Research School for Operations
Management and Logistics

Anticipatory Freight Selection in Intermodal Long-haul Round-trips

A.E. Pérez Rivera and M.R.K. Mes

Department of Industrial Engineering and Business Information Systems,
University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Beta Working Paper series 492

BETA publicatie	WP 492 (working paper)
ISBN	
ISSN	
NUR	
Eindhoven	December 2015

Anticipatory Freight Selection in Intermodal Long-haul Round-trips

A.E. Pérez Rivera*and M.R.K. Mes

Department of Industrial Engineering and Business Information Systems,
University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Abstract

We consider a Logistic Service Provider (LSP) that transports freight periodically in a long-haul round-trip. At the start of a round-trip, the LSP consolidates freights in a long-haul vehicle and delivers them to multiple destinations in a region. Within this region, the LSP picks up freights using the same vehicle and transports them back to the starting location. The same region is visited every period, independent of which freights were consolidated. Consequently, differences in costs between two periods are due to the destinations visited (for delivery and pickup of freights) and the use of an alternative transport mode. Freights have different time-windows and become known gradually over time. The LSP has probabilistic knowledge about the arrival of freights and their characteristics. Using this knowledge, the goal of the LSP is to consolidate freights in a way that minimizes the total costs over time. To achieve this goal, we propose the use of a look-ahead policy, which is computed using an Approximate Dynamic Programming (ADP) algorithm. We test our solution method using information from a Dutch LSP that transports containers daily, by barge, from the East of the country to different terminals in the port of Rotterdam, and back. We show that, under different instances of this real-life information, the use of an ADP policy yields cost reductions up to 25.5% compared to a benchmark policy. Furthermore, we discuss our findings for several network settings and state characteristics, thereby providing key managerial insights about look-ahead policies in intermodal long-haul round-trips.

Keywords*: Intermodal transportation; synchromodal planning; long-haul consolidation; anticipatory routing; approximate dynamic programming.*

1 Introduction

In a world with increasing trade and increasing environmental consciousness, Logistic Service Providers (LSPs) are constantly looking for new ways of organizing their long-haul transportation processes. Nowadays, LSPs are aiming towards the efficiency of an entire transportation network in order to satisfy the increasing, and diverse, demands from customers while maximizing profitability. This aim brings many challenges in the control of LSPs transportation processes. In this paper, we study one of these challenges faced by an LSP in The Netherlands. This Dutch LSP transports containers from the Eastern part of the country to the Port of Rotterdam, and vice versa, in daily long-haul round-trips. In the first part of the round-trip, a barge transports containers from a single inland terminal to different deep-sea terminals spread over a distance of 40km in the Port of Rotterdam (e.g., export containers). In the second part of the round-trip, the same barge picks up containers from the terminals in the Port of Rotterdam, and transports them back to the inland terminal where it started (e.g., import containers). Alternatively, the LSP has trucks to transport urgent containers that are not transported with the barge. The challenge consists on how to assign the new orders that come every day (i.e., new containers to deliver or to pick up, each with different destinations and time-windows) either to the barge or to the trucks, in a way that maximizes the efficiency of the entire network.

Ideally, the barge would visit as few terminals as possible in each round-trip (both for delivery or pickup of containers) and the truck alternative would be seldom used. However, the variability in the containers that arrive each day (and their characteristics) makes this ideal situation difficult to achieve. Each day, the LSP must carefully decide which containers to transport, and which ones to postpone, if it

*Corresponding author: a.e.perezrivera@utwente.nl

wants operations to be as close to ideal over time. For example, postponing the delivery (and/or pickup) of containers to a given terminal today might result in a consolidation opportunity with the delivery (and/or pickup) of containers to that terminal tomorrow. Also, visiting an additional, nearby, terminal today might save a longer tour of terminals visited tomorrow. The proper balance of consolidation and postponement decisions in each round-trip will result in a better performance over a period of time, and thus over the entire transportation network.

In a more generic description, we study the decision problem that arises when an intermodal transportation company carries freight in long-haul round-trips, periodically. In every period, a single long-haul round-trip is performed. In each long-haul round trip, freight is transported (i) from a single origin to multiple locations within a far away region (i.e., delivery), and (ii) from locations in that region back to the origin (i.e., pickup), using a high capacity vehicle (i.e., multiple freights). Since the region is far away from the origin, but locations within the region are close to each other, the long-haul is the same in every round-trip and every period. For this reason, differences in costs between two periods arise due to the locations visited in the round-trip corresponding to each period (either for delivery or pickup of freight), and the use of an alternative transportation mode. This alternative transportation mode is more expensive than the high-capacity vehicle and can only be used for freights with immediate due-day. New freights, with different characteristics, arrive each period. Each freight has a given destination and a given time-window restricting its transportation. Although the number of freights, and their characteristics, vary from day to day, there is information about their probability distribution. Since time-windows are stochastic, freights that can be transported in future periods become known gradually over time. The objective of the company is to reduce its total costs over a multi-period horizon, by selecting the right freights (for the long-haul round-trips) each period.

For three reasons, selecting the freights (for both parts of the long-haul round-trip) that minimize the costs over a multi-period horizon is not straightforward. First, the freights that arrive in each period, either for delivery or for pickup, are uncertain. The uncertainty is not only on the number of freights that arrive, but also on their characteristics. Second, freights have different time-windows which (i) restrict the periods in which they can be consolidated (i.e., release-day), and (ii) restrict the periods to which they can be postponed (i.e., due-day). Third, the cost advantage of using the high capacity vehicle for a given period (i.e., selecting as many freights to the same location as possible), might be conflicting with the objective of minimizing costs over a multi-period horizon. To overcome these challenges, and reduce costs over time, we model the freight selection in intermodal long-haul round-trips as a Markov Decision Process (MDP), and propose an Approximate Dynamic Programming (ADP) algorithm to solve it.

Our goal is to provide insight in how the diverse problem characteristics influence the anticipatory freight selection/postponement decisions. By modeling the problem as an MDP, we can incorporate all characteristics of the stochastic freight demand and make the dilemma of selecting/postponing freights measurable over a multi-period horizon. This measurement provides the optimal tradeoff between carrying as many freights in the long-haul vehicle as possible and reducing costs for the entire horizon. Furthermore, with this modeling approach, time-evolving systems (such as the time-windows of freights) can be directly analyzed. As with many optimal approaches, MDPs are usually not applicable to realistic problem instances. To overcome this shortcoming, we propose an ADP solution. An ADP solution approximates the solution to the MDP model while retaining all stochastic and time-evolving characteristics. The approximation makes it applicable for realistic instances, thus allowing us to provide insights based on information from the Dutch LSP.

The remainder of this paper is structured as follows. In Section 2, we briefly review the relevant literature and specify the contribution of our research to it. In Section 3, we introduce the notation and formulate the problem as an MDP. In Section 4, we present the ADP solution algorithm. In Section 5, we evaluate various designs for the ADP algorithm, and provide a comparison with exact and benchmark policies. Finally, we conclude this paper in Section 6 with the important theoretical observations and key managerial insights about modeling and solving the anticipatory freight selection problems in intermodal long-haul round trips.

2 Literature Review

The literature on freight consolidation in intermodal transportation networks is vast. In this brief review of it, we focus on two problem classes: (i) problems concerning assignment of freights to transport modes in an intermodal network, and (ii) problems concerning anticipatory routing of vehicles or dynamic selection of loads in transportation. In the first class, we summarize the key points and shortcomings of models and solution approaches proposed for Dynamic Service Network Design (DSND) problems in

intermodal transportation networks. In the second class, we provide examples on how the dynamic and stochastic nature of demand in transportation has been captured in routing and transportation problems, and what kind of solutions have been proposed for them. For an extensive review on research about the first problem class, we refer the reader to SteadieSeifi et al. [2014] and Crainic and Kim [2007]; and for the second class, to Pillac et al. [2013] and Powell et al. [2007].

Decision problems in DSND involve the choice of transportation services (or modes) for freight, over a multi-period horizon, where at least one problem characteristic varies over time [SteadieSeifi et al., 2014]. However, two of the shortcomings in most DSND studies are that: (i) they do not incorporate time-dependencies (e.g., time-windows and in-advance information), even with multi-period horizons [Crainic and Kim, 2007], or (ii) they assume deterministic demand [SteadieSeifi et al., 2014]. Furthermore, it seems that studies that tackle these shortcomings do it one at a time, leaving room for further improvement. For example, studies that model time dependencies, such as Andersen et al. [2009b], and consolidation opportunities, such as Moccia et al. [2011], frequently assume deterministic demand. Research that models uncertainty in the demand, such as Hoff et al. [2010], is usually developed for different transportation services of a single transportation mode (e.g, truck), although services offered by LSPs are increasingly becoming intermodal.

To some extent, the aforementioned shortcomings have been tackled one at a time due to the solution approaches used. Classical graph theory and meta-heuristics, which have been extensively applied to solve DSND problems [SteadieSeifi et al., 2014, Wieberneit, 2008], are less suitable for dealing with time-dependencies and stochastic demands. To deal with time-dependencies, mathematical programming techniques such as cycle-based variables [Andersen et al., 2009a], branch-and-price Andersen et al. [2011], and digraphs formulations[Moccia et al., 2011] have been proposed. In a similar way, meta-heuristics, such as Tabu Search [Crainic et al., 2000, Verma et al., 2012], are used to tackle time-dependencies in large problems [SteadieSeifi et al., 2014]. However, integrating stochasticity in these techniques and heuristics is not straightforward. Further designs, such as stochastic scenarios [Hoff et al., 2010], are necessary to model the variability in some of the DSND problem parameters. Still, the possible gains of incorporating stochasticity into DSND models and solution approaches has been acknowledged in practice [Lium et al., 2009] and an increasing number of studies have been performed in the last years to determine the value of exploiting stochastic information [Zuidwijk and Veenstra, 2015].

In contrast to the first problem class, the second class of concerning dynamic and stochastic freight transportation has been studied extensively for routing of a *single mode* [Pillac et al., 2013, Powell et al., 2007]. Although our problem contains *multiple modes* (vehicle types), work done in this second class provides valuable insights for our work. One of these insights concerns the value of dynamic, in-advance, information itself. Knowledge of load information, one or two days in advance, has been shown to improve performance in trucking companies [Zolfagharinia and Haughton, 2014]. Another of these insights concerns stochastic freight (load) demand, where demand reveals dynamically with time or with events. Two strategies have been widely used to tackle problems with such demand: (i) sampling strategies, and (ii) stochastic modeling [Pillac et al., 2013]. Both strategies yield solutions that anticipate on the realization of the stochastic variables (i.e., policies for the possible realizations of the random variables) and that perform better than reactive (non-anticipatory) approaches.

Each strategy, however, has its own difficulties. The first strategy requires procedures capable of correctly sampling the random variables, which come with some form of bias and are heuristic in nature, such as the Indifference Zone Selection approach used by Ghiani et al. [2009]. The second strategy requires complete analytical models of the evolution of the system and its variability, which are usually non-scalable or non-applicable to real-life instances, such as the Markov Decision Process model used by Novoa and Storer [2009]. To overcome the difficulties of each strategy, several techniques have been proposed in the literature [Pillac et al., 2013]. To reduce the bias of sampling methods, Multiple Scenario Approaches with algorithms based on consensus, expectation, or regret of the probabilistic knowledge, have shown significant benefits [Bent and Hentenryck, 2004]. To reduce the dimensionality issues of stochastic modeling, Approximate Dynamic Programming based on roll-out procedures and value function approximation has been used [Novoa and Storer, 2009, Simao et al., 2009].

To summarize, previous research about intermodal and stochastic freight transportation had different perspectives. Within the first problem class, there has been little research about large stochastic multi-period problems [Lium et al., 2009, SteadieSeifi et al., 2014, Wieberneit, 2008]. Within the second problem class, research about multiple modes and pickup and delivery has been studied less in comparison to a single-mode or task [Berbeglia et al., 2010, Pillac et al., 2013]. For these reasons, our work has four contributions to the literature about intermodal and stochastic freight transportation. First, we propose an MDP model that includes stochastic freight demand (and its characteristics) for an intermodal network,

handles complex time-dependencies, and measures performance over a multi-period horizon. Second, we propose an ADP algorithm to solve the model for large (realistic) problem instances. Third, we provide methodological insights on the design process of an ADP algorithm for intermodal transportation networks. Fourth, we provide managerial insights on the selection/postponement decisions for several intermodal network settings and as day-to-day (state) characteristics.

3 Mathematical Model

In this section, we formulate a mathematical model of the optimization problem using Markov Decision Process (MDP) theory. First, we introduce the notation for the stochastic, multi-period, and discrete problem characteristics mentioned in Section 1. Next, we model these characteristics as an MDP using stages, states, decisions, transitions, and the optimality equations. Finally, we discuss the dimensionality issues of this model.

