Real-Time Vehicle Repositioning Through Fast Optimal Transport

Sakshum Sharma[1], **Suresh Chavhan**[1], **Deepak Gupta**[2]

[1]Indian Institute of Science Education and Research Thiruvananthapuram, India
[2]Maharaja Agrasen Institute of Technology, Delhi, India
sakshum.sharma.1@gmail.com, suresh@iisertvm.ac.in, deepakgupta@mait.ac.in

Abstract

In urban mobility services such as ride-hailing and delivery logistics, vehicles must be strategically positioned during low-demand periods to serve sudden demand surges in specific geographic areas. To ensure that the demand is sufficiently met, there needs to be proactive as well as reactive planning. This paper proposes the formulation of dynamic vehicle routing problem(DVRP) with time-budget constraints where stochastic demand requires balancing reactive service of immediate requests with proactive positioning for anticipated future demand. The proposed approach is an adaptive repositioning framework that combines zone-level entropic optimal transport for vehicle-request assignment with online learning of spatial demand patterns. The Greenkhorn algorithm efficiently solves the regularized transport problem at each decision epoch, while exponential smoothing continuously updates zone preference weights that drive threshold-based repositioning of idle vehicles to anticipated high-demand areas. Further, the approach provides formal stability analysis under stationary stochastic demand, proving almost sure queue boundedness via Foster-Lyapunov drift conditions. Experiments on instances based on a real street network demonstrate effective proactive repositioning as compared to myopic, learning-based baselines without requiring offline training or explicit demand forecasting models.

Introduction

The vehicle routing problem (VRP) is a foundational challenge in operations research and logistics optimization, with applications ranging from last-mile delivery and emergency response to public transportation and supply chain management. Conventional VRP approaches assume complete knowledge of customer requests, with the objective of minimizing routing costs while serving all possible locations within environmental constraints. Despite many advances, this static nature diverges from the operations in urban systems where customer requests arrive unpredictably throughout service periods, traffic conditions fluctuate, and observed demand levels vary across time. With the rise of on-demand services, such as emergency response, ride-hailing, instant delivery, and real-time communication, logistics have undergone a transformation from offline planning to continuous, real-time decision-making processes under uncertainty. This gap between static models and dynamic operations necessitates Dynamic Vehicle Routing Problem (DVRP) frameworks, where routing decisions evolve continuously based on the revealed information.

Proactive repositioning guided by learned demand patterns is essential for anticipating real-time demand bursts. Throughout the day, new requests arrive, depending on the time and the requirement in a particular area. To serve the maximum number of requests throughout the day, the operators must update the assignments and reposition them in advance. Proactive vehicle repositioning in dynamic vehicle routing is challenging due to the uncertainty in future demand and the combinatorial state-action space of the problem. Modelling the time-budget constrained DVRP with spike demands as a complete undirected graph representing an urban road network. Unlike traditional DVRP formulations, vehicles operate without capacity constraints or customer time windows; the time budget constraints the routing decision. The system consists of a central depot where vehicles begin and return after completing routes to reset their time budgets. Between completion of the routes, the vehicles remain in their last-served position. This helps during idle periods to strategically place vehicles near anticipated high-demand areas rather than defaulting to depot returns. The objective is to maximize the number of requests served subject to route time budget constraints.

Existing methods mostly focus on routing heuristics (Psaraftis, Wen, and Kontovas 2016) or learning routing policies through reinforcement learning (Nazari et al. 2018). However, in practice, solution algorithms must incorporate strategic assignment. When deciding which vehicles should be placed during the idle period before requests arrive. Poorly assigned vehicles often lead to them being stranded in areas with lower demand, regardless of the routing quality. In this paper, we propose a framework that decomposes the problem into zone-level vehicle-to-demand assignment via entropic optimal transport (EOT) and proactive repositioning enabled by online demand learning, and the contributions are (1) Online learning approach that adapts to non-stationary demand requests using exponential smoothing, avoiding the need for specific offline policy training; (2) Formal stability analysis that proves almost sure boundedness under stationary stochastic demand using Foster-Lyapunov drift conditions; and (3) Experimental validation on a real

urban network showing substantial service rate gains compared to reactive and learning-based baselines.

Related Works

Early research on DVRP with stochastic requests follows a set of rules and does not incorporate probabilistic knowledge about future requests into its decision-making. (Bent and Van Hentenryck 2004b) and (Voccia, Campbell, and Thomas 2019) present a multiple scenario approach (MSA) that computes and reoptimizes plans for multiple scenarios, which corresponds to a static VRP involving known requests and randomly sampled future requests. In (Bent and Van Hentenryck 2004a), the optimal plan is selected using consensus, which returns the plan with the largest score or the lowest cost. (Ferrucci and Bock 2016), (Klapp, Erera, and Toriello 2018) decomposed the problem into sequential static VRPs. These methods schedule decision epochs to occur at the end of fixed time intervals, producing sufficient computation time for reoptimizing static VRPs. However, new requests cannot be scheduled or rejected before the current time period ends.

A Markov decision process (MDP) based approach is seen as a potential solution here (Pillac et al. 2013), (Psaraftis, Wen, and Kontovas 2016), (Ritzinger, Puchinger, and Hartl 2016), (Soeffker, Ulmer, and Mattfeld 2022). However, these approaches suffer from the curse of dimensionality, making them unsuitable for real-time applications. Approximate Dynamic Programming(ADP) has been developed to mitigate such issues. Look-ahead is one such ADP type that estimates future cumulative rewards through explicit scenario sampling and forward simulation. (Ulmer et al. 2019) integrates offline VFA into the online roll-out algorithm. Unfortunately, the advantages achieved by this method appear weak, as it simulates only 16 scenarios at each decision epoch.

Machine learning-based approaches (Peng, Wang, and Zhang 2019) solve combinatorial optimization problems using deep RL and graph neural networks, where extensive training is done to identify optimal routes. (Nazari et al. 2018), (Kool, Van Hoof, and Welling 2018), (Vinyals, Fortunato, and Jaitly 2015), (Bengio, Lodi, and Prouvost 2021). However, these methods require extensive training and struggle with distribution shifts when deployment scenarios differ from those in training.

(Jain 2010) surveys common clustering approaches, including K-means, DBSCAN, hierarchical clustering (single-link and MST), and spectral clustering. The cluster-first, route-second approach (Fisher and Jaikumar 1981) partitions customers into capacity-feasible clusters by solving a generalized assignment problem (GAP) that minimizes distances to cluster seeds, and then optimizes routes within each cluster independently. (Kumari et al. 2024) uses Clarke-Wright (CW) heuristics with a two-phase approach by first clustering customers to depots, then optimizing routes within clusters in multi-depot Capacitated VRP with Time Windows (MD-CVRPTW). While these methods achieve computational tractability through spatial aggregation, they usually rely on improvised clustering heuristics (distance-based k-means, sweeps) rather than optimization-based assignment mechanisms. Most focus on reactive policies without proactive repositioning for anticipated demand.

