 59.52	$+4.0$, $p = 0.50$, $d = 0.09$

The comparison against the greedy heuristic is more revealing, and its outcome differs by scenario in a way that is itself informative. In the two single-purpose scenarios the PPO policy significantly outperforms the greedy heuristic, by 29.6 points in the people scenario and 21.7 points in the parcels scenario, with medium effect sizes ($d = 0.72$ and $d = 0.54$ respectively). In the mixed scenario, by contrast, the two policies are statistically indistinguishable: the difference of 4.0 points is not significant ($p = 0.50$), and the effect size is negligible ($d = 0.09$). The learned policy thus matches, but does not exceed, the greedy heuristic when both types of demand are present.

5.2.2 Interpreting the Scenario-Dependent Advantage

That the PPO advantage is largest in the single-purpose scenarios and vanishes in the mixed one may at first appear counter-intuitive, since the mixed scenario is the most complex. The explanation lies in the nature of the demand each scenario presents. The people scenario is sparse as passengers appear infrequently, and a myopic dispatcher that simply drives to the nearest request spends much of its time idle or repositioning re-actively. The empty-driving ratio of the people scenario, discussed in Section 5.3, is by far the highest of the three, which leaves ample room for an anticipatory policy to improve on the heuristic by positioning itself in advance of demand. The learned policy exploits exactly this room, which is why its advantage is greatest where demand is sparsest.

The mixed scenario, in contrast, is dense. With both passengers and parcels arriving, there is almost always a worthwhile task nearby, and the greedy heuristic's rule of serving the nearest appropriate request is close to optimal under such conditions. The opportunity for anticipatory positioning shrinks when the next task is rarely far away, and the learned policy, while it reaches the same level of return, cannot exceed a heuristic that is already near-optimal for the situation.

This scenario-dependent pattern also sets the appropriate expectation for the remainder of the chapter. The case for the co-modal policy does not rest on its outperforming a strong heuristic in return, which it does not always do, but on the operational character of its behavior and on the system-level benefits it delivers, which are the subjects of the following sections.

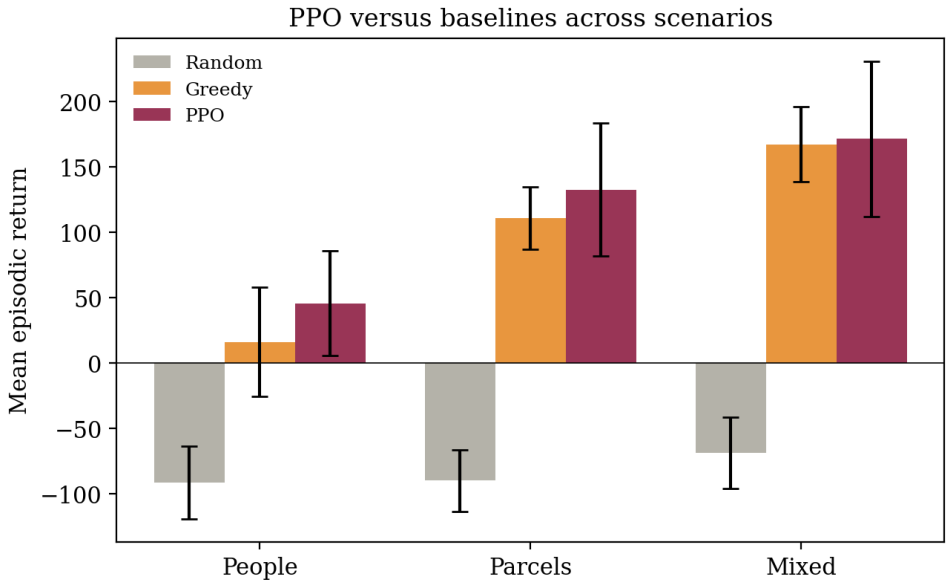

Figure 5.2: Mean episodic return (± one standard deviation) of the random, greedy, and PPO policies across the three scenarios, evaluated over 100 shifts each.

5.2.3 Service Quality

Beneath the return, the learned policy achieves a high standard of service in all three scenarios. In the mixed scenario it delivers 86.7% of the passengers and 70.6% of the parcels that appear, for a combined delivery rate of 78.7%. In the single-purpose scenarios the relevant rates are 89.6% for passengers in the people scenario and 80.6% for parcels in the parcels scenario. The slight reduction in each type's delivery rate when moving from a dedicated scenario to the mixed one, from 89.6% to 86.7% for passengers and from 80.6% to 70.6% for parcels, is the expected cost of dividing a single vehicle's attention between two streams of demand. The central question, addressed in the next section, is whether the gains of serving both streams with one vehicle outweigh this modest loss in per-stream service.

Table 5.3: Delivery rates of the PPO policy by scenario (fraction of the demand that appeared that was served).

Scenario	Passengers	Parcels	Combined
People	89.6%	—	89.6%
Parcels	—	80.6%	80.6%
Mixed	86.7%	70.6%	78.7%

5.3 System-Level Benefit of Co-Modal Operation

The previous section established that the learned mixed policy serves both types of demand competently, matching a strong heuristic in return while incurring only a modest reduction in per-

stream delivery rates. This section addresses the question that motivates the entire system, which is whether serving people and parcels with a single shared vehicle is better, at the level of the transport system, than operating two dedicated single-purpose vehicles. The comparison is framed as a choice an operator might face, namely whether to deploy one mixed-purpose vehicle or a pair of specialized vehicles, one for passengers and one for parcels, and it is evaluated on the throughput, efficiency, environmental, and economic metrics of Section 4.7.

5.3.1 The Comparison

The dedicated fleet is represented by the two single-purpose PPO policies of the people and parcels scenarios, operating independently. Its combined performance is obtained by summing the two vehicles' contributions, and its per-vehicle figures by averaging across the two. The mixed fleet is the single PPO policy of the mixed scenario. Table 5.4 and Figure 5.3 set the two side by side.

Table 5.4: One mixed-purpose vehicle compared with a dedicated fleet of two single-purpose vehicles (one for passengers, one for parcels), per shift. The dedicated fleet is shown both as a two-vehicle total and as a per-vehicle average; the final column expresses the per-vehicle advantage of the mixed configuration.

Metric	**Mixed (1 veh.)**	**Dedicated (2 veh.)**	**Dedicated (per veh.)**	**Per-vehicle advantage**
Passengers delivered	5.89	6.94	3.47	+2.42
Parcels delivered	7.20	12.83	6.42	+0.78
Total deliveries	13.09	19.77	9.89	+32.4%
Empty-driving ratio	29.1%	55.1%	55.1%	−26.0 pp
Emissions per delivery	21.16	24.23	24.23	−12.7%
Profit per shift	44.61	14.50	7.25	+515%
Vehicles required	1	2	—	−50%

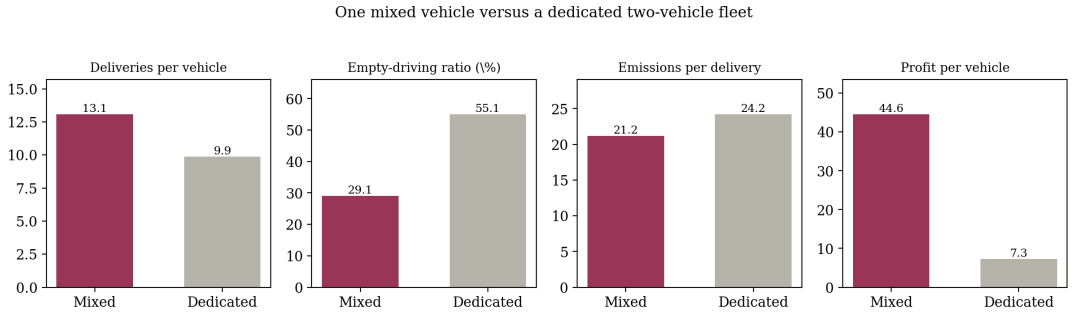

Figure 5.3: System-level comparison of a single mixed-purpose vehicle against a dedicated two-vehicle fleet (one for passenger-only, one for parcel-only). Panels show deliveries per vehicle, empty-driving ratio, emissions per delivery, and profit per vehicle.