3.1 Notation

We consider a finite multi-period horizon $\mathcal{T} = \{0, 1, 2, \ldots, T^{max} - 1\}$. At each period $t \in \mathcal{T}$, one high capacity vehicle performs a round-trip, traveling from a single origin to a group of far-away locations $\mathcal{D}' \subseteq \mathcal{D}$ within a region, and back. Each round-trip is divided in two phases: (i) the *delivery* and (ii) the *pickup*. In the delivery phase, freights are transported from the origin to multiple location. In the pickup phase, freights are transported from multiple location to the origin. Sometimes, these phases occur "simultaneously" within a round-trip, i.e., when there are both freights to be delivered and to be picked up at the same location. Since only one round-trip is planned each period, a total of T^{max} consecutive round-trips are considered. Each period, the planner selects freights to consolidate in both parts of the round-trip. Each freight has a location $d \in \mathcal{D}$ where it must be delivered to, or picked up from. For simplicity, in the remainder of this paper we refer to a period as a "day", and to a location as a "destination" that a vehicle must visit in a round-trip. Furthermore, we name freights as delivery freights and as pickup freights, depending on the phase of the round-trip they must be transported in.

Each delivery and pickup freight must be transported within a specific time-window. Time-windows are characterized by a release-day $r \in \mathcal{R} = \{0, 1, 2, \cdots, R^{max}\}$ and a time-window length of $k \in \mathcal{K} = \{0, 1, 2, \cdots, K^{max}\}$ days. For modeling purposes, the release-day of a freight is relative to the current day (e.g., a freight that has $r = 1$ today will be released tomorrow), and the time-window length of a freight is relative to its release-day (e.g., a freight that has $k = 0$ has to be transported the same day it is released). Note that r is the number of days in advance that the LSP knows about a freight before it can be transported. Also note that k is the number of days within which the LSP has to transport a freight, once it has been released. Thus, the larger k is, the more flexibility the LSP has for consolidating this freight in different round-trips.

Although freights (and their different characteristics) are known only after they arrive, the LSP has probabilistic knowledge about their arrival. This probabilistic knowledge comes in the form of eight empirical probability distributions. In between two consecutive days, $f \in \mathcal{F}$ delivery freights arrive with probability p_f^F; and $g \in \mathcal{G}$ pickup freights arrive with probability p_g^G. A delivery freight has to be delivered to destination $d \in \mathcal{D}$ with probability $p_d^{D,F}$. A pickup freight has to be picked up at destination $d \in \mathcal{D}$ with probability $p_d^{D,G}$. Furthermore, an arriving delivery freight has release-day $r \in \mathcal{R}$ with probability $p_r^{R,F}$ and time-window length $k \in \mathcal{K}$ with probability $p_k^{K,F}$. Similarly, an arriving pickup freight has release-day $r \in \mathcal{R}$ with probability $p_r^{R,G}$ and time-window length $k \in \mathcal{K}$ with probability $p_k^{K,G}$. All the probabilities describing the characteristics of a freight are independent of the day and of other freights. All of the aforementioned stochastic characteristics of the demand are summarized in Table 1. Note that the first letter in the superscript of a probability denotes the characteristic of a freight (D for destination, R for release-day, and K for time-window length) and the second letter denotes which part of the round-trip it represents (F for the delivery and G for the pickup).

We now present the parameters related to the transportation resources. The costs of the high capacity vehicle depend on the group of destinations it visits, for both delivery and pickup of freights. We denote a group (i.e., a combination) of destinations with $\mathcal{D}' \subseteq \mathcal{D}$, and define its associated cost with $C_{\mathcal{D}'}$. In this definition of costs, we do not take into account the sequence or the number of times in which destinations are visited (either for delivery or pickup of freight). Nevertheless, if costs of permutation, or repetition, of destinations are necessary, they can easily be incorporated as we will show in Section 5.3. In addition to $C_{\mathcal{D}'}$, there is a cost B_d per freight with destination d consolidated in the high capacity vehicle. This

Table 1: Demand related parameters and their notation

Parameter	**Set**	**Probabilities**
Number of delivery freights	$\mathcal{F} \subseteq \mathbb{N}$	$p_f^F \ \forall f \in \mathcal{F}$
Number of pickup freights	$\mathcal{G} \subseteq \mathbb{N}$	$p_g^G \ \forall g \in \mathcal{G}$
Locations	$\mathcal{D}$	$p_d^{D,F}, p_d^{D,G} \ \forall d \in \mathcal{D}$
Release-days	$\mathcal{R} = \{0, 1, 2, ..., R^{max}\}$	$p_r^{R,F}, p_r^{R,G} \ \forall r \in \mathcal{R}$
Time-window lengths	$\mathcal{K} = \{0, 1, 2, ..., K^{max}\}$	$p_k^{K,F}, p_k^{K,G} \ \forall k \in \mathcal{K}$

vehicle has a maximum transport capacity of Q freights for each part of the round-trip. There is also an alternative transport option (e.g., truck) for the delivery, or the pickup, of freight at each destination d at a cost of A_d per freight. The use of the alternative option is restricted for freights whose due-day is immediate (i.e., $r = k = 0$).

3.2 Formulation

Each day of the planning horizon corresponds to a *stage* in the MDP. Thus, stages are discrete, consecutive, and denoted by t. At each stage t, the LSP has information about delivery and pickup freights; this information is modeled using the integer variables $F_{t,d,r,k}$ and $G_{t,d,r,k}$, for all destinations d, release-days r, and time-window lengths k. $F_{t,d,r,k}$ represents the number of delivery freights and $G_{t,d,r,k}$ represents the number of pickup freights that are known at stage t. The *state* of the system at stage t is given by the vector $\boldsymbol{S}_t$ containing all the freight variables, as seen in (1). We denote the state space of the system by $\mathcal{S}$, i.e., $\boldsymbol{S}_t \in \mathcal{S}$.

$$\boldsymbol{S}_t = [(F_{t,d,r,k}, G_{t,d,r,k})]_{\forall d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \tag{1}$$

At each stage t, the LSP's *decision* consists of which delivery and pickup freights from $\boldsymbol{S}_t$ to consolidate in the long-haul vehicle of that stage's round-trip. This decision has two restrictions: (i) the release-day of freights (i.e., only freights that have been released can be transported), and (ii) the capacity of the long-haul vehicle (i.e., no more than Q containers can be consolidated in each part of the round-trip). Even though the state $\boldsymbol{S}_t$ contains all known freights (released and not released), for the decision we only consider those freights that have been released (i.e., $r = 0$). We model the decision using the non-negative integer variables $x_{t,d,k}^F$ and $x_{t,d,k}^G$, for all destinations d and time-window lengths k. $x_{t,d,k}^F$ represents the number of delivery freights consolidated in the round-trip corresponding to stage t and $x_{t,d,k}^G$ represents the number of pickup freights consolidated in the same round-trip. The decision vector $\boldsymbol{x}_t$ containing all decision variables and restrictions at stage t is defined in (2a), subject to constraints (2b) to (2f) which define the feasible decision space $\mathcal{X}_t$.

$$\boldsymbol{x}_t = \left[\left(x_{t,d,k}^F, x_{t,d,k}^G\right)\right]_{\forall d \in \mathcal{D}, k \in \mathcal{K}} \tag{2a}$$

$$\text{s.t.}$$

$$0 \leq x_{t,d,k}^F \leq F_{t,d,0,k}, \ \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{2b}$$

$$0 \leq x_{t,d,k}^G \leq G_{t,d,0,k}, \ \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{2c}$$

$$\sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} x_{t,d,k}^F \leq Q, \tag{2d}$$

$$\sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} x_{t,d,k}^G \leq Q, \tag{2e}$$

$$x_{t,d,k}^F, x_{t,d,k}^G \in \mathbb{Z}^+ \cup \{0\} \tag{2f}$$

To measure the direct costs of a decision, we need to know the group of destinations visited with the long-haul vehicle and the freights that must be transported using the alternative transport option. For this reason, we introduce two additional variables, which depend on the freights that were consolidated in the long-haul vehicle (for the two phases of a round-trip) as follows. First, we define $y_{t,d} \in \{0, 1\}$ as the binary variable that gets a value of 1 if any delivery or pickup freight with destination d is consolidated

in the long-haul vehicle at stage t and 0 otherwise. Second, we define $z_{t,d}$ as the non-negative integer variable that counts how many urgent delivery or pickup freights to destination d were not transported using the long-haul vehicle. These variables are defined as a function of the state and decision variables as seen in (3b) and (3c). The total costs at stage t are defined then as a function of the decision vector $\boldsymbol{x}_t$ and the state $\boldsymbol{S}_t$, using the auxiliary variables, as shown in (3a).

$$\begin{aligned} C\left(\boldsymbol{S}_t, \boldsymbol{x}_t\right) = & \sum_{\mathcal{D}' \subseteq \mathcal{D}} \left(C_{\mathcal{D}'} \cdot \prod_{d' \in \mathcal{D}'} y_{t,d'} \cdot \prod_{d'' \in \mathcal{D} \setminus \mathcal{D}'} \left(1 - y_{t,d''}\right) \right) \\ & + \sum_{d \in \mathcal{D}} \left(A_d \cdot z_{t,d}\right) \\ & + \sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} \left(B_d \cdot \left(x^F_{t,d,k} + x^G_{t,d,k}\right)\right) \end{aligned} \tag{3a}$$

s.t.

$$y_{t,d} = \begin{cases} 1, & \text{if } \sum_{k \in \mathcal{K}} \left(x^F_{t,d,k} + x^G_{t,d,k}\right) > 0 \\ 0, & \text{otherwise} \end{cases}, \ \forall d \in \mathcal{D} \tag{3b}$$

$$z_{t,d} = F_{t,d,0,0} - x^F_{t,d,0} + G_{t,d,0,0} - x^G_{t,d,0}, \ \forall d \in \mathcal{D} \tag{3c}$$

The objective of the problem is to minimize the transportation costs over the planning horizon, i.e., the sum of (3a) over all $t \in \mathcal{T}$. However, there is uncertainty in the arrival of freights (and their characteristics) within this horizon, meaning that the states $\boldsymbol{S}_t$ are also uncertain. For this reason, the objective of our model must be expressed in terms of expected costs over the horizon and an optimal decision for every possible state in the horizon must be found. Since for every possible state there is an optimal decision, the output of the model is a policy. We use π to denote a policy, i.e., a function that maps each possible state $\boldsymbol{S}_t$ to a decision vector $\boldsymbol{x}^\pi_t$, for every $t \in \mathcal{T}$. We denote the set of all such policies by Π. The objective is to find the policy $\pi^* \in \Pi$ that minimizes the expected costs over the planning horizon, given an initial state $\boldsymbol{S}_0$, as seen in (4).

$$\min_{\pi \in \Pi} \mathbb{E}\left\{ \sum_{t \in \mathcal{T}} C\left(\boldsymbol{S}_t, \boldsymbol{x}^\pi_t\right) \middle| \boldsymbol{S}_0 \right\} \tag{4}$$

Using Bellman's principle of optimality, the minimal expected costs (and thus the optimal policy) can be computed through a set of recursive equations. These recursive equations are expressed in terms of the current-stage and the expected next-stage costs, as seen in Equation (5). Before solving these equations, we need to define two other aspects of the model: (i) the transition (i.e., evolution) of the system from state $\boldsymbol{S}_t$ to state $\boldsymbol{S}_{t+1}$, and (ii) the transition probabilities of moving from one state to another one, given some decision. We now elaborate on these two aspects.

$$V_t\left(\boldsymbol{S}_t\right) = \min_{\boldsymbol{x}_t \in \mathcal{X}} \left(C\left(\boldsymbol{S}_t, \boldsymbol{x}_t\right) + \mathbb{E}\left\{V_{t+1}\left(\boldsymbol{S}_{t+1}\right)\right\}\right), \ \forall \boldsymbol{S}_t \in \mathcal{S} \tag{5}$$

The transition from $\boldsymbol{S}_t$ to $\boldsymbol{S}_{t+1}$ is influenced by the decision $\boldsymbol{x}_t$, as well as by the freights that arrive after this decision. To define this transition, we first focus on the arriving freights. Eight probability distributions (i.e., random variables) describe the arrival of freights, and their characteristics, over time (see Table 1). We merge these random variables into two "new information" variables: $\widetilde{F}_{t,d,r,k}$ and $\widetilde{G}_{t,d,r,k}$, for all destinations d, release-days r, and time-window lengths k. These variables represent the delivery and pickup freights, respectively, whose information arrived between stages $t-1$ and t. The vector $\boldsymbol{W}_t$ containing all the new information variables at stage t makes up the so-called *exogenous information* of the model [Powell, 2007], as seen in (6).

$$\boldsymbol{W}_t = \left[\left(\widetilde{F}_{t,d,r,k}, \widetilde{G}_{t,d,r,k}\right)\right]_{\forall d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \tag{6}$$

Using this new vector, we can define a state $\boldsymbol{S}_t$ at stage t as the result of the state of the previous stage $\boldsymbol{S}_{t-1}$, the decision vector of the previous stage $\boldsymbol{x}_{t-1}$, and the exogenous information $\boldsymbol{W}_t$ that arrived

between the stages. Remind that, to model the time-windows of freights, release-days r are indexed relative to day t and time-window lengths k are indexed relative to release-days r. Naturally, once a freight has been released, the time-window length must be decreased by one every day that passes until the freight is transported. All of these factors, and index-relations, are used to capture the transition of the system. We represent them using the *transition function* S^M shown in (7a). This function works as follows.