A recent survey shows that optimal transport can make informed decisions in transportation and routing (Moradi 2025). (Lei and Wu 2022) applies optimal transport to the ride-hailing setting by representing vehicle and request distributions onto spatial grids, solving for minimum Wasserstein distance transport plans using the Sinkhorn algorithm. (Haasler et al. 2024) models multi-vehicle dynamic routing as multimarginal OT over graphs, applying Sinkhorn iterations for efficient large-scale routing computations. (Wiedemann, Uscidda, and Raubal 2025) proposes GeOT, which combines EOT with neural networks for geospatial task prediction, where Sinkhorn divergence is used as the training loss function and evaluation criteria.

While most DVRP papers handle vehicle positioning as a reactive approach for servicing requests, repositioning-based approaches explicitly optimize idle vehicle placement to meet future demand. (Alonso-Mora et al. 2017) employs a two-tier framework that initially assigns current requests and then repositions idle vehicles toward predicted high-demand zones. (Ackermann and Rieck 2023) introduces overlapping zones and decides when and where to reposition vehicles from parking lots based on undersupply probabilities. (Brar et al. 2025) uses XGBoost explicitly for demand forecasting. (Bischoff and Maciejewski 2020) models vehicle rebalancing as the dynamic transportation problem. (Wallar et al. 2018) addresses repositioning of idle vehicles to restore spatial equilibrium (rebalancing) in shared-ride mobility-on-demand systems. (Guo et al. 2023) combines graph convolution network with column-and-constraint generation to model demand uncertainty. This paper divides the approach into rebalancing decisions and modelling worst-case demand.

Problem Statement

Identical vehicles are initially located at a central depot at the start of the planning horizon. Each vehicle operates under a time budget per route, with all routes starting and ending at the depot. Throughout the planning horizon, customer requests arrive dynamically from a stochastic process exhibiting temporal bursts. The time budget, when exhausted or below the decided threshold, the vehicle resets to the depot with a new budget. To ensure that vehicles are repositioned properly, the vehicles remain at the last customer location after completing service. The objective here is to maximize the service rate, which represents the number of served requests, in accordance with the vehicle time budget constraints. Balancing reactive service of immediate arrivals with proactive repositioning for anticipated demand. Our focus here is to fulfill the dynamic requests rather than cancellations; this request insertion needs anticipatory fleet positioning and real-time optimization under capacity constraints.

Model Formulation

Optimal Transport(OT) problems imply finding an efficient way to transform some initial distribution to target distribution(Santambrogio 2015). The standard OT formulation assumes equal total mass in both distributions, which

doesn't hold in our case as dynamic routing where vehicle availability and request arrivals fluctuate. OT problem becomes computationally intractable for large-size problems due to high number of variables. To address this, (Cuturi 2013) introduces an entropic barrier term in the objective function, this computes an approximate solution of large transportation problems via Sinkhorn iterations. Building on this work, (Altschuler, Weed, and Rigollet 2018) introduces greenkhorn algorithm which updates the single most-violating row/column per iteration instead of all rows and columns simultaneously.

Decision Epoch(DE): At each time-step t, the system receives newly-arrived requests, updates vehicle states (positions, remaining time budgets, route completion status). It executes a two-stage simultaneosu decision process: 1) exponential smoothing updates zone-level demand weights θ_t based on observed spatial demand patterns, (2) idle vehicles proactively reposition to high-demand zones before Greenkhorn assigns pending requests to available vehicles via zone-level optimal transport. This structure balances reactive service of immediate arrivals with anticipatory positioning for future demand across the planning horizon.

Actions: The system state at decision epoch k is represented as the tuple:

$$S_k = (t(k), v_{\text{pos}}(k), B(k), Q(k), \theta(k)), \tag{1}$$

where $t(k)$ denotes the current time, $v_{\text{pos}}(k) = (v_{\text{pos},i}(k))_{i=1}^{K}$ represents the spatial positions of all vehicles, $B(k) = (B_i(k))_{i=1}^{K}$ captures their remaining time budgets, $Q(k) = (r_j(k))_{j\in\text{pending}}$ contains all unserved requests, and $\theta(k) \in \Delta^{n_z}$ denotes the learned zone preference weights. After assignment and repositioning decisions, vehicle positions $v_{\text{pos}}(k)$ update based on movements, budgets $B(k)$ decrease as services begin, the queue $Q(k)$ removes served requests, and the zone weights $\theta(k)$ are updated via exponential smoothing from observed spatial demand. Figure 1 illustrates the complete framework. For conceptual clarity, the diagram uses continuous time notation t, corresponding to decision epoch k in the formal model.

Methodology

Adaptive repositioning framework with Greenkhorn assignment and online learning as shown in Fig. 1. The framework uses Greenkhorn, which is a sparse variant of the Sinkhorn algorithm that solves EOT problems by iteratively scaling rows and columns of the Gibbs kernel $K = \exp(-C/\varepsilon)$ to match prescribed marginal constraints. Unlike standard Sinkhorn, which updates all entries simultaneously, Greenkhorn identifies the most violating constraint at each iteration and updates only the corresponding row or column, achieving $O(n^2 \log n/\varepsilon^2)$ complexity for sparse settings.

In our setting, Greenkhorn takes the current supply distribution μ_t (vehicle positions across zones) and demand distribution ν_t (pending requests per zone) as inputs, then computes the transport plan π_t that minimizes expected travel costs $\langle C, \pi_t\rangle$ subject to entropic regularization. The Merton-inspired learning (Merton 1969) component updates zone preference weights θ_t via exponential smoothing based on observed spatial demand patterns, identifying regions with increased expected demand. These learned weights guide proactive vehicle repositioning: idle vehicles in low-demand zones ($\theta_i < \text{mean}(\theta)$) physically move to high-demand target zones ($\theta_j > 1.1\,\theta_i$), changing the fleet's spatial configuration and altering future supply distributions $\mu_{t'}$.

This enables anticipatory positioning for demand bursts rather than purely reactive service. The approach combines strategic repositioning with Greenkhorn's principled assignment optimization, both operating over the fixed zone cost matrix C.

