Case Studies on Transport Policy 25 (2026) 101877

Contents lists available at ScienceDirect

Case Studies on Transport Policy

journal homepage: www.elsevier.com/locate/cstp

Dispatch optimization under inventory-driven delays: A case study using Approximate Dynamic Programming

Lucas J. Verhofstad [a], Mahnam Saeednia [b], Bilge Atasoy [c,*]

[a] MSc Transport, Infrastructure & Logistics, TU Delft, Netherlands
[b] Department of Transport and Planning, Faculty of Civil Engineering and Geosciences, TU Delft, Netherlands
[c] Department of Maritime & Transport Technology, Faculty of Mechanical Engineering, TU Delft, Netherlands

ARTICLE INFO

Keywords:
Inventory-driven delays
Markov decision process
Dispatch optimization
Inventory-dependent service time
Approximate dynamic programming
Partial value function approximation

ABSTRACT

Inventory capacity constraints represent real-world space scarcity in warehouses. When only full truckloads are permitted, these constraints are typically enforced by blocking deliveries that would violate the capacity. However, when the inventory is simultaneously too high to allow a delivery while also too low to serve all demand, this method leads to unmet demand. This study therefore introduces inventory-driven delays, where vehicles are held at the destination until enough inventory capacity is available, as a method to enforce capacity constraints. To capture these delays, the dispatch problem is formulated as a discrete-time Markov Decision Process (MDP) with inventory-dependent service times. The problem is solved using Approximate Dynamic Programming (ADP), which accounts for opportunity costs associated with waiting vehicles, allowing the policy to handle the state-dependent costs introduced by inventory-driven delays. A case study is developed based on data from an Ethiopian brewery. The proposed method is compared against multiple heuristics across four different replenishment strategies. Results show that one full truckload of unmet demand is recovered per ten cumulative days of inventory-driven delays, suggesting that allowing such delays can serve as a strategic buffer in inventory-capacitated, full-truckload systems. Future work may address dynamic order generation, parameter uncertainty, and extensions toward the general Inventory Routing Problem (IRP).

1. Introduction

Many logistics systems involve a centralized vendor distributing goods to independently operated retailers. Replenishment strategies vary, ranging from Vendor Managed Inventory (VMI), where the vendor decides when to deliver, to Retail Managed Inventory (RMI), where each retailer independently places orders. Another approach is a lead-time-based strategy, where an order is triggered automatically — for example, once the inventory level reaches a defined reorder point. In the studied problem the retailers place orders and the vendor is responsible for transportation, deciding when to dispatch a vehicle to an order. The vendor aims to minimize the use of trucks to reduce costs. Additionally, there is a penalty when a retailer's inventory reaches zero due to potential lost sales. Optimizing this decision of when to deliver can be done using the dispatch problem.

Retailers in real-world supply chains typically have limited storage space, so some operations research models include inventory capacity constraints to reflect this. These constraints are typically handled by limiting the delivered quantity to the remaining capacity or by blocking deliveries that would exceed it. In some cases, only full truckloads are allowed to ensure an optimal load rate for the vehicles. In this case, limiting the delivered quantities is not permitted, leaving blocking as the logical mechanism to enforce the warehouse inventory capacity. However, when the inventory can simultaneously be too high to allow a replenishment while also too low to serve all demand, this method will cause unmet demand. To the best of the authors' knowledge, no research has addressed this issue.

This paper introduces inventory-driven delays as a third method to enforce warehouse inventory capacity constraints. When a vehicle arrives with more goods than the remaining capacity, it waits on-site until sufficient space becomes available. During this time, the inventory of the truck can be used by the retailer, thus serving as a temporary buffer at the cost of occupying the vehicle. The inventory-driven delays are included as inventory-dependent service times in the problem formulation. By formulating the problem as a Markov Decision Process (MDP), developing a solution method using Approximate Dynamic Programming (ADP), and implementing and testing it in a case study, this work demonstrates how inventory-driven delays can be used effectively to constrain the inventory capacity. Additionally, by

* Corresponding author.
E-mail address: b.atasoy@tudelft.nl (B. Atasoy).

https://doi.org/10.1016/j.cstp.2026.101877
Received 19 December 2025; Received in revised form 18 May 2026; Accepted 8 June 2026
Available online 14 June 2026

comparing the results to two heuristics and a similar algorithm which includes a blocking alternative, the benefits of using this method are shown.

In the remainder of this paper, first a literature review is conducted as shown in Section 2. Then the problem is described in Section 3 and the method to address this problem is given in Section 4. The case-study implementation and results are then given in Section 5. Lastly, the conclusion is given in Section 6 with a future outlook.

2. Literature review

This paper studies the dispatch problem, in which researchers have explored a range of directions. Many focus on dynamic settings (Lan et al., 2022b), stochastic variants (Moradi-Afrapoli and Askari-Nasab, 2020), or combinations of both (Chen et al., 2023b; Dehghan et al., 2023; Delgado, 2022; Guo and Xu, 2020; de Jong et al., 2024). Of particular relevance is the dispatch problem by de Jong et al. (2024) that incorporates inventory tracking, being closest to the direction of this work. However, a key limitation is that these models do not include inventory capacity constraints, which is the focus of this work.

The Inventory Routing Problem (IRP) is a time-dependent extension of the Vehicle Routing Problem (VRP) (Mor and Speranza, 2022; Pappalardo, 2021), characterized by the tracking of inventory levels (Pappalardo, 2021). Nearly all IRP formulations assume a VMI replenishment strategy (Archetti and Ljubić, 2022; Baller et al., 2020; Bertazzi et al., 2025; Brinkmann et al., 2020), which is however not strictly required by definition (Bertazzi and Speranza, 2012). Since the dispatch problem can be represented as a VRP on a star-shaped network, constraints from the IRP are likely applicable to dispatch problems. Inventory capacities are commonly incorporated into IRP formulations, such as the work of Archetti and Ljubić (2022), Bertazzi et al. (2025), Brinkmann et al. (2020), though typically by limiting delivery quantities or blocking deliveries rather than by delaying service time until space becomes available. No formulation within the IRP has been found that accounts for inventory-driven delays.

Within the broader VRP framework, numerous variants exist, including stochastic VRP (Iklassov et al., 2024; Marković et al., 2020; Shahparvari and Abbasi, 2017; Shahparvari et al., 2017; Honda and Nakade, 2024; Yernar and Turan, 2025; Ammouriova et al., 2023; Pappalardo, 2021), dynamic VRP (Pillac et al., 2013; Ammouriova et al., 2023; Pappalardo, 2021), time-dependent VRP (Adamo et al., 2024; Emanab, 2023; Gendreau et al., 2015; Gmira et al., 2021), and real-time VRP (Muñoz-Villamizar et al., 2024; Gmira et al., 2021). Service times in these models are sometimes modeled as stochastic (Honda and Nakade, 2024; Dávila et al., 2021; Yernar and Turan, 2025) or time-dependent (Keskin et al., 2019; Petropoulos et al., 2024), showcasing that it is possible to include variable service times, which could be used for including inventory-driven delays as inventory-dependent service times. However, this extended formulation is not yet available in the literature.

Many relevant IRP models (Brinkmann et al., 2020; McKenna et al., 2020) and dispatch optimization models (de Jong et al., 2024) formulate the problem as a Markov Decision Process (MDP). Moreover, Approximate Dynamic Programming (ADP) provides a scalable solution technique by approximating value functions or policies, while preserving the structure of the MDP (Mardešic et al., 2024). One such approach is value function approximation (VFA), which estimates the downstream value of a state to guide action selection based on long-term expected rewards (Rempel and Cai, 2021; Mardešic et al., 2024).

While some dispatch models track inventory over time, no formulation is present that includes inventory-capacity constraints. The problem can be formulated as an IRP, and many IRP formulations include inventory-capacity constraints. These constraints are typically handled by limiting the delivery quantity or by blocking deliveries that would violate inventory capacity constraints. Lastly, within the broader VRP formulation, some models use service times which depend on a variable. By making the service time depend on the inventory level, a formulation can account for inventory-driven delays, however no existing formulation of this mechanism has been found in the literature. Therefore, this study fills this gap. The placement of this study within the literature is provided in Table 1.