The transition of delivery ($F_{t,d,r,k}$) and pickup ($G_{t,d,r,k}$) freights with destination d, release-day r, and time-window length k, from $\boldsymbol{S}_{t-1}$ to $\boldsymbol{S}_t$, is defined having four considerations. First, freights that are already released at day t (i.e., $r = 0$) and have a time-window length $k < K^{max}$, are the result of freights from the previous day $t-1$ that were already released (i.e., $r = 0$), had a time-window length $k+1$, and were not consolidated in the previous round-trip; in addition to freights from the previous day $t-1$ that had a next day release (i.e., $r = 1$) and the same time-window length k, and the freights that arrived between the previous and the current day with release-day 0 and time-window length k, as seen in (7b) and (7c). Second, freights that are already released at day t and have a time-window length $k = K^{max}$ are the result of freights from the previous day $t-1$ that had a next day release (i.e., $r = 1$) and the same time-window length K^{max}, in addition to the freights that arrived between the previous and the current day with the same characteristics (i.e., $r = 0$ and $k = K^{max}$), as seen in (7d) and (7e). Third, freights that are not released at day t, do not have the maximum release-day (i.e., $0 < r < R^{max}$), and have time-window length k, are the result of freights from the previous day $t-1$ that had a release-day $r+1$ and a time-window length k, in addition to the freights that arrived between the previous and the current day with the same characteristics (i.e., r and k), as shown in (7f) and (7g). Fourth, freights that are not released at day t, have the maximum release-day R^{max}, and have time-window length k, are the result only of the freights that arrived between the previous and the current day with release-day R^{max} and time-window length k, as seen in (7h) and (7i).

$$\boldsymbol{S}_t = S^M\left(\boldsymbol{S}_{t-1}, \boldsymbol{x}_{t-1}, \boldsymbol{W}_t\right),\ \forall t \in \mathcal{T} | t > 0 \tag{7a}$$

where

$$F_{t,d,0,k} = F_{t-1,d,0,k+1} - x^F_{t-1,d,k+1} + F_{t-1,d,1,k} + \widetilde{F}_{t,d,0,k}, \tag{7b}$$

$$G_{t,d,0,k} = G_{t-1,d,0,k+1} - x^G_{t-1,d,k+1} + G_{t-1,d,1,k} + \widetilde{G}_{t,d,0,k}, \tag{7c}$$

$$\forall d \in \mathcal{D},\ \text{and}\ k \in \mathcal{K} | k < K^{max}\ .$$

$$F_{t,d,0,K^max} = F_{t-1,d,1,K^{max}} + \widetilde{F}_{t,d,0,K^{max}}, \tag{7d}$$

$$G_{t,d,0,K^max} = G_{t-1,d,1,K^{max}} + \widetilde{G}_{t,d,0,K^{max}}, \tag{7e}$$

$$\forall d \in \mathcal{D}.$$

$$F_{t,d,r,k} = F_{t-1,d,r+1,k} + \widetilde{F}_{t,d,r,k}, \tag{7f}$$

$$G_{t,d,r,k} = G_{t-1,d,r+1,k} + \widetilde{G}_{t,d,r,k} \tag{7g}$$

$$\forall d \in \mathcal{D},\ r \in \mathcal{R} | 0 < r < R^{max},\ \text{and}\ k \in \mathcal{K}\ .$$

$$F_{t,d,R^{max},k} = \widetilde{F}_{t,d,R^{max},k} \tag{7h}$$

$$G_{t,d,R^{max},k} = \widetilde{G}_{t,d,R^{max},k} \tag{7i}$$

$$\forall d \in \mathcal{D},\ \text{and}\ k \in \mathcal{K}\ .$$

Using the transition function S^M, we can rewrite (5) in terms of the arriving information $\boldsymbol{W}_{t+1}$ as shown in (8). In (8), the only stochastic variable at stage t is the vector $\boldsymbol{W}_t$, i.e., the exogenous information. As explained earlier, this vector contains all new information $\widetilde{F}_{t,d,r,k}$ and $\widetilde{G}_{t,d,r,k}$, which are based on the eight discrete and finite random variables describing the arrival of freights (and their characteristics). The vector $\boldsymbol{W}_t$ is thus a random vector with discrete and finite possible realizations. We denote the set containing all these realizations with Ω, i.e., $\boldsymbol{W}_t \in \Omega$, $\forall t \in \mathcal{T}$. For each realization $\omega \in \Omega$, there is an associated probability p^{Ω}_{ω}. Using these probabilities, the expectation in (8) can be rewritten, as seen in (9).

$$V_t\left(\boldsymbol{S}_t\right) = \min_{\boldsymbol{x}_t \in \mathcal{X}} \left(C\left(\boldsymbol{S}_t, \boldsymbol{x}_t\right) + \mathbb{E}\left\{V_{t+1}\left(S^M\left(\boldsymbol{S}_t, \boldsymbol{x}_t, \boldsymbol{W}_{t+1}\right)\right)\right\}\right),\ \forall \boldsymbol{S}_t \in \mathcal{S} \tag{8}$$

$$V_t\left(\boldsymbol{S}_t\right) = \min_{\boldsymbol{x}_t \in \mathcal{X}} \left(C\left(\boldsymbol{S}_t, \boldsymbol{x}_t\right) + \sum_{\omega \in \Omega} \left(p_\omega^\Omega \cdot V_{t+1}\left(S^M\left(\boldsymbol{S}_t, \boldsymbol{x}_t, \omega\right)\right)\right)\right),\ \forall \boldsymbol{S}_t \in \mathcal{S} \tag{9}$$

The probability p_ω^Ω depends in three aspects of the realization $\omega \in \Omega$. First, it depends on the total number of delivery and pickup freights arriving, which we denote with f and g respectively. Second, it depends on the probability that $\widetilde{F}_{d,r,k}^\omega$ delivery freights and $\widetilde{G}_{d,r,k}^\omega$ pickup freights will have destination d, release-day r and time-window length k. Third, it depends on a multinomial coefficient β [Riordan, 2002] that counts the number ways of assigning the total number of arriving delivery freights f and pickup freights g to each variable $\widetilde{F}_{d,r,k}^\omega$ and $\widetilde{G}_{d,r,k}^\omega$, respectively. This coefficient is necessary since the order in which freights arrive does not matter and freight characteristics are allowed to "repeat". With these three aspects, the probability p_ω^Ω can be computed with (10a).

$$p_\omega^\Omega = \beta \cdot p_f^F p_g^G \cdot \prod_{d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \left(\left(p_d^{D^F} p_r^{R^F} p_k^{K^F}\right)^{\widetilde{F}_{d,r,k}^\omega} \left(p_d^{D^G} p_r^{R^G} p_k^{K^G}\right)^{\widetilde{G}_{d,r,k}^\omega}\right) \tag{10a}$$

where

$$f = \sum_{d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \widetilde{F}_{d,r,k}^\omega \tag{10b}$$

$$g = \sum_{d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \widetilde{G}_{d,r,k}^\omega \tag{10c}$$

$$\beta = \frac{f!}{\prod_{d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \left(\widetilde{F}_{d,r,k}^\omega!\right)} \cdot \frac{g!}{\prod_{d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \left(\widetilde{G}_{d,r,k}^\omega!\right)} \tag{10d}$$

Solving the recursive equations in (9), using the probabilities from (10a), will yield the minimum expected costs and the optimal policy for the entire planning horizon. However, the computational effort to do this increases exponentially with increasing domains of the eight random variables. We look into more detail on the dimensionality issues in the next section.

3.3 Dimensionality Issues

The optimality equations in (9) suffer from what Powell [2007] calls "three curses of dimensionality". The first curse comes from the set of all possible realizations of the exogenous information Ω. For each possible decision $\boldsymbol{x}_t \in \mathcal{X}$, the calculation of the expectation requires the next-stage value of $|\Omega|$ states. The second issue comes from evaluating all possible decisions. At each state, finding the decision that minimizes the sum of the direct and and expected downstream costs involves the evaluation of all possible combinations of the freights that are released (i.e., $F_{t,d,0,k}$ and $G_{t,d,0,k}$, for all destinations d and time-window lengths k). The third, and most difficult of all dimensionality issues, comes from the set of all possible states $\mathcal{S}$. In our model, the number of possible states increases with increasing domains of the freight demand parameters seen in Table 1, and specially with the number of release-days $|\mathcal{R}|$ since this determines the degree in which freights can be accumulated. To provide the reader with a measurable idea on these issues, we elaborate on the first curse of dimensionality.

Each realization $\omega \in \Omega$ is basically a combination of the values that the exogenous information variables $\widetilde{F}_{t,d,r,k}$ and $\widetilde{G}_{t,d,r,k}$ can have. Note that there are a total of $\left(|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}|\right)^2$ variables in a realization ω. Since both $\widetilde{F}_{t,d,r,k}$ and $\widetilde{G}_{t,d,r,k}$ can have a value greater than one (i.e., multiple freights with the same characteristics), the number of possible realizations of exogenous information $|\Omega|$ depends not only on the number of different characteristics, but also on the number of freights that can have the same characteristics, as seen in (11).

$$\begin{aligned} |\Omega| &= \sum_{n=0}^{|\mathcal{F}|} \binom{|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| + n - 1}{n} \cdot \sum_{n=0}^{|\mathcal{G}|} \binom{|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| + n - 1}{n} \\ &= \sum_{n=0}^{|\mathcal{F}|} \frac{(|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| + n - 1)!}{n!\,(|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| - 1)!} \cdot \sum_{n=0}^{|\mathcal{G}|} \frac{(|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| + n - 1)!}{n!\,(|\mathcal{D}| \cdot |\mathcal{R}| \cdot |\mathcal{K}| - 1)!} \end{aligned} \tag{11}$$

The more possible freight characteristics there are, the larger the set Ω becomes. In a similar way, but with an even stronger relation, the set of all states $\mathcal{S}$ grows exponentially with an increasing number of possible freight characteristics. The set of possible decisions $\mathcal{X}$ grows fast as well, but the optimal action can be obtained (for realistic problems and in reasonable time) through an Integer Linear Program (ILP), as will be shown later on. Due to these dimensionality issues, an exact solution to (9) (e.g., using backward dynamic programming) is only feasible in problems with a small number of destinations, possible release-days, and possible time-window lengths. Nevertheless, the model provides the foundation for our proposed Approximate Dynamic Programming (ADP) algorithm for realistic problem sizes, which we explain in the following section.

4 Solution Algorithm

Our proposed solution algorithm approximates the optimal solution of the model presented in the previous section. This algorithm is based on the modeling framework of Approximate Dynamic Programming (ADP), which contains several methods for tackling the curses of dimensionality. The output of the ADP algorithm is both an approximation of the expected costs and a policy based on this approximation. The general idea of ADP is to modify the Bellman's equations with a series of components and algorithmic manipulations in order to approximate their solution. In this section, we elaborate on the additional components and algorithmic manipulations we use to find the approximated expected costs and their policy, as shown in Algorithm 1. Note that all components in this algorithm are indexed with a superscript n, to denote the iterations performed. This section is divided in three parts. First, we introduce the concepts of *post-decision state* and *forward dynamic programming*, which tackle the first and third dimensionality issues mentioned in Section 3.3. Second, we introduce the concept of *basis functions* as an approximation of the value of the post-decision states. Finally, we describe a way of tackling the second dimensionality issue of finding the optimal action for a single stage.