Optimization Objective: Our formulation requires assessing two competing goals simultaneously: maximize service rate by balancing cost-efficient assignments with anticipatory repositioning for future demand. The stochastic aspect arises from uncertainty in demand arrivals and service times, which makes future assignment costs random rather than deterministic. Given the observed supply distribution μ_t and demand distribution ν_t, the transport plan is obtained by minimizing the expected transportation cost under entropic regularization:

$$\min_{\pi_t\in\Pi(\mu_t,\nu_t)} \big\langle C, \pi_t\big\rangle + \varepsilon\,\mathrm{KL}(\pi_t \,\|\, K). \tag{2}$$

The feasible set is the transport polytope

$$\Pi(\mu_t, \nu_t) = \big\{\, \pi \geq 0 : \pi\mathbf{1} = \mu_t,\ \pi^\top\mathbf{1} = \nu_t \,\big\}, \tag{3}$$

which enforces the supply and demand marginals. Here, $C \in \mathbb{R}_+^{n_z\times n_z}$ is the fixed zone-to-zone cost matrix representing expected travel times, $K = \exp(-C/\varepsilon)$ is the Gibbs kernel, and $\varepsilon > 0$ is the entropic regularization parameter.

The regularization term $\varepsilon\,\mathrm{KL}(\pi_t \,\|\, K)$ provides numerical stability, enables efficient computation via Greenkhorn's sparse scaling updates, and yields smooth probabilistic assignments that handle supply–demand imbalances.

Planning Strategy: The approach combines two mechanisms operating(as detailed in Model Formulation). First, the demand pattern learning and assignment optimization(DE1: Learning). Second, exponential smoothing continuously updates zone preference weights θ_t based on observed spatial demand, identifying regions with increased expected requests (DE2: Repositioning). These learned weights inform proactive repositioning decisions for idle vehicles, physically moving them from low-demand zones to anticipated high-demand areas. Subsequently, Greenkhorn solves the zone-level optimal transport problem given the repositioned fleet configuration, assigning vehicles to current pending requests via cost-minimizing transport plans.

Exponential Smoothing for Demand Learning: Merton's portfolio optimization (Merton 1969), which proactively repositions financial capital across assets based on learned risk-return profiles to maximize long-term utility under uncertainty, we develop an adaptive approach that applies analogous principles to fleet positioning in dynamic vehicle routing. We take the core intuition of anticipatory resource calculation from statistical learning into a computationally tractable online learning. Our approach combines exponential smoothing to continuously learn spatial demand

patterns θ_t from observed arrivals d_t, with threshold-based repositioning rules that physically move idle vehicles from low-demand zones to high-demand targets. This creates an adaptive strategy where repositioning decisions evolve dynamically as θ_t tracks shifting demand distributions: during non-rush hour periods, θ_t remains relatively uniform and triggers minimal repositioning; during pre-rush hour periods when certain zones exhibit increased θ values, it proactively concentrates vehicles in anticipated hotspots before the actual demand materializes. The adaptivity arises from the coupling between learning (θ_t responds to recent observations with exponential memory decay) and action (repositioning thresholds automatically adjust as the learned distribution changes), enabling the system to transition between reactive and proactive modes based on the current demand regime.

Repositioning Decision: The proposed repositioning strategy has a threshold which balances optimality against operational feasibility in the following ways: 1) Condition $\theta_i < \text{mean}(\theta_t)$ identifies vehicles in zones with below-average demand concentration, filtering out vehicles that are already well-positioned. 2) Ratio threshold $\theta_j/\theta_i > 1.1$ ensures that repositioning targets offer substantive demand improvement (at least 10% higher), avoiding marginal moves that consume travel time without strategic benefit. 3) Reachability constraint $C_{ij} < 35$ minutes prevents long-distance repositioning that would strand vehicles during transit when nearby demand materializes.

Algorithm 1 presents the complete adaptive repositioning framework for our setting. The computational complexity per decision epoch is $O(n^2 \log n/\varepsilon^2)$ due to Greenkhorn assignment, with exponential smoothing adding $O(n_z+K\cdot n_z)$ overhead. Thus, the overall complexity is $O(n^2 \log n + n_z)$.

Algorithm 1: Adaptive Repositioning for Dynamic VRP

Require: Zone structure (n_z, C), fleet size K, horizon T, parameters (α, ε)
Ensure: Served requests

Initialize $\theta_0 \in \Delta^{n_z}$, vehicle positions at depot
for $t = 0, \Delta t, 2\Delta t, \ldots, T$ **do**
  Observe arrivals d_t, update queue_t
  // Learn demand distribution
  $\theta_{t+1} \leftarrow \alpha \cdot \frac{d_t}{\|d_t\|_1} + (1-\alpha) \cdot \theta_t$
  // Proactive repositioning
  for each idle vehicle i in zone z_i with $\theta_{t+1}(z_i) < \text{mean}(\theta_{t+1})$ **do**
    Find target zone z_j such that:
      $\theta_{t+1}(z_j)/\theta_{t+1}(z_i) > 1.1$ and $C_{z_i,z_j} < 35$
    Move vehicle i to zone z_j
  end for
  // Compute zone-level assignment
  Aggregate supply μ_t and demand ν_t per zone
  $\pi_t \leftarrow \text{Greenkhorn}(\mu_t, \nu_t, C, \varepsilon)$ // Solve entropic OT
  // Execute assignments and routing
  Assign requests to vehicles using π_t
  Route each vehicle using the cheapest insertion
  Update vehicle states and served requests
end for
return Served requests

Stability Analysis

Assumption 1 (Demand Regularity). The demand process $\{d_t(z)\}_{t\geq 0,\, z\in[n_z]}$ satisfies: (1) **Bounded expectation:** $\mathbb{E}[d_t(z)] \leq \lambda_{\max} < \infty$ for all t, z. (2) **Bounded variance:** $\text{Var}(d_t(z)) \leq \sigma^2 < \infty$ for all t, z. (3) **Finite second moment:** $\mathbb{E}[\|d_t\|^2] \leq M^2 < \infty$. (4) **Positive lower bound:** $\mathbb{P}(\|d_t\|_1 \geq \lambda_{\min}) = 1$ for some $\lambda_{\min} > 0$.

Assumption 2 (Network Connectivity). The zone cost matrix C is irreducible: for any pair of zones (i, j), there exists a sequence of zones $i = z_0, z_1, \ldots, z_m = j$ such that $C_{z_k,z_{k+1}} < \infty$ for all k.

Assumption 3 (Gibbs Kernel Regularity). The Gibbs kernel $K = \exp(-C/\varepsilon)$ satisfies: (1) **Boundedness:** $0 < k_{\min} \leq K_{ij} \leq k_{\max} < \infty$ for all i, j. (2) **Primitivity:** K is primitive (there exists an m such that K^m is strictly positive). (3) **Condition number:** The ratio $\kappa = k_{\max}/k_{\min}$ is finite. The contraction rate of Greenkhorn satisfies $\rho_G \approx 1 - O(1/\kappa)$. For small regularization ε, the Gibbs kernel becomes ill-conditioned (i.e., large κ), which requires stabilization through log-domain Sinkhorn updates or adaptive regularization.