Clearly, the dedicated fleet appears to be superior in raw throughput as two vehicles deliver 19.77 items per shift against the single vehicle's 13.09. This comparison is, however, between one

vehicle and two, and the operator's decision concerns the resources required to serve demand, not the output of an arbitrary number of vehicles. The meaningful comparison is therefore on a per-vehicle basis, and on that basis the picture reverses because the single mixed vehicle completes 13.09 deliveries, against an average of 9.88 per vehicle in the dedicated fleet, a per-vehicle advantage of 32.4%. One mixed vehicle does substantially more useful work than the average dedicated vehicle, because it is always looking for one type of demand it is permitted to serve.

5.3.2 Efficiency and Emissions

The source of this advantage is visible in the empty-driving ratio. The mixed vehicle travels without any cargo for only 29.1% of its time, against an average of 55.1% for the dedicated vehicles. The dedicated passenger vehicle spends most of its time empty, because passenger demand alone is too sparse to keep it occupied while the dedicated parcel vehicle fares better but still idles. The mixed vehicle fills these gaps by carrying parcels through the periods when no passenger is available and by collecting parcels along the routes it travels for passengers, converting what would otherwise be empty running into productive delivery. This is the operational essence of co-transport, and the empty-driving ratio is its clearest single measure.

Because the environmental cost is approximated by distance traveled, the reduction in empty running translates directly into an environmental gain. The mixed vehicle emits the equivalent of 21.16 distance units per delivery, against 24.23 for the dedicated fleet, a reduction of 12.7% in emissions per unit of service. The mixed policy is thus not merely doing more work per vehicle but doing it more cleanly per delivery, because a larger share of every kilometer traveled is spent carrying something useful. For a city seeking to reduce the environmental burden of its transport system, this per-delivery reduction is the relevant figure.

5.3.3 Economic Outcome

The economic model of Section 4.7.3 combines these effects into the profit per shift. The mixed vehicle earns a profit of 44.61 per shift, against a combined 14.50 for the two dedicated vehicles, and on a per-vehicle basis the advantage is large: the dedicated fleet earns only 7.25 per vehicle, so the mixed vehicle is more than five times as profitable per vehicle. This striking figure must be interpreted with care, and it is reported with an explicit caveat. The magnitude of the multiple is sensitive to the chosen tariff and operating cost, and in particular to the fact that the dedicated vehicles operate close to break-even under the calibration adopted, which inflates any ratio taken against their profit. The direction of the result is robust, since it follows directly from the higher utilization and throughput of the mixed vehicle, but the precise multiple should be read as illustrative rather than as a precise economic forecast.

The more defensible economic statement is therefore the one expressed in absolute terms and in vehicle count. A single mixed vehicle generates more total profit than two dedicated vehicles combined, while requiring half the fleet, half the drivers, and roughly half the capital. Even setting aside the sensitivity of the profit multiple, the reduction from two vehicles to one to serve the

same demand is a first-order operational saving whose value does not depend on the fine detail of the tariff. Taken together, the throughput, efficiency, emissions, and economic results provide a consistent answer to the third research question: co-modal operation delivers a genuine system-level benefit, concentrated not in the raw volume of service but in the efficiency with which a vehicle is used.

5.3.4 Re-framing the Viability Result

It is worth connecting this result back to the viability comparison of Section 5.2. There, the learned mixed policy only matched the greedy heuristic in return, which might have seemed a disappointing outcome. The fleet comparison re-frames that finding. The value of the co-modal policy was never going to manifest as a higher return against a strong heuristic in a dense scenario. It manifests instead as the ability to serve two streams of demand with one vehicle at high utilization, which is a property of the co-modal formulation itself rather than of the margin by which one policy beats another. The contribution of learning, established in Section 5.2, is that a single policy can be trained to serve this combined demand competently at all and the contribution of the co-modal design, established here, is that doing so is markedly more efficient than the dedicated alternative. The two findings are complementary, and together they make the case for the approach.

5.4 The Co-Transport Mechanism

The preceding sections established that the mixed policy is efficient at the level of the system, but efficiency alone does not reveal how the policy achieves it. This section examines the behavior directly, in order to confirm that the efficiency arises from genuine co-transport, that is, from bundling parcels into the journeys the vehicle makes to serve passengers, rather than from the vehicle simply alternating between two tasks in sequence. The detour factor introduced in Section 4.7.2 provides the decisive evidence, because it measures how far a delivered request travels relative to the shortest path between its origin and destination, and is therefore a direct signature of whether a request is carried along an indirect shared route.

5.4.1 The Detour Signature

The argument rests on comparing the parcel detour in the mixed scenario with the parcel detour in the dedicated parcels scenario, where co-transport is by construction impossible. In the parcels-only scenario the vehicle serves parcels exclusively, so each parcel is carried along a route chosen solely for its own delivery. The observed parcel detour of 155% in that scenario therefore represents the baseline indirection of a policy that is optimizing parcel delivery in isolation, including the moderate detours incurred when several parcels are batched together. In the mixed scenario, on the contrary, the parcel detour increases to 279%, almost twice as large. A parcel delivered by

the mixed policy travels, on average, far further relative to its direct route than a parcel delivered by the dedicated policy.

This increase is the behavioral fingerprint of co-transport. The only structural difference between the two scenarios is the presence of passengers in the mixed case, so the additional indirection that parcels experience there can only be explained by parcels being carried along routes that are shaped by passenger service. The mixed policy collects a parcel and retains it while it diverts to collect or deliver a passenger, releasing the parcel at its locker when the route happens to pass nearby. The parcel consequently follows a long, passenger-driven path rather than a direct one. The dedicated policy, having no passengers to serve, has no reason to impose such detours and delivers parcels comparatively directly.

5.4.2 Asymmetry Between Passengers and Parcels

The passenger detour behaves quite differently, and the contrast is itself informative. In the mixed scenario the passenger detour is 57.8%, far smaller than the parcel detour of 279%, and comparable to the level a passenger-only policy exhibits. The mixed policy thus keeps passenger trips relatively direct while allowing parcel trips to absorb the indirection required to combine the two services. This asymmetry is exactly the behavior the reward function was designed to encourage, and it reflects the structure of the problem on two counts. First, passengers are the more valuable demand and are subject to the greater urgency, so carrying them directly protects the more important service. Second, the vehicle may hold only one passenger but several parcels, so it is the parcels that have the flexibility to be carried along shared routes, while the single passenger seat is best occupied for as short a time as possible.