Algorithm 1 Approximate Dynamic Programming Solution Algorithm

Initialize $\bar{V}_t^0,\ \forall t \in \mathcal{T}$
$n := 1$
while $n \leq N$ **do**
  $\boldsymbol{S}_0^n := \boldsymbol{S}_0$
  for $t = 0$ **to** $T^{max} - 1$ **do**
    $\hat{v}_t^n := \min_{\boldsymbol{x}_t^n} \left(C\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right) + \bar{V}_t^{n-1}\left(S^{M,x}\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right)\right)\right)$
    $\boldsymbol{x}_t^{n*} := \arg\min_{\boldsymbol{x}_t^n} \left(C\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right) + \bar{V}_t^{n-1}\left(S^{M,x}\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right)\right)\right)$
    $\boldsymbol{S}_t^{n,x*} := S^{M,x}\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^{n*}\right)$
    $\boldsymbol{W}_{t+1}^n := \text{RandomFrom}\left(\Omega\right)$
    $\boldsymbol{S}_{t+1}^n := S^M\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^{n*}, \boldsymbol{W}_{t+1}^n\right)$
  end for
  for $t = T^{max} - 1$ **to** 1 **do**
    $\bar{V}_{t-1}^n(\boldsymbol{S}_{t-1}^{n,x*}) := U^V(\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x*}), \boldsymbol{S}_{t-1}^{n,x*}, \hat{v}_t^n)$
  end for
  $n := n + 1$
end while
return $\left[\bar{V}_t^N\right]_{\forall t \in \mathcal{T}}$

4.1 Post-decision State and Forward Dynamic Programming

To tackle the first dimensionality issue (i.e., the large set of realizations of the exogenous information Ω), we introduce two new components into the model: (i) a post-decision state $\boldsymbol{S}_t^{n,x}$, and (ii) an approximated next-stage cost $\bar{V}_t^n(\boldsymbol{S}_t^{n,x})$. The post-decision state is the state of the system directly after a decision $\boldsymbol{x}_t^n$ has been made (given state $\boldsymbol{S}_t^n$) but before the exogenous information $\boldsymbol{W}_t^n$ arrives, at iteration n of the algorithm. In a similar way to the freight variables of a state, the post-decision freight variables $F_{t,d,r,k}^{n,x}$ and $G_{t,d,r,k}^{n,x}$ form the post-decision state $\boldsymbol{S}_t^{n,x}$, as seen in (12).

$$\boldsymbol{S}_t^{n,x} = \left[\left(F_{t,d,r,k}^{n,x}, G_{t,d,r,k}^{n,x}\right)\right]_{\forall d \in \mathcal{D}, r \in \mathcal{R}, k \in \mathcal{K}} \tag{12}$$

Second, the approximated next-stage cost $\bar{V}_t^n(\boldsymbol{S}_t^{n,x})$ serves as an estimated measurement for all future costs (i.e., $\bar{V}_t^n(\boldsymbol{S}_t^{n,x}) \approx \mathbb{E}\left\{V_{t+1}\left(\boldsymbol{S}_{t+1}\right)\right\}$). We elaborate on how this measurement replaces the standard expectation in the optimality equations (5) later on. For now, we focus on the post-decision state.

To define a post-decision state $\boldsymbol{S}_t^{n,x}$ at time t and iteration n, we define a function $S^{M,x}$ that relates the post-decision freight variables $\boldsymbol{S}_t^{n,x}$ with the state $\boldsymbol{S}_t^n$ and decision $\boldsymbol{x}_t^n$, as shown in (13a). The workings of this function are similar to the transition function S^M defined in (7a), leaving out the exogenous information $\boldsymbol{W}_{t+1}^n$. Remind that the decision $\boldsymbol{x}_t^n$ is restricted to released freights, thus this only appears when $r = 0$ in $S^{M,x}$.

$$\boldsymbol{S}_t^{n,x} = S^{M,x}\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right),\ \forall t \in \mathcal{T} \tag{13a}$$

where

$$F_{t,d,0,k}^{n,x} = F_{t,d,0,k+1}^n - x_{t,d,k+1}^n + F_{t,d,1,k}^n, \tag{13b}$$

$$G_{t,d,0,k}^{n,x} = G_{t,d,0,k+1}^n - x_{t,d,k+1}^n + G_{t,d,1,k}^n, \tag{13c}$$

$$\forall d \in \mathcal{D},\ \text{and}\ k \in \mathcal{K} | k < K^{max}\ .$$

$$F_{t,d,0,K^{max}}^{n,x} = F_{t,d,1,K^{max}}^n, \tag{13d}$$

$$G_{t,d,0,K^{max}}^{n,x} = G_{t,d,1,K^{max}}^n, \tag{13e}$$

$$\forall d \in \mathcal{D}\ .$$

$$F_{t,d,r,k}^{n,x} = F_{t,d,r+1,k}^n, \tag{13f}$$

$$G_{t,d,r,k}^{n,x} = G_{t,d,r+1,k}^n, \tag{13g}$$

$$\forall d \in \mathcal{D},\ \forall r \in \mathcal{R} | r \geq 1,\ \text{and}\ k \in \mathcal{K}. \tag{13h}$$

$$\text{and}\ k \in \mathcal{K}\ .$$

$$F_{t,d,R^{max},k}^{n,x} = 0, \tag{13i}$$

$$G_{t,d,R^{max},k}^{n,x} = 0, \tag{13j}$$

$$\forall d \in \mathcal{D},\ \text{and}\ k \in \mathcal{K}\ .$$

To tackle the third dimensionality issue (i.e., the enormous set of all possible states $\mathcal{S}$), we use the algorithmic manipulation of "forward dynamic programming". In contrast to backward dynamic programming, forward dynamic programming starts at the first stage (rather than the last one) and, at each stage, solves a forward optimality equation for only one state, as seen in (14).

$$\begin{aligned}\hat{v}_t^n &= \min_{\boldsymbol{x}_t^n}\left(C\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right) + \bar{V}_t^{n-1}\left(\boldsymbol{S}_t^{n,x}\right)\right) \\ &= \min_{\boldsymbol{x}_t^n}\left(C\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right) + \bar{V}_t^{n-1}\left(S^{M,x}\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right)\right)\right)\end{aligned} \tag{14}$$

This forward optimality equation follows the same reasoning as the Bellman's equation from (5), with two differences: (i) the next-stage costs are approximated, and (ii) each feasible decision $\boldsymbol{x}_t^n$ has only one corresponding post-decision state (avoiding all possible realizations of exogenous information). Remind that feasible decisions depend on the state at hand, as defined in (2a) to (2f). In Algorithm 1, forward dynamic programming takes places from line 5 to line 11.

Besides solving the forward optimal equations in line 6 in Algorithm 1, the optimal decision and optimal post-decision states are stored within the forward dynamic programming method, as shown in line 7 and line 8, respectively. This is done with the goal of improving the approximation, as will be explained later in Section 4.2. Before stepping forward from stage t to $t+1$, a random realization $\boldsymbol{W}_{t+1}^n$ from the set of exogenous information Ω is obtained, as seen in line 9. This is done in a Monte Carlo simulation of all random variables. Using all of these aspects, the algorithm steps forward in time, to the next state, using the transition mechanism S^M, defined in (7a), as seen in line 10.

Certainly, the simulation of exogenous information introduces variability in the forward dynamic programming steps. However, this variability represents the inherent stochasticity of demand in the problem and can be used to improve the approximation of the expected costs. For this reason, the aforementioned steps in forward dynamic programming are repeated a total of N times (i.e., number of iterations), as seen in line 3 of Algorithm 1. In each iteration n, the algorithm begins with the same initial conditions (i.e., state), as seen in line 4, then forward dynamic programming is performed, and at last, the approximated next-stage cost $\bar{V}_t^n(\boldsymbol{S}_t^{n,x})$ is updated (improved) retrospectively, as seen in lines 12

through 14. In the following section, we explain how the approximation works and how it is updated in every iteration.

4.2 Basis Functions and Non-stationary Least-Squares Update Function

The approximated next-stage cost $\bar{V}_t^n(\boldsymbol{S}_t^{n,x})$ represents the future costs (after stage t) estimated at iteration n. There are two challenging questions to define the approximated next-stage cost: (i) how to build a good approximation based on a post-decision state, and (ii) how to update the approximation with every iteration. For the first challenge, we use a "basis functions" approach. Basis functions are quantitative characteristics, or features, of a post-decision state, which explain, to some extent, the next-stage costs. Examples of basis functions are the sum of all freights in a post-decision state, the number of destinations of the released-freights, the product of two post-decision freight variables, etc. We denote a basis function as $\phi_a(\boldsymbol{S}_t^{n,x})$, where a belongs to the set of all basis functions or features $\mathcal{A}$. The approximated next-stage cost of a post-decision stage $\bar{V}_t^n(\boldsymbol{S}_t^{n,x})$ is a weighted sum of all basis functions, as shown in (15), where $\theta_a \in \mathbb{R}$ is the weight of each basis function $a \in \mathcal{A}$,. Defining the right set of basis functions, requires both creativity and careful analysis through experimentation, see Section 5.

$$\bar{V}_t^n(\boldsymbol{S}_t^{n,x}) = \sum_{a \in \mathcal{A}} (\theta_a \cdot \phi_a(\boldsymbol{S}_t^{n,x})) \tag{15}$$

For the second challenge, an updating step after every iteration is necessary. Since in every iteration a Monte Carlo simulation of the exogenous information is performed, the costs of the newly seen state can be used to improve our knowledge of the next-stage costs. However, due to the post-decision relation to stages in the horizon, the approximated next-stage cost has to be updated retrospectively (i.e., the current costs are not used to update the current post-decision state, but the previous one). We define a function U^V to denote the process that updates the approximated costs $\bar{V}_{t-1}^n(\boldsymbol{S}_{t-1}^{n,x})$ at iteration n, using (i) the approximated costs, from the previous iteration, of the post-decision state of the current iteration and of the previous stage $\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x})$, (ii) the post-decision state of the current iteration and the previous stage itself ($\boldsymbol{S}_{t-1}^{n,x}$), and (iii) the solution to the forward optimality equation $\hat{v}_t^n$ corresponding to the current iteration and of the current stage, as seen in (16).

$$\bar{V}_{t-1}^n(\boldsymbol{S}_{t-1}^{n,x}) = U^V(\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x}), \boldsymbol{S}_{t-1}^{n,x}, \hat{v}_t^n) \tag{16}$$

The logic behind the retrospective update is that, at stage t of iteration n, the system has moved from stage $t-1$ to stage t with a realization of the exogenous information (via the Monte Carlo simulation). As a result of this, the system is now in a state $\boldsymbol{S}_t^n$, which has optimal costs $\hat{v}_t^n$. These costs are the "realized" next-stage costs of the previous-stage post-decision state that the algorithm had estimated in the previous iteration, i.e., $\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x})$. In other words, the approximated next-stage cost that was calculated at the previous stage $t-1$ (using the previous iteration $n-1$ estimate) has now been observed at stage t in iteration n. The update takes places in a "double-pass" way (lines 12 to 14 in Algorithm 1). For more information on double-pass, see Powell [2007].

To apply the update function to our approximated next-stage costs, we need to modify the weights of each basis function at each iteration. This use of basis functions and weights in the approximated next-stage costs is comparable to a linear regression of costs as a function of all basis functions (features). Usually, linear regression models aim to reduce the squared value of some performance indicator (e.g., residuals, percentage, etc.) using a number of "observations". However, in our ADP algorithm, observations are generated each iteration and are not all known at once. A suitable updating mechanisms for sequential observations is the non-stationary least square method [Powell, 2007], as seen in (17).

$$\theta_a^n = \theta_a^{n-1} - (G^n)^{-1}\phi_a(\boldsymbol{S}_t^{n,x})\left(\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x}) - \hat{v}_t^n\right) \tag{17}$$

Broadly speaking, the non-stationary least squares method updates the approximated next-stage costs based on the observed error $\left(\bar{V}_{t-1}^{n-1}(\boldsymbol{S}_{t-1}^{n,x}) - \hat{v}_t^n\right)$ and the basis function value $\phi_a(\boldsymbol{S}_t^{n,x})$ itself. For example, if the basis function at some stage in some iteration has a value of zero (i.e., a state does not have that feature), then the weight of this basis function will not be updated (i.e., it will remain the same as in the previous iteration). However, if the basis function has a large value, the observed error will ensure it gets updated in the right direction. The matrix $(G^n)^{-1}$ makes sure all weights are updated with a magnitude that minimizes the squared errors in the non-stationary data.

