

## **Service and transfer selection for freights in a synchromodal network**

Arturo Pérez Rivera, Martijn Mes

Beta Working Paper series 504

BETA publicatie	WP 504 (working paper)
ISBN	
ISSN	
NUR	804
Eindhoven	April 2016

# Service and transfer selection for freights in a sychromodal network

Arturo Pérez Rivera\* and Martijn Mes

Department Industrial Engineering and Business Information Systems  
University of Twente, The Netherlands

April 1, 2016

## Abstract

We study the planning problem of selecting services and transfers in a synchronomodal network to transport freights with different characteristics, over a multi-period horizon. The evolution of the network over time is determined by the decisions made, the schedule of the services, and the new freights that arrive each period. Although freights become known gradually over time, the planner has probabilistic knowledge about their arrival. Using this knowledge, the planner balances current and future costs at each period, with the objective of minimizing the total costs over the entire horizon. To model this stochastic and multi-period tradeoff, we propose a Markov Decision Process (MDP) model. To overcome the computational complexity of solving the MDP, we propose an Approximate Dynamic Programming (ADP) approach. Using different problem settings, we show that our look-ahead approach has significant benefits compared to a benchmark heuristic.

## 1 Introduction

We consider the problem of selecting services and transfers in a synchronomodal network, to transport freights from their origin to their destination, while minimizing costs over a multi-period horizon. In a synchronomodal setting, all freights are booked “mode-free”, meaning that there are no restrictions for selecting a transportation mode or deciding the number transfers among the intermodal terminals. As an example, consider Figure 1. In this figure, the star freight can be brought from its origin to its destination directly by truck, or via a combination of any of five intermodal terminals that have different transportation modes, schedules, and times. Although

---

\*Corresponding author. E-mail: a.e.perezrivera@utwente.nl; Tel.: +31 534894715; Fax: +31 534892159.

there is flexibility in the selection of services and transfers, all decisions are encumbered by various time restrictions and by the variability in the arrival of freights over time. In this paper, we study how these challenges can be tackled, specially for large-sized problems, in order to select the services and transfers that achieve the lowest expected costs.

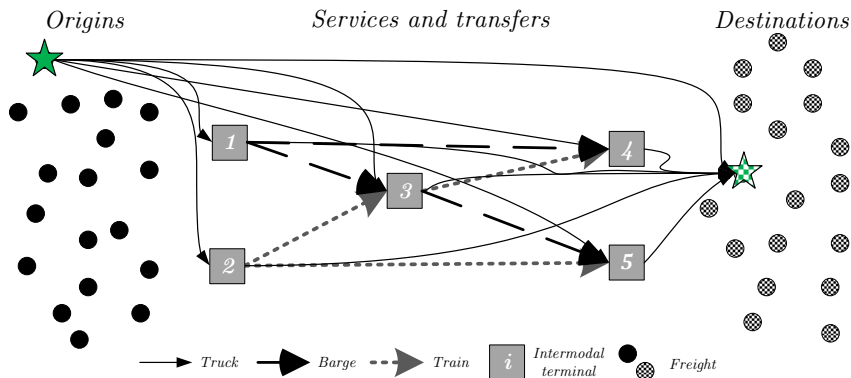


Figure 1: Example of a synchronomodal network

In synchronomodal planning, it is possible to change the transportation plan, i.e., the services and transfers needed to bring a freight from its origin to its destination, at any point in time. Even though the planner might have a complete plan at a given moment, only the first part of such a plan is implemented. The next decision moment, the planner has the flexibility to change the original plan if necessary. Consequently, at each decision moment, the planner can make three types of decisions for available freights at each location: (i) transport a freight to its final destination, (ii) transport a freight to an intermediate terminal, and (iii) postpone the transport of a freight. All types of decisions incur some form of costs. The first and the second type incur *direct* costs, which are costs realized by the services required for the transportation of a freight. The third type has direct costs only in case of holding costs. Since the problem is to minimize costs over a multi-period horizon, the second and third type also incur *future* costs, which are costs that are not incurred on the moment the decision was made, but on a posterior moment within the planning horizon. Naturally, there is uncertainty about future costs since these depend on the decisions that will be made in the future, which in turn depend on new arriving freights. The optimal balance between direct and future costs guarantees the best performance for the multi-period horizon. However, anticipating future costs is challenging.

The decisions and the evolution of the network over time are influenced by two types of time restrictions. The first type corresponds to the durations and schedules of services and transfers. As an example, consider

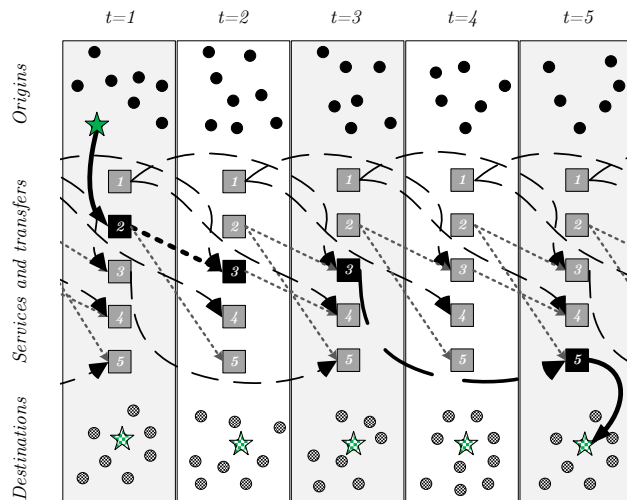


Figure 2: Time evolution and planning example of service and transfer selection corresponding to the synchronodal network of Figure 1.

Figure 2, which shows a possible plan spanning 5 days using both train and barge. Note that the network corresponds to that of Figure 1. Although the plan spans a 5 day horizon, only the first decision is implemented, and as time progresses, re-planning decisions can be made if necessary. The second type corresponds to the time-windows of freights. In combination with service schedules and durations, time-windows limit the feasible transportation services and transfers, and thus the feasible decisions. In addition to the time restrictions, the variability in the number of freights that arrive each day and their characteristics (i.e., origin, destination, time-window), also influence the evolution of the network. These freights and their characteristics are unknown beforehand, but there is probabilistic information about their variability. Every day, the planner must consider all these network characteristics and select which freights use the services available that day. Although we mention “days” in this paragraph, time can be discretized into any arbitrary interval.