The policy has therefore discovered, from the reward signal alone, a division of roles between the two cargo types that mirrors the logic a human logistician might apply which is keeping the premium, time-sensitive, single-occupancy service direct, and use the flexible, lower-priority, multi-occupancy cargo to fill the spare capacity along the way. That this division emerges without being explicitly programmed, as a consequence of the load-conditioned reward and the on-board capacity constraints, is one of the more substantial findings of this work.

5.4.3 Relation to the Preliminary Study

This result also reinforces, in the stochastic setting, the design lesson drawn from the deterministic reward sweep of Section 5.1. There, the balanced co-modal regime was characterized by direct passenger trips and moderately extended parcel trips, and it was selected as the target for the stochastic reward configuration because it represented genuine integration rather than shielding or hoarding. The detour signature observed here, direct passengers and substantially detoured parcels, is the stochastic counterpart of that balanced regime, confirming that the reward design transferred its intended behavioral character from the controlled deterministic environment to the far more demanding stochastic one. The next section strengthens this claim into a causal one,

by showing that removing the load-conditioned structure of the reward removes some of the co-transport behavior as well.

5.5 Ablation: The Causal Role of the Load-Conditioned Reward

The evidence presented so far is consistent with the hypothesis that the load-conditioned step penalty is responsible for the co-transport behavior, but it remains correlational: the policies examined were all trained with the tiered penalty, so the possibility cannot be excluded that any reasonable penalty would produce the same behavior. This section closes that gap with a controlled ablation. The tiered penalty is replaced by a single flat penalty of -0.20 applied to every move regardless of the load carried, while every other element of the environment, the network, and the training procedure is held fixed, and a new policy is trained for one hundred million steps under this flat scheme. By isolating the load-conditioned structure as the single manipulated variable, the ablation determines whether that structure is the cause of the integrated behavior or merely incidental to it.

5.5.1 A Note on Comparability

It should also be noted that the flat penalty of -0.20 is not the arithmetic mean of the three tiered penalties (-0.40, -0.20, -0.10, whose mean is -0.233). The value was chosen to match the intermediate tier (the parcel-only penalty) so that the total penalty budget experienced by a typical episode is comparable between the two schemes, rather than introducing a confound through a substantially different aggregate cost. The fact that delivery rates remain virtually unchanged between the two schemes (Section 5.5.3) confirms that both penalty levels provide a sufficient learning signal for the basic task, and that the behavioral difference is attributable to the gradient between load states rather than to a difference in overall penalty magnitude.

5.5.2 Behavioral Comparison

Table 5.5 compares the policy trained with the tiered penalty against the policy trained with the flat penalty, both in the mixed scenario and under otherwise identical conditions.

The behavioral differences are clear and they all point in the same direction. The parcel detour falls from 279% under the tiered penalty to 190% under the flat one, a substantial reduction in the very quantity that Section 5.4 identified as the signature of co-transport. The empty-driving ratio rises from 29.1% to 35.7%, indicating that the flat policy leaves the vehicle without cargo for a larger share of its time. The passenger waiting time increases from 12.6 to 16.3 time units, showing that the flat policy is slower to collect passengers. In short, when the gradient between load states is removed, the policy bundles parcels less, drives empty more, and attends to passengers less promptly.

Table 5.5: Behavioral comparison of the tiered and flat penalty schemes in the mixed scenario. Episodic return is omitted because it is not comparable across the two reward functions.

Behavioral metric	**Tiered**	**Flat**
Passenger service		
Passenger detour	58%	51%
Passenger waiting time	12.6	16.3
Passenger delivery rate	86.7%	87.8%
Parcel service		
Parcel detour	279%	190%
Parcel delivery rate	70.6%	72.3%
Overall efficiency		
Empty-driving ratio	29.1%	35.7%
Combined delivery rate	78.7%	80.0%

5.5.3 What the Ablation Establishes

The most important feature of Table 5.5 is the contrast between the behavioral metrics and the delivery rates. While the detour, empty-driving, and waiting metrics change markedly between the two schemes, the delivery rates barely move: passengers are delivered at 86.7% against 87.8%, parcels at 70.6% against 72.3%, and the combined rate at 78.7% against 80.0%. The two policies therefore deliver essentially the same volume of demand; what differs is the manner in which they do so. This dissociation is precisely the point. The tiered penalty does not determine what the agent accomplishes, since any sensible penalty suffices to learn the basic task, but it does determine how the agent accomplishes it, shaping the routing into the integrated, low-empty-running pattern that constitutes co-transport.

This is the causal evidence the first research question requires. The load-conditioned structure of the reward is not incidental to the co-transport behavior but its cause: removing the structure, while leaving a penalty of the same nominal magnitude in place, removes the behavior, even though the agent continues to deliver demand at the same rate. The flat policy still learns to serve passengers and parcels, because the large delivery rewards make delivery worthwhile under any penalty scheme, but it does so as a sequence of largely independent errands rather than as an integrated operation, with more empty running and less bundling. The integration that gives co-modal operation its system-level advantage is therefore attributable specifically to the gradient between the load states, confirming the design hypothesis on which this work is built.

It is instructive, finally, to read this ablation alongside the deterministic reward sweep of Section 5.1. The sweep showed that the relative sizes of the penalties select among distinct behavioral regimes; the ablation shows that the existence of a gradient at all is what separates integrated co-transport from a flat, errand-by-errand policy. The two studies are complementary halves of a single argument about reward design: the gradient must exist for integration to emerge, and it must be balanced for that integration to be of the desired, non-degenerate kind. Together they establish that the behavior of a co-modal agent is governed, in a precise and demonstrable way, by

the structure of the reward it is given.

5.6 Human-Centric Service Constraints

The experiments reported so far assume that demand is endlessly patient meaning that a passenger waits indefinitely to be collected, and a parcel may be carried for any length of time before delivery. Real transport demand is not so forgiving, and Section 4.4.3 introduced two time-based constraints to capture this, namely passenger patience and a parcel carry deadline. This section reports the experiments conducted with these constraints. It is, by design, also an account of a difficult and instructive design process, because introducing time pressure exposed a sequence of reward-design pitfalls that had to be diagnosed and addressed in turn. Reporting that process in full, including the configurations that failed, is more faithful to the work actually carried out than presenting only the final design would be, and each failure reinforces the central thesis that the structure of the reward and the constraints governs the behavior that is learned.

Figure 5.4 summarizes the parcel delivery rate and detour factor across all patience design iterations. Each label on the horizontal axis denotes one configuration in the design sequence, described in turn in Sections 5.6.1–5.6.4 and specified in full in Appendix Table A.4. The following subsections describe each step.