By implementing the PVFA with inventory-dependent service times, this study demonstrates that it is possible to incorporate inventory capacities by delaying vehicles until sufficient space becomes available, thereby accounting for inventory-driven delays. This approach avoids the need for hard inventory capacity constraints used in other models, such as formulations that block entire deliveries (Rempel and Cai, 2021) or those that limit delivery quantities (Archetti and Ljubić, 2022). Furthermore, the dispatch model is applied to four different replenishment strategies. This contrasts with most inventory-routing and dispatch models, which typically assume a VMI strategy (Bertazzi and Speranza, 2012; Bertazzi et al., 2025; Brinkmann et al., 2020; Baller et al., 2020; Archetti and Ljubić, 2022). This study demonstrates that dispatch optimization is also feasible under an RMI replenishment strategy, thereby expanding its applicability.

3. Problem description

In the studied problem, there is a single centralized vendor that serves multiple independent retailers. The retailers have capacitated inventories that deplete over time as they sell products. To replenish their stock, retailers can place an order at the vendor, resulting in a so-called RMI replenishment strategy. The vendor has the knowledge of all retailer inventory levels and receives their orders. The vendor operates a fleet of homogeneous vehicles that can be used to fulfill orders. At each decision moment, the vendor can choose which outstanding orders the available vehicles are dispatched to. When a vehicle arrives at a retailer with a load larger than the remaining inventory capacity, the vehicle is required to wait on-site until the retailer has depleted enough stock, incurring an inventory-driven delay. The vendor aims to deliver orders to prevent stockouts; however, dispatching vehicles too early can result in inventory-driven delays, thereby reducing logistics efficiency and possibly causing overall lower performance. The vendor seeks to dispatch vehicles optimally while accounting for this trade-off, which can be achieved by solving the dispatch problem.

The formulation accounts for inventory-driven delays; therefore, retailers have limited inventory capacities, and vehicles are required to wait on site until sufficient inventory space becomes available. This is captured through inventory-dependent service times. Note that, the inventory level depends on dispatch decisions, whose costs in turn depend on the inventory level. Hence, the model includes state-dependent decision variables, resulting in nonlinear behavior. To focus on dispatch behavior, the model adopts several simplifications: input parameters are deterministic and known in advance, the vendor has unlimited stock and the vehicle fleet is homogeneous. In the robustness analyses, demand is treated as an exogenous stochastic variable following a normal distribution.

4. Methodology

To address the described problem, a discrete-time MDP is formulated. A rolling-horizon approach is adopted, where future actions are evaluated to include future costs in the decision-making process; however, only the action at the current epoch is executed. The MDP state tracks, for each vehicle, which order it is serving, what stage of service it is in, and how long it will remain in that stage, allowing the system to track the status of each vehicle. Furthermore, for each retailer, the inventory level is included in the state, enabling the identification of bottlenecks and the calculation of inventory-dependent service times. Lastly, for each retailer, a list of outstanding orders is tracked to construct the vendor's backlog. The decision-maker, i.e., the vendor,

Table 1
Positioning of our work.

Source	Type	Dynamic	Stochastic	Inventory tracking	Inventory capacity	Inventory-dependent service times
Moradi-Afrapoli and Askari-Nasab (2020)	dispatch		✓			
Zou et al. (2020)	dispatch					
Wang et al. (2021)	dispatch					
Delgado (2022)	dispatch	✓	✓			
Guo and Xu (2020)	dispatch	✓	✓			
Lan et al. (2022b)	dispatch					
Lan et al. (2022a)	dispatch	✓				
Wang et al. (2022)	dispatch					
Chen et al. (2023a)	dispatch	✓	✓			
Chen et al. (2023b)	dispatch	✓	✓			
Dehghan et al. (2023)	dispatch	✓	✓			
W. et al. (2023)	dispatch					
de Jong et al. (2024)	dispatch	✓	✓	✓		
Mirzaei-Nasirabad et al. (2023)	dispatch	✓				
Bertazzi and Speranza (2012)	IRP			✓	✓	
McKenna et al. (2020)	IRP		✓	✓	✓	
Brinkmann et al. (2020)	IRP	✓	✓	✓	✓	
Archetti and Ljubić (2022)	IRP			✓	✓	
Bertazzi et al. (2025)	IRP	✓		✓	✓	
Pillac et al. (2013)	VRP	✓				
Gendreau et al. (2015)	VRP					
Shahparvari and Abbasi (2017)	VRP		✓			
Shahparvari et al. (2017)	VRP		✓			
Keskin et al. (2019)	VRP					
Marković et al. (2020)	VRP		✓			
Dávila et al. (2021)	VRP					
Gmira et al. (2021)	VRP					
Pappalardo (2021)	VRP	✓		✓		
Ammouriova et al. (2023)	VRP	✓	✓			
Emanab (2023)	VRP					
Adamo et al. (2024)	VRP					
Honda and Nakade (2024)	VRP		✓			
Iklassov et al. (2024)	VRP		✓			
Petropoulos et al. (2024)	VRP					
Yernar and Turan (2025)	VRP		✓			
This study	dispatch			✓	✓	✓

must decide which outstanding orders to dispatch at each decision epoch. Another option is to keep available vehicles idle. Based on the current state and the chosen action, a new state is generated using a state transition function. A reward function is defined to evaluate how well a certain state performs, based on the previously mentioned indicators of performance. These rewards are discounted by a discount factor to prioritize solutions with immediate positive effects, and are summed into an objective function that is maximized.

4.1. Mathematical formulation

The system consists of a set of retailers $\mathcal{N}$, indexed by i, and a set of vehicles $\mathcal{K}$, indexed by k. The system evolves over a finite set of time steps $\mathcal{T} = \{0, 1, \ldots, T\}$, indexed by t with $t = 0$ indicating the current decision epoch and T indicating the time horizon.

The problem is formulated as a discrete-time MDP, defined by the tuple $(\mathcal{S}, \mathcal{A}, f, r, \gamma)$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ the action space, f the state transition function, r the reward function, and γ the discount factor.

The state space $\mathcal{S}$ captures all possible configurations of the system and includes variables defined for each retailer and each vehicle. At each decision epoch t, the state S_t consists of:

- $S_t^{\text{tar},k}$: retailer currently assigned to vehicle k (zero if idle at the brewery),
- $S_t^{\text{stg},k}$: operational stage of vehicle k, with values indicating idle at vendor (0), en route (1), at retailer (2), or returning (3),
- $S_t^{\text{dur},k}$: remaining time steps to complete the current stage of vehicle k,
- $S_t^{\text{inv},i}$: inventory level at retailer i,

$$S_{t+1}^{\text{tar},k} = \begin{cases} x_{t+1}^{k} & \text{if } S_t^{\text{stg},k} = 0 \\ S_t^{\text{tar},k} & \text{otherwise} \end{cases} \qquad \forall k \in \mathcal{K} \tag{1}$$

$$S_{t+1}^{\text{stg},k} = \begin{cases} 0 & \text{if } S_{t+1}^{\text{tar},k} = 0 \\ 0 & \text{if } S_t^{\text{dur},k} \le 1,\ S_t^{\text{stg},k} = 3 \\ S_t^{\text{stg},k} + 1 & \text{if } S_t^{\text{dur},k} \le 1,\ S_t^{\text{stg},k} \ne 3 \\ S_t^{\text{stg},k} & \text{otherwise} \end{cases} \qquad \forall k \in \mathcal{K} \tag{2}$$

$$S_{t+1}^{\text{dur},k} = \begin{cases} \tau^{2,i} + \dfrac{\max(S_{t+1}^{\text{inv},i} + y_{t+1}^{k} - I^{\text{cap},i},\ 0)}{\delta^i} & \text{if } q = 1,\ q' = 2 \\ \max(S_t^{\text{dur},k} - 1,\ 0) & \text{if } q = q' \\ \tau^{q',i} & \text{otherwise} \end{cases} \qquad \begin{array}{l} \forall k \in \mathcal{K},\ i = S_{t+1}^{\text{tar},k}, \\ q = S_t^{\text{stg},k},\ q' = S_{t+1}^{\text{stg},k} \end{array} \tag{3}$$

$$S_{t+1}^{\text{inv},i} = S_t^{\text{inv},i} + \left|\mathcal{K}_t^{\text{arr},i}\right| - \delta^i \qquad \forall i \in \mathcal{N} \tag{4}$$

$$S_{t+1}^{\text{ord},i} = \left(S_t^{\text{ord},i} \setminus S_{t+1}^{\text{ord},i,-}\right) \cup S_{t+1}^{\text{ord},i,+} \qquad \forall i \in \mathcal{N} \tag{5}$$

Box I.

- $S_t^{\text{ord},i}$: outstanding orders at retailer i, represented as a list of order timestamps.

While derived independently, similarities with the work of Lam et al. (2017) were found, as that source also uses a vehicle state in which 0 = idle, 1 = en route, 2 = at location, and 3 = returning. In both cases, vehicles are tracked relative to their destination. The work of Lam et al. (2017) differs from the present work in other respects, most notably because it includes no mechanisms to track inventory levels.