4.3 Single-stage Decision Problem

The remaining dimensionality curse we tackle in our ADP algorithm is the one of finding the optimal action for a single stage. In the instances we consider, this dimensionality issue is relatively small. Nevertheless, for states with a large number of different freights, enumerating all possible decisions might be computationally difficult. For this reason, we developed a mixed-integer linear program (MILP) for the single-stage decision problem, as seen in (18a-18w).

$$\min C\left(\boldsymbol{S}_t^n, \boldsymbol{x}_t^n\right) = \sum_{\mathcal{D}' \subseteq \mathcal{D}} \left(C_{\mathcal{D}'} \cdot w_{t,\mathcal{D}'}\right) + \sum_{d \in \mathcal{D}} \left(A_d \cdot z_{t,d}\right) + \sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} \left(B_d \cdot \left(x_{t,d,k}^F + x_{t,d,k}^G\right)\right) + \sum_{a \in \mathcal{A}} \left(\theta_a \cdot \phi_a(\boldsymbol{S}_t^{n,x})\right) \tag{18a}$$

s.t.

$$\sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} x_{t,d,k}^F \leq Q \tag{18b}$$

$$\sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} x_{t,d,k}^G \leq Q \tag{18c}$$

$$x_{t,d,0}^F + x_{t,d,0}^G + z_{t,d} = F_{t,d,0,0}^n + G_{t,d,0,0}^n, \ \forall d \in \mathcal{D} \tag{18d}$$

$$\sum_{k \in \mathcal{K}} x_{t,d,k}^F - \sum_{k \in \mathcal{K}} \left(F_{t,d,0,k}\right) \cdot y_d \leq 0, \ \forall d \in \mathcal{D} \tag{18e}$$

$$\sum_{k \in \mathcal{K}} x_{t,d,k}^G - \sum_{k \in \mathcal{K}} \left(G_{t,d,0,k}\right) \cdot y_d \leq 0, \ \forall d \in \mathcal{D} \tag{18f}$$

$$x_{t,d,k+1}^F + F_{t+1,d,0,k}^{n,x} = F_{t,d,0,k+1}^n + F_{t,d,1,k}^n, \ \forall d \in \mathcal{D}, k \in \mathcal{K} | k < K^{max} \tag{18g}$$

$$x_{t,d,k+1}^G + G_{t+1,d,0,k}^{n,x} = G_{t,d,0,k+1}^n + G_{t,d,1,k}^n, \ \forall d \in \mathcal{D}, k \in \mathcal{K} | k < K^{max} \tag{18h}$$

$$F_{t+1,d,0,K^{max}}^{n,x} = F_{t,d,1,K^{max}}^n \ \forall d \in \mathcal{D} \tag{18i}$$

$$G_{t+1,d,0,K^{max}}^{n,x} = G_{t,d,1,K^{max}}^n \ \forall d \in \mathcal{D} \tag{18j}$$

$$F_{t+1,d,r,k}^{n,x} = F_{t,d,r+1,k}^n \ \forall d \in \mathcal{D}, k \in \mathcal{K}, r \in \mathcal{R} | r < R^{max} \tag{18k}$$

$$G_{t+1,d,r,k}^{n,x} = G_{t,d,r+1,k}^n \ \forall d \in \mathcal{D}, k \in \mathcal{K}, r \in \mathcal{R} | r < R^{max} \tag{18l}$$

$$w_{t,\mathcal{D}'} - y_{t,d'} \leq 0, \ \forall \mathcal{D}' \subseteq \mathcal{D}, d' \in \mathcal{D}' \tag{18m}$$

$$w_{t,\mathcal{D}'} + y_{t,d'} \leq 1, \ \forall \mathcal{D}' \subseteq \mathcal{D}, d' \in \mathcal{D} \setminus \mathcal{D}' \tag{18n}$$

$$w_{t,\mathcal{D}'} - \sum_{d' \in \mathcal{D}'} y_{t,d'} + \sum_{d'' \in \mathcal{D}' \setminus \mathcal{D}} y_{t,d''} \geq 1 - \left|\mathcal{D}'\right|, \ \forall \mathcal{D}' \subseteq \mathcal{D} \tag{18o}$$

$$\sum_{\mathcal{D}' \subseteq \mathcal{D}} w_{t,\mathcal{D}'} = 1 \tag{18p}$$

$$x_{t,d,k}^F \in \mathbb{Z} \cap \left[0, F_{t,d,0,k}^n\right], \ \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{18q}$$

$$x_{t,d,k}^G \in \mathbb{Z} \cap \left[0, F_{t,d,0,k}^n\right], \ \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{18r}$$

$$y_{t,d} \in \{0,1\}, \ \forall d \in \mathcal{D} \tag{18s}$$

$$z_{t,d} \in \left[0, F_{t,d,0,0}^n + G_{t,d,0,0}^n\right], \ \forall d \in \mathcal{D} \tag{18t}$$

$$w_{t,\mathcal{D}'} \in [0,1], \ \forall \mathcal{D}' \subseteq \mathcal{D} \tag{18u}$$

$$F_{t+1,d,0,k}^{n,x} \in \left[0, F_{t,d,0,k+1}^n + F_{t,d,1,k}^n\right], \ \forall d \in \mathcal{D}, k \in \mathcal{K} | k < K^{max} \tag{18v}$$

$$G_{t+1,d,0,k}^{n,x} \in \left[0, G_{t,d,0,k+1}^n + G_{t,d,1,k}^n\right], \ \forall d \in \mathcal{D}, k \in \mathcal{K} | k < K^{max} \tag{18w}$$

The objective in (18a) is to minimize the sum of (i) a linearized version of the direct costs of a decision, as shown in (3a), and (ii) the approximated next-stage cost with the basis functions, as shown in (15). Constraints (18b) to (18f) define the feasible decision spaces and the auxiliary variables. Constraints (18g) to (18l) define the post-decision freight variables. Constraints (18m) to (18p) make the objective function linear through the use of a binary variable $w_{t,\mathcal{D}'}$ that gets a value of 1 if the subset of destinations $\mathcal{D}'$ is visited with the long-haul vehicle and 0 otherwise. Constraints (18q) to (18w) define the domain of all variables in the MILP model. Note that not all post-decision freight variables are included in the model, only the ones that are modified by the decision. Furthermore, the basis functions $\phi_a(\boldsymbol{S}_t^{n,x})$ for all $a \in \mathcal{A}$ are assumed to be linear in the decision variables of the MILP model. For examples on how to incorporate basis functions in the MILP, see A.

5 Numerical Experiments

In this section, we analyze the performance of our ADP algorithm under various transportation settings. The settings are based on the operations of the Dutch LSP that participates in this research. The analysis consists of two phases. In the first phase, we test how well the ADP policy approximates the solution of the Markov model (i.e., the optimal solution). In the second phase, we evaluate the performance of the ADP policy compared to a benchmark policy. With these typification of experiments, we are able to test the theoretical and the practical relevance of our approach, respectively. The section is divided as follows. First, we present our experimental setup. For each experiment, we present in detail the means (e.g., input parameters, algorithm settings, etc.) and the goals (e.g., research questions, hypothesis, etc.) we want to achieve with it. Second, we analyze the results of our experiments from the two phases. Finally, we summarize the principal findings and provide a discussion on the benefits and the shortcomings of our approach.

5.1 Experimental Setup

In the first phase of our experimental setup, we study the approximation quality of the ADP policy. Our goal in these experiments is to find the set of features and algorithm settings that result in a proper approximation. To measure approximation quality, we need to know the optimal expected costs, which can be obtained by solving the MDP model the algorithm is based upon. Naturally, this is only possible for small instances of the problem. Remind that an instance consists of all random variables (and their distribution), the cost structure of all subsets of destinations, the planning horizon, and the long-haul vehicle capacity. We create two small-size instances: I_1 and I_2, considering the smallest number of problem characteristics that are still somewhat representative of the Dutch LSP operations. The main input parameters for these two instances are shown in Table 2.

The two instances for the first phase of experiments differ in the definition of the random variables. Instance I_1 represents a "balanced" network, where the stochasticity in delivery freight characteristics is the same as the one in pickup freights, and Instance I_2 represents an "unbalanced" network. By testing these two opposite instances, we aim to define an approximation that is robust to the distribution of the random variables. Although the definition of the random variables differs in I_1 and I_2, the cost setup is the same in both cases. This cost setup is based upon three cost considerations. First, there are costs only if the long-haul (i.e., high capacity) vehicle departs or the alternative transport mode is used. Second, the long-haul vehicle costs depend predominantly on the subset of destinations visited. This means that the costs for delivering (or picking up) an additional container at a terminal already scheduled to be visited are small compared to visiting an additional destination. Third, using the alternative transport mode, for the same number of freights that a long-haul vehicle can carry, is much more expensive than using the long-haul vehicle. This reflects the economies of scales achieved through consolidation in high-capacity vehicles. In Table 2, we show the ranges of all costs involved. Remind that the long-haul vehicle has cost $C_{\mathcal{D}'}$ if it visits the group of destinations $\mathcal{D}' \subseteq \mathcal{D}$ in a round-trip (independent of the number of freights consolidated), and cost B_d per freight with destination d consolidated for delivery or pickup. The use of the alternative mode has a cost of A_d per freight.

Defining a set of features that properly capture future costs within an ADP algorithm is both a science and an art. With scientific approaches such as factor analysis and regression analysis, one can test how "good" a feature is. However, defining the right set of features requires creativity about potential causes of future costs. We build three different sets of features based on a common "job" description used in transportation settings: *MustGo*, *MayGo*, and *Future* freights. In this case, freights refer to either delivery or pickup freights. In our case, MustGo freights are those released freights whose due-day is immediate. MayGo freights are those released freights whose due-day is not immediate. Future freights are those that have not yet been released. We use the MustGo, MayGo and Future adjectives in destinations as well, with an analogous meaning to those of freight. In Table 3 we show the three sets of features, which we name Value Function Approximation (VFA) 1, 2, and 3. All feature types in this table are related to the freights of a post-decision state. The symbol "•" denotes a VFA set containing a feature type. All feature types are numerical, and either indicate (i.e., 1 if yes, 0 if no), count (1,2,...), number (add), or multiply (i.e., product between two numbers) the different type of freights and destinations. Between parentheses we show the number of basis functions (i.e., independent variables) that a feature type has for I_1 and I_2. For example, there is one post-decision state variable per destination, per time-window length, both for the delivery and the pickup, thus all post-decision state variables are 3*3*2=18.

Now that we have defined the instances and the candidate VFAs, we explain our three-step methodology to choose the best VFA and the algorithm parameters (e.g., number of iterations, step size, etc.).

Table 2: Input parameters of the numerical experiments

Description	Parameter	Instance							
		I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
Destinations	$\mathcal{D}$	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,2,\ldots,12\}$	$\{1,2,\ldots,12\}$	$\{1,2,\ldots,12\}$	$\{1,2,\ldots,12\}$	$\{1,2,\ldots,12\}$	$\{1,2,\ldots,12\}$
Destination probability	p_d^{DF}	$\{\frac{1}{10},\frac{8}{10},\frac{1}{10}\}$	$\{\frac{1}{3},\frac{1}{3},\frac{1}{3}\}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$
	p_d^{DG}	$\{\frac{1}{10},\frac{8}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{8}{10},\frac{1}{10}\}$	$\approx \frac{6^d}{d!}e^{-6}$	$\frac{1}{12},\ \forall d \in \mathcal{D}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$	$\approx \frac{6^d}{d!}e^{-6}$
Release-days	$\mathcal{R}$	$\{0\}$	$\{0\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$
Release-day probability	p_r^{RF}	$\{1\}$	$\{1\}$	$\{\frac{3}{10},\frac{3}{10},\frac{4}{10}\}$	$\{\frac{3}{10},\frac{3}{10},\frac{4}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$
	p_r^{RG}	$\{1\}$	$\{1\}$	$\{\frac{3}{10},\frac{3}{10},\frac{4}{10}\}$	$\{\frac{3}{10},\frac{3}{10},\frac{4}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$
Time-window lengths	$\mathcal{K}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$
Time-window length probability	p_k^{KF}	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{1}{3},\frac{1}{3},\frac{1}{3}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$
	p_k^{KG}	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{2}{10},\frac{3}{10},\frac{5}{10}\}$	$\{\frac{8}{10},\frac{1}{10},\frac{1}{10}\}$	$\{\frac{1}{10},\frac{1}{10},\frac{8}{10}\}$
Freights	$\mathcal{F}=\mathcal{G}$	$\{1\}$	$\{1\}$	$\{1,2,\ldots,10\}$	$\{1,2,\ldots,10\}$	$\{1,2,\ldots,10\}$	$\{1,2,\ldots,10\}$	$\{1,2,\ldots,10\}$	$\{1,2,\ldots,10\}$
Freight probability	p_f^F	$\{1\}$	$\{1\}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$
	p_f^G	$\{1\}$	$\{1\}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{2^f}{f!}e^{-2}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$	$\approx \frac{4^f}{f!}e^{-4}$
Number of states	$\lvert\mathcal{S}\rvert$	19,321	19,321	$\gg 7.96\cdot 10^{27}$	$\gg 7.96\cdot 10^{27}$	$\gg 7.96\cdot 10^{27}$	$\gg 7.96\cdot 10^{27}$	$\gg 7.96\cdot 10^{27}$	$\gg 7.96\cdot 10^{27}$
Time horizon	T^{max}	5	5	5	5	5	5	5	5
Long-haul capacity	Q	2	2	10	10	10	10	10	10
Range cost long-haul vehicle	$C_{\mathcal{D}'}$	$[250,1000]$	$[250,1000]$	$[250,2150]$	$[250,2150]$	$[250,2150]$	$[250,2150]$	$[250,2150]$	$[250,2150]$
Range cost individual freight	B_d	0	0	$[50,150]$	$[50,150]$	$[50,150]$	$[50,150]$	$[50,150]$	$[50,150]$
Range cost alternative vehicle	A_d	$[500,1000]$	$[500,1000]$	$[300,800]$	$[300,800]$	$[300,800]$	$[300,800]$	$[300,800]$	$[300,800]$