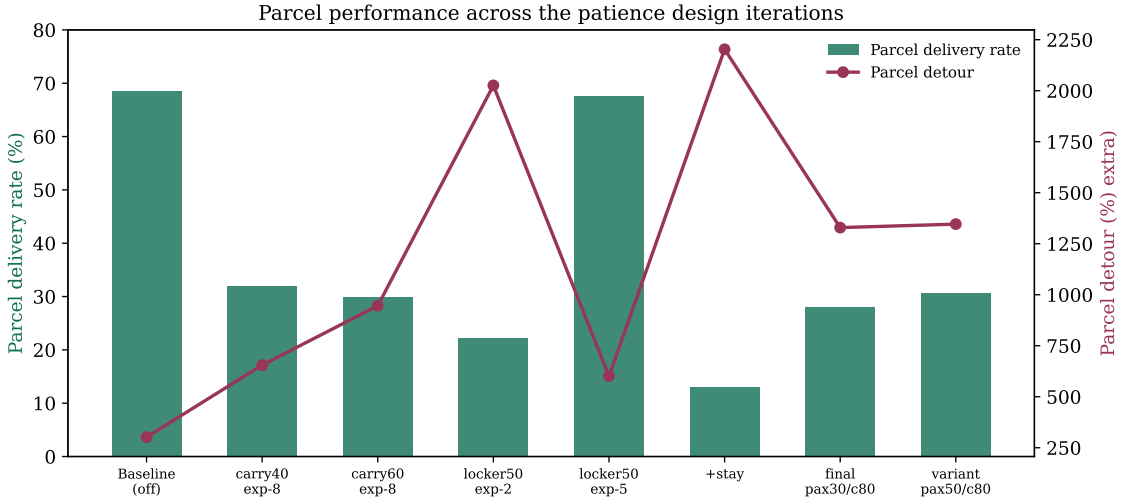

Figure 5.4: Parcel delivery rate (bars, left axis) and parcel detour factor (line, right axis) across the patience design iterations. Each label encodes the carry-deadline, expiry-penalty and patience settings of one iteration (detailed in Sections 5.6.1–5.6.4 and Appendix Table A.4); the baseline (patience off) is shown for reference.

5.6.1 First Attempt: The Carry Deadline and the Abandonment of Parcels

The first design applied a patience window to waiting passengers, who canceled if not collected within thirty time units, and a carry deadline to parcels, which expired if not delivered within forty time units of collection, with penalties of -10 and -8 respectively. Trained under this scheme

for fifty million steps, the agent reacted in an unexpected and revealing way because rather than learning to deliver parcels promptly to beat the deadline, it largely stopped collecting parcels at all. The parcel delivery rate collapsed to 31.9%, while passengers continued to be served at 81.4%, and the empty-driving ratio rose sharply to 64.0% as the vehicle spent much of its time carrying nothing. Of the parcels that were collected, 28.1% expired before delivery.

The reason, on reflection, is a perverse incentive. Collecting a parcel started a countdown that, if not met, incurred a penalty, whereas leaving the parcel at its locker carried no such risk. Faced with a tight deadline and the difficulty of guaranteeing timely delivery amid competing passenger demand, the agent learned that the safest way to avoid the expiry penalty was never to expose itself to it, that is, never to pick the parcel up. Relaxing the deadline from forty to sixty time units changed little as the parcel delivery rate remained essentially flat at 29.9%, the empty-driving ratio stayed close to 59.4%, and the agent continued to treat parcels as a liability to be avoided rather than an opportunity to be served.

5.6.2 Diagnosis and Redesign: Locker Patience and Slot Recycling

A closer inspection of the demand statistics revealed a second, compounding problem. Because parcels in this design never expired while waiting at their locker, a parcel that the agent declined to collect remained in its slot indefinitely, blocking that slot from generating new demand. The effective rate at which parcels appeared therefore fell, and the environment quietly drifted away from the intended demand profile. The constraint had been placed on the wrong part of the parcel's life cycle as it penalized carrying a parcel too long but did nothing about a parcel that was never collected.

The design was accordingly revised. A patience window was added to parcels at the locker, mirroring passenger patience, so that an uncollected parcel expires after fifty time units and releases its slot. The carry deadline was relaxed to two hundred time units, large enough that it ceased to be the binding constraint and the expiry penalty was first reduced to -2. This restored the flow of parcel demand, since slots no longer remained blocked, but it introduced yet another behavior: with the expiry penalty so small, the agent found it cheaper to let parcels expire than to make the effort to deliver them. The parcel delivery rate remained low at 22.2%, while fully 56.3% of the parcels that appeared expired at their locker, unattended. The penalty was therefore raised to -5, a level at which letting a parcel expire became genuinely costly relative to the effort of delivering it.

5.6.3 The Hoarding-as-Discount Exploit

With the expiry penalty at -5, the locker patience at fifty time units, and the carry deadline at two hundred, and trained for one hundred million steps, the agent at last delivered parcels well, reaching a parcel delivery rate of 67.6% with an expiry rate of only 10.2% and a strong return. Inspection of the resulting behavior, however, revealed a subtler exploit that the aggregate delivery

rate had concealed. The parcel detour factor had risen to 601%, well into the territory of the parcel-hoarding regime identified in the deterministic sweep of Section 5.1. The agent was collecting a parcel early and then retaining it for a large part of the shift, delivering it only eventually and only just within the generous carry window, because under the load-conditioned penalty a vehicle carrying a parcel pays a smaller per-step cost than an empty one. The parcel was being used less as a delivery to be completed than as a permanent discount on the cost of movement.

To remove the incentive behind this behavior, an additional action was introduced that allowed the vehicle to remain stationary at a small fixed cost of -0.05, on the reasoning that an agent able to wait cheaply would no longer need to hoard a parcel merely to reduce its movement penalty. The intervention did not have the intended effect, and in fact made matters worse. Equipped with the option to wait, and still facing the difficulty of serving both demands under time pressure, the agent converged to a qualitatively different but equally undesirable strategy: it specialized in passengers, served them at 85.4%, and used the new waiting action to idle through the periods when no passenger was available, allowing parcels to expire at a rate of 69.8% while their delivery rate collapsed to 13.0%. Replacing the tiered penalty with a flat one under the same stay-enabled configuration produced essentially the same degenerate outcome, with a parcel delivery rate of 12.1% and an expiry rate of 71.2%. The waiting action was therefore judged to enlarge the space of degenerate strategies without resolving the underlying tension, and although it was retained in the subsequent patience experiments for continuity, it was excluded from the main unconstrained results of the preceding sections, where it had been shown to be purely harmful.

5.6.4 Final Design and Outcome

The design that was ultimately adopted abandoned the locker patience and the very long carry window in favor of a more direct pairing: passenger patience at thirty time units and a parcel carry deadline of eighty time units, with an expiry penalty of -5. The carry deadline of eighty units was calibrated to be generous relative to the time required to cross the grid, yet tight enough to discourage the indefinite retention seen in the hoarding regime. Trained for one hundred million steps, this configuration served passengers at 88.0% and delivered 28.1% of parcels, with a passenger cancellation rate of 4.0% and a parcel expiry rate of 21.2%. These figures represent a genuine compromise: the agent no longer refuses parcels outright, as in the first attempt, nor hoards them indefinitely, as in the long-window design, but it also does not approach the 70.6% parcel delivery rate of the unconstrained mixed policy. The parcel detour, at 1329%, remains high, indicating that the residual parcel deliveries are still accomplished through long, passenger-driven routes rather than direct ones.