At each time step, a feasible action space $\mathcal{A}(S_t) \subseteq \mathcal{A}$ is defined from which an action vector $A_t \in \mathcal{A}(S_t)$ is selected. Each element in A_t denotes a retailer to which a vehicle is dispatched. Each value of zero implies that a vehicle remains idle.

The state transition function f determines the next state S_{t+1} based on the current state S_t and the chosen action A_t: $S_{t+1} = f(S_t, A_t)$. It is composed of five subfunctions, given in Eqs. (1)–(5), each responsible for updating a separate state variable, which are detailed in the following paragraphs. See Eqs. (1)–(5) in Box I.

The target retailer of vehicle k is updated as shown in Eq. (1) by assigning each idle vehicle ($S_t^{\text{stg},k} = 0$) to a retailer from the dispatch vector A_t. Here, x_{t+1}^k denotes the retailer to which vehicle k is dispatched, and is selected such that the vector x_{t+1} contains the same elements as A_t, thereby ensuring that each dispatched vehicle receives exactly one assignment.

The stage of vehicle k is updated as shown in Eq. (2), based on its current stage and remaining duration. The stage advances if the remaining duration falls below one, while it resets to zero when the vehicle becomes idle at the brewery.

The time component of vehicle k is updated as shown in Eq. (3) by decreasing the remaining duration by one if the vehicle stays in the same stage, or by assigning a new duration $\tau^{q,i}$ for a vehicle serving retailer i in stage q otherwise. A special case occurs when a vehicle begins service ($q = 2$), in which case the duration is extended to incorporate potential waiting time due to limited inventory space at the retailer. This delay depends on the inventory capacity $I^{\text{cap},i}$ and the average depletion rate δ^i, which represent the maximum inventory level and the average daily inventory reduction at retailer i, respectively. We define y_{t+1}^k as the unloading rank of vehicle k, starting at 1 for a single delivery. This rank accounts for the cumulative inventory of both the current vehicle and its predecessors arriving in the same epoch to correctly calculate the required service time extensions.

To the best of our knowledge no existing VRP, IRP, or dispatch formulation uses a relationship between inventory levels and delays to handle inventory capacities. In this work a new mechanism is introduced in which trucks incur additional waiting time when insufficient inventory space is available at the retailer, ensuring that inventory capacity constraints are not violated.

The inventory level of retailer i is updated as shown in Eq. (4) by accounting for both incoming deliveries and ongoing depletion. Inventory is reduced by the average depletion rate δ^i and increased by one unit for each arriving vehicle, identified by the set

$$\mathcal{K}_t^{\text{arr},i} = \left\{ k \in \mathcal{K} \;\middle|\; S_t^{\text{stg},k} = 1,\ S_{t+1}^{\text{stg},k} = 2,\ S_{t+1}^{\text{tar},k} = i \right\}$$

The order backlog of retailer i is updated as shown in Eq. (5). While technically a vector, set notation is used to describe its behavior. Each entry in $S_t^{\text{ord},i}$ represents a time at which an order is placed for which no vehicle has yet been dispatched, allowing the model to track the age of an order. At each epoch, new orders are added to the backlog, while the oldest entries are removed corresponding to orders for which a vehicle has been dispatched. Let $S_{t+1}^{\text{ord},i,+}$ denote the newly placed orders at time $t+1$, with d_t^i elements equal to $t+1$, and let $S_{t+1}^{\text{ord},i,-}$ denote the earliest outstanding orders, one for each vehicle dispatched to retailer i.

The reward function r, defined in Eq. (6), assigns a numerical value to the new state S_{t+1}, capturing its quality. The reward function consists of two components: the losses due to stockouts is given by R_{t+1}^{los} in Eq. (7) and penalties for idle vehicles by R_{t+1}^{idl} in Eq. (8). Each component is weighted by a corresponding penalty parameter L^{los} for unmet demand and L^{idl} for idle vehicles.

$$r(S_{t+1}) = R_{t+1}^{\text{los}} + R_{t+1}^{\text{idl}} \tag{6}$$

$$R_{t+1}^{\text{los}} = \sum_{\substack{i \in \mathcal{N} \\ S_{t+1}^{\text{inv},i} < 0}} \min\left(-S_{t+1}^{\text{inv},i},\ \delta^i\right) \cdot L^{\text{los}} \tag{7}$$

$$R_{t+1}^{\text{idl}} = \sum_{\substack{k \in \mathcal{K} \\ S_{t+1}^{\text{tar},k} = 0}} L^{\text{idl}} \tag{8}$$

The discount factor $\gamma \in [0, 1]$ captures how future rewards are valued relative to immediate ones.

4.2. Solution method

A MDP can be solved using the Bellman Equation shown in Eq. (9), which characterizes the optimal objective value $V_t(S_t)$ of being in state

$$\Delta\hat{V}_t^{o} = \Delta\hat{V}_t^{\text{los},o} + \Delta\hat{V}_t^{\text{idl},o} + \Delta\hat{V}_t^{\text{opp},o} \qquad \forall o \in \mathcal{O}_t \tag{11}$$

$$\eta_t^{o} = |\mathcal{K}_t^{\text{enr},i}| + (n-1) \qquad \forall o \in \mathcal{O}_t \tag{12}$$

$$\hat{S}_{t,\text{arr}}^{\text{inv},o} = S_t^{\text{inv},i} + \eta_t^{o} - \delta^{i} \cdot \tau^{1,i} \qquad \forall o \in \mathcal{O}_t \tag{13}$$

$$w_t^{o} = \max\left(\frac{\hat{S}_{t,\text{arr}}^{\text{inv},o} I^{\text{cap},i} + 1}{\delta^{i}}, 0\right) \qquad \forall o \in \mathcal{O}_t \tag{14}$$

$$\Delta\hat{V}_t^{\text{los},o} = \gamma^{\tau^{1,i}} \sum_{\substack{m\in\{1,\dots,H\}\\ \hat{S}_{t,\text{ arr}}^{\text{inv},o} < m\cdot\delta^{i}}} \frac{\min\left(-\hat{S}_{t,\text{arr}}^{\text{inv},o} + m \cdot \delta^{i},\, \delta^{i}\right)}{m^2} L^{\text{los}} \qquad \forall o \in \mathcal{O}_t \tag{15}$$

$$\Delta\hat{V}_t^{\text{idl},o} = \begin{cases} L^{\text{idl}} & \text{if } w_t^{o} > 0 \\ L^{\text{idl}} - L^{\text{idl}} \cdot \gamma^{(\tau^{1,i}+\tau^{2,i}+\tau^{3,i})} & \text{otherwise} \end{cases} \qquad \forall o \in \mathcal{O}_t \tag{16}$$

$$\Delta\hat{V}_t^{\text{opp},o} = \begin{cases} \lambda \cdot \gamma^{(\tau^{1,i}+\tau^{2,i}+\tau^{3,i})} & \text{if } w_t^{o} > 0 \\ 0 & \text{otherwise} \end{cases} \qquad \forall o \in \mathcal{O}_t \tag{17}$$

Box II.

S_t at time t.

$$V_t(S_t) = \max_{A_t\in\mathcal{A}(S_t)} \left\{ r(S_{t+1}) + \gamma \cdot V_{t+1}(S_{t+1}) \right\} \tag{9}$$

Let $S_t^{\text{ord}} = \left[S_t^{\text{ord},1}, S_t^{\text{ord},2}, \dots, S_t^{\text{ord},|\mathcal{N}|}\right]$ denote the vector indicating for each outstanding order at time t the index of its retailer. Let $\mathcal{O}_t = \{1, 2, \dots, |S_t^{\text{ord}}|\}$ denote the index set of all outstanding orders indexed by o, and let $\mathcal{K}_t^{\text{ven}} = \{k \in \mathcal{K} \mid S_t^{\text{stg},k} = 0\}$ denote the set of vehicles at the vendor which can be dispatched. A feasible dispatch decision, including the possibility for each vehicle to remain idle, consists of distributing $|\mathcal{K}_t^{\text{ven}}|$ indistinguishable vehicles over $|\mathcal{O}_t| + |\mathcal{K}_t^{\text{ven}}|$ distinguishable options. Using the classical stars and bars formulation, the number of possible actions per epoch can be determined as follows:

$$|\mathcal{A}_t(S_t)| = \binom{|\mathcal{K}_t^{\text{ven}}| + \left(|\mathcal{O}_t| + |\mathcal{K}_t^{\text{ven}}|\right) - 1}{\left(|\mathcal{O}_t| + |\mathcal{K}_t^{\text{ven}}|\right) - 1}$$

Additionally, since each action directly influences subsequent states, all possible future decision sequences must be evaluated. This leads to exponential growth over the length of the planning horizon and combinatorial growth with the sizes of the order and vehicle sets. This computational explosion showcases the necessity of employing approximation methods.