The objective of this paper is twofold: (i) to design a model and look-ahead solution method that capture all problem characteristics and their effect on the planning objective, and (ii) to explore the use of look-ahead decision methods under several settings. We model the decision problem and the evolution of the network using a Markov Decision Process (MDP) model. With this model, the optimal trade-off between the three types of decisions, over time and under uncertain demand, can be obtained. However, and as with many optimal approaches, solving MDP models become unmanageable as problem instances grow larger. To overcome this, we use Approximate Dynamic Programming (ADP). ADP combines simulation, optimization,

and statistical techniques to approximate the solution of an MDP model without losing any of its characteristics.

The remainder of this paper is organized as follows. In Section 2, we briefly mention the relevant literature and specify our contribution to it. In Section 3, we introduce the MDP model. In Section 4, we explain our ADP solution approach. In Section 5, we test various designs within the ADP algorithm, and provide a comparison with benchmark heuristics. Finally, we close in Section 6 with conclusions and key research insights about modeling and solving the problem of selecting services and transfers for freights in a synchmodal network.

## 2 Literature Review

In this section, we briefly comment on the literature about synchmodal planning. We focus our attention on the literature about planning problems in dynamic and flexible intermodal transportation networks. Extensive literature reviews about this area can be found in [2] and [15].

Synchmodal planning is the proactive organization and control of intermodal transportation services based on the latest information available [15]. In such a planning paradigm, decision methods must balance the demand with all available services and intermodal transfers each time new information becomes known [13]. Although research about such methods in synchmodal planning problems is on its infancy, several studies show how existing methods for intermodal transport planning can be extended to such problem settings [17] and how significant gains can be achieved in practice [9, 14, 17].

In intermodal transport planning, Dynamic Service Network Design problems (DSND) are the closest to the synchmodal planning problems. DSND involves the selection of transportation services and modes for freights, where at least one feature of the network varies over time [15]. Due to the time-space nature of DSND problems, graph theory and mathematical programming approaches are commonly used in this area. However, these approaches have computational limitations for large and complex time-evolving problem instances [16], which are characteristics common to synchmodality [13]. To overcome these limitations, additional designs, such as decomposition algorithms [5], receding horizons [6], and model predictive control [10], are necessary. These additional designs are less suitable for including probabilistic information in the decisions, which may explain why most DSND studies assume deterministic demand [15] even though the need to incorporate it has been recognized [7].

To incorporate stochasticity in DSND approaches, techniques such as scenario generation [3, 7], two-stage stochastic programming [1, 8], and approximate dynamic programming (ADP) [4, 11] have been used. Although these approaches perform better than their deterministic counterpart, they

have limitations when considering synchromodal planning. In the scenario generation technique, plans do not change as new information becomes available. In the two-stage stochastic programming approach, explicit probabilistic constraints and high computational requirements limit their applicability to large instances. In ADP, a proper design and validation of the approximation algorithm is crucial and challenging. Nevertheless, the ADP approach allows generic modeling of complex, time-revealing, stochastic networks and a fast response time for updating plans.

To summarize, research in DSND problems provides a useful base for synchromodal planning. Considering all challenges and opportunities mentioned before, we believe that our contribution to the scientific literature of stochastic DSND problems and synchromodal planning has three key points. First, we design a Markov Decision Process (MDP) model and solution method based on Approximate Dynamic Programming (ADP) that capture all problem characteristics and their effect on the planning objective. Second, we explore the use of such a look-ahead approach, under different problem settings, and provide design and validation insights. Third, we compare the ADP approach against an advanced sampling procedure and specify further research directions based on the insights.

### 3 Optimization Model

In this section, we first present the notation for our problem. As noted before, our problem falls into the class of DSND problems, which are commonly modeled using a time-space representation of the transportation network in a directed graph. Following this convention, we denote all problem characteristics using a directed graph and present the MDP model for our stochastic planning problem. We end with a discussion on the computational limitations of the model.

#### 3.1 Notation

We define a directed graph  $G_t = (\mathcal{N}_t, \mathcal{A}_t)$ , where  $t \in \mathcal{T} = \{0, 1, \dots, T^{\max} - 1\}$  represents the finite planning horizon (i.e.,  $T^{\max}$  decision periods),  $\mathcal{N}_t$  represents the set of all nodes at time  $t$ , and  $\mathcal{A}_t$  represents the set of all directed arcs at time  $t$ . In the remainder of the paper, we refer to a time period  $t$  as a day, although it is important to note that time can be discretized in any arbitrary interval. In the remainder of the model description, all notation and formulations indexed by  $t$  correspond to that day. Nodes  $\mathcal{N}_t$  represent physical locations where freight can begin or end transportation, i.e., origins, intermodal terminals, and destinations. We denote the set of origin nodes as  $\mathcal{N}_t^O$ , the set of destination nodes as  $\mathcal{N}_t^D$ , and the set of intermodal terminal nodes as  $\mathcal{N}_t^I$ . These three sets are mutually exclusive, make up the set of all

nodes, and are all indexed with  $i, j$  and  $d$ . Arcs  $\mathcal{A}_t$  represent all transportation services in the network. Similar to the node classification, we classify the arcs into three types. The set of arcs between an origin and an intermodal node is denoted as  $\mathcal{A}_t^O = \{(i, j) | i \in \mathcal{N}_t^O \text{ and } j \in \mathcal{N}_t^I\}$ . The set of arcs between two intermodal terminal nodes is denoted as  $\mathcal{A}_t^I = \{(i, j) | i, j \in \mathcal{N}_t^I\}$ . The set of arcs between an origin or an intermodal node, and a destination, is denoted as  $\mathcal{A}_t^D = \{(i, d) | i \in \mathcal{N}_t^O \cup \mathcal{N}_t^I \text{ and } d \in \mathcal{N}_t^D\}$ .