A short sensitivity check was conducted to test whether the imperfect parcel performance was an artifact of overly tight passenger patience. Relaxing the passenger patience from thirty to fifty time units, with all else held fixed, reduced passenger cancellations almost to zero, from 4.0% to 0.9%, and raised passenger service to 91.6%, confirming that the patience window controls cancellations as intended. Crucially, however, the parcel delivery rate barely moved, rising only from 28.1% to 30.7%, and the parcel detour remained essentially unchanged. The limited parcel

performance is therefore not a consequence of passenger time pressure but a structural feature of how a single agent resolves the competition between the two demand streams under deadlines, which lends weight to the interpretation offered below.

5.6.5 Lessons from the Patience Experiments

A notable feature of the constrained setting is that the greedy heuristic remained strongly competitive with, and in several configurations superior to, the learned policy. This is consistent with the interpretation advanced in Section 5.2.2: the heuristic's simple rule of always serving the nearest appropriate task is a robust strategy that degrades gracefully under time pressure, whereas the learned policy must discover how to balance urgency against efficiency from the reward alone, and is correspondingly more prone to settling into one of the degenerate strategies described above.

The patience experiments are best read not as a clean success but as an honest exploration of the limits of the approach, and they yield three lessons. First, they reinforce the thesis of this dissertation from the opposite direction: where the reward sweep and the flat-penalty ablation showed how reward structure can be used deliberately to produce desirable behavior, the patience experiments show how easily a well-intentioned constraint can produce undesirable behavior, from refusing to collect parcels, to letting them expire, to hoarding them, to idling away the time. In every case the agent did exactly what its reward and constraints made most profitable, and in every case that differed from what was intended and the gap between the two is the recurring subject of this work.

Second, they locate a concrete boundary of the method as applied here. Under tight, competing time constraints, a single agent trained with a standard policy-gradient algorithm and a compact network tends to resolve the tension by specializing, typically in the more valuable passenger demand, rather than by mastering the joint scheduling of both streams. The sensitivity check confirmed that this is structural rather than an artifact of the passenger patience setting. Overcoming it would plausibly require richer machinery than was used here, such as a more expressive policy architecture, a curriculum that introduces the constraints gradually, or an explicitly multi-objective formulation which are identified as future work in Chapter 6. Third, and more practically, the experiments justify the decision to treat the unconstrained setting as the primary object of study in the earlier sections as it is the setting in which the co-transport behavior can be elicited and analyzed cleanly, without the confound of an agent retreating into specialization under pressure. The constrained setting remains a valuable probe of realism and of the method's limits, and it is reported here in that regard.

Chapter 6

Conclusions

This chapter concludes the dissertation by summarizing its main contributions, formally revisiting each research question in the light of the evidence gathered, and identifying the limitations and directions for future work that arise from the findings.

6.1 Main Contributions

The work presented in this dissertation makes the following contributions to the intersection of deep reinforcement learning and urban co-modal transport:

1. **Reward–behavior decoupling.** Through a systematic reward-shaping sweep in a deterministic environment, this work demonstrated that cumulative return is not a reliable proxy for operational behavior in co-modal settings as the policy that maximizes return is not necessarily the policy that behaves correctly. This finding, first reported in (5), is confirmed and extended in the stochastic setting.

2. **Causal identification of the co-transport mechanism.** A controlled ablation showed that removing the load-conditioned gradient from the step penalty, while leaving a penalty of the same nominal magnitude in place, eliminates the integrated co-transport behavior without affecting delivery rates. This establishes that the tiered penalty structure is the cause of co-transport, not merely a correlate.

3. **System-level quantification of co-modal value.** A single mixed-purpose vehicle, controlled by the learned policy, delivers 32.4% more per vehicle than the average vehicle in a dedicated two-vehicle fleet, while driving empty 29.1% of the time (versus 55.1% for the dedicated fleet), emitting 12.7% less per delivery, and generating substantially higher profit.

4. **Behavioral signature of co-transport.** The parcel detour factor nearly doubles when moving from a dedicated parcels scenario (155%) to the mixed scenario (279%), providing a direct, measurable fingerprint of parcels being bundled into passenger routes. The passenger detour remains comparatively direct (58%), confirming that the policy preserves the quality of the premium service while using the flexible cargo to fill spare capacity.

5. **Identification of method limits under time pressure.** The patience experiments revealed that, under competing time-based service constraints, a single agent trained with a standard policy-gradient algorithm and a compact network tends to specialize in the more valuable demand stream rather than mastering joint scheduling. This marks a concrete boundary of the approach and motivates the future-work directions below.

6.2 Addressing the Research Questions

RQ1 — Reward design (mechanism). The structure of the reward function governs the behavior a single DRL agent learns in a precise and demonstrable way. The relative sizes of the load-conditioned step penalties select among qualitatively distinct behavioral regimes (shielding, hoarding, balanced co-modal), and the existence of a gradient between load states is what separates integrated co-transport from a flat, errand-by-errand policy. Reward maximization alone is an unreliable guide to good behavior as the regime with the highest return is one of the two failure modes, while the balanced regime earns the lowest.

RQ2 — Viability and competitiveness. A single-agent DRL policy can serve combined passenger and parcel demand under stochastic, congested conditions. It decisively beats a random baseline everywhere (all comparisons significant at $p < 0.01$ with large effect sizes) and significantly outperforms a strong greedy nearest-task heuristic in the two single-purpose scenarios (people and parcels). In the dense mixed scenario, PPO matches the heuristic in return (difference not significant, $p = 0.50$, negligible effect size), but achieves that return through a fundamentally different, more efficient routing strategy.

RQ3 — System-level benefit. Co-modal operation delivers genuine system-level benefit concentrated in per-vehicle efficiency. One mixed vehicle completes 13.09 deliveries per shift, against 9.88 per vehicle in the dedicated fleet (+32.4%). Its empty-driving ratio is 29.1% versus 55.1%, its emissions per delivery are 12.7% lower, and it generates more total profit than the two dedicated vehicles combined, using half the resources.

RQ4 — Human-centric constraints. Time-based service constraints (passenger patience, parcel carry deadlines) inject realism and trade-offs but also make the task markedly harder. Under tight, competing deadlines, the agent converges to a passenger-specialist strategy, achieving high passenger service (88%) but low parcel delivery (28%). This tendency is structural rather than an artifact of any single parameter as relaxing passenger patience from 30 to 50 ticks reduces cancellations to near zero but barely moves the parcel delivery rate.

6.3 Limitations

The findings of this work are subject to the following limitations:

- **Scale and abstraction.** The experiments were conducted on a 10×10 grid with a single vehicle and two slots per demand type. While sufficient to exhibit the behaviors of interest, this remains a stylized abstraction. The extension to larger grids and real-world road networks was explored (20×20) but proved unreliable within a practical training budget.
- **Single agent.** The formulation models one vehicle in isolation. The coordination problems that arise when multiple co-modal vehicles share a city are outside its scope.
- **Economic model.** The fare and cost parameters are deliberately simple and calibrated so that dedicated operation sits near break-even. The direction of the profit and emissions results is robust to this choice, but their precise magnitudes are illustrative rather than predictive.
- **Single training seed.** Each reported policy is the product of a single training run. The confidence intervals capture evaluation variability but not inter- run variability, which the deterministic sweep suggested can be appreciable.
- **Policy architecture.** A standard two-layer MLP was used throughout. More expressive architectures were not explored and might alleviate the limitations observed under time pressure.