Lemma 1 (Greenkhorn Feasibility). Let $K \in \mathbb{R}_{++}^{n_z\times n_z}$ be a positive, primitive Gibbs kernel satisfying Assumption 3. For any supply distribution $\mu \in \mathbb{R}_+^{n_z}$ and demand distribution $\nu \in \mathbb{R}_+^{n_z}$ with $\sum_i \mu_i = \sum_j \nu_j$, the Greenkhorn algorithm computes a transport plan π such that:

$$\|\pi\mathbf{1} - \mu\|_1 + \|\pi^\top\mathbf{1} - \nu\|_1 \;\leq\; \varepsilon_{\text{fast}},$$

in

$$T_{\text{fast}} \;=\; O\Big(\varepsilon_{\text{fast}}^{-2} \log\Big(\frac{n_z}{\kappa}\Big)\Big),$$

where $\kappa = k_{\max}/k_{\min}$ is the condition number of K and $\varepsilon_{\text{fast}} > 0$ is the desired feasibility tolerance.

Proof. This follows from Theorem 2 of (Altschuler, Weed, and Rigollet 2018), which establishes that Greenkhorn achieves ε-approximate feasibility of the marginal constraints with iteration complexity $O(\varepsilon^{-2}\log(n/\kappa))$. The algorithm iteratively selects the row or column with the maximum marginal violation and updates the corresponding dual variable to reduce that violation.

Remark. The transport plan π computed at each time t provides a near-feasible assignment between supply μ_t and demand ν_t. While this plan is not necessarily the exact minimizer of $\langle C, \pi\rangle$, the entropic regularization ensures that π is a smooth, numerically stable approximation to the optimal assignment.

Lemma 2 (Simplex Constraint Preservation and Compactness). For any initial condition $\theta_0 \in \Delta^{n_z}$, the exponential smoothing update preserves the simplex constraint almost surely:

$$\theta_t \in \Delta^{n_z} \qquad \text{for all } t \geq 0,\ \mathbb{P}\text{-a.s.}$$

Moreover, since Δ^{n_z} is compact, the sequence $\{\theta_t\}$ is uniformly bounded almost surely:

$$\|\theta_t\|_2 \leq 1 \quad \text{and} \quad \|\theta_t\|_1 = 1 \qquad \text{for all } t \geq 0,\ \mathbb{P}\text{-a.s.}$$

Proof. We proceed by induction. *Base case:* By assumption, $\theta_0 \in \Delta^{n_z}$, so $\sum_z \theta_0(z) = 1$ and $\theta_0(z) \geq 0$ for all z.

Inductive step: Assume $\theta_t \in \Delta^{n_z}$. The update is

$$\theta_{t+1}(z) = \alpha \frac{d_t(z)}{\|d_t\|_1} + (1-\alpha)\theta_t(z). \tag{4}$$

Since $d_t(z) \geq 0$, $\theta_t(z) \geq 0$ (inductive hypothesis), and $\alpha \in (0,1)$, we have $\theta_{t+1}(z) \geq 0$.

For the sum,

$$\begin{aligned}\sum_z \theta_{t+1}(z) &= \alpha \sum_z \frac{d_t(z)}{\|d_t\|_1} + (1-\alpha)\sum_z \theta_t(z) \\ &= \alpha \cdot 1 + (1-\alpha)\cdot 1 = 1.\end{aligned} \tag{5}$$

Thus $\theta_{t+1} \in \Delta^{n_z}$, completing the induction.

Remark. The simplex constraint is preserved automatically because the update is a convex combination; hence no projection is required.

For boundedness, if $x \in \Delta^{n_z}$, then

$$\|x\|_2^2 = \sum_z x(z)^2 \leq (\sum_z x(z))^2 = 1, \tag{6}$$

by the Cauchy–Schwarz inequality. Thus $\|\theta_t\|_2 \leq 1$. The ℓ_1 norm satisfies $\|\theta_t\|_1 = 1$ by definition. Since

$$\Delta^{n_z} = \{x \in \mathbb{R}_+^{n_z} : \sum_i x_i = 1\} \tag{7}$$

is closed and bounded in $\mathbb{R}^{n_z}$, it is compact by the Heine–Borel theorem.

Assumption 4 (Observed Marginals). At each time t, the marginals (μ_t, ν_t) are $\mathcal{F}_t$-measurable; that is, they are deterministic functions of the system state (vehicle positions and arriving requests) at time t.

Assumption 5 (Service Feasibility). There exists a constant $p_{\text{service}} > 0$ such that any available vehicle can successfully serve any pending request with probability at least p_{service}, given sufficient time budget. This follows from network connectivity (Assumption 2) and the primitivity of the Gibbs kernel (Assumption 3). This assumption can be violated under extreme operational scenarios: (1) catastrophic demand surges and (2) spatial demand isolation

Practical Bounds on p_{service}: For budget-feasible zone pairs ($C_{ij} + s_{\max} < T_{\text{budget}}$), which consists 95% of zone combinations, a vehicle available has the capacity to reach and serve the request if assigned. However, p_{service} accounts for assignment competition: Greenkhorn may allocate the vehicle anywhere else, or the vehicle may serve other requests first, exhausting remaining budget.

Lemma 3 (Bounded Tracking). Under Assumptions 1 and 4, the sequence $\{\theta_t\}$ generated by the exponential smoothing update remains bounded with finite tracking error. Specifically, there exists a constant $C_{\text{tracking}} < \infty$ depending on system parameters $(\alpha, \sigma^2, \lambda_{\max}, \lambda_{\min}, n_z)$ such that

$$\sup_{t \geq 0} \mathbb{E}\big[\|\theta_t\|^2\big] \leq C_{\text{tracking}}. \tag{8}$$

Proof. From Lemma 2, we have $\theta_t \in \Delta^{n_z}$ for all t almost surely. Since the simplex is compact and satisfies $\|\theta_t\|_2 \leq 1$, it follows that

$$\mathbb{E}\big[\|\theta_t\|^2\big] \leq \mathbb{E}[1] = 1 \qquad \text{for all } t. \tag{9}$$

Hence $C_{\text{tracking}} = 1$ suffices.

Remark on Tracking Error. Although the sequence $\{\theta_t\}$ does not converge to a fixed point (due to the circular dependence where θ_t influences future demand observations via repositioning decisions), we can characterize its tracking behavior.