Table 3: Various sets of features (basis functions of a post-decision state)

Feature type	VFA 1	VFA 2	VFA 3
All post-decision state variables (18)	•	•	•
All post-decision state variables squared (18)	•	-	-
Count of MustGo destinations (1)	•	•	•
Number of MustGo freights (1)	•	•	•
Product of MustGo destinations and MustGo freights (1)	•	-	-
Count of MayGo destinations (1)	•	•	•
Number of MayGo freights (1)	•	•	•
Product of MayGo destinations and MayGo freights (1)	•	-	-
Count of Future destinations (1)	•	•	•
Number of Future freights (1)	•	•	•
Product of Future destinations and Future freights (1)	•	-	-
Indicator MustGo freights per destination (3)	-	•	-
Indicator MayGo freights per destination (3)	-	•	-
Indicator Future freights per destination (3)	-	•	-
Number of all freights (1)	•	•	•
Constant (1)	•	•	•

First, we apply the ADP algorithm to each state of each instance, using each VFA. Remind that an instance represents a transportation network with many possible states, as seen in Table 2. We use the algorithm settings recommended in the literature [Pérez Rivera and Mes, 2015, Powell, 2007]. Second, we test the resulting ADP policy (of each state, per instance and VFA combination) in a simulation of 500 replications of the time horizon. For each state, we compare the resulting average costs of the simulation of the ADP policy with the optimal expected costs. Finally, we calculate the average difference between each VFA (for all states in each instance) and the optimal expected costs, and decide upon the best VFA in both instances. With the chosen VFA, we further tune the algorithm parameters that we use in the second phase of our numerical experiments.

The second phase evaluates the cost-reduction capabilities of our approach. Our goal in these experiments is to determine the performance of the ADP policy compared to a competing policy in realistic transportation networks. Moreover, we assess the differences in performance of our approach under different network settings (in a sensitivity analysis fashion) and under different "states", or day-to-day situations within a network. The competing policy we use as a benchmark is one commonly used in practice: solve the transportation problem to optimality for the current state. In other words, the policy is to look at all freights known (both released and not released) and transport the ones for which the minimum transportation costs are achieved for that day. To test the cost-reduction capabilities, we create six realistic, normal-sized, instances: I_3 to I_8, as seen in Table 2.

The instances considered in the second phase differ with respect to the distribution of the random variables (i.e., probabilities of the freights, of the destinations, and of the time-window parameters). The rationale behind building different instances is to test the "balance" of a network (i.e., delivery and pickup freight characteristics are the same or different), the "in-advance" information (i.e., freights are known long, or shortly, before they are released), and the "urgency" of freight (i.e., freights must be carried the same day they are announced or some days later). In this rationale, Instance I_3 represents a totally balanced network, while Instance I_4 represents a totally unbalanced network. Instances I_5 through I_8 represent balanced networks with different time-window characteristics. Freights are mostly released "immediately" in Instance I_5 (no in-advance information), whereas in Instance I_6 they are mostly released the "in the future" (substantial in-advance information). In Instance I_7, freights are mostly urgent (i.e., immediate release and due-day), whereas in Instance I_8 freights are mostly non-urgent (i.e., future release and due day). All instances have the same cost setup, which follows the same logic as in the first phase of experiments.

The instances of the second phase are much larger than the ones of the first phase. In each instance, a lower bound on the number of states is $7.96 \cdot 10^{27}$ states. For this reason, we cannot evaluate the performance for all initial states as we did in the first phase; we need to consider smaller subset of the state space. However, choosing a subset of states has two complications: (i) results might hold only for the chosen sample of states, and (ii) states have different characteristics that influence the performance of the ADP policy [Pérez Rivera and Mes, 2015]. To measure different regions of the state space in a representative way, we build a subset of states as follows. First, we generate a random sample of 10,000 states for each network I_3 to I_8, which are "commonly encountered" in an LSP having such a network. To achieve a commonly encountered state, we simulate the day-to-day operations of the LSP (i.e., the

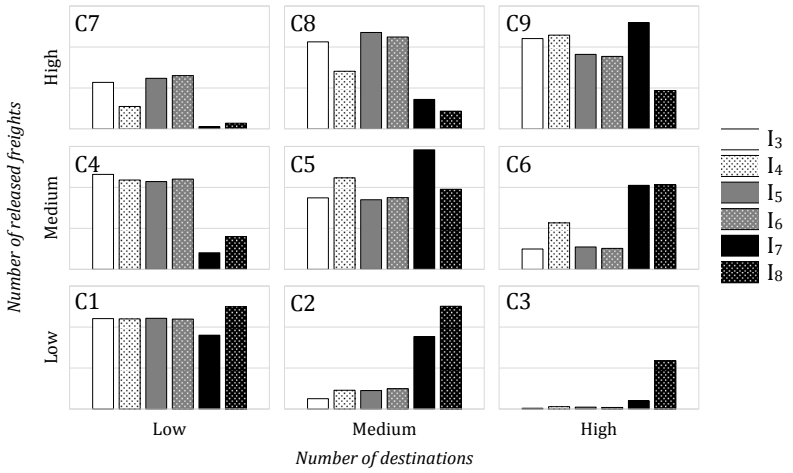

Figure 1: Categorization of the 10,000 sampled states of the Instances I_3 to I_8

benchmark policy)l We begin with a random initial state, and each day of a horizon of a week, we simulate the decisions and arrival of new freights. After the simulated week, we stop the simulation and store the state in which the system is at that moment. Naturally, the resulting sample contains various types of states representing various regions of the entire state space. To categorize these regions, we use two state-characteristics that are of great importance in practice: the number of released freights, and the number of different destinations of the released freights. We define three levels in each characteristic: Low, Medium, and High, and categorize the 10,000 states of each instance into nine different categories C1 to C9, as seen in Figure 1. Since the distribution of state characteristics differ per instance, we define the boundary of the levels (e.g., how many freights is "low", how many destinations is "high"), individually for each instance as seen in B, Table 7. These boundaries are computed with the objective of minimizing the differences between the category with the largest number of states and the one with the lowest number of states.

Now that we have defined the instances and the categorization of states within them, we explain our methodology to determine the performance and to compare the ADP policy to the competing policy. In a similar way to the first phase of the numerical experiments, we perform three steps in the large instances. First, we choose one state per category, for each instance. The state is chosen randomly among those states close to the center of the cluster of states from the given category (center in terms of the two aforementioned characteristics for each category). Second, we apply the ADP algorithm to each of these states, using the best VFA and the best settings from the first phase of experiments for 500 iterations. Remind that applying the ADP algorithm means learning the weights for the linear combination of the basis functions. These weights represent the future costs, which are used as a decisions rule or policy. In the last step, we simulate the ADP policy and the benchmark heuristic using common random numbers. This allows us to do a pairwise analysis of differences among the two policies.

5.2 Results

We now present the results of the two phases of our numerical experiments separately. Remind that ADP policy performance is the result of a simulation of 500 replications of the weights (decision rule) learned by the ADP algorithm in 500 iterations.

5.2.1 Experiments of Phase I

In the first phase, we show how the different sets of features (i.e., VFAs in Table 3) perform. We measure performance as the difference between (i) the average costs of the ADP policy (obtained in a simulation of the policy obtained with the ADP algorithm), and (ii) the optimal expected costs (obtained by solving the MDP). In Table 4, we show the average performance over all states in the state space. Furthermore, we show the coefficient of determination (R^2) of a linear regression of the sets of features over the optimal expected costs. This linear regression is "a-posteriori" with the total expected costs for the horizon, while

the ADP regression is "a-priori" with the future downstream costs at each day of the horizon.

Table 4: Performance of the different VFAs in instances I_1 and I_2

Instance	VFA 1		VFA 2		VFA 3	
	R^2	Diff.	R^2	Diff.	R^2	Diff.
I_1	0.63	5.6%	0.69	5.9%	0.55	5.6%
I_2	0.64	6.6%	0.68	7.7%	0.55	6.8%
I_1-delivery	0.89	16%	0.89	14%	0.89	8%
I_2-delivery	0.89	8%	0.90	7%	0.90	7%

In Table 4, we show two additional instances: I_1-delivery and I_2-delivery. These instances correspond to the delivery part of Instances I_1 and I_2, respectively, in a similar problem to the one presented by Pérez Rivera and Mes [2015]. We observe that the delivery-only instances have a better a-posteriori fit (higher R^2) than the round-trip ones, but have a worse performance (higher difference between optimal costs and ADP policy costs). This result shows that, although the use of features in our ADP algorithm is related to a linear regression of costs, performing a-posteriori linear regression to define the best set of features might not yield the desired result. For example, in I_1, we would have chosen VFA 2 due to its highest R^2, while VFA 3 with a lower R^2 performs better than VFA 2. In addition, we observe that VFA 2 has a slightly worse performance compared to VFA 1 and VFA 3 in the round-trip instances, even though it has more detailed features and one would expect it to better capture costs. Again, these results show that adding more variables to capture costs does not necessarily lead to a better approximation of the optimal costs. We discuss these, and other challenges of designing an accurate ADP algorithm in Section 5.3.

With respect to defining the best VFA, we notice that VFA 1 and VFA 3 perform the best in I_1 and I_2. We choose VFA 3 as the best set of features for two reasons: (i) it only contains linear features, which make it directly applicable to the ILP for the single-stage decision, and (ii) it contains the least number of features, improving the computational time of the updating function within the ADP algorithm.

Before going to the second phase of the experiments, we dive into more detail on the performance of VFA 3 for all states in the state space of I_2 in Figure 2. Two aspects of this figure are noteworthy: (i) the ADP algorithm always over-estimates the optimal expected costs, and never under-estimates them, and (ii) the distribution of the percentage difference over all states has a long tail. Always over-estimating the optimal costs is not necessarily an issue as long as the relative difference between two states remains the same. However, the long tail of the differences suggests that some states are more difficult to estimate than others, and thus the relative difference between values of two states might be different with the ADP estimates compared to the optimal values. We elaborate more on this challenge of estimating different states within the same instance in Section 5.3.

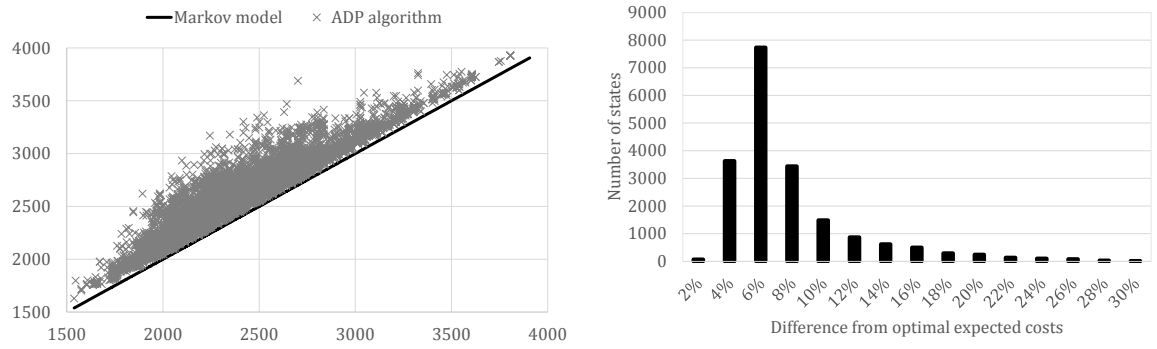

Figure 2: Accuracy of VFA 3 in Instance I_2

5.2.2 Experiments of Phase II

In the second phase of experiments, we show the performance of the ADP algorithm on I_3 through I_8, using VFA 3. We measure performance in these experiments as the pairwise difference between the simulated costs of using the ADP-policy (which results from first applying the ADP algorithm) and the simulated costs of using the competing policy mentioned in Section 5.1. By pairwise difference we mean the use of common random numbers in the replications of the simulations, to rule out variability in the network as a cause for differences. The results for each category of each instance can be seen in Table 5. In this table, the average differences between the costs from the ADP policy and the costs from the competing policy are given (as a percentage). A negative percentage in this table can be interpreted as how much lower the ADP policy costs are compared to the competing policy costs (i.e., savings).