6.4 Future Work

The limitations above motivate several directions for future research:

1. **Scaling to realistic environments.** Extending the grid to realistic city sizes (e.g., 50×50 or beyond), with non-uniform traffic patterns, demand hot-spots, and multiple vehicle types, would test whether the co-transport mechanism transfers to more complex settings.
2. **Multi-agent coordination.** Deploying a fleet of co-modal vehicles, each controlled by an independent or cooperative policy, raises new questions about territory partitioning, demand competition, and emergent coordination that a single-agent formulation cannot address.
3. **Richer policy architectures.** Attention-based or graph-neural architectures might enable the agent to reason more effectively about urgency, spatial structure, and multi-hop parcel routing, potentially overcoming the specialization observed in the patience experiments.
4. **Curriculum learning for time constraints.** Introducing patience and deadlines gradually during training, rather than from the start, may allow the agent to first master the basic co-transport skill and then adapt to time pressure without retreating into specialization.
5. **Real-world validation.** Partnering with a ride-hailing or logistics operator to test the approach on real demand traces and road networks would be the ultimate validation of the co-modal concept.

References

[1] Abubakr O. Al-Abbasi, Arnob Ghosh, and Vaneet Aggarwal. DeepPool: Distributed model-free algorithm for ride-sharing using deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 20(12):4714–4727, 2019.

[2] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016.

[3] Irwan Bello, Hieu Pham, Quoc V. Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2017.

[4] Judd Cramer and Alan B. Krueger. Disruptive change in the taxi business: The case of Uber. *American Economic Review*, 106(5):177–182, 2016.

[5] Gonçalo Ferreira, Rosaldo Rossetti, António Costa, and Tânia Fontes. Shaping co-transport behaviour through reward design: A deep reinforcement learning study. Submitted to the 2026 IEEE International Smart Cities Conference (ISC2); preliminary study preceding this dissertation, 2026.

[6] Waldy Joe and Hoong Chuin Lau. Deep reinforcement learning approach to solve dynamic vehicle routing problem with stochastic customers. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, pages 394–402, 2020.

[7] Wouter Kool, Herke van Hoof, and Max Welling. Attention, learn to solve routing problems! In *International Conference on Learning Representations (ICLR)*, 2019.

[8] Victoria Krakovna, Jonathan Uesato, Vladimir Mikulik, Matthew Rahtz, Tom Everitt, Ramana Kumar, Zac Kenton, Jan Leike, and Shane Legg. Specification gaming: The flip side of AI ingenuity. DeepMind Blog, 2020. Available at https://deepmind.google/discover/blog/specification-gaming-the-flip-side-of-ai-ingenuity/.

[9] Baoxiang Li, Dmitry Krushinsky, Hajo A. Reijers, and Tom Van Woensel. The share-a-ride problem: People and parcels sharing taxis. *European Journal of Operational Research*, 238 (1):31–40, 2014.

[10] Kaixiang Lin, Renyu Zhao, Zhe Xu, and Jiayu Zhou. Efficient large-scale fleet management via multi-agent deep reinforcement learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1774–1783, 2018.

[11] Yang Liu et al. Piggyback on idle ride-sourcing drivers for integrated on-demand and flexible intracity parcel delivery services. *Transportation Science*, 2025. Published online April 2025.

[12] Kalyan Manchella, Ashwin Umtri, and Vaneet Aggarwal. FlexPool: A distributed model-free deep reinforcement learning algorithm for joint passengers and goods transportation. *IEEE Transactions on Intelligent Transportation Systems*, 22(4):2035–2047, 2021.

[13] Kalyan Manchella, Ashwin Umtri, and Vaneet Aggarwal. PassGoodPool: Joint passengers and goods fleet management with reinforcement learning aided pricing, matching, and route planning. *IEEE Transactions on Intelligent Transportation Systems*, 23(3):2484–2494, 2022.

[14] Alaa Mourad, Jakob Puchinger, and Chengbin Chu. A survey of models and algorithms for optimizing shared mobility. *Transportation Research Part B: Methodological*, 123:323–346, 2019.

[15] Mohammadreza Nazari, Afshin Oroojlooy, Lawrence V. Snyder, and Martin Takać. Reinforcement learning for solving the vehicle routing problem. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[16] Andrew Y. Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the 16th International Conference on Machine Learning (ICML)*, pages 278–287, 1999.

[17] Matthew J. Page, Joanne E. McKenzie, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, Jennifer M. Tetzlaff, Elie A. Akl, Sue E. Brennan, et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372:n71, 2021.

[18] Mingqi Pan et al. Comprehensive overview of reward engineering and shaping in advancing reinforcement learning applications. *arXiv preprint arXiv:2408.10215*, 2024.

[19] Guan-Yu Qin et al. Deep reinforcement learning for demand driven services in logistics and transportation systems: A survey. *ACM Computing Surveys*, 2025. Published online February 2025.

[20] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021.

[21] Luigi Ranieri, Serena Digiesi, Bartolomeo Silvestri, and Michele Roccotelli. A review of last mile logistics innovations in an externalities cost reduction vision. *Sustainability*, 10(3): 782, 2018.

[22] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[23] Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[24] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 2nd edition, 2018.

[25] Alejandro Tirachini and Oded Cats. COVID-19 and public transportation: Current assessment, prospects, and research needs. *Journal of Public Transportation*, 22(1):1–21, 2020.

[26] Da-Hai Zeng et al. DHDRDS: A deep reinforcement learning-based ride-hailing dispatch system for integrated passenger–parcel transport. *Sustainability*, 17(9):4012, 2025.

Appendix A

Supplementary Material

A.1 Environment Configuration Parameters

Table A.1 presents the complete configuration of the stochastic environment used in the main experiments (Chapter 5), including all reward parameters, spawn probabilities, and patience/deadline settings.

A.2 PPO Training Hyperparameters

Table A.2 repeats the PPO configuration from Section 4.5.2 in a single reference table for convenience.

A.3 Economic Model Parameters

Table A.3 presents the parameters of the economic model used for reporting in Chapter 5. These values do not affect training; they are applied post hoc to evaluation episodes.

A.4 Patience Experiment Configurations

Table A.4 summarizes the full sequence of patience design iterations discussed in Section 5.6, including the exact configuration and key outcome metrics for each run. All values are verified against committed evaluation JSON files (see `VERIFICATION_CHECKLIST.md` in the project repository).

A.5 Greedy Baseline Priority Ordering

The greedy nearest-task heuristic used as a non-learning baseline (Section 4.6.2) follows a fixed priority ordering at every step:

1. Deliver onboard passenger (highest priority — never extend a passenger's trip).

Table A.1: Complete environment and reward configuration for the main experiments (baseline, no patience).