Let

$$\text{target}_t = \frac{d_t}{\|d_t\|_1} \tag{10}$$

denote the normalized observed demand at time t. The update can be written as

$$\theta_{t+1} = (1-\alpha)\,\theta_t + \alpha\,\text{target}_t. \tag{11}$$

Since both θ_t and target_t lie in Δ^{n_z}, whose diameter is bounded by $\sqrt{2}$, the tracking error satisfies

$$\mathbb{E}\big[\|\theta_t - \text{target}_t\|^2\big] \leq 2 \qquad \text{for all } t. \tag{12}$$

The sequence $\{\theta_t\}$ does not converge, but instead tracks the time-varying demand distribution with bounded error. The fixed step size $\alpha = 0.5$ yields fast adaptation to regime changes while maintaining a steady-state tracking error. The bounded tracking property ensures that the demand-weight dynamics remain well-behaved and do not amplify variability, this is an important requirement for the stability result in Theorem 1.

Theorem 1 (System Stability)

Statement. Under Assumptions 1–5, suppose the fleet size K satisfies $K > \lambda_{\max} n_z \Delta t$ (i.e., total capacity exceeds the aggregate expected arrival rate). Then the system state remains bounded almost surely:

$$\mathbb{P}\left(\sup_{t \geq 0} \text{queue}_t < \infty\right) = 1,$$

where queue_t denotes the number of pending requests at time t.

Proof. See Appendix. □

Remark on Greenkhorn's Role. The $\varepsilon_{\text{fast}}$-feasibility of the transport plan (Lemma 1) ensures that the assignment mechanism does not systematically mismatch supply and demand. While the plan may not minimize transport cost exactly, the small marginal constraint violations, $\|\pi_t \mathbf{1} - \mu_t\|_1 \leq \varepsilon_{\text{fast}}$, guarantee that vehicles in zones with capacity are assigned to zones with requests, preventing deadlock. For practical values such as $\varepsilon_{\text{fast}} = 10^{-9}$, these marginal errors are negligible relative to fleet size and demand, ensuring that the computed assignments translate directly into service.

Remark on Repositioning. The stability result implicitly relies on the repositioning preventing "spatial starvation" situations where vehicles cluster in low-demand zones while

high-demand zones remain underserved. Although we do not formally analyze exponential smoothing (e.g., proving minimum-supply guarantees over sliding windows), the bounded tracking behavior of θ_t (Lemma 3) ensures that repositioning decisions remain responsive to demand patterns without diverging. Empirical performance further supports the effectiveness of this strategy in practice.

Remark on Capacity Assumption. The condition $K > \lambda_{\max} n_z \Delta t$ is sufficient but not necessary for stability. In practice, stability depends on the spatial distribution of demand and the effectiveness of repositioning. The theorem provides a conservative bound guaranteeing stability whenever total fleet capacity exceeds the peak aggregate arrival rate across all zones.

Empirical Results

Experiment Scenario and Design

Our experiments utilize the Vienna network from (Zhang et al. 2023), which is built on OpenStreetMap data and comprises 16,080 intersections and 36,424 road segments. The depot is centrally located, with a network partitioned.The CTD stochastic request distribution from (Zhang et al. 2023), have three demand intensities ($\Lambda \in \{0.2, 0.4, 0.8\}$ req/min) with 5 instance files intensity. Each instance is augmented with artificial burst patterns consisting of 50 requests at 2.5 requests per minute within designated 20-minute windows, with preceding idle time windows. The Greenkhorn solver uses a regularization parameter $\epsilon = 0.5$ with a tolerance of 10^{-9} and a maximum of $10{,}000$ iterations. Exponential smoothing uses $\alpha = 0.5$. Vehicle repositioning is triggered when zone potential ratios exceed 1.1 and travel costs are below 35 minutes, with vehicles traveling at an average speed of 40 km/h.

The spatial partitioning of the Vienna road network was done into 30 zones using k-means clustering based on geographic coordinates. Each zone is represented by a different color, with each centroid marked by a red cross. This zoning enables Greenkhorn to operate efficiently with a 30×30 cost matrix. The 30 zones had the perfect computational tractability while preserving the essential spatial structure of the network. Increasing the zones had a negligible improvement on Greenkhorn's work. The maximum inter-zone diameter of 3,352.9 minutes ensures all zone pairs remain reachable within the planning horizon, satisfying the network connectivity assumption required for system stability. Single-burst files concentrate the demands in a singular peak window, modeling a three-phase temporal structure consisting of baseline demand, an idle window with reduced arrival rates, and a concentrated burst period representing peak demand. For double bursts, two spikes are used to model morning and evening rush periods within a single service day, separated by a 160-minute repositioning window. The bursts mimic realistic demand patterns where rush-hour surges concentrate in high-density urban areas such as business districts or transportation hubs. It exhibits spatial clustering with 98% concentration in high-density areas. Non-burst periods maintain steady background demand, allowing comparison between reactive performance and proactive gains.

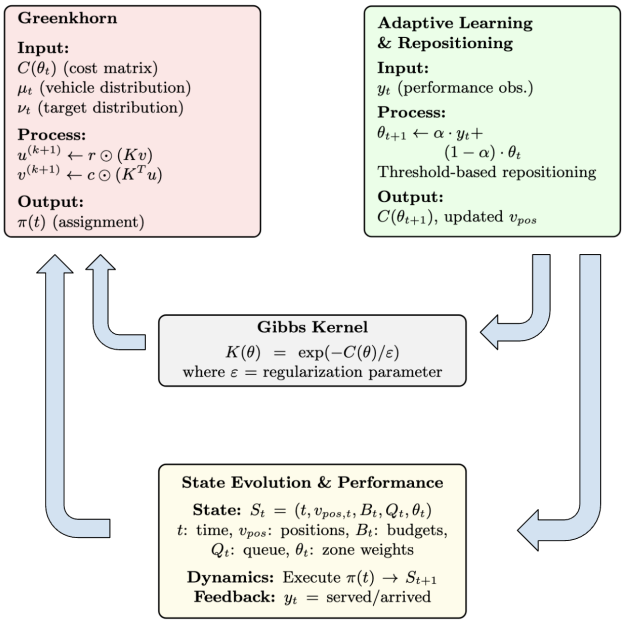

Figure 1: Adaptive repositioning framework with Greenkhorn assignment and online learning.

Baselines

We evaluated the performance of our Greenkhorn formulation against the following baselines. One myopic baseline(greedy) and the learning based baseline(DRLSA)(Joe and Lau 2020).