We make three modeling assumptions with respect to the transportation services between different types of locations. First, we assume that services beginning at an origin, i.e.,  $\mathcal{A}_t^O$ , as well as services ending in a destination, i.e.,  $\mathcal{A}_t^D$ , are available every day and are realized by truck. This assumption correspond to the usual pre- and end-haulage operations in an synchromodal network. Second, we assume that services between two intermodal terminals, i.e.,  $\mathcal{A}_t^I$ , are done by high-capacity modes and never by truck. Although this is a simplification of the network, trucks between intermodal terminals are seldom used. Third, we assume there is at most one service between two intermodal terminal nodes. Naturally, multiple services between two intermodal terminals can be modeled using more than one pair of nodes representing those terminals. Note that the services between two intermodal terminals are not necessarily the same every day to represent the schedules for the high-capacity modes.

Transportation services in the network have their starting and ending location modeled as nodes within  $G_t$ . However, there are three other characteristics that are relevant for the planning. First, there is a *maximum capacity*  $Q_{i,j,t}$  in the service between two intermodal terminals  $(i, j) \in \mathcal{A}_t^I$ , measured in number of freights. For all services involving an origin or a destination, we assume there is unlimited number of trucks. Second, all services  $(i, j) \in \mathcal{A}_t$  have a *service duration* of  $L_{i,j,t}^A$  days, which lasts at least one day. Remind that time can be discretized into any arbitrary interval. All transfer/handling operations at each location  $i \in \mathcal{N}_t$  have a duration of  $L_{i,t}^N$  days. To measure the total time required for the service between two locations, we define the auxiliary parameter  $M_{i,j,t} = L_{i,t}^N + L_{i,j,t}^A + L_{j,t}^N$ . We assume that traveling directly to a destination by truck is always faster than going through an intermodal terminal, i.e.,  $L_{i,d,t}^A < \min_{j \in \mathcal{N}_t^I} \{M_{i,j,t} + L_{j,d,t}^A\}, \forall (i, d) \in \mathcal{A}_t^D$ . This assumption works in a similar way as the triangle inequality in routing problems. Third, all relevant costs from a service  $(i, j) \in \mathcal{A}_t$  are captured in the *cost function*  $C_{i,j,t}$ . This means that, although pre- and end-haulage decisions (e.g., first- and last-mile routing), as well as freight handling decisions (e.g., container stacking), are outside the scope of the planner, their costs can be captured with the function  $C_{i,j,t}$ .

Each day  $t$ , freights with different attributes become known to the planner. These freights are characterized by an origin  $i \in \mathcal{N}_t^O$ , a destination

$d \in \mathcal{N}_t^D$ , a release day  $r \in \mathcal{R}_t = \{0, 1, 2, \dots, R_t^{\max}\}$ , and a time-window length  $k \in \mathcal{K}_t = \{0, 1, 2, \dots, K_t^{\max}\}$ , where  $R_t^{\max}$  and  $K_t^{\max}$  are the maximum release day and time-window length, respectively, that a freight can have. Note that the absolute due-day is  $k$  days after  $r$ . Even though new freights and their characteristics are only known until they arrive, there is probabilistic knowledge about their arrival. In between two consecutive days  $t-1$  and  $t$ , a total of  $f \in \mathbb{N}$  freights arrive into the system with probability  $p_{f,t}^F$ . A freight that arrives has origin  $i \in \mathcal{N}_t^O$  with probability  $p_{i,t}^O$ , destination  $d \in \mathcal{N}_t^D$  with probability  $p_{d,t}^D$ , release-day  $r \in \mathcal{R}_t$  with probability  $p_{r,t}^R$ , and time-window length  $k \in \mathcal{K}_t$  with probability  $p_{k,t}^K$ .

### 3.2 MDP Model

The stages of the MDP are defined by  $t \in \mathcal{T}$ . To model freights in the network, we introduce the variable  $F_{i,d,r,k,t} \in \mathbb{Z}^+$  that represents the number of freights at location  $i \in \mathcal{N}_t^O \cup \mathcal{N}_t^I$ , that have destination  $d \in \mathcal{N}_t^D$ , release day  $r \in \mathcal{R}_t$ , and time-window length  $k \in \mathcal{K}_t$ . The state  $S_t$  of the system consists of all freights variables as seen in (1). The state space is denoted as  $\mathcal{S}_t$ , i.e.,  $S_t \in \mathcal{S}_t$ .

$$S_t = [F_{i,d,r,k,t}]_{\forall i \in \mathcal{N}_t^O \cup \mathcal{N}_t^I, d \in \mathcal{N}_t^D, r \in \mathcal{R}'_t, k \in \mathcal{K}_t} \quad (1)$$

Note that we use a new set  $\mathcal{R}'_t$  for the release days. The release day definition at origin nodes remains the same. The release day at an intermodal terminal, however, is now used to represent the days “left” for a freight to arrive at that node. For example, if a released freight is sent to an intermodal terminal  $j$  on a barge whose total service duration is four days, this freight will appear the day after it was sent, as a freight with  $r = 3$  at location  $j$ . This new set, which is defined as  $\mathcal{R}'_t = \left\{0, 1, 2, \dots, \max \left\{ R_t^{\max}, \max_{(i,j) \in \mathcal{A}_t^I} M_{i,j,t} \right\} \right\}$ , allows us to model multi-day durations of services without the need of remembering decisions from more than one day ago, i.e., to be more computationally efficient. Note that, in case no total service duration is larger than  $R_t^{\max}$ , then  $\mathcal{R}_t = \mathcal{R}'_t$ . Time-window lengths  $k$  still model the number of days after the release-day  $r$ , within which the freight has to be at its final destination. We will elaborate more on the evolution of the network over time later on in this section.