Parameter	**Value**	**Description**
Environment		
Grid side (N)	10	City grid dimension
Passenger slots	2	Max simultaneous passengers
Parcel slots	2	Max simultaneous parcels
Shift duration (T)	300	Time units per episode
Passenger spawn probability	0.02	Per step, per empty slot
Parcel spawn probability	0.10	Per step, per empty slot
Traffic multiplier (street)	1	Clear road
Traffic multiplier (cross-street)	2	Moderate congestion
Traffic multiplier (avenue)	3	Heavy congestion
Congested-corridor spacing	every 5 cells	Arterial pattern
Max passengers on board	1	Exclusive ride
Max parcels on board	unbounded	Shared cargo space
Reward function (tiered, baseline)		
Passenger delivery reward	$+20$	Per delivery
Parcel delivery reward	$+15$	Per delivery
Passenger pickup reward	0	No immediate reward
Step penalty (empty)	-0.40	Per tick, $\times$ traffic
Step penalty (parcel only)	-0.20	Per tick, $\times$ traffic
Step penalty (passenger)	-0.10	Per tick, $\times$ traffic
Patience (when enabled)		
USE_PATIENCE	False	Master toggle (baseline)
Passenger patience limit	30	Ticks before cancellation
Parcel locker patience	0	Disabled
Parcel carry deadline	80	Ticks after pickup
Passenger canceled penalty	-10	Per cancellation
Parcel expired penalty	-5	Per expiry
Flat-penalty ablation		
USE_FLAT_PENALTY	True	Ablation toggle
Flat step penalty	-0.20	All moves, $\times$ traffic

2. Pick up waiting passenger (passengers are higher-reward and more urgent).

3. Deliver onboard parcel (already invested in carrying it).

4. Pick up waiting parcel (lowest priority — parcels are patient at the locker).

Within each tier, ties are broken by Manhattan distance (nearest target wins). The heuristic is patience-blind: it does not consult urgency observations and follows the same priority ordering whether or not time-based constraints are active. When no active tasks exist, a random movement action is selected.

Table A.2: PPO hyperparameters used across all experiments.

Hyperparameter	**Value**
Policy architecture	MLP, two hidden layers of 512 units
Parallel environments	64
Rollout length (per env)	4096 steps
Optimization epochs	10
Minibatch size	1024
Learning rate	3×10^{-4}
Discount factor γ	0.99
Entropy coefficient	0.01
PPO clip parameter ε	0.2 (default)
Training budget (single-purpose)	50×10^6 steps
Training budget (mixed)	100×10^6 steps
Device	CUDA (GPU)

A.6 Full Statistical Comparison

Table A.5 provides the complete statistical results for all pairwise policy comparisons reported in Chapter 5. Each comparison uses 100 paired episodes evaluated on the same random seeds. Both a parametric paired t-test and a non-parametric Mann–Whitney U test are reported; agreement between the two confirms that the conclusions do not depend on distributional assumptions.

The sole discrepancy between the two tests occurs in the mixed PPO-vs-Greedy comparison, where the t-test yields $p = 0.501$ (not significant) while the Mann–Whitney test yields $p = 0.014$ (significant at the 0.05 level but not at 0.01). Given the negligible effect size ($d = 0.08$) and the high variance of the PPO policy in this scenario ($\sigma = 59.5$ vs. 28.8 for Greedy), the conservative interpretation, adopted in Chapter 5, is that the two policies are statistically indistinguishable in the mixed scenario.

Table A.6 reports the 95% bootstrap confidence intervals for the mean episodic return of each policy, computed via 10,000 bootstrap resamples.

A.7 Per-Scenario Complete Metrics

Table A.7 presents the complete set of domain and economic metrics for the PPO policy across all three scenarios, providing a single-point reference for all quantitative claims in Chapter 5.

A.8 Reward Computation Walkthrough

This section provides a concrete step-by-step example of how the per-step reward is computed, using the tiered (baseline) configuration.

Table A.3: Economic and environmental model parameters (reporting only).

Parameter	**Value**
Passenger base fare	5.0
Passenger fare per unit distance	1.0 per tile
Parcel flat fee	6.0
Operating cost per tile	0.3
Emissions proxy	Total tiles traveled (VKT)

Scenario. The vehicle is at position $(3,5)$ carrying one parcel (no passenger on board). It selects action "right" and moves to cell $(4,5)$, which is a main avenue cell (traffic multiplier $m = 3$). No request origin or destination coincides with $(4,5)$.

Computation.

1. **Delivery reward:** No delivery occurs at this step. $r_t^{\text{deliver}} = 0$.
2. **Load state:** The vehicle carries parcels but no passenger. Load state = *parcel-only*.
3. **Step penalty:** The parcel-only penalty is $c_t = -0.20$.
4. **Traffic multiplier:** The destination cell is an avenue, so $m_t = 3$.
5. **Total step penalty:** $c_t \times m_t = -0.20 \times 3 = -0.60$.
6. **Total reward:** $R(s_t, a_t) = 0 + (-0.60) = \mathbf{-0.60}$.

Contrast: same move while empty. If the vehicle had been empty, the penalty would be $-0.40 \times 3 = -1.20$, twice as costly. This difference is what incentivizes the agent to carry cargo rather than run empty.

Contrast: delivery on arrival. Now suppose cell $(4,5)$ is the destination of the carried parcel. The delivery is triggered automatically:

1. $r_t^{\text{deliver}} = +15$ (parcel delivery reward).
2. Step penalty remains $-0.20 \times 3 = -0.60$.
3. Total reward: $R(s_t, a_t) = 15 + (-0.60) = \mathbf{+14.40}$.

The net effect of a single successful delivery ($+14.40$) dominates approximately 24 penalty-free moves through ordinary streets ($24 \times -0.60 = -14.40$), which gives the agent a substantial incentive to complete deliveries efficiently.

Table A.4: Complete patience experiment configurations and results. Rates are percentages; detour is percentage extra distance. The "Baseline (off)" row refers to the patience-enabled starting point with an early carry deadline and penalty, not the main unconstrained baseline of Sections 5.3–5.6 (which uses no patience and achieves a parcel rate of 70.6%)

Run	**Patience (pax)**	**Carry deadline**	**Exp. penalty**	**Parcel rate**	**Parcel detour**	**Reward**
Baseline (off)[†]	–	40	−8	68.5%	302%	165.19
carry40/exp-8	30	40	−8	31.9%	655%	27.36
carry60/exp-8	30	60	−8	29.9%	946%	47.05
locker50/exp-2	30	200	−2	22.2%	2026%	67.53
locker50/exp-5	30	200	−5	67.6%	601%	158.70
+stay, tiered	30	200	−5	13.0%	2203%	88.62
+stay, flat	30	200	−5	12.1%	2012%	80.74
Final (pax30/c80)	30	80	−5	28.1%	1329%	107.63
Variant (pax50/c80)	50	80	−5	30.7%	1346%	113.73

[†]This row is the pre-patience baseline used as the starting point for the patience experiment sequence. It uses an early carry-deadline/penalty configuration (carry = 40, penalty = −8) that differs from the main unconstrained system (carry = 80, penalty = −5) reported in Sections 5.2–5.5.

A.9 Software and Hardware Environment

Table A.8 lists the software versions used for all experiments. The environment was developed as a custom Gymnasium environment; training and evaluation used the Stable-Baselines3 implementation of PPO.

