UNIVERSITY OF TWENTE.

DINALOG
Dutch Institute
for Advanced Logistics

INTEGRATED SCHEDULING OF DRAYAGE AND LONG-HAUL TRANSPORTATION IN SYNCHROMODALITY

Arturo E. Pérez Rivera & Martijn R.K. Mes

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

Odysseus 2018 - Thursday, June 7th
Cagliari, Italy

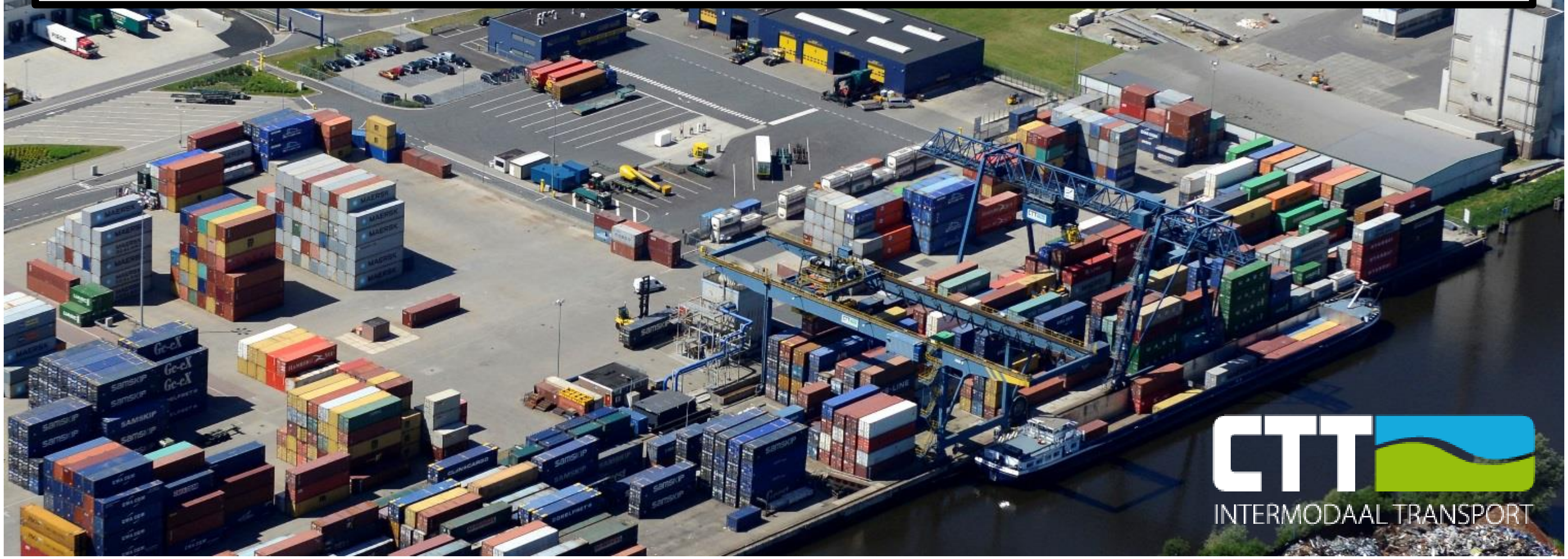
CONTENTS

INTERMODAL TRANSPORTATION CHAIN