At each stage, the planner must decide how many released freights to transport and to postpone, for all locations. Remind that, in a synchro-modal network, only the first part of the plan to transport a freight to its destination is implemented at each decision moment. Consequently, at every stage, the decision to transport a freight can be either to send it directly to its final destination, or to send it to an intermodal terminal. To model this decision, we introduce the variable  $x_{i,j,d,k,t} \in \mathbb{Z}^+$ , which represents the number of freights having destination  $d \in \mathcal{N}_t^D$  and time-window length  $k \in \mathcal{K}_t$



that are transported from location  $i$  to location  $j$  using service  $(i, j) \in \mathcal{A}_t$ . Thus, the decision vector  $x_t$  consists of all transported freights in the network, as seen in (2a).

$$x_t = [x_{i,j,d,k,t}]_{\forall (i,j) \in \mathcal{A}_t, d \in \mathcal{N}_t^D, k \in \mathcal{K}_t} \quad (2a)$$

s.t.

$$\sum_{j \in \mathcal{N}_t^I \cup \{d\}} x_{i,j,d,k,t} \leq F_{i,d,0,k,t}, \quad \forall i \in \mathcal{N}_t^O \cup \mathcal{N}_t^I, d \in \mathcal{N}_t^D, k \in \mathcal{K}_t \quad (2b)$$

$$x_{i,d,d,L_{i,d,t}^A} \geq F_{i,d,0,L_{i,d,t}^A}, \quad \forall (i,d) \in \mathcal{A}_t^D, k \in \mathcal{K}_t \quad (2c)$$

$$x_{i,j,d,k,t} = 0, \quad \forall (i,j) \in \mathcal{A}_t, d \in \mathcal{N}_t^D, k \in \mathcal{K}_t | k < M_{i,j,t} + M_{j,d,t} \quad (2d)$$

$$\sum_{d \in \mathcal{N}_t^D} \sum_{k \in \mathcal{K}_t} x_{i,j,d,k,t} \leq Q_{i,j,t}, \quad \forall (i,j) \in \mathcal{A}_t^I \quad (2e)$$

Naturally, the decision  $x_t$  depends on the state  $S_t$ , as well as on the capacity of the long-haul transportation services. The feasible decision space  $\mathcal{X}_t$ , with  $x_t \in \mathcal{X}_t$ , has four constraints. First, the number of freights transported from one location to all other locations cannot exceed the number of released freights at hand at the start location, as seen in (2b). Second, released freights whose time-window length is as long as the duration of direct transport (i.e., trucking) must be transported using this service, as seen in (2c). Third, freights whose time-window length is smaller than the duration of the shortest path between an intermodal terminal and their destination cannot be transported via that terminal, as seen in (2d). Fourth, transport between two intermodal terminals cannot exceed the capacity of the long-haul vehicle, as seen in (2e).

After making a decision  $x_{t-1}$ , but before entering the state  $S_t$ , new freights become known to the planner. We represent new freights with origin  $i \in \mathcal{N}_t^O$ , destination  $d \in \mathcal{N}_t^D$ , release day  $r \in \mathcal{R}_t$ , and time-window length  $k \in \mathcal{K}_t$ , by  $\tilde{F}_{i,d,r,k,t}$ . We denote the vector of all new freights that arrive between stages  $t-1$  and  $t$  by  $W_t$ , as seen in (3). This vector represents the exogenous information that became known between stages  $t-1$  and  $t$ .

$$W_t = [\tilde{F}_{i,d,r,k,t}]_{\forall i \in \mathcal{N}_t^O, d \in \mathcal{N}_t^D, r \in \mathcal{R}_t, k \in \mathcal{K}_t} \quad (3)$$

The evolution of the network over time is influenced by decisions, exogenous information, and various time relations. To model this evolution, we introduce the transition function  $S^M$  as seen in (4a). The general idea of  $S^M$  is to define the freights at  $S_t$  using only the previous-stage decision  $x_{t-1}$  and the exogenous information  $W_t$ . Although decisions can span more than one day, we use freight release days and time-windows lengths to avoid the need of remembering a decision for more than one stage. Naturally, when freights are not transported, they remain at the same location and their release days

and time-window lengths decrease. However, when freights are transported from a given location  $i$  to an intermodal terminal  $j$ , they are modeled as freights whose release day increases and their time-window length decreases in line with the total duration of transport  $M_{i,j,t}$ . To model all these relations,  $S^M$  classifies freight variables  $F_{t,i,d,r,k}$  into seven categories, as shown in (4b) to (4h). To exemplify in detail the workings of these categories, consider (4c). These constraints apply to released freights at an intermodal terminal  $i$  with destination  $d$  and time-window length  $k$ . These freights are the result of three types of freights: (i) released freights in the same terminal, from the previous stage, that had the same destination, that had one additional day in the time-window, and that were not transported to any other node (i.e.,  $F_{t-1,i,d,0,k+1} - \sum_{j \in \mathcal{A}_t} x_{t-1,i,j,d,k+1}$ ); (ii) freights in the same node, from the previous stage, that had the same destination, that had a release-day of one, and that had the same time-window length (i.e.,  $F_{t-1,i,d,1,k}$ ); and (iii) freights that arrived from other locations to  $i$ , that have the same destination, whose total duration of transportation was one period, and whose time-window length was  $k + M_{j,i,t}$  at the moment of the decision  $x_{t-1}$  (i.e.,  $\sum_{j \in \mathcal{A}_t | M_{j,i,t}=1} x_{t-1,j,i,d,k+M_{j,i,t}}$ ). All other constraints work in a similar fashion.

$$S_t = S^M(S_{t-1}, x_{t-1}, W_t) \quad (4a)$$

s.t.