- **Greedy:** This type of assignment evaluates all available vehicles, and then selects the vehicle with the lowest intermediate cost. It does not consider future requests, workload balancing across zones, or route feasibility beyond simply checking capacity/time constraints. Once the best vehicle is identified, the request is appended to that vehicle's current route (no reordering or planning ahead). The decision is myopic: it optimizes only the current request, not the system-wide future demand. This often overloads the most central or nearest vehicles while leaving others underutilized.
- **Deep reinforcement learning-based vehicle routing:**(Joe and Lau 2020) The implementation does *not* use simulated annealing—this component is disabled. Normally, simulated annealing is a metaheuristic that explores the search space by occasionally accepting worse solutions with a decreasing probability; however, in this case, the algorithm uses a deterministic cheapest-insertion routing heuristic instead, to have comparable results.
- **Pure Greenkhorn:** This reactive baseline does not account for the proactive repositioning. It isolates the greenkhorn working in this problem setting. Its arrival requests. Vehicles are assigned based on their current position after completing previous routes, with no strategic repositioning during idle periods. While this provides su-

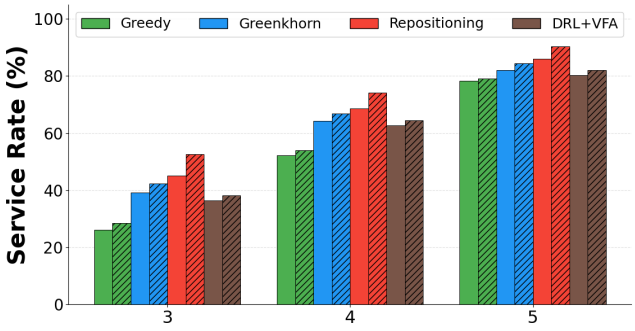

Figure 2: Service rate comparison across fleet sizes($\Lambda = 0.2$).

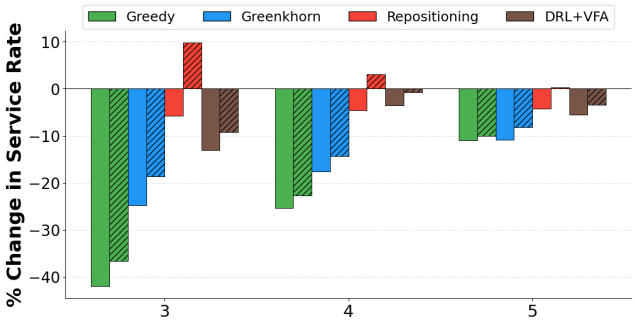

Figure 3: Service rate degradation under burst patterns compared to the non-burst baseline.

perior load balancing compared to greedy (vehicles distributed proportionally to zone-level flows rather than individually selected), it cannot proactively position the fleet for anticipated demand increase.

Results

Service Rates: The objective is to serve the maximum number of requests in the planning horizon. The service rate is provided in Figure 2 for fleet sizes of 3, 4, and 5 vehicles for $\Lambda = 0.2$ instance for single bursts. The repositioning formulation outperforms all baselines across all fleet sizes, with varying percentage differences. Figure 3 provides service rate degration. **Single Burst:** In this pattern there is a one demand increase beginning at t=200 after an idle period preceding this time period. In the lower request instance, the repositioning approach shows better results than baselines by 4-6% in varying fleet size. DRL+VFA approach closes the gap when fleet size is highest. Ensuring our methods efficacy to learning based method. Reactive only approach trails closes to DRL+VFA by a margin of 1-2%. Similar pattern can be seen in middle request instances. In the higher request instances, repositioning fares better than other approaches by approximately 5% showing good resilience in handling higher requests instances. At the highest fleet size of 5 vehicles, all methods approach saturation as sufficient resources enable even reactive strategies to partially succeed. The consistent advantage at constrained capacity of

Method	K / Λ	Single burst scenarios			Double burst scenarios		
		0.2	0.4	0.8	0.2	0.4	0.8
Greedy	3	26 ± 0.0	26 ± 0.0	28 ± 0.0	20 ± 0.0	20 ± 0.0	28 ± 0.0
	4	52 ± 0.0	52 ± 0.0	48 ± 0.0	34 ± 3.1	35 ± 3.5	48 ± 2.9
	5	68 ± 2.4	71 ± 2.9	72 ± 2.6	45 ± 2.3	49 ± 2.7	54 ± 3.4
Greenkhorn	3	39 ± 5.9	46 ± 4.2	35 ± 3.8	42 ± 5.9	44 ± 4.2	37 ± 3.8
	4	64 ± 5.1	67 ± 5.1	58 ± 4.3	66 ± 4.5	68 ± 5.1	60 ± 4.3
	5	82 ± 3.2	83 ± 3.6	78 ± 4.1	84 ± 3.2	88 ± 3.6	82 ± 4.1
Repositioning	3	45 ± 4.8	49 ± 5.2	52 ± 4.6	52 ± 4.8	62 ± 5.2	61 ± 4.6
	4	68 ± 4.1	75 ± 3.9	71 ± 4.4	74 ± 4.1	75 ± 3.9	72 ± 4.4
	5	86 ± 2.8	89 ± 2.5	83 ± 3.3	90 ± 2.8	94 ± 2.5	88 ± 3.3
DRL+VFA	3	36 ± 4.5	48 ± 4.8	52 ± 4.6	38 ± 1.5	40 ± 1.8	38 ± 1.6
	4	62 ± 4.1	65 ± 4.4	70 ± 5.0	64 ± 2.1	54 ± 2.4	46 ± 2.0
	5	80 ± 5.9	79 ± 5.2	82 ± 5.7	79 ± 2.9	70 ± 3.2	64 ± 2.7

Table 1: Service rate comparison (mean $\pm$ std) across fleet sizes (K) and demand intensities (Λ) under single and double burst scenarios.

fleet size 3,4 validates that proactive learning-based repositioning provides meaningful gains even for single burst scenarios. All the values with instances are provided in Table 1. **Double Burst:** Two concentrated demand surges occur sequentially with preceding idle window. In this case our approach outperforms all the methods by approximately 7% gains in initial request files. This gap widens more in the highest instances by 9% when fleet size is 3. Repositions the vehicles during idle time period and arrives pre-positioned when the second burst commences. In contrast, reactive methods (Greedy, Pure Greenkhorn) must respond to each burst independently without leveraging information from prior surges, while DRL+VFA's offline-trained policy cannot adapt online between bursts. As evident from the single bursts, all methods approach saturation when the fleet size is 5.

Across both the patterns, our method consistently outperforms baselines by 4-6 % further by 10-12% in double bursts condition, validating robust generalization to multiple demand surges.