Table 5: Average cost difference between the ADP policy and the competing policy

Category	I_3	I_4	I_5	I_6	I_7	I_8	Average
C1	-5.9%	-8.6%	-9.4%	-5.5%	-0.6%	-5.2%	-5.9%
C2	-9.1%	-12.3%	-4.0%	-2.7%	-0.6%	-11.0%	-6.6%
C3	-1.9%	-6.7%	-8.2%	-3.1%	1.1%	-7.2%	-4.3%
C4	-14.9%	-25.5%	-5.2%	-11.8%	-1.5%	-8.0%	-11.2%
C5	-15.1%	-1.5%	-9.7%	-25.9%	-0.4%	-9.7%	-10.4%
C6	1.3%	-4.5%	-3.8%	-10.6%	-2.0%	-7.8%	-4.6%
C7	-4.4%	-3.7%	-24.2%	-0.1%	-11.0%	-7.3%	-8.4%
C8	-2.3%	-16.7%	-2.1%	-7.1%	-0.6%	-3.3%	-5.3%
C9	-0.9%	2.3%	-4.4%	-11.0%	4.7%	-7.6%	-2.8%
Average	-5.9%	-8.6%	-7.9%	-8.6%	-1.2%	-7.5%	-6.6%
Weighted Average	-7.0%	-8.7%	-7.6%	-10.1%	0.3%	-8.0%	-6.9%

On average, the ADP policy achieves 6.6% savings over all categories and instances, as seen in Table 5. To begin our analysis, we focus on the different categories (rows in Table 5). The savings range from 2.8% to 11.2%. However, when we look into the performance of a category in the various instances, two observations stand out. First, in C4, the ADP policy achieves the largest savings of all categories in all instances (25.5% in I_4), as well as the largest average savings among all instances (11.2%). Second, in C9, the ADP policy achieves the worst performance of all categories in all instances (4.7% higher costs than the competing policy in I_7), as well as the smallest average savings among all instances (2.8%). In these experiments we show that, although savings are achieved in all categories on average over the test instances, a careful analysis most be done for some of the categories of the states that the system can be in. We elaborate more on this difference in performance among categories in Section 5.3.

Focusing on the instances (columns in Table 5), we observe that, on average, savings larger than 5% are achieved in all instances except I_7. This is more noticeable when we weight the average with the percentage of states from the 10,000 random states that belong to each category (see Figure 1). In this weighted case, the ADP policy costs in I_7 are even 0.3% higher than the costs of using the competing policy. Furthermore, we observe that in I_7, the ADP policy generates higher costs than the competing policy in two categories (C3 and C9). In I_3 and I_4, the ADP policy generates higher costs than the competing policy in only one category (C6 and C9, respectively).

To study more into detail the pairwise differences, we calculate the confidence intervals for all differences, as seen in Appendix C. In most instances and categories (except the ones mentioned before, I_6-C7, I_7-C1, and I_7-C2), significant savings are achieved. In addition to the two categories of I_7 that had no savings on average (C3 and C9), we have two categories were the savings are not significant (C1 and C2). We further elaborate on why the ADP policy under-performs in these cases, when compared to the others, in Section 5.3.

5.3 Discussion

Before applying our approach in practice, several input parameters must be defined. First, all random variables in our model are defined as empirical distributions. Such empirical distributions can be obtained from historical data on freights and their characteristics. Furthermore, the "finite" nature of our model is in line with the constraints that hold in practice (e.g., size of the container yard, maximum number

of days a leased container can be used, etc.), and thus presents no extra difficulty to define. Second, the costs of a long-haul vehicle are modeled to depend on the subset of destinations visited rather than the route (sequence) in which they are visited. The reasoning of this choice is due to the destination characteristics of our problem. The region that is visited periodically (e.g., terminals in a port), contains only a small number of destinations. Thus, the optimal route for every combination of destinations can be calculated beforehand (i.e., solving a traveling salesman problem) and used once the freights are selected. In addition, differences in waiting time, CO_2 emissions, or other indicators of terminals, can be easily incorporated with the full definition of all subsets of destinations. In case there are many destinations in the region of the round-trip, then indeed the costs can be troubling to compute. In the case of the Dutch LSP considered, the number of container terminals within the Port of Rotterdam is small enough and nicely structured (grouped in a limited number of port areas), such that the computation of an optimal route for each subset of destinations is done within reasonable time.

The first phase of experiments raised interesting discussion points about accuracy. In these experiments we saw that, besides the creativity required to define "features" of a state, there are two major challenges in constructing an accurate approximation algorithm for our problem. On the one hand, there is a challenge in measuring accuracy with respect to the optimal solution. We can only solve the MDP for small instances in which stochastic time-window interactions of freights have a relatively small effect on the future costs. As Powell [2007] points out, performance might not be as expected if the problem does not benefit from looking into the future. This issue make it hard to determine whether all significant interactions were present when deciding for the best set of features to use. On the other hand, there is the question of which states to test for accuracy. In our case, we tested the entire state space in the first phase of experiments. However, not all states achieved the same accuracy. To know more about the states where a low accuracy was achieved, we used the same categorization procedure of states of the second phase of experiments (see Figure 1), and calculated some descriptive statistics about the performance of the ADP policy in I_2, as seen in Table 6.

Table 6: Statistics about cost-differences in Instance I_2

Category	Freights	Destinations	Min	Max	Average	95^{th} Percentile	States
C1	Low	Low	2.8%	18.2%	7.2%	9.6%	282
C2	Low	Medium	2.5%	39.8%	7.5%	15.5%	2244
C3	Low	High	1.9%	31.5%	7.5%	17.9%	1416
C4	Medium	Low	2.8%	26.4%	6.5%	10.2%	150
C5	Medium	Medium	2.0%	36.2%	6.9%	16.0%	2940
C6	Medium	High	1.8%	33.4%	6.9%	15.8%	4008
C7	High	Low	2.1%	38.3%	5.8%	8.5%	75
C8	High	Medium	1.8%	41.3%	6.7%	17.5%	2550
C9	High	High	1.5%	36.5%	6.4%	15.6%	5656

It is interesting to see that all categories from I_2 have similar average and min-max ranges of differences in Table 6. In this instance, it seems that inaccurate states do not have specific characteristics (at least not in terms of number of freights and number of destinations) that makes them difficult for the ADP to estimate. Furthermore, we see that 95% of all states in each category are below three times the average difference, while the largest 5% is up to six times the average difference. To investigate this 5%, we separated the states were the ADP policy performance was more than 20% away from the optimal one. This separation resulted in a set of 417 states (2.2% of the entire state space). On average, the number of freights (5.19) and the number of destinations (2.52) in this "difficult" subset was approximately the same as the number of freights and number of destinations in the entire state space (5.18 and 2.55 respectively). However, the percentage of urgent freights with respect to all released freights was larger in this subset (38.5%) than for the entire state space (49.2%). It seems that when no direct costs are observed immediately (due to the smaller number of urgent freights or MustGo's), the estimate is worse. This observation is also supported by the fact that all categories with freights in the low level in Table 6 perform the worse on average (i.e., higher costs).

The second phase of experiments also raised interesting discussion points. In contrast to the categories in I_2, performance of the ADP policy in the categories of the normal-sized instances (I_3 through I_8) *differed* among the categories. Remind that performance, in this second phase, is measured as the difference between the ADP policy (with the VFA weights fixed after 500 iterations) and the competing policy. In C4, which has a medium number of freights and a low number of destinations, the ADP policy performed the best. In C9, which has a large number of freights and a large number of destinations, the ADP policy performed the worst. These results are to be expected when one looks at the possible actions in each of

these two categories, with respect to the capacity of the long-haul vehicle. In C4, which has a medium number of freights, it is possible to send a full barge, or a less than full barge. Naturally, these two actions differ in immediate costs, but might differ even more in future costs (due to future consolidation possibilities). The ADP policy seems to capture better the future consolidation possibilities, meaning that although it might perform worse than the competing policy on one day, it performs better in the entire horizon. In C9, which has a large number of freights, the action of using the alternative for many freights is necessary. In this case, the daily costs will be large anyways, meaning that optimizing for each day individually (i.e., competing policy) coincides with the optimal policy.

In the second phase, we also observed that the ADP policy performs better in some instances than others. In I_4 and I_6, the ADP policy performs at least 8% better on average than the competing policy. In I_7, there is almost no difference between the ADP policy and the competing policy. Instance I_4 represents an "unbalanced" network in terms of number of freights and destinations. It seems that, when relations between the number of freights arriving (and their destinations) for delivery and for pickup are more complex, the ADP policy particularly pays off. Instance I_6 is the instance that represents an "in-advance information" network. Most freights in this instance are announced two days before they are released, as seen in Table 2. The ADP policy seems to exploit this information characteristic in the day-to-day decisions (e.g., by postponing freights today for consolidation of known freights that will be released tomorrow or the day after). In contrast to I_6, I_7 represents a "very urgent" network. Freights are mostly due on the day they are released, and they are most released the same day their information is known. This characteristic leaves little to no "in-advance" information, diminishing the additional benefits of the ADP policy. This comparison between instances reveals that when freights must be carried the same day they are announced (i.e., mostly urgent orders every day), myopic optimization policies work just fine. In all other cases, particularly with *unbalanced* networks on which freights are announced *in advance*, significant savings can be achieved with a look-ahead policy such as the one calculated with our proposed ADP algorithm.

6 Conclusions

In this paper, we provided an MDP model for the anticipatory freight selection problem in intermodal long-haul round-trips, and an ADP algorithm to solve it. With our proposed model, we studied the optimal tradeoff between selecting freights for today's round-trip and postponing them for future trips, under stochastic demand and various time-dependencies, for a multi-period horizon. With our proposed solution algorithm, we solved the model for realistic problem instances, and tested how the diverse problem characteristics influence decisions. We performed our experimental tests in two phases, the first one related to the approximation quality of our ADP algorithm, and the second one related to the cost-reduction capabilities of the ADP policy.

In the first phase of experiments, we provided methodological insights on the design decisions related to the ADP approach. We tested several sets of basis functions (i.e., features) for the ADP algorithm, using small sized problem instances, and compared their performance against the optimal solution of the MDP model. Using regression analysis, we noticed that a low coefficient of determination from a set of features does not necessarily results in a poor performance of these features within an ADP algorithm. Furthermore, through a test of overlapping sets of features, we observed that having more features not always results in better performance. Although we noted that some states are more difficult to approximate than others, our chosen set of features achieves costs that are, on average, 5.6% away from the optimal ones.

In the second phase of experiments, we provided managerial insights on the tradeoff between selection and postponement of freights, for several intermodal network settings and state categories (i.e., day-to-day situations). We tested the performance of the ADP policy against a common practice heuristic that optimizes the freights at hand only. We noticed that in networks having a high number of urgent freights (i.e., immediate release and due-day), the benchmark policy seems to be optimal in most state categories, and thus there are no benefits of using the ADP policy. In all other networks settings tested, the use of the ADP policy results in savings between 6.9% and 10.1%, on average over the state-categories, compared to the benchmark policy. More important, we noted that in networks with more information in advance (i.e., most freights are released in the future), the ADP algorithm performs the best, resulting in cost reduction up to 25.5% more than the benchmark policy.

References

J. Andersen, T. G. Crainic, and M. Christiansen. Service network design with asset management: Formulations and comparative analyses. *Transportation Research Part C: Emerging Technologies*, 17(2):197 – 207, 2009a. ISSN 0968-090X. doi: http://dx.doi.org/10.1016/j.trc.2008.10.005. URL http://www.sciencedirect.com/science/article/pii/S0968090X08000867. Selected papers from the Sixth Triennial Symposium on Transportation Analysis (TRISTAN VI).

J. Andersen, T. G. Crainic, and M. Christiansen. Service network design with management and coordination of multiple fleets. *European Journal of Operational Research*, 193(2):377 – 389, 2009b. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2007.10.057. URL http://www.sciencedirect.com/science/article/pii/S0377221707011186.

J. Andersen, M. Christiansen, T. G. Crainic, and R. Gronhaug. Branch and price for service network design with asset management constraints. *Transportation Science*, 45(1):33–49, 2011. doi: 10.1287/trsc.1100.0333. URL http://dx.doi.org/10.1287/trsc.1100.0333.

R. W. Bent and P. V. Hentenryck. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987, 2004. doi: 10.1287/opre.1040.0124. URL http://dx.doi.org/10.1287/opre.1040.0124.

G. Berbeglia, J.-F. Cordeau, and G. Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8 – 15, 2010. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2009.04.024. URL http://www.sciencedirect.com/science/article/pii/S0377221709002999.

T. G. Crainic and K. H. Kim. Chapter 8 Intermodal Transportation. In C. Barnhart and G. Laporte, editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, pages 467 – 537. Elsevier, 2007. doi: http://dx.doi.org/10.1016/S0927-0507(06)14008-6. URL http://www.sciencedirect.com/science/article/pii/S0927050706140086.