$$F_{t,i,d,0,k} = F_{t-1,i,d,0,k+1} - \sum_{j \in \mathcal{A}_t} x_{t-1,i,j,d,k+1} + F_{t-1,i,d,1,k} + \tilde{F}_{t,i,d,0,k}, \quad (4b)$$

$$\forall i \in \mathcal{N}_t^O, d \in \mathcal{N}_t^D, k+1 \in \mathcal{K}_t$$

$$F_{t,i,d,0,k} = F_{t-1,i,d,0,k+1} - \sum_{j \in \mathcal{A}_t} x_{t-1,i,j,d,k+1} + F_{t-1,i,d,1,k} \quad (4c)$$

$$+ \sum_{j \in \mathcal{A}_t | M_{j,i,t}=1} x_{t-1,j,i,d,k+M_{j,i,t}},$$

$$\forall i \in \mathcal{N}_t^I, d \in \mathcal{N}_t^D, k+1 \in \mathcal{K}_t$$

$$F_{t,i,d,0,K_t^{\max}} = F_{t-1,i,d,1,K_t^{\max}} + \tilde{F}_{t,i,d,0,K_t^{\max}}, \quad (4d)$$

$$\forall i \in \mathcal{N}_t^O, d \in \mathcal{N}_t^D$$

$$F_{t,i,d,r,k} = F_{t-1,i,d,r+1,k} + \tilde{F}_{t,i,d,r,k}, \quad (4e)$$

$$\forall i \in \mathcal{N}_t^O, d \in \mathcal{N}_t^D, r+1 \in \mathcal{R}_t | r \geq 1, k \in \mathcal{K}_t$$

$$F_{t,i,d,r,k} = F_{t-1,i,d,r+1,k} + \sum_{j \in \mathcal{A}_t | M_{j,i,t}=r+1} x_{t-1,j,i,d,k+M_{j,i,t}}, \quad (4f)$$

$$\forall i \in \mathcal{N}_t^I, d \in \mathcal{N}_t^D, r+1 \in \mathcal{R}'_t | r \geq 1, k \in \mathcal{K}_t$$

$$F_{t,i,d,|\mathcal{R}'_t|,k} = \sum_{j \in \mathcal{A}_t | M_{j,i,t}=|\mathcal{R}'_t|+1} x_{t-1,j,i,d,k+M_{j,i,t}}, \quad (4g)$$

$$\begin{aligned}
& \forall i \in \mathcal{N}_t^I, d \in \mathcal{N}_t^D, k \in \mathcal{K}_t \\
& F_{t,i,d,R_t^{\max},k} = \tilde{F}_{t,i,d,R_t^{\max},k}, \\
& \forall i \in \mathcal{N}_t^O, d \in \mathcal{N}_t^D, k \in \mathcal{K}_t
\end{aligned} \tag{4h}$$

The goal is to minimize the total costs over a multi-period horizon, considering all possible states that can occur in each day of the horizon. We define a set of decisions for all possible states as a policy  $\pi$ , i.e., a function that maps each possible state  $S_t \in \mathcal{S}_t$  to a decision  $x_t^\pi \in \mathcal{X}_t$ . Consequently, the objective is to determine the policy  $\pi$  from the set of all policies  $\Pi$  that minimizes the expected costs over the planning horizon, given an initial state  $S_0$ , as seen in (5):

$$\min_{\pi \in \Pi} \mathbb{E} \left[ \sum_{t \in \mathcal{T}} C_t(x_t^\pi) = \sum_{t \in \mathcal{T}} \sum_{(i,j) \in \mathcal{A}_t} \left( C_{i,j,t} \cdot \sum_{d \in \mathcal{N}_t^D} \sum_{k \in \mathcal{K}_t} x_{i,j,d,k,t}^\pi \right) \middle| S_0 \right] \tag{5}$$

To solve the sequential decision problem, we transform (5) into the Bellman's equations of (6). In these equations, the expected next-stage costs is computed using the value of the next-stage state  $S_{t+1}$  (obtained using  $S^M$ ), the decision  $x_t^\pi$ , a realization of the exogenous information  $\omega \in \Omega_{t+1}$ , and the associated probability  $p_\omega^{\Omega_{t+1}}$ . The solution to all recursive equations of (6) provide the optimal policy for the MDP at stage  $t$ , and by iterating backwards through time, from the end of the planning horizon, the objective in (5) is achieved.

$$V_t(S_t) = \min_{x_t^\pi \in \mathcal{X}_t} \left( C_t(x_t^\pi) + \sum_{\omega \in \Omega_{t+1}} p_\omega^{\Omega_{t+1}} \cdot V_{t+1}(S^M(S_t, x_t^\pi, \omega)) \right), \forall S_t \in \mathcal{S}_t \tag{6}$$

However, solving the Bellman equations in (6) for large problems is computationally challenging. The state space  $\mathcal{S}_t$ , decision space  $\mathcal{X}_t$ , and the realizations of the exogenous information in  $\Omega_t$  grow larger with an increasing size of the problem instance. Due to these three ‘‘curses of dimensionality’’ [12], our MDP model is solvable only for tiny problem instances. Notwithstanding, the MDP model serves as a base for our large instance solution approach.

## 4 Solution Approach

Our solution approach is based on Approximate Dynamic Programming (ADP). ADP approximates the expected next-stage costs in (6) through an algorithmic strategy that steps forward through time and uses simulation for the exogenous information  $\Omega_{t+1}$ . For such an approach, a more natural

form of the Bellman's equations in (6) is the expectational form given by:

$$V_t(S_t) = \min_{x_t^\pi \in \mathcal{X}_t} (C_t(x_t^\pi) + \mathbb{E}^\omega [V_{t+1}(S^M(S_t, x_t^\pi, \omega))]) \quad \forall S_t \in \mathcal{S}_t \quad (7)$$

In our ADP approach, the entire expectation  $\mathbb{E}^\omega[\cdot]$  in (7) is replaced by an approximate value function  $\bar{V}_t^n(S_t^{x,n})$ , where  $S_t^{x,n}$  is the so-called post-decision state, i.e., the state after a decision has been made but before the new exogenous information becomes known. As seen in (8), this construct avoids the dimensionality issue of the large number of realizations of the exogenous information.

$$V_t^n(S_t^n) = \min_{x_t^\pi \in \mathcal{X}_t} (C_t(x_t^\pi) + \bar{V}_t^n(S_t^{x,n})) \quad (8)$$

To avoid the large state space, the optimality equations in (8) are solved for one state at each stage, starting from the initial state  $S_0$ . The transition from one state to the next uses a sample from  $\Omega_{t+1}$ , obtained through a Monte Carlo simulation, and the transition function  $S^M$  defined in (4a). This process is performed for the entire planning horizon, and repeated for  $N$  iterations, hence the superscript  $n$  in the approximate value function and post-decision state.