Computation time: We evaluate computational efficiency using median processing time per request across all methods. Computation time is logged every 5 minutes at each decision epoch in the simulation, documenting the wall-clock time needed for both assignment (Greedy/Greenkhorn solver) and repositioning logic (repositioning's θ-weight updates). The median is calculated across all decision epochs from 5 instance files to offer a stable central tendency measure that is unaffected by outliers from warm-up phases or garbage collection incidents. Figure 4 displays results in grouped scatter plots, where filled markers represent single-burst situations and hollow markers signify double-burst situations, accompanied by error bars illustrating the minimum and maximum ranges across instances. Greedy achieves the quickest assignment durations (0.6-0.9 ms/request) thanks to its straightforward nearest-neighbor approach, whereas DRL+VFA shows moderate inference latency (3.4-3.8 ms/request) even though it doesn't need any online optimization. Greenkhorn faces incnreased expenses (5.2-5.8 ms/request) due to resolving the optimal transport issue through Sinkhorn iterations. Repositioning has

an overhead of just 17-20% relative to Greenkhorn (6.1-6.7 ms/request), resulting in about 1 ms extra cost per request for θ-weight exponential smoothing and repositioning threshold validations. Importantly, all approaches function effectively within real-time constraints: even Repositioning's worst-case scenario of 7 ms per decision is insignificant in relation to the 300,000 ms (5-minute) planning period between decision points. In contrast to DRL+VFA, which requires significant offline training (approximately 1000 episodes), repositioning functions without training and has a polynomial-time complexity of $O(n^2 \log n + n_z)$ for each epoch, enabling immediate deployment in operational systems. The slight computational expense yields significant improvements in service rates (15-25 percentage points higher than Greenkhorn), indicating a beneficial cost-benefit balance for real-world DVRP applications.

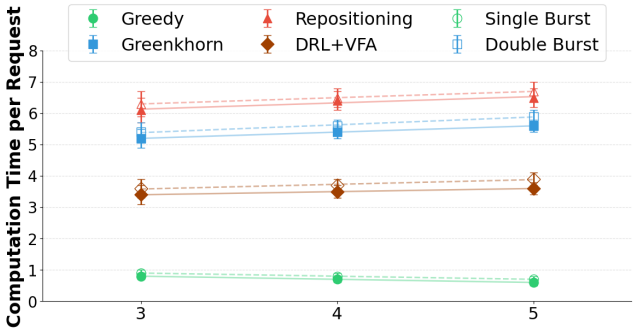

Figure 4: Computation time under burst patterns and median computation time across fleet sizes.

Sensitivity analysis: We perform univariate sensitivity analysis, varying each hyperparameter while holding others at baseline values ($\alpha = 0.5$, $\varepsilon = 0.5$, $\theta_{\text{ratio}} = 1.1$, $C_{\max} = 35$ min). Sensitivity analysis across $\alpha \in \{0.3, 0.5, 0.7\}$, $\theta_{\text{ratio}} \in \{1.05, 1.1, 1.2\}$, $C_{\max} \in \{25, 35, 45\}$ minutes, and $\varepsilon \in \{0.1, 0.5, 1.0\}$ shows that the method is robust to hyperparameter choices, with service rate variations $< 2\%$ across all tested ranges ($K = 3$, $\Lambda = 0.2, 0.4, 0.8$, single burst scenarios). Repositioning thresholds (θ_{ratio}, $C_{\max}$) demonstrate particular stability, varying $< 0.5\%$ between tested values, indicating that zone-level demand disparities during bursts are large enough to trigger repositioning regardless of the threshold choice. We select default values $\alpha = 0.5$, $\varepsilon = 0.5$ balancing performance and computational efficiency.

Conclusion

In this paper, a novel assignment strategy for DVRP is presented, combining both reactive and proactive strategies. Greenkhorn-based optimal transport with online demand learning through exponential smoothing. By adapting Merton's portfolio theory to hedge our bets on highly probable zones for repositioning during idle periods, the proposed approach outperforms the baselines in varying conditions. Theoretical analysis established stability guarantees: Greenkhorn achieves feasible transport plans, exponential smoothing preserves simplex constraints with bounded tracking error, and Foster-Lyapunov drift conditions prove almost sure queue boundedness under stationary stochastic demand. Future work includes extending this framework to other variants of DVRP, like Capacitated DVRP(CVRP).

Appendix

Proof of Theorem 1 (System Stability)

Proof. We construct a Lyapunov argument using the queue length.

1. Lyapunov function:

$$V_t = \text{queue}_t. \tag{13}$$

2. Queue evolution:

$$\text{queue}_{t+1} = \text{queue}_t + \text{arrivals}_t - \text{served}_t \tag{14}$$

$$\mathbb{E}[\text{queue}_{t+1} \mid \mathcal{F}_t] = \text{queue}_t + \mathbb{E}[\text{arrivals}_t \mid \mathcal{F}_t] \tag{15}$$

$$-\mathbb{E}[\text{served}_t \mid \mathcal{F}_t] \tag{16}$$

$$\mathbb{E}[\text{arrivals}_t \mid \mathcal{F}_t] \le \lambda_{\max} n_z \Delta t. \tag{17}$$

Let available_t denote the number of idle vehicles at time t, $\gamma \in (0, 1)$ the minimum fleet availability fraction, and $\beta = \frac{\gamma K}{2} p_{\text{service}} - \lambda_{\max} n_z \Delta t > 0$ the negative drift coefficient (positive by the capacity assumption $K > \lambda_{\max} n_z \Delta t$).

3. Service capacity lower bound:

$$\|\pi_t \mathbf{1} - \mu_t\|_1 + \|\pi_t^\top \mathbf{1} - \nu_t\|_1 \le \varepsilon_{\text{fast}} \tag{18}$$

$$\mathbb{E}[\text{served}_t \mid \mathcal{F}_t] \ge \min(\text{queue}_t, \text{available}_t) p_{\text{service}} \tag{19}$$

4. Fleet availability:

$$\text{available}_t \ge \gamma K/2 \quad \text{(with high probability)}. \tag{20}$$

5. Drift conditions:

Case 1: Large queue ($\text{queue}_t \ge \gamma K$):

$$\begin{aligned} &\mathbb{E}[\text{served}_t \mid \mathcal{F}_t] \ge (\gamma K/2)\, p_{\text{service}}, \\ &\mathbb{E}[\text{queue}_{t+1} \mid \mathcal{F}_t] \le \text{queue}_t + \lambda_{\max} n_z \Delta t - (\gamma K/2) p_{\text{s}}, \\ &\mathbb{E}[\text{queue}_{t+1} \mid \mathcal{F}_t] \le \text{queue}_t - \beta. \end{aligned} \tag{21}$$

Case 2: Small queue ($\text{queue}_t < \gamma K$):

$$\mathbb{E}[\text{queue}_{t+1} \mid \mathcal{F}_t] \le \text{queue}_t (1 - p_{\text{service}}) + \lambda_{\max} n_z \Delta t, \tag{22}$$

$$\mathbb{E}[\text{queue}_{t+1} \mid \mathcal{F}_t] \le C_{\text{queue}}, \quad C_{\text{queue}} = \frac{\lambda_{\max} n_z \Delta t}{p_{\text{service}}}. \tag{23}$$

6. Almost sure boundedness: The system state $z_t = (\text{queue}_t, \theta_t, \text{vehicle\_positions}_t)$ evolves as a discrete-time Markov process. By the drift condition (iii) of Theorem 14.0.1 in Meyn and Tweedie (2009, p. 335) (Meyn and Tweedie 2012), the established Foster–Lyapunov drift inequality implies positive recurrence: there exists $M < \infty$ such that

$$\mathbb{P}\left(\limsup_{t \to \infty} \text{queue}_t \le M \right) = 1 \tag{24}$$

□

Acknowledgements

We thank the anonymous reviewers for their insightful comments that strengthened this paper.