T. G. Crainic, M. Gendreau, and J. M. Farvolden. A simplex-based tabu search method for capacitated network design. *INFORMS Journal on Computing*, 12(3):223–236, 2000. doi: 10.1287/ijoc.12.3.223.12638. URL http://dx.doi.org/10.1287/ijoc.12.3.223.12638.

G. Ghiani, E. Manni, A. Quaranta, and C. Triki. Anticipatory algorithms for same-day courier dispatching. *Transportation Research Part E: Logistics and Transportation Review*, 45(1):96 – 106, 2009. ISSN 1366-5545. doi: http://dx.doi.org/10.1016/j.tre.2008.08.003. URL http://www.sciencedirect.com/science/article/pii/S1366554508001142.

A. Hoff, A.-G. Lium, A. Lokketangen, and T. Crainic. A metaheuristic for stochastic service network design. *Journal of Heuristics*, 16(5):653–679, 2010. ISSN 1381-1231. doi: 10.1007/s10732-009-9112-8. URL http://dx.doi.org/10.1007/s10732-009-9112-8.

A.-G. Lium, T. G. Crainic, and S. W. Wallace. A study of demand stochasticity in service network design. *Transportation Science*, 43(2):144–157, 2009. doi: 10.1287/trsc.1090.0265. URL http://dx.doi.org/10.1287/trsc.1090.0265.

L. Moccia, J.-F. Cordeau, G. Laporte, S. Ropke, and M. P. Valentini. Modeling and solving a multimodal transportation problem with flexible-time and scheduled services. *Networks*, 57(1):53–68, 2011. ISSN 1097-0037. doi: 10.1002/net.20383. URL http://dx.doi.org/10.1002/net.20383.

C. Novoa and R. Storer. An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 196(2):509 – 515, 2009. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2008.03.023. URL http://www.sciencedirect.com/science/article/pii/S0377221708003172.

A. Pérez Rivera and M. Mes. Dynamic multi-period freight consolidation. In F. Corman, S. Voß, and R. R. Negenborn, editors, *Computational Logistics*, volume 9335 of *Lecture Notes in Computer Science*, pages 370–385. Springer International Publishing, 2015. ISBN 978-3-319-24263-7. doi: 10.1007/978-3-319-24264-4_26. URL http://dx.doi.org/10.1007/978-3-319-24264-4_26.

V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1 – 11, 2013. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2012.08.015. URL http://www.sciencedirect.com/science/article/pii/S0377221712006388.

W. B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, volume 1. John Wiley & Sons, 2007.

W. B. Powell, B. Bouzaiene-Ayari, and H. P. Simao. Chapter 5 Dynamic Models for Freight Transportation. In C. Barnhart and G. Laporte, editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, pages 285 – 365. Elsevier, 2007. doi: http://dx.doi.org/10.1016/S0927-0507(06)14005-0. URL http://www.sciencedirect.com/science/article/pii/S0927050706140050.

J. Riordan. *Introduction to combinatorial analysis*. Courier Dover Publications, 2002.

H. P. Simao, J. Day, A. P. George, T. Gifford, J. Nienow, and W. B. Powell. An approximate dynamic programming algorithm for large-scale fleet management: A case application. *Transportation Science*, 43(2):178–197, 2009. doi: 10.1287/trsc.1080.0238. URL http://dx.doi.org/10.1287/trsc.1080.0238.

M. SteadieSeifi, N. Dellaert, W. Nuijten, T. V. Woensel, and R. Raoufi. Multimodal freight transportation planning: A literature review. *European Journal of Operational Research*, 233(1):1 – 15, 2014. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2013.06.055. URL http://www.sciencedirect.com/science/article/pii/S0377221713005638.

M. Verma, V. Verter, and N. Zufferey. A bi-objective model for planning and managing rail-truck intermodal transportation of hazardous materials. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):132 – 149, 2012. ISSN 1366-5545. doi: http://dx.doi.org/10.1016/j.tre.2011.06.001. URL http://www.sciencedirect.com/science/article/pii/S1366554511000809. Select Papers from the 19th International Symposium on Transportation and Traffic Theory.

N. Wieberneit. Service network design for freight transportation: a review. *OR Spectrum*, 30(1):77–112, 2008. ISSN 0171-6468. doi: 10.1007/s00291-007-0079-2. URL http://dx.doi.org/10.1007/s00291-007-0079-2.

H. Zolfagharinia and M. Haughton. The benefit of advance load information for truckload carriers. *Transportation Research Part E: Logistics and Transportation Review*, 70:34 – 54, 2014. ISSN 1366-5545. doi: http://dx.doi.org/10.1016/j.tre.2014.06.012. URL http://www.sciencedirect.com/science/article/pii/S1366554514001070.

R. A. Zuidwijk and A. W. Veenstra. The value of information in container transport. *Transportation Science*, 49(3):675–685, 2015. doi: 10.1287/trsc.2014.0518. URL http://dx.doi.org/10.1287/trsc.2014.0518.

A Basis functions in the MILP

To model the various sets of features seen in Table 3, we introduce two additional variables into the MILP shown in (18a) through (18w). The binary variable $u_{t,d}$ gets a value of 1 if destination d has any MustGo freight, and 0 otherwise. The binary variable $v_{t,d}$ gets a value of 1 if destination d has any MayGo freight, and 0 otherwise. To define these variables within the MILP, we introduce constraints (19b) through (19g). To account for the future costs (i.e., weights from the ADP algorithm), we need to construct weights for the MILP decision variables depending on the set of features $\mathcal{A}$ used, as shown in (19a). For example, in VFA 2, we have the feature "Indicator of MustGo freights per destination", whereas in VFA 3 we do not have this feature. In both VFA 2 and VFA 3 we have the feature "Count of MustGo destinations". When using VFA 2, the weight $\theta_d^{u,\mathcal{A}}$ in the MILP, will be the sum of weights corresponding to the feature a "Indicator of MustGo freights per destination" and the feature a' "Count of MustGo destinations" (i.e., $\theta_d^{u,\mathcal{A}} = \theta_a + \theta_{a'}$). When using VFA 3, the weight $\theta_d^{u,\mathcal{A}} = \theta_{a'}$, since this set of features only has the feature a' "Count of MustGo destinations", related to the variable $u_{t,d}$. At the end of (19a), we add the constant δ_t^n, which takes into account the total costs of features that are not dependent on the MILP variables, but only on the state itself (i.e., "Number of future freights", "Constant", etc.).

$$\begin{aligned}\sum_{a\in\mathcal{A}}(\theta_a\cdot\phi_a(\boldsymbol{S}_t^{n,x})) = &\left[\theta_{d,k}^{F,\mathcal{A}}\cdot F_{t+1,d,0,k}^{n,x}\right]_{\forall d\in\mathcal{D},k\in\mathcal{K}} + \left[\theta_{d,k}^{G,\mathcal{A}}\cdot G_{t+1,d,0,k}^{n,x}\right]_{\forall d\in\mathcal{D},k\in\mathcal{K}} \\ &+ \left[\theta_d^{u,\mathcal{A}}\cdot u_{t,d}\right]_{\forall d\in\mathcal{D}} + \left[\theta_d^{v,\mathcal{A}}\cdot v_{t,d}\right]_{\forall d\in\mathcal{D}} \\ &+ \delta_t^n \end{aligned} \tag{19a}$$

$$F_{t+1,d,0,0}^{n,x} - \left(F_{t,d,0,1}^n + F_{t,d,1,0}^n\right)\cdot u_{t,d} \le 0,\ \forall d\in\mathcal{D} \tag{19b}$$

$$G_{t+1,d,0,0}^{n,x} - \left(G_{t,d,0,1}^n + G_{t,d,1,0}^n\right)\cdot u_{t,d} \le 0,\ \forall d\in\mathcal{D} \tag{19c}$$

$$\sum_{k\in\mathcal{K}|k>0} F_{t+1,d,0,k}^{n,x} - \left(\sum_{k\in\mathcal{K}\setminus K^{max}}\left(F_{t,d,0,k+1}^n\right) + \sum_{k\in\mathcal{K}}\left(F_{t,d,1,k+1}^n\right)\right)\cdot v_{t,d} \le 0,\ \forall d\in\mathcal{D} \tag{19d}$$

$$\sum_{k\in\mathcal{K}|k>0} G_{t+1,d,0,k}^{n,x} - \left(\sum_{k\in\mathcal{K}\setminus K^{max}}\left(G_{t,d,0,k+1}^n\right) + \sum_{k\in\mathcal{K}}\left(G_{t,d,1,k+1}^n\right)\right)\cdot v_{t,d} \le 0,\ \forall d\in\mathcal{D} \tag{19e}$$

$$u_{t,d}\in\{0,1\},\ \forall d\in\mathcal{D} \tag{19f}$$

$$v_{t,d}\in\{0,1\},\ \forall d\in\mathcal{D} \tag{19g}$$

B Categorization of states in normal-sized experiments

Table 7: Categorization of states

Category	Freights (F)	Destinations (D)	I_3		I_4		I_5		I_6		I_7		I_8	
			F	D	F	D	F	D	F	D	F	D	F	D
C1	Low	Low	[0,13)	[0,4)	[0,11)	[0,4)	[0,14)	[0,4)	[0,14)	[0,4)	[0,8)	[0,2)	[0,22)	[0,4)
C2	Low	Medium	[0,13)	[4,5)	[0,11)	[4,5)	[0,14)	[4,5)	[0,14)	[4,5)	[0,8)	[2,3)	[0,22)	[4,5)
C3	Low	High	[0,13)	[5,9)	[0,11)	[5,10)	[0,14)	[5,8)	[0,14)	[5,9)	[0,8)	[3,7)	[0,22)	[5,9)
C4	Medium	Low	[13,20)	[0,4)	[11,18)	[0,4)	[14,20)	[0,4)	[14,20)	[0,4)	[8,14)	[0,2)	[22,30)	[0,4)
C5	Medium	Medium	[13,20)	[4,5)	[11,18)	[4,5)	[14,20)	[4,5)	[14,20)	[4,5)	[8,14)	[2,3)	[22,30)	[4,5)
C6	Medium	High	[13,20)	[5,9)	[11,18)	[5,10)	[14,20)	[5,8)	[14,20)	[5,9)	[8,14)	[3,7)	[22,30)	[5,9)
C7	High	Low	[20,57)	[0,4)	[18,45)	[0,4)	[20,52)	[0,4)	[20,47)	[0,4)	[14,37)	[0,2)	[30,54)	[0,4)
C8	High	Medium	[20,57)	[4,5)	[18,45)	[4,5)	[20,52)	[4,5)	[20,47)	[4,5)	[14,37)	[2,3)	[30,54)	[4,5)
C9	High	High	[20,57)	[5,9)	[18,45)	[5,10)	[20,52)	[5,8)	[20,47)	[5,9)	[14,37)	[3,7)	[30,54)	[5,9)

C Confidence intervals of the performance in normal-sized experiments

Table 8: Confidence intervals of the difference between the benchmark heuristic and the ADP algorithm (VFA 3)

State	I_3	I_4	I_5	I_6	I_7	I_8
C1	[-7.0%,-4.8%]	[-9.6%,-7.5%]	[-10.3%,-8.4%]	[-6.1%,-4.9%]	[-1.3%,0.0%]	[-5.9%,-4.5%]
C2	[-9.7%,-8.4%]	[-13.1%,-11.6%]	[-4.8%,-3.3%]	[-3.6%,-1.8%]	[-1.2%,0.1%]	[-11.6%,-10.4%]
C3	[-2.7%,-1.2%]	[-7.2%,-6.1%]	[-9.1%,-7.4%]	[-3.8%,-2.4%]	[0.5%,1.7%]	[-7.7%,-6.7%]
C4	[-16.0%,-13.8%]	[-26.5%,-24.6%]	[-6.2%,-4.1%]	[-12.5%,-11.2%]	[-2.2%,-0.7%]	[-8.4%,-7.6%]
C5	[-15.9%,-14.3%]	[-2.0%,-0.9%]	[-10.5%,-8.8%]	[-26.5%,-25.3%]	[-1.0%,0.1%]	[-10.3%,-9.2%]
C6	[0.5%,2.1%]	[-5.1%,-3.9%]	[-4.5%,-3.1%]	[-11.1%,-10.0%]	[-2.6%,-1.4%]	[-8.2%,-7.3%]
C7	[-4.7%,-4.0%]	[-4.3%,-3.0%]	[-25.0%,-23.5%]	[-0.6%,0.4%]	[-12.2%,-9.8%]	[-7.9%,-6.8%]
C8	[-2.9%,-1.7%]	[-17.1%,-16.3%]	[-2.5%,-1.6%]	[-7.5%,-6.7%]	[-0.9%,-0.2%]	[-3.7%,-2.9%]
C9	[-1.5%,-0.3%]	[1.8%,2.8%]	[-5.4%,-3.5%]	[-11.4%,-10.7%]	[3.9%,5.4%]	[-7.9%,-7.2%]