The general outline of an ADP algorithm can be found in Figure 4.7, page 141, of the book of [12]. We now focus on two designs (i.e., variations) we propose for that algorithm. Our first design uses a commonly proposed ADP setup. We use *basis functions* for  $\bar{V}_t^n(S_t^{x,n})$  and the *non-stationary least squares* method for updating this function. A basis function  $\phi_a(S_t^{x,n})$  is a quantitative characteristic of a given feature  $a$  of a post-decision state  $S_t^{x,n}$  that describes, to some extent, the value of that post-decision state. Examples of features in our problem are the number of freights for a given destination and the number of freights at a given intermodal terminal. Given a set of features  $\mathcal{A}$ , the approximated next-stage costs in (8) are the result of the product between the basis function  $\phi_a(S_t^{x,n})$  and the weight  $\theta_{a,t}^n$  for each feature  $a \in \mathcal{A}$ , as seen in (9).

$$\bar{V}_t^{x,n}(S_t^{x,n}) = \sum_{a \in \mathcal{A}} \theta_{a,t}^n \phi_a(S_t^{x,n}) \quad (9)$$

The weight  $\theta_{a,t}^n$  depends on the iteration  $n$  because, at iteration  $n$ , costs from the previous approximation at  $n - 1$  have been observed, and can be used to update the weights. We use a Non-stationary Least Squares (NLS) method for updating these weights since it gives more emphasis to the recent observation than to the previous one. This emphasis is necessary at early iterations, where initial conditions might bias the approximation and the result of the ADP approach. The weights  $\theta_{a,t}^n$ , for all  $a \in \mathcal{A}$ , are updated each iteration  $n$  using the observed error (i.e., difference between the next-stage estimate from the previous iteration  $\bar{V}_{t-1}^{n-1}(S_{t-1}^{x,n})$  and the current

estimate  $\widehat{v}_t^n$ , the value of all basis functions  $\phi_a(S_t^{x,n})$ , the optimization matrix  $H^n$ , and the previous weights  $\theta_{a,t}^{n-1}$ , as seen in (10). For a comprehensive explanation on the NLS method, we refer to [12].

$$\theta_{a,t}^n = \theta_{a,t}^{n-1} - H_n \phi_a(S_t^{x,n}) \left( \overline{V}_{t-1}^{n-1}(S_{t-1}^{x,n}) - \widehat{v}_t^n \right) \quad (10)$$

The first design considers downstream costs only through a one-step estimate. Since estimates can be off, especially in early iterations, it might be beneficial to do “look” at more than one step ahead. To do this, our second design builds on the first one and uses two additional constructs. First, we add a valid inequality to the decision space  $\mathcal{X}_t$  as follows. If a direct service (i.e., truck) for a freight between its origin and its destination is cheaper than going from its origin to a given intermodal terminal and subsequently to its destination, we prevent this freight from going to that intermodal terminal when its time-window length allows only a direct service after the intermodal terminal. Second, we add another estimate to  $\overline{V}_t^n(S_t^{x,n})$ , as seen in (11). In this new approximate value function,  $\overline{C}_t^n(S_t^{x,n})$  is a cost estimate about all costs through the end of the planning horizon obtained with a sampling method, and  $\alpha$  is a weight to balance the use of basis functions and samples for  $\overline{V}_t^n(S_t^{x,n})$ .

$$\overline{V}_t^{x,n}(S_t^{x,n}) = \alpha \sum_{a \in \mathcal{A}} \theta_{a,t}^n \phi_a(S_t^{x,n}) + (1 - \alpha) \overline{C}_t^n(S_t^{x,n}) \quad (11)$$

At last, the output of our two ADP designs are the weights  $\theta_{a,t}^N$ . The resulting policy  $\pi$  maps state  $S_t \in \mathcal{S}_t$  to decision  $x_t^\pi$  as seen in (12).

$$x_t^\pi = \arg \min \left( C_t(x_t^\pi) + \sum_{a \in \mathcal{A}(S_t^x)} \theta_{a,t}^N \phi_a(S_t^x) \right) \quad (12)$$

## 5 Numerical Experiments

In this section, we explore the value of our ADP designs through a series of numerical experiments. Using three small instances, we compare the costs achieved by our ADP approach in three small instances against a benchmark policy and an advance sampling procedure. The section is divided as follows. First, we introduce our experimental setup. Second, we show, analyze, and discuss the results of our experiments.

### 5.1 Experimental Setup

For the three instances, we use a network containing a single origin, three intermodal terminals, and three destinations over a planning horizon of 15 days. Each day, there are three services between the intermodal terminals,

with capacities and durations as shown in Figure 3. The fixed costs of these services are of  $C_{1,2}^F = C_{2,3}^F = 100$  and  $C_{1,3}^F = 150$ . The variable costs range between 36 and 44, and are equal to the Euclidean distance between the terminals in a plane of 100x50 distance units, as shown to scale in Figure 3. In addition, every day there is a direct service between the origin and the terminals, between the origin and the destinations, and between the terminals and the destinations; and they all have duration of one day. There are no fixed costs for the direct services, and the variable cost range varies between 241 and 927, and are equal to ten times the Euclidean distance between the two locations they connect. The number of freights that arrives each day varies between  $f = \{0, 1, \dots, 4\}$ , with probability  $p_f^F$  as shown in Figure 3. In the three instances, each freight has destination  $d \in \{4, 5, 6\}$  with probability  $p_d^D$  as shown in Figure 3, and is always released (i.e.,  $p_0^R = 1$ ). Each freight has a time-window length  $k = \{1, 2, \dots, 5\}$  with probability  $p_k^K$  according to the instances considered, as shown in Figure 3.