ADP is used to reduce the computational effort. A policy $\pi(S_t)$ is used to select an action $A_t = \pi(S_t)$, and using this policy, the Bellman equation (Eq. (9)) can be rewritten to the policy Bellman equation given as Eq. (10). The value function under a fixed policy π is, by definition, less than or equal to the optimal value function ($V_t^{\pi}(S_t) \leq V_t(S_t)$).

$$V_t^{\pi}(S_t) = r(S_{t+1}) + \gamma \cdot V_{t+1}^{\pi}(S_{t+1}) \tag{10}$$

The policy π has to be formulated. This study applies a Partial Value Function Approximation (PVFA), based on the value function approximation in the literature by Rempel and Cai (2021) and Mardešic et al. (2024). In a value function approximation, the impact of an action on the value function is estimated. In this study, the impact of delaying a single order by one epoch is estimated individually using a value of delay. This order-specific evaluation motivates the term PVFA and distinguishes our study from other approaches in the literature. The value of delay is denoted by $\Delta\hat{V}_t^{o}$ for order o. This value has two purposes:

1. If $\Delta\hat{V}_t^{o} \geq 0$, no vehicle should be dispatched to order o, even if there are idle vehicles. A value above zero indicates that it is more efficient to dispatch later, most likely due to inventory-driven delays.
2. To optimize the value function, it is best to dispatch to orders with the lowest value of delay ($\Delta\hat{V}_t^{o}$). Therefore, this value can be used to rank orders.

In practice, all orders with $\Delta\hat{V}_t^{o} < 0$ can be ranked based on this value, and vehicles are dispatched in that order. For each order o, the value of delay is calculated as shown in Eq. (11) (see Box II). The first two terms estimate the direct impact of different penalties and rewards on delaying an order. While these terms account for the impact of serving an order on the objective value, they do not capture the cost of using a vehicle. This cost, referred to as the opportunity cost and denoted by $\Delta\hat{V}_t^{\text{opp},o}$, reflects the benefit lost when a vehicle is occupied by a particular order and thus cannot be used for other orders. Inventory-driven delays increase this cost, as the vehicle remains occupied for a longer period. To our knowledge, no existing work explicitly quantifies this order-specific opportunity cost. Throughout the remainder of this section, the following notation is adopted to simplify expressions and improve readability:

- Let $o \in \mathcal{O}_t$ represent an order at time t.
- Let $i \in \mathcal{N}$ denote the retailer associated with order o.
- Let $n \in \mathbb{N}$ denote the position of order o in retailer i's backlog vector $S_t^{\text{ord},i}$.

Let $\mathcal{K}_t^{\text{enr},i} = \{k \in \mathcal{K} \mid S_t^{\text{tar},k} = i,\ S_t^{\text{stg},k} = 1\}$ denote the set of vehicles currently en route to retailer i. The number of deliveries expected to arrive prior to order o, denoted by η_t^{o}, is defined in Eq. (12). This value is utilized in Eq. (13) to estimate the expected inventory level upon arrival $\hat{S}_{t,\text{arr}}^{\text{inv},o}$ and in Eq. (14) to determine the potential service time extension, w_t^{o}. The impact of stockouts on the value of delivery, $\Delta\hat{V}_t^{\text{los},o}$, is formulated in Eq. (15). Because vehicle unavailability in subsequent periods may cause a single-epoch dispatch delay to propagate into a multi-epoch delay, the formulation accounts for stockouts over $m \in \{1, \dots, H\}$ epochs. These are discounted by m^2 to reflect the increasing opportunity for stockout mitigation in later periods. Furthermore, an idle bonus $\Delta\hat{V}_t^{\text{idl},o}$ is incorporated under the assumption that the vehicle would otherwise remain idle. If no service time extension occurs ($w_t^{o} = 0$), this bonus is forfeited once the vehicle returns to the brewery.

The true opportunity cost of vehicle utilization is state-dependent, as it depends on the outstanding orders, fleet availability, and inventory levels. As this future state is unknown, it cannot be calculated in advance. To address this, a constant λ is introduced, representing the value of vehicle availability in a future epoch. $\Delta\hat{V}_t^{\text{opp},o}$ denotes the impact of the opportunity cost on the value of delay as shown in Eq. (17). If an order incurs an inventory-driven delay ($w_t^{o} > 0$), the opportunity cost λ is added to the value of delay.

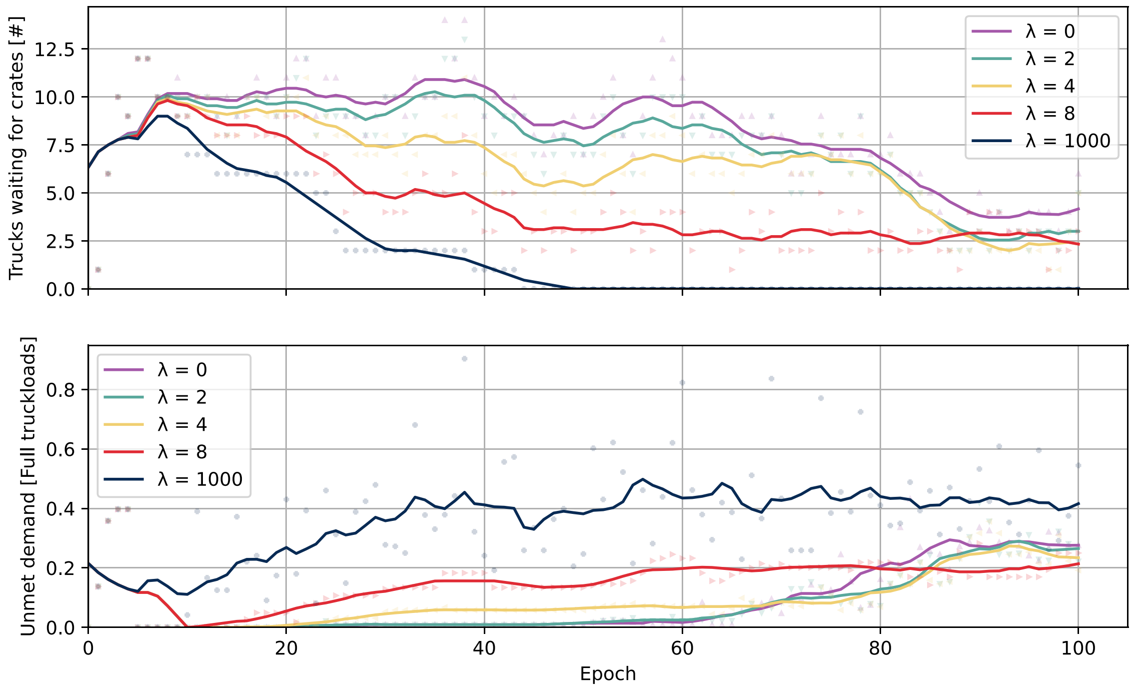

Fig. 1. Sensitivity analyses of λ.

5. Experimental results

In this section, we present the case study that is developed for evaluating the performance of the proposed approach and present comparative results looking at different KPIs.

5.1. Case study overview

For this study, we developed a case study at Habesha Breweries S.C., a large Ethiopian brewery that operates a homogeneous fleet of 76 vehicles to deliver its beer nationwide to resellers. Resellers can order a full truckload at Habesha, after which it is expected that a vehicle will be dispatched. The beer is transported in returnables, and both Habesha and the resellers own a set of these returnables. To ensure that everyone maintains the same number, each delivered full returnable is matched with an empty one, creating a closed-loop, full-truckload delivery system. Trucks arriving ahead of empty crate availability must wait on-site to complete this mandatory one-for-one exchange. Habesha tracks its vehicles and stores the itineraries. Furthermore, depletion is recorded at the resellers, and inventories are counted weekly, allowing Habesha to reconstruct the inventory levels at the resellers over time.

Due to a shortage of returnables, resellers maintain a limited stock. Additionally, high fuel prices mean Habesha does not allow less-than-truckload deliveries. Because this combination of a limited crate supply and mandatory full truckloads can lead to unmet demand, the available data motivates the study of inventory-driven delays as a potential solution. The problem formulation proposed in this paper is applied to this case study where Habesha is the vendor and the resellers are the retailers. The number of returnables at the resellers represents the inventory capacity of the retailers.