References

Ackermann, C.; and Rieck, J. 2023. A novel repositioning approach and analysis for dynamic ride-hailing problems. *EURO Journal on Transportation and Logistics*, 12: 100109.

Alonso-Mora, J.; Samaranayake, S.; Wallar, A.; Frazzoli, E.; and Rus, D. 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, 114(3): 462–467.

Altschuler, J.; Weed, J.; and Rigollet, P. 2018. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. arXiv:1705.09634.

Bengio, Y.; Lodi, A.; and Prouvost, A. 2021. Machine learning for combinatorial optimization: a methodological tour d'horizon. *European Journal of Operational Research*, 290(2): 405–421.

Bent, R.; and Van Hentenryck, P. 2004a. The Value of Consensus in Online Stochastic Scheduling. In *ICAPS*, volume 4, 219–226.

Bent, R. W.; and Van Hentenryck, P. 2004b. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6): 977–987.

Bischoff, J.; and Maciejewski, M. 2020. Proactive empty vehicle rebalancing for Demand Responsive Transport services. *Procedia Computer Science*, 170: 739–744.

Brar, A. S.; Su, R.; Li, Y.; and Zardini, G. 2025. Vehicle Rebalancing Under Adherence Uncertainty. arXiv:2412.16632.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Ferrucci, F.; and Bock, S. 2016. Pro-active real-time routing in applications with multiple request patterns. *European Journal of Operational Research*, 253(2): 356–371.

Fisher, M. L.; and Jaikumar, R. 1981. A generalized assignment heuristic for vehicle routing. *Networks*, 11(2): 109–124.

Guo, X.; Xu, H.; Zhuang, D.; Zheng, Y.; and Zhao, J. 2023. Fairness-Enhancing Vehicle Rebalancing in the Ride-hailing System. arXiv:2401.00093.

Haasler, I.; Ringh, A.; Chen, Y.; and Karlsson, J. 2024. Scalable computation of dynamic flow problems via multimarginal graph-structured optimal transport. *Mathematics of Operations Research*, 49(2): 986–1011.

Jain, A. K. 2010. Data clustering: 50 years beyond K-means. *Pattern recognition letters*, 31(8): 651–666.

Joe, W.; and Lau, H. C. 2020. Deep reinforcement learning approach to solve dynamic vehicle routing problem with stochastic customers. In *Proceedings of the international conference on automated planning and scheduling*, volume 30, 394–402.

Klapp, M. A.; Erera, A. L.; and Toriello, A. 2018. The dynamic dispatch waves problem for same-day delivery. *European Journal of Operational Research*, 271(2): 519–534.

Kool, W.; Van Hoof, H.; and Welling, M. 2018. Attention, learn to solve routing problems! *arXiv preprint arXiv:1803.08475*.

Kumari, V.; Parsawar, V.; Srivastava, K.; Ochawar, A.; Ochawar, R. S.; Dayani, V.; and Singh, K. 2024. Dynamic Clustering for Multi-Depot Capacitated Vehicle Routing with Time Windows: A CW Heuristic Approach. In *2024 OITS International Conference on Information Technology (OCIT)*, 304–309. IEEE.

Lei, D.; and Wu, Y. 2022. Order-Dispatching Strategy Induced by Optimal Transport Plan for an Online Ride-Hailing System. *Transportation Research Record*, 2676(6): 156–169.

Merton, R. C. 1969. Lifetime portfolio selection under uncertainty: The continuous-time case. *The review of Economics and Statistics*, 247–257.

Meyn, S. P.; and Tweedie, R. L. 2012. *Markov chains and stochastic stability*. Springer Science & Business Media.

Moradi, S. 2025. A Survey on Algorithmic Developments in Optimal Transport Problem with Applications. arXiv:2501.06247.

Nazari, M.; Oroojlooy, A.; Snyder, L.; and Takác, M. 2018. Reinforcement learning for solving the vehicle routing problem. *Advances in neural information processing systems*, 31.

Peng, B.; Wang, J.; and Zhang, Z. 2019. A deep reinforcement learning algorithm using dynamic attention model for vehicle routing problems. In *International Symposium on Intelligence Computation and Applications*, 636–650. Springer.

Pillac, V.; Gendreau, M.; Guéret, C.; and Medaglia, A. L. 2013. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1): 1–11.

Psaraftis, H. N.; Wen, M.; and Kontovas, C. A. 2016. Dynamic vehicle routing problems: Three decades and counting. *Networks*, 67(1): 3–31.

Ritzinger, U.; Puchinger, J.; and Hartl, R. F. 2016. A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 54(1): 215–231.

Santambrogio, F. 2015. Optimal Transport for Applied Mathematicians: Calculus of Variations, PDEs, and Modeling, volume 87 of Progress in Nonlinear Differential Equations and Their Applications.

Soeffker, N.; Ulmer, M. W.; and Mattfeld, D. C. 2022. Stochastic dynamic vehicle routing in the light of prescriptive analytics: A review. *European Journal of Operational Research*, 298(3): 801–820.

Ulmer, M. W.; Goodson, J. C.; Mattfeld, D. C.; and Hennig, M. 2019. Offline–online approximate dynamic programming for dynamic vehicle routing with stochastic requests. *Transportation Science*, 53(1): 185–202.

Vinyals, O.; Fortunato, M.; and Jaitly, N. 2015. Pointer networks. In Advances in Neural Information Processing Systems 28, p.

Voccia, S. A.; Campbell, A. M.; and Thomas, B. W. 2019. The same-day delivery problem for online purchases. *Transportation Science*, 53(1): 167–184.

Wallar, A.; Van Der Zee, M.; Alonso-Mora, J.; and Rus, D. 2018. Vehicle rebalancing for mobility-on-demand systems with ride-sharing. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 4539–4546. IEEE.

Wiedemann, N.; Uscidda, T.; and Raubal, M. 2025. GeOT: A spatially explicit framework for evaluating spatio-temporal predictions. arXiv:2410.11709.

Zhang, J.; Luo, K.; Florio, A. M.; and Van Woensel, T. 2023. Solving large-scale dynamic vehicle routing problems with stochastic requests. *European Journal of Operational Research*, 306(2): 596–614.