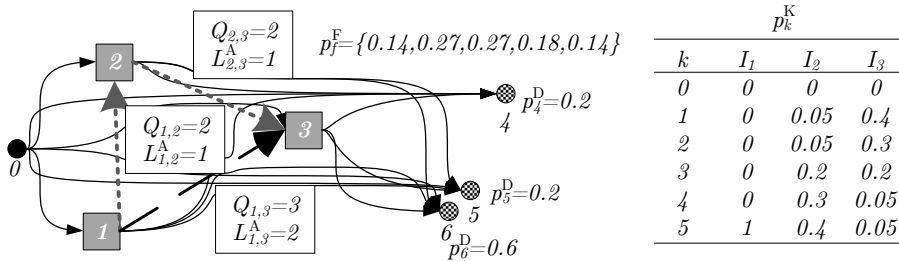


Figure 3: Network characteristics for instances  $I_1$  and  $I_2$

Using the two problem instances, we test four planning methods: a benchmark heuristic, our two ADP designs (named ADP 1 and ADP 2), and an advance sampling procedure. The set of features  $\mathcal{A}$  consists of all state variables and a constant of 1. The weight  $\alpha$  for ADP 2 is defined as  $\alpha = \max \{25 / (25 + n - 1), 0.05\}$  and the sampling method is the same as the advance sampling procedure introduced in the next paragraph. The ADP algorithm runs for 100 iterations and the NLS parameters used are those recommended by [11].

The benchmark heuristic strikes for a balance between using the intermodal services efficiently (consolidate as many freights as possible) and the postponement of freight. It consists of four steps: (i) define the shortest and second shortest path for each freight to its final destination, without considering fixed costs for services between terminals, (ii) calculate the savings between the shortest and second shortest path and define these as savings of the first intermodal service used in the shortest path, (iii) sort all freights in non-decreasing time-window length, i.e., closest due-day first, and (iv) for each freight in the sorted list, check whether the savings of the first in-

termodal service of its shortest path are larger than the fixed cost for this service; if so, use this service for the freight, if not, postpone the transport of the freight. Naturally, all capacities, durations, and time-windows must be checked while doing these steps.

The sampling procedure consists of three steps: (i) enumerate all feasible decisions, (ii) for each feasible decision, estimate future costs by sampling, in a Monte Carlo fashion and using common random numbers across the decisions, realizations of the exogenous information for the remainder of the planning horizon, and simulating the use of the benchmark heuristic for making decisions with these samples, and (iii) choose the decision with the lowest sum of direct and estimated future costs. Although heavily computationally intensive (i.e., not applicable to larger instances), this procedure exploits all possible benefits of looking-ahead in decision making.

The tests are done in a simulation of each planning method consisting of 100 replications of the planning horizon, using common random numbers, and using ten commonly encountered states in each instance. Note that these 100 replications are different from the 100 iterations of the ADP algorithm. Thus, we test the ADP approach in two phases: (i) learning phase through 100 iterations and (ii) simulation phase of using the resulting policy in (12) for 100 replications. To define the ten test states in each instance, we do a simulation of the benchmark heuristic, beginning with an empty sate, for a horizon of 15 days. We save the state at the end of the horizon. We replicate this procedure 10,000 times, and choose the ten states that were observed the most.

## 5.2 Experimental Results and Discussion

First, we analyze Instance  $I_1$ . This is the most flexible instance of the three, since all freights that arrive have a time-window length of 5 days. The results of the ten chosen states are shown in Table 5.2. We show the costs for the benchmark heuristic, and the relative savings, as a percentage, of the other planning methods when compared to the benchmark. In addition, we show characteristics of the initial state. Note that these initial states are ordered by decreasing number of observations during our experimental setup.

On average, ADP 1 achieves savings of 12.8%, ADP 2 of 29.2%, and the advanced sampling procedure of 41.2% when compared to the benchmark heuristic. All three methods that explicitly look-ahead in their decisions perform better than the benchmark that does so only implicitly. The sampling method performs the best, at a high computational expense. For large instances, or even small ones where time is discretized into smaller intervals, this method would not be applicable. ADP 2 performs second best, at a higher computational expense during the learning phase than ADP 1. How-

Table 1: Results for Instance  $I_1$ 

State	Freights			Benchmark	ADP 1	ADP 2	Sampling
	Total	$k < 3$	$k \geq 3$				
1	4	2	2	12221	-13.6%	-33.9%	-43.3%
2	7	3	4	14684	-12.8%	-32.7%	-39.9%
3	5	2	3	13042	-13.1%	-27.5%	-41.5%
4	6	3	3	13863	-12.3%	-25.9%	-39.0%
5	6	2	4	13863	-12.0%	-30.0%	-42.3%
6	6	2	4	13863	-10.4%	-31.3%	-42.9%
7	5	2	3	13042	-12.6%	-23.4%	-41.5%
8	4	3	1	12221	-14.7%	-25.0%	-38.9%
9	2	1	1	10579	-14.9%	-29.9%	-42.4%
10	5	3	2	13042	-11.2%	-32.9%	-40.6%

ever, during the updating of decisions in the planning horizon, both ADP designs take the same time, which is significantly faster the sampling procedure (e.g., milliseconds against minutes per decision in our experiments). ADP 1 lowest savings indicate that a one-step look-ahead is not sufficient for achieving the best performance. Furthermore, the difference between the two ADP designs suggests that further research that explicitly considers a few stages in advance, such as rolling-horizon procedures within the ADP framework, can improve performance significantly.

The average results across the test states of  $I_2$  and  $I_3$  are shown in Tables 5.2, respectively. Note that each instance has its own set of test states, which differs from the other instances. Furthermore, note that  $I_2$  and  $I_3$  have significantly less flexibility than  $I_1$ : due to their time-window length, only 40% and 0.05% of arriving freights can use any intermodal connection, respectively.