A.10 Full Deterministic Reward Sweep Results

Table A.9 presents the complete results of the 24-configuration reward sweep described in Section 5.1. Each row is the mean across three independent training seeds ($\{42, 123, 777\}$), each evaluated over 1,000 episodes. Configurations are sorted by descending mean reward. The regime classification uses the 100% detour threshold defined in Section 5.1.2: passenger-shielding (PS) if $\delta_{\text{pax}} > 100\%$; parcel-hoarding (PH) if $\delta_{\text{par}} > 100\%$ and $\delta_{\text{pax}} < 100\%$; balanced co-modal (B) otherwise.

The table confirms the patterns discussed in Section 5.1: (i) all configurations with $p_s = -0.05$ (the cheapest passenger penalty) are classified as passenger-shielding and occupy the top of the reward ranking; (ii) all configurations with $p_p = -0.15$ (the cheapest parcel penalty) exhibit parcel hoarding; and (iii) the balanced configurations cluster at the lower end of the reward scale, confirming the reward–behavior decoupling.

Table A.5: Full statistical comparison of policies across all scenarios. CI = 95% bootstrap confidence interval; t = paired t-statistic; U = Mann–Whitney U statistic; d = Cohen's d.

Scenario	**Comparison**	**Δ Mean**	t	p (**t-test**)	U	p (**MW**)	d
People	PPO vs Random	+137.1	27.48	< 0.001	9940	< 0.001	3.97
People	PPO vs Greedy	+29.6	7.61	< 0.001	7033	< 0.001	0.72
Parcels	PPO vs Random	+222.6	38.64	< 0.001	9922	< 0.001	5.58
Parcels	PPO vs Greedy	+21.7	4.26	< 0.001	7770	< 0.001	0.54
Mixed	PPO vs Random	+239.8	35.99	< 0.001	9969	< 0.001	5.15
Mixed	PPO vs Greedy	+4.0	0.67	0.501	6006	0.014	0.08

Table A.6: Bootstrap 95% confidence intervals for mean episodic return.

Scenario	**Policy**	**Mean**	**95% CI**
People	PPO	45.94	[38.05, 53.71]
People	Greedy	16.32	[8.15, 24.53]
People	Random	−91.15	[−96.54, −85.75]
Parcels	PPO	132.73	[122.19, 142.13]
Parcels	Greedy	111.01	[106.32, 115.75]
Parcels	Random	−89.82	[−94.43, −85.13]
Mixed	PPO	171.52	[159.32, 182.68]
Mixed	Greedy	167.54	[161.91, 173.07]
Mixed	Random	−68.32	[−73.61, −63.01]

Table A.7: Complete domain and economic metrics for the PPO policy across all three scenarios. All values are means over 100 evaluation shifts.

Metric	**People**	**Parcels**	**Mixed**
Service			
Passengers delivered / shift	6.94	0.00	5.89
Parcels delivered / shift	0.00	12.83	7.20
Total deliveries / shift	6.94	12.83	13.09
Passengers spawned / shift	7.66	0.00	6.68
Parcels spawned / shift	0.00	15.74	9.74
Passenger delivery rate	89.6%	—	86.7%
Parcel delivery rate	—	80.6%	70.6%
Efficiency			
Empty-driving ratio	73.0%	37.2%	29.1%
Vehicle utilization	27.0%	62.8%	70.9%
Passenger waiting time (ticks)	11.4	—	12.6
Passenger detour	53%	—	58%
Parcel detour	—	155%	279%
Economic			
Revenue / shift	81.22	76.98	111.48
Operating cost / shift	74.86	68.83	66.87
Profit / shift	6.36	8.15	44.61
Distance / shift (tiles)	249.5	229.4	222.9
Emissions per delivery (tiles/del)	42.81	22.68	21.16
Reward			
Mean episodic return	45.94	132.73	171.52
Std episodic return	39.87	50.86	59.52

Table A.8: Software and hardware environment.

Component	**Version / Specification**
Python	3.8.10
PyTorch	2.4.1
Stable-Baselines3	2.4.1
Gymnasium	1.1.1
NumPy	1.24.4
SciPy	1.10.1
Matplotlib	3.7.5
Operating system	Ubuntu 20.04.6 LTS
GPU	NVIDIA RTX A6000
CPU	i7-13700K
RAM	64GB
Training time (50M steps)	$\approx$ 2 hours
Training time (100M steps)	$\approx$ 4 hours

Table A.9: Full deterministic reward sweep: all 24 valid configurations sorted by mean reward. p_e = empty penalty, p_p = parcel penalty, p_s = passenger penalty. Regime: PS = passenger-shielding, PH = parcel-hoarding, B = balanced.

p_e	p_p	p_s	$\bar{R}$	D_{pax}	D_{par}	E	δ_{pax}	δ_{par} **Regime**
−0.40	−0.25	−0.05	49.92	97.5%	96.7%	24.4%	225.0%	54.5% PS
−0.45	−0.25	−0.05	47.56	95.5%	93.9%	23.9%	224.3%	62.2% PS
−0.45	−0.20	−0.05	47.38	94.7%	93.2%	22.6%	221.1%	76.8% PS
−0.35	−0.25	−0.05	47.09	95.2%	93.9%	25.6%	191.1%	55.6% PS
−0.40	−0.20	−0.05	46.11	94.0%	91.4%	23.5%	203.9%	80.7% PS
−0.45	−0.15	−0.05	44.94	94.9%	92.7%	23.7%	51.9%	250.4% PH
−0.45	−0.20	−0.15	44.79	96.2%	93.9%	29.4%	0.2%	209.3% PH
−0.45	−0.25	−0.10	44.55	94.5%	91.4%	29.5%	122.3%	85.2% PS
−0.35	−0.25	−0.15	44.35	96.8%	95.7%	41.2%	0.2%	71.3% B
−0.35	−0.15	−0.05	44.33	96.2%	94.1%	34.1%	0.1%	155.4% PH
−0.40	−0.15	−0.10	44.15	95.7%	93.3%	29.6%	0.1%	207.6% PH
−0.40	−0.25	−0.15	44.10	96.5%	95.1%	40.9%	0.2%	72.3% B
−0.35	−0.15	−0.10	44.04	96.1%	93.7%	32.3%	0.2%	172.3% PH
−0.45	−0.20	−0.10	43.91	96.0%	93.1%	29.5%	0.3%	213.1% PH
−0.35	−0.25	−0.10	43.67	96.5%	94.6%	40.7%	0.3%	77.3% B
−0.40	−0.15	−0.05	43.26	94.4%	92.5%	28.5%	0.4%	217.4% PH
−0.35	−0.20	−0.15	43.12	96.2%	94.0%	39.9%	0.3%	89.9% B
−0.45	−0.25	−0.15	43.04	95.8%	92.8%	36.9%	0.3%	114.3% PH
−0.35	−0.20	−0.05	42.92	95.6%	92.5%	39.4%	2.8%	79.6% B
−0.40	−0.25	−0.10	42.85	95.5%	93.4%	39.0%	0.3%	97.8% B
−0.40	−0.20	−0.10	42.35	95.4%	92.1%	37.9%	0.4%	100.3% PH
−0.35	−0.20	−0.10	41.84	95.5%	92.7%	40.2%	0.2%	87.7% B
−0.40	−0.20	−0.15	41.47	94.4%	91.5%	34.5%	0.3%	137.2% PH
−0.45	−0.15	−0.10	41.42	93.8%	91.3%	29.0%	0.4%	222.8% PH