5.2. Implementation

The proposed model is implemented in Python. The implementation is verified using structured test cases. The penalty parameters L_{idl} and L_{los} were initialized to reflect the local economic landscape of Ethiopia and the priorities of Habesha. Given the relative high margin on the product and the low wages for truck drivers, reducing unmet demand is prioritized over reducing the truck usage. Therefore, a value of being idle is set to $L_{\text{idl}} = 1$ while the value of unmet demand is set to $L_{\text{los}} = -100$. An epoch length of half a day is used; a longer epoch length introduces biases due to the rounding of activities, while a shorter epoch length introduces other biases, as trips typically take multiple days, and a natural rounding to half-days occurs in the data. The rest of the model parameters are set through sensitivity analyses. Due to the nonlinearity between parameters each parameter was sequentially tuned and re-evaluated each time another parameter was changed.

The model's sensitivity to the discount factor was evaluated for $\gamma \in \{0.7, 0.8, 0.9, 0.95, 1.0\}$. The analysis revealed that system performance is robust for values of $\gamma \geq 0.9$. Lower values of γ induced myopic behavior; in these instances the objective function over-prioritized immediate cost savings at the expense of long-term demand fulfillment, leading to a decrease in overall efficiency. While $\gamma = 0.9$ was found to be the lower bound of stability, $\gamma = 0.95$ was selected for the final model to provide a stability buffer.

The look-ahead horizon H determines how early a potential future stockout is integrated into the expected reward calculation. Initially, the model considered only the immediate epoch ($H = 1$); however, this approach proved to be highly sensitive to demand uncertainty in the robustness analyses. To address this, a value of $H = 3$ is selected, allowing it to anticipate and mitigate stock-out risks multiple steps in advance.

The parameter λ serves as a soft constraint on inventory-driven delays. The tuning process as shown in Fig. 1 demonstrates a clear trade-off: High λ acts as a blocking constraint, excessively restricting flow, while a low λ permits unrealistic inventory-driven delays, causing performance to drop in the final stages where shipment prioritization is needed. A value of $\lambda = 2$ was chosen due to better performance.

To provide empirical insights into the contribution of different components to the performance of our approach, a study is performed where parts of the PVFA are disabled systematically. $L^{\text{los}} = 0$ means $\Delta\hat{V}_t^{\text{los},o} \equiv 0$, $L^{\text{idl}} = 0$ means $\Delta\hat{V}_t^{\text{idl},o} \equiv 0$ and lastly $\lambda = 0$ means $\Delta\hat{V}_t^{\text{opp},o} \equiv 0$. The results shown in Fig. 2 reveal that $\Delta\hat{V}_t^{\text{los},o}$ has a large impact, which is logical given that this is what motivates the model to prevent stockouts. Furthermore, it can be seen that both $\Delta\hat{V}_t^{\text{idl},o}$ and $\Delta\hat{V}_t^{\text{opp},o}$ mainly reduce service time extensions.

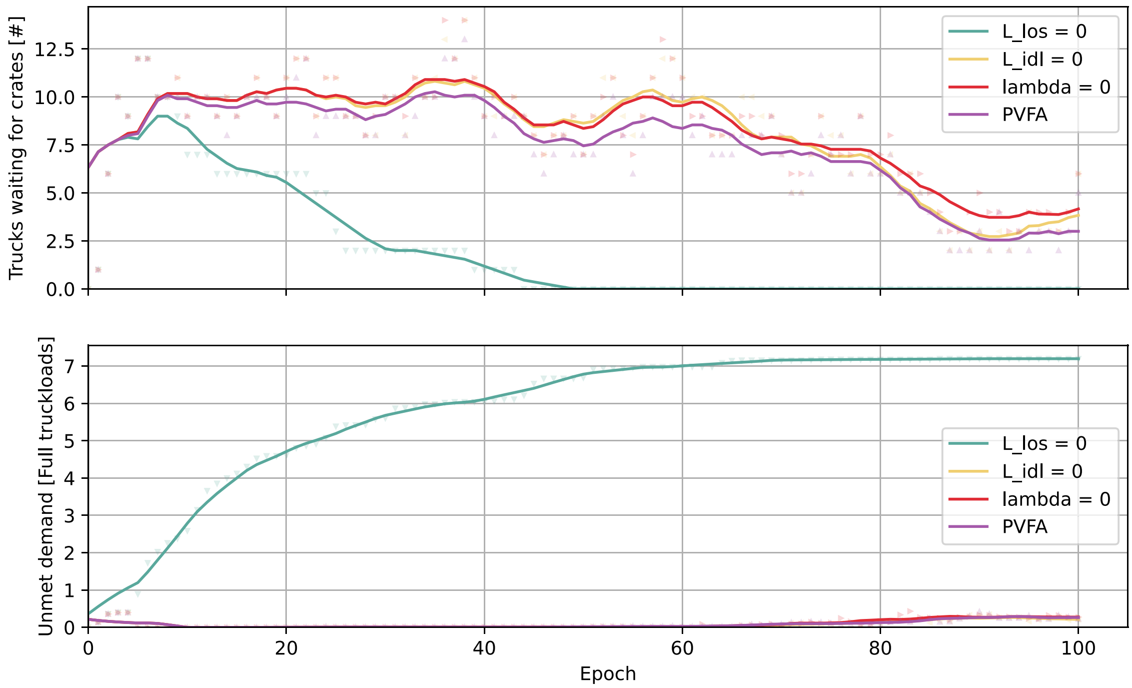

Fig. 2. Contribution of different model components.

5.3. Comparison of different replenishment strategies

To ensure a fair comparison beyond FIFO, we include several non-trivial heuristics that differ in how they trade off early dispatch against the risk of inventory-driven delays. Importantly, all heuristics allow for delayed dispatch decisions, making them structurally compatible with the proposed model.

Efficiency-first strategies prioritize resource utilization by preventing inventory-driven delays, which occupy vehicles longer than strictly necessary. This orientation is represented by the Earliest Deadline (ED) with blocking policy. Under this strategy, a dispatch is only permitted if it will not result in an inventory-driven delay. Among valid dispatches, the policy prioritizes orders based on the number of epochs remaining until a stockout becomes unavoidable. Additionally PVFA blocking is tested, a version of the proposed method restricted by the same blocking constraint to isolate the specific impact of the delay-capable formulation.

In contrast, demand-first strategies prioritize the prevention of unmet demand at all costs, treating vehicle efficiency as a secondary concern. This is implemented via a Just-in-Time delay-capable heuristic, which defers each dispatch unless it is essential to prevent a stockout. Consequently, dispatches and thus inventory-driven delays are only permitted when immediately necessary to prevent stockouts. When multiple orders require a dispatch, the heuristic prioritizes the dispatch with the minimal predicted waiting time. The policies are evaluated under four replenishment strategies:

1. **RMI** with stochastic order arrivals, modeled using a Poisson distribution.
2. **Fixed lead-time (five-day)** ordering, where retailers place orders sufficiently in advance to maintain a five-day stock buffer, effectively using a reorder point to trigger orders.
3. **Fixed lead-time (three-day)** ordering, using the same reorder point mechanism as the five-day strategy, but maintaining only a three-day stock buffer.
4. **VMI**, where the vendor determines dispatch timing based on observed stock levels.

To compare the performance of the policies across different replenishment strategies, a set of key performance indicators (KPIs) is evaluated. Due to the stochastic nature of order generation in the RMI strategy, each policy is simulated nine times. To visualize the variability in results, the standard deviation (SD) is used as a measure of spread. Bar heights represent the mean values across runs, and error bars indicate ± 1 SD.

The objective value represents the weighted performance of the model, reflecting the total rewards discounted by the epoch in which they are obtained. As shown in Fig. 3, the PVFA achieves the highest objective value across all scenarios. This consolidated metric confirms that the PVFA successfully balances the trade-offs between reducing inventory-driven delays and recovering unmet demand.

As shown in Fig. 4, the PVFA blocking and ED blocking yield the theoretical minimum for service time extensions due to their blocking mechanisms. In contrast, the Just-in-time policy exhibits the highest cumulative delays, which is expected given its priority of preventing unmet demand even at the cost of significant inventory-driven delays. The PVFA yields slightly lower service time extensions than the Just-in-time heuristic. Detailed analyses of the scenarios revealed that the Just-in-time heuristic incurs its service time delays primarily in early epochs, whereas the PVFA shifts these delays toward later epochs. An example of this behavior in the VMI scenario is illustrated in Fig. 5.

Fig. 6 presents the total volume of unmet demand across all strategies. As expected, a baseline of unmet demand is unavoidable due to initial stock levels and, in the RMI scenario, random order generation that may occur too late to prevent a stockout. The fact that PVFA blocking incurs significantly more unmet demand than the standard PVFA demonstrates that allowing inventory-driven delays is essential for recovering unmet demand. Furthermore, a detailed epoch-by-epoch analysis (such as shown in the VMI example in Fig. 5) reveals that the Just-in-time heuristic prioritizes zero unmet demand in early epochs at the expense of high vehicle waiting times. This reactive prioritization leads to a loss of fleet efficiency that eventually causes a spike in unmet demand during later stages as the system does not have enough vehicles to dispatch to new orders. In contrast, the PVFA proves more effective by proactively allowing marginal unmet demand early on. This preserves vehicle availability and prevents the fleet from becoming

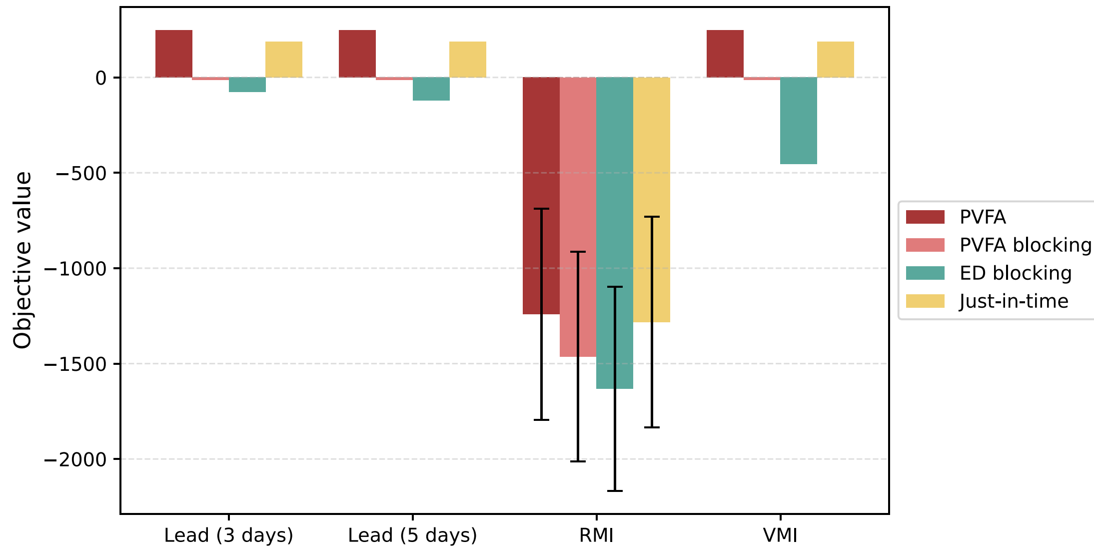

Fig. 3. Comparative performance — objective value per strategy.

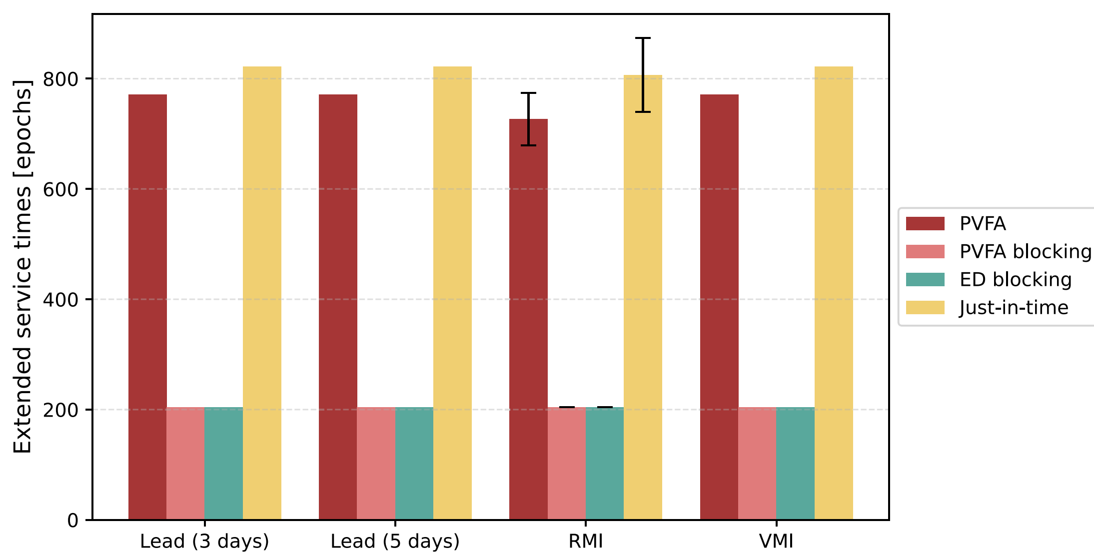

Fig. 4. Total extension of service times per strategy.

trapped in an inefficient state, ultimately resulting in lower cumulative unmet demand than the Just-in-time policy.

In summary, the PVFA reduces unmet demand by 24.67 to 28.24 full truckloads compared to the PVFA blocking policy, depending on the replenishment strategy. This improvement comes at the cost of 523 to 567 epochs of additional service time extensions. Given that one epoch represents half a day, this indicates that removing the blocking constraint allows the system to recover one full truckload of unmet demand for every 10.0 to 10.6 days of inventory-driven delay.

5.4. Robustness to uncertainty

While the primary model assumes deterministic parameters, real-world logistics systems are subject to stochasticity. The PVFA is inherently robust to dynamic order generation because it does not require access to future order logs to make dispatch decisions. Furthermore, travel time uncertainty could impact the results. However, as all travel times are rounded up to half-day increments, the discretization naturally provides a buffer against minor fluctuations. Additionally, travel times typically follow right-skewed and fat-tailed distributions; significant delays are often event-based and fall outside the scope of steady-state robustness.

Additionally, demand uncertainty is another important form of uncertainty, which is treated as exogenous within this study. This uncertainty leads to deviations between expected inventory levels under mean demand and realized inventory levels. As the inventory level also depends on dispatch decisions, this induces endogenous effects in the resulting outcomes. Both the predicted unmet demand as well as the predicted extended service times depend on the predicted inventory level. A robustness analysis is conducted where the PVFA is provided with the mean demand as input, while the actual transitions in the MDP use demand values drawn from a normal distribution. This variability is defined by the Coefficient of Variation. Notably, demand was not truncated at zero. While this allows for theoretical negative demand values, it ensures that the mean remains constant across all tests, allowing for a pure evaluation of the system's sensitivity to variance. Table 2 presents the results of this analysis. The model demonstrates high resilience to

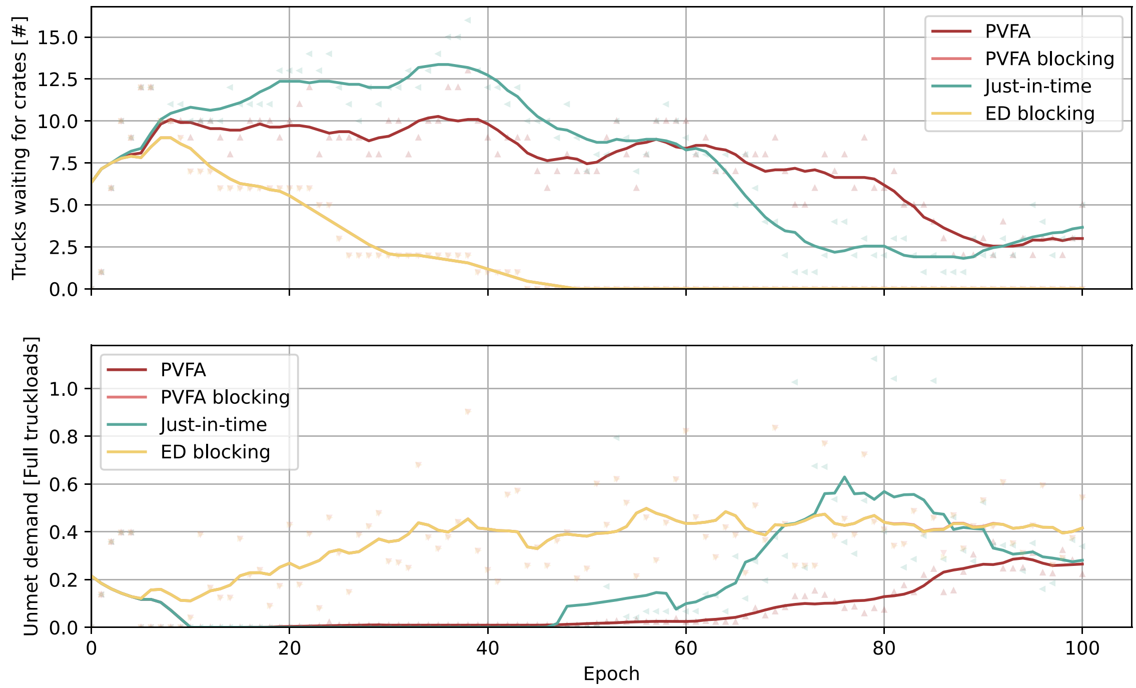

Fig. 5. Number of trucks waiting for crates and unmet demand per epoch for the VMI replenishment strategy.

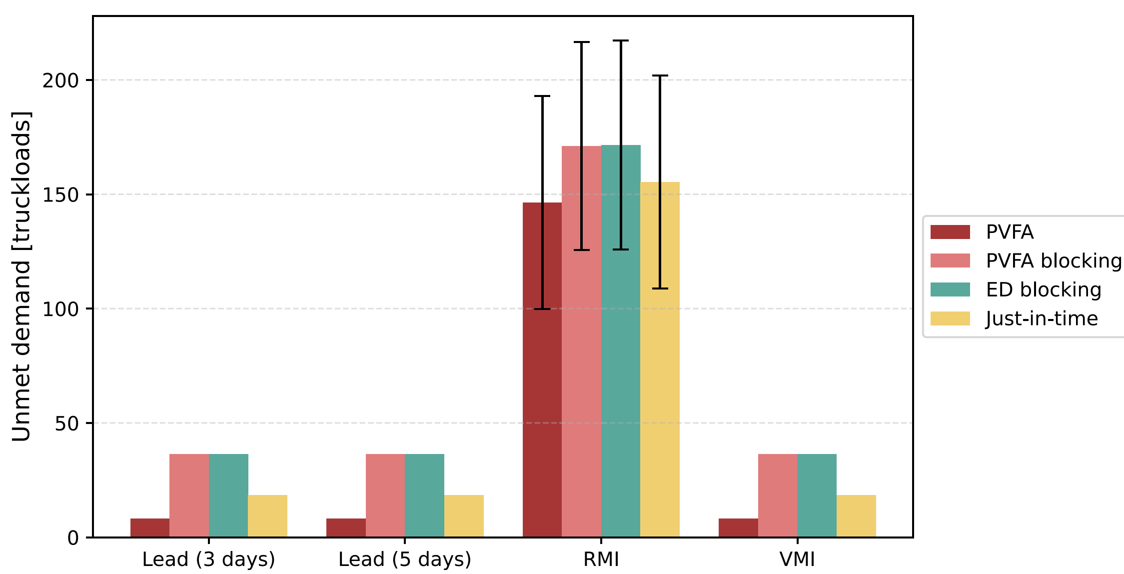

Fig. 6. Total amount of unmet demand per strategy.

Table 2
Robustness to demand variability of the 5 days lead time strategy, based on 20 runs.

Coefficient of variation	Objective value	Unmet demand	Service time extensions
0%	249	8	771
25%	243 ± 16	9 ± 1	796 ± 18
50%	200 ± 35	19 ± 5	811 ± 34
100%	−426 ± 408	93 ± 32	856 ± 64

moderate demand uncertainty, with only minor performance decreases observed for a CV up to 25%. At a 50% variation, the model begins to show increased unmet demand and service extensions. However, a significant performance drop occurs when the CV reaches 100%; at this level of volatility, the unmet demand increases more than tenfold compared to the deterministic baseline. This suggests that while the PVFA is robust against typical forecasting errors and fluctuations in demand, extreme demand volatility will significantly offset the outcome.

6. Conclusion

By formulating the dispatch problem as an MDP with inventory-dependent service times, this paper provides a framework to allow inventory-driven delays. We demonstrate how an opportunity cost within a PVFA approach serves as a steering mechanism, allowing the system to selectively accept service extensions rather than relying on hard blocking constraints.

The evidence for this method's contribution is found through a comparative performance analysis between the proposed PVFA and the heuristics. As both the PVFA and Just-in-time policies yield lower unmet demand than the Earliest deadline and PVFA blocking, it is shown that removing blocking constraints can improve demand fulfillment. Furthermore, by outperforming the Just-in-time strategy, the PVFA demonstrates an ability to prioritize certain unmet demand, strategically allowing marginal unmet demand in early epochs to preserve fleet efficiency for later stages. A clear trade-off between the PVFA and PVFA blocking is observed: the system recaptures one full truckload of unmet demand for every ten cumulative days of inventory-driven delays. This finding suggests that inventory-capacitated, full-truckload vendor-to-retailer systems, where inventories can be simultaneously too high to allow a replenishment while also too low to serve all demand, can benefit from treating inventory-driven delays as a strategic buffer rather than using a binary blocking constraint. By using inventory-driven delays instead of the hard binary rejection used by blocking constraints, a soft method to constrain dispatches to retailers at capacity is developed. Consequently, this method serves as a decision-support framework for systems where the value of unmet demand must be weighed against the cost of vehicle idleness.

Several limitations remain that offer opportunities for future research. While the proposed method successfully recaptures unmet demand within the context of an Ethiopian brewery, the results are currently tied to a case-specific environment. Future work should evaluate the generalizability of inventory-driven delays in other logistics settings, such as those with heterogeneous vehicles where waiting-time trade-offs may vary. Furthermore, this study assumes deterministic depletion rates, delays, and travel times. In reality, the unpredictability of demand, travel times, and delays introduces significant variance that static models cannot fully capture. Future research could explore dynamic or stochastic variants to include this unpredictability. A stochastic variant is a particularly interesting direction for future research because it addresses the problem of binary rejection. In an uncertain state, a blocking constraint is difficult to implement because it is unclear if a delivery is allowed. Inventory-driven delays replace this with a soft method where a vehicle can be dispatched at the risk of a delay rather than the risk of a total failure. Finally, incorporating a routing mechanism would expand the model into a more general Inventory Routing Problem (IRP) formulation to capture how localized delays impact broader network performance.

CRediT authorship contribution statement

Lucas J. Verhofstad: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Mahnam Saeednia:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Bilge Atasoy:** Writing – review & editing, Validation, Supervision, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge Habesha Breweries S.C. for their support and collaboration during the case study, and in particular, the efforts of the Supply Chain team for their contribution in carrying out this research. The authors also would like to thank the reviewers for their constructive feedback across different rounds of the revision.

References

Adamo, T., Gendreau, M., Ghiani, G., Guerriero, E., 2024. A review of recent advances in time-dependent vehicle routing. European J. Oper. Res. 319 (1), 1–15.

Ammouriova, M., Herrera, E.M., Neroni, M., Juan, A.A., Faulin, J., 2023. Solving vehicle routing problems under uncertainty and in dynamic scenarios: From simheuristics to agile optimization. Appl. Sci. 13 (1).

Archetti, C., Ljubić, I., 2022. Comparison of formulations for the inventory routing problem. European J. Oper. Res. 303 (3), 997–100816.

Baller, A.C., van Ee, M., Hoogeboom, M., Stougie, L., 2020. Complexity of inventory routing problems when routing is easy. Networks 75, 113–123.

Bertazzi, L., Chagas, G.O., Coelho, L.C., Laganà, D., Vocaturo, F., 2025. Online algorithms for the multi-vehicle inventory-routing problem with real-time demands. Transp. Res. Part C 170, 104892.

Bertazzi, L., Speranza, G., 2012. Inventory routing problems: an introduction. EURO J. Transp. Logist. 1, 307–326.

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2020. The multi-vehicle stochastic-dynamic inventory routing problem for bike sharing systems. Bus. Res. 30, 69–92.

Chen, X., Bai, R., Qu, R., Dong, H., 2023a. Cooperative double-layer genetic programming hyper-heuristic for online container terminal truck dispatching. IEEE Trans. Evol. Comput. 27 (5), 1220–1234.

Chen, X., Bao, F., Qu, R., Dong, J., Bai, R., 2023b. Neural network assisted genetic programming in dynamic container port truck dispatching. In: IEEE Conference on Intelligent Transportation Systems, Proceedings. ITSC, pp. 2246–2251.

Dávila, S., Alfaro, M., Fuertes, G., Vargas, M., Camargo, M., 2021. Vehicle routing problem with deadline and stochastic service times: Case of the ice cream industry in Santiago city of Chile. Mathematics 9, 2750.

de Jong, N., Aslan, A., Bakir, I., 2024. Dynamic service of geographically dispersed time-sensitive demands. Transp. Res. Part C: Emerg. Technol. 163 (104625).

Dehghan, A., Cevik, M., Bodur, M., 2023. Neural approximate dynamic programming for the ultra-fast order dispatching problem. http://dx.doi.org/10.48550/arXiv.2311.12975.

Delgado, F., 2022. Dynamic multiline vehicle dispatching strategy in transit operations. IEEE Trans. Intell. Transp. Syst. 23 (12), 24918–24928.

Emanab, Z.E., 2023. Prediction of Travel Time Using Deep Learning Algorithms to Solve TDVRPTW (Master's thesis). University of Manitoba.

Gendreau, M., Ghiani, G., Guerriero, E., 2015. Time-dependent routing problems: A review. Comput. Oper. Res. 64, 189–197.

Gmira, M., Gendreau, M., Lodi, A., Potvin, J.-Y., 2021. Managing in real-time a vehicle routing plan with time-dependent travel times on a road network. Transp. Res. Part C: Emerg. Technol. 132.

Guo, G., Xu, Y., 2020. A deep reinforcement learning approach to ride-sharing vehicle dispatching in autonomous mobility-on-demand systems. IEEE Intell. Transp. Syst. Mag. 99.

Honda, Y., Nakade, K., 2024. A heuristic algorithm for the vehicle routing problem with stochastic travel and service times. In: Proceedings of the 3rd International Conference on Smart Manufacturing, Industrial and Logistics Engineering (SMILE 2023) and the 7th Asian Conference of Management Science and Applications (ACMSA 2023). In: Lecture Notes in Mechanical Engineering, pp. 549–563, Chengdu, China, 17–19 November 2023.

Iklassov, Z., Sobirov, I., Solozabal, R., Takáč, M., 2024. Reinforcement learning for solving stochastic vehicle routing problem. In: Yanı koğlu, B., Buntine, W. (Eds.), Proceedings of the 15th Asian Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 222, PMLR, pp. 502–517.

Keskin, M., Laporte, G., Çatay, B., 2019. Electric vehicle routing problem with time-dependent waiting times at recharging stations. Comput. Oper. Res. 107, 77–94.

Lam, S.S.W., Ng, C.B.L., Nguyen, F.N.H.L., Ng, Y.Y., Ong, M.E.H., 2017. Simulation-based decision support framework for dynamic ambulance redeployment in Singapore. Int. J. Med. Inform. 106, 37–47.

Lan, Y.-L., Liu, F.-G., Huang, Z., Ng, W.W.Y., Zhong, J., 2022a. Two-echelon dispatching problem with mobile satellites in city logistics. IEEE Trans. Intell. Transp. Syst. 23 (1), 84–96.

Lan, Y.-L., Liu, F., Ng, W.W.Y., Gui, M., Lai, C., 2022b. Multi-objective two-echelon city dispatching problem with mobile satellites and crowd-shipping. IEEE Trans. Intell. Transp. Syst. 23 (9), 15340–15353.

Mardešic, N., Erdeli, T., Caric, T., Durasevic, M., 2024. Review of stochastic dynamic vehicle routing in the evolving urban logistics environment. Mathematics 12, 28.

Marković, D., Petrović, G., Ćojbašić, v., Stanković, A., 2020. The vehicle routing problem with stochastic demands in an urban area–a case study. Facta Univ. Ser.: Mech. Eng. 18 (1), 107–120.

McKenna, R.S., Robbins, M.J., Lunday, B.J., McCormack, I.M., 2020. Approximate dynamic programming for the military inventory routing problem. Ann. Oper. Res. 288, 391–416.

Mirzaei-Nasirabad, H., Mohtasham, M., Askari-Nasab, H., Alizade, B., 2023. An optimization model for the real-time truck dispatching problem in open-pit mining operations. Optim. Eng. 24, 2449–2473.

Mor, A., Speranza, M.G., 2022. Vehicle routing problems over time: a survey. Ann. Oper. Res. 314, 255—-275.

Moradi-Afrapoli, A., Askari-Nasab, H., 2020. A stochastic integrated simulation and mixed integer linear programming optimisation framework for truck dispatching problem in surface mines. Int. J. Min. Miner. Eng. 11 (4), 257–284.

Muñoz-Villamizar, A., Faulin, J., Reyes-Rubiano, L., Henriquez-Machado, R., Solano-Charris, E., 2024. Integration of google maps API with mathematical modeling for solving the real-time VRP. Transp. Res. Procedia 78, 32–39, 25th Euro Working Group on Transportation Meeting.

Pappalardo, M., 2021. Dynamic and Stochastic Vehicle Routing Problem: a Simulation Tool for a Large-Scale Study Case (Ph.D. thesis). Politecnico di Torino.

Petropoulos, F., Laporte, G., Aktas, E., Alumur, S.A., Archetti, C., Ayhan, H., Battarra, M., Bennell, J.A., Bourjolly, J.-M., Boylan, J.E., Breton, M., Canca, D., Charlin, L., Chen, B., Cicek, C.T., Jr, L.A.C., Currie, C.S., Demeulemeester, E., Ding, L., Disney, S.M., Ehrgott, M., Eppler, M.J., Erdoğan, G., Fortz, B., Franco, L.A., Frische, J., Greco, S., Gregory, A.J., Hämäläinen, R.P., Herroelen, W., Hewitt, M., Holmström, J., Hooker, J.N., Işık, T., Johnes, J., Kara, B.Y., Karsu, Ö., Kent, K., Köhler, C., Kunc, M., Kuo, Y.-H., Letchford, A.N., Leung, J., Li, D., Li, H., Lienert, J., Ljubić, I., Lodi, A., Lozano, S., Lurkin, V., Martello, S., McHale, I.G., Midgley, G., Morecroft, J.D., Mutha, A., Oğuz, C., Petrovic, S., Pferschy, U., Psaraftis, H.N., Rose, S., Saarinen, L., Salhi, S., Song, J.-S., Sotiros, D., Stecke, K.E., Strauss, A.K., Tarhan, İ., Thielen, C., Toth, P., Woensel, T.V., Berghe, G.V., Vasilakis, C., Vaze, V., Vigo, D., Virtanen, K., Wang, X., Weron, R., White, L., Yearworth, M., Yıldırım, E.A., Zaccour, G., Zhao, X., 2024. Operational research: methods and applications. J. Oper. Res. Soc. 75 (3), 423–617.

Pillac, V., Gendreau, M., Guéret, C., Medaglia, A.L., 2013. A review of dynamic vehicle routing problems. European J. Oper. Res. 225 (1), 1–11.

Rempel, M., Cai, J., 2021. A review of approximate dynamic programming applications within military operations research. Oper. Res. Perspect. 8, 100204.

Shahparvari, S., Abbasi, B., 2017. Robust stochastic vehicle routing and scheduling for bushfire emergency evacuation: An Australian case study. Transp. Res. Part A: Policy Pr. 104, 32–49.

Shahparvari, S., Abbasi, B., Chhetri, P., 2017. Possibilistic scheduling routing for short-notice bushfire emergency evacuation under uncertainties: An Australian case study. Omega 72, 96–117.

W., X., Li, P., Huang, W., Liu, Z., Liu, Q., 2023. Two-stage vehicle pair dispatch in multi-hop ridesharing. In: Proceedings of the 2023 26th International Conference on Computer Supported Cooperative Work in Design. IEEE, pp. 255–260.

Wang, X., Fodjo, I.D., Chu, X., Chen, K., Dong, F., Hirota, K., 2022. A hierarchical multiplex structure plus model with fuzzy inference for VRSDP/MD practical transportation problems. Appl. Math. Model. Comput. Simul. 132–151.

Wang, X., Xing, Z., Wu, W., Chen, X., 2021. A hybrid algorithm to solve periodic vehicle dispatching problem with heterogeneous vehicles in logistics transportation. In: Proceedings of the 2021 IEEE International Conference on Networking, Sensing and Control. ICNSC, IEEE.

Yernar, A., Turan, C., 2025. Recent developments in vehicle routing problem under time uncertainty: a comprehensive review. Bull. Electr. Eng. Informatics 14 (2), 1263–1275.

Zou, W.-Q., Pan, Q.-K., Meng, T., Gao, L., Wang, Y.-L., 2020. An effective discrete artificial bee colony algorithm for multi-AGVs dispatching problem in a matrix manufacturing workshop. Expert Syst. Appl. 161, 113675.