Table 2: Average results for Instance  $I_2$  and  $I_3$ 

Instance	Benchmark	ADP 1	ADP 2	Sampling
$I_2$	11078	-5.2%	-9.8%	-31.2%
$I_3$	12874	2.9%	0.4%	-3.3%

The larger savings from all look-ahead methods  $I_1$  and  $I_2$ , compared to  $I_3$ , indicate that the more flexibility there is, and the more freights a state has, the better it is to look-ahead in the decision making. In  $I_2$ , similar results to  $I_1$  are achieved, but with significantly less cost savings. In  $I_3$ , the benchmark heuristic performs better than the ADP approach, and the sampling achieves small savings. In most states of  $I_3$ , the only feasible option (time-wise) for freights is to use a direct service via truck. In such a setting, decision making methods that focus on current costs, such as the benchmark heuristic, perform well since there are hardly consolidation opportunities to anticipate for. However, a robust ADP design should be



able to learn such a policy, as the sampling method seems to do. Further research on adaptations of the ADP algorithm for such settings, such as aggregate functions and state representatives, is necessary.

## 6 Conclusions

We developed an MDP model and an ADP algorithm for selecting services and transfers for freights in a synchromodal network. With the MDP model, the optimal balance between transporting and postponing freights, in different locations of the network, over time, and under uncertain demand, can be achieved. With the ADP algorithm, the computational burden of the MDP model is reduced while preserving all of its modeling functionalities.

Through numerical experiments, we explored the value of using look-ahead decisions in our planning problem and reflected on the value and the limitations of our ADP designs. We observed that the more time-window flexibility and number of freights there are, the better the look-ahead methods perform. We also observed that the two methods that look-ahead more than one stage performed better than the standard one-step look-ahead ADP approach. Further research about ADP designs that explicitly consider a few stages in advance (e.g., rolling horizon, sampling, approximate policy iteration) and other, possibly non-linear, value function approximations, are relevant for synchromodal planning.

## References

- [1] Bai, R., Wallace, S.W., Li, J., Chong, A.Y.L.: Stochastic service network design with rerouting. *Transportation Research Part B: Methodological* 60, 50 – 65 (2014)
- [2] Caris, A., Macharis, C., Janssens, G.K.: Decision support in intermodal transport: A new research agenda. *Computers in Industry* 64(2), 105 – 112 (2013), decision Support for Intermodal Transport
- [3] Crainic, T.G., Hewitt, M., Rei, W.: Scenario grouping in a progressive hedging-based meta-heuristic for stochastic network design. *Computers & Operations Research* 43, 90 – 99 (2014)
- [4] Dall’Orto, L.C., Crainic, T.G., Leal, J.E., Powell, W.B.: The single-node dynamic service scheduling and dispatching problem. *European Journal of Operational Research* 170(1), 1 – 23 (2006)
- [5] Ghane-Ezabadi, M., Vergara, H.A.: Decomposition approach for integrated intermodal logistics network design. *Transportation Research Part E: Logistics and Transportation Review* 89, 53 – 69 (2016)

- [6] Li, L., Negenborn, R.R., Schutter, B.D.: Intermodal freight transport planning – a receding horizon control approach. *Transportation Research Part C: Emerging Technologies* 60, 77 – 95 (2015)
- [7] Lium, A.G., Crainic, T.G., Wallace, S.W.: A study of demand stochasticity in service network design. *Transportation Science* 43(2), 144–157 (2009)
- [8] Lo, H.K., An, K., hua Lin, W.: Ferry service network design under demand uncertainty. *Transportation Research Part E: Logistics and Transportation Review* 59, 48 – 70 (2013)
- [9] Mes, M.R.K., Iacob, M.E.: Synchronodal transport planning at a logistics service provider. In: Zijm, H., Klumpp, M., Clausen, U., Hompel, t.M. (eds.) *Logistics and Supply Chain Innovation: Bridging the Gap between Theory and Practice*, pp. 23–36. Springer International Publishing (2016)
- [10] Nabais, J., Negenborn, R., Benítez, R.C., Botto, M.A.: Achieving transport modal split targets at intermodal freight hubs using a model predictive approach. *Transportation Research Part C: Emerging Technologies* 60, 278 – 297 (2015)
- [11] Pérez Rivera, A., Mes, M.: Dynamic multi-period freight consolidation. In: Corman, F., Voß, S., Negenborn, R.R. (eds.) *Computational Logistics: 6th International Conference, ICCL 2015, Lecture Notes in Computer Science*, vol. 9335, pp. 370–385. Springer International Publishing (2015)
- [12] Powell, W.B.: *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. John Wiley & Sons Inc., second edition edn. (2011)
- [13] Riessen, B., Negenborn, R.R., Dekker, R.: Synchronodal container transportation: An overview of current topics and research opportunities. In: Corman, F., Voß, S., Negenborn, R.R. (eds.) *Computational Logistics: 6th International Conference, ICCL 2015*, pp. 386–397. *Lecture Notes in Computer Science*, Springer International Publishing (2015)
- [14] Riessen, B.V., Negenborn, R.R., Dekker, R., Lodewijks, G.: Service network design for an intermodal container network with flexible transit times and the possibility of using subcontracted transport. *International Journal of Shipping and Transport Logistics* 7(4), 457–478 (2015)
- [15] SteadieSeifi, M., Dellaert, N., Nuijten, W., Woensel, T.V., Raoufi, R.: Multimodal freight transportation planning: A literature review. *European Journal of Operational Research* 233(1), 1 – 15 (2014)

- [16] Wieberneit, N.: Service network design for freight transportation: a review. *OR Spectrum* 30(1), 77–112 (2008)
- [17] Zhang, M., Pel, A.: Synchromodal hinterland freight transport: Model study for the port of rotterdam. *Journal of Transport Geography* 52, 1 – 10 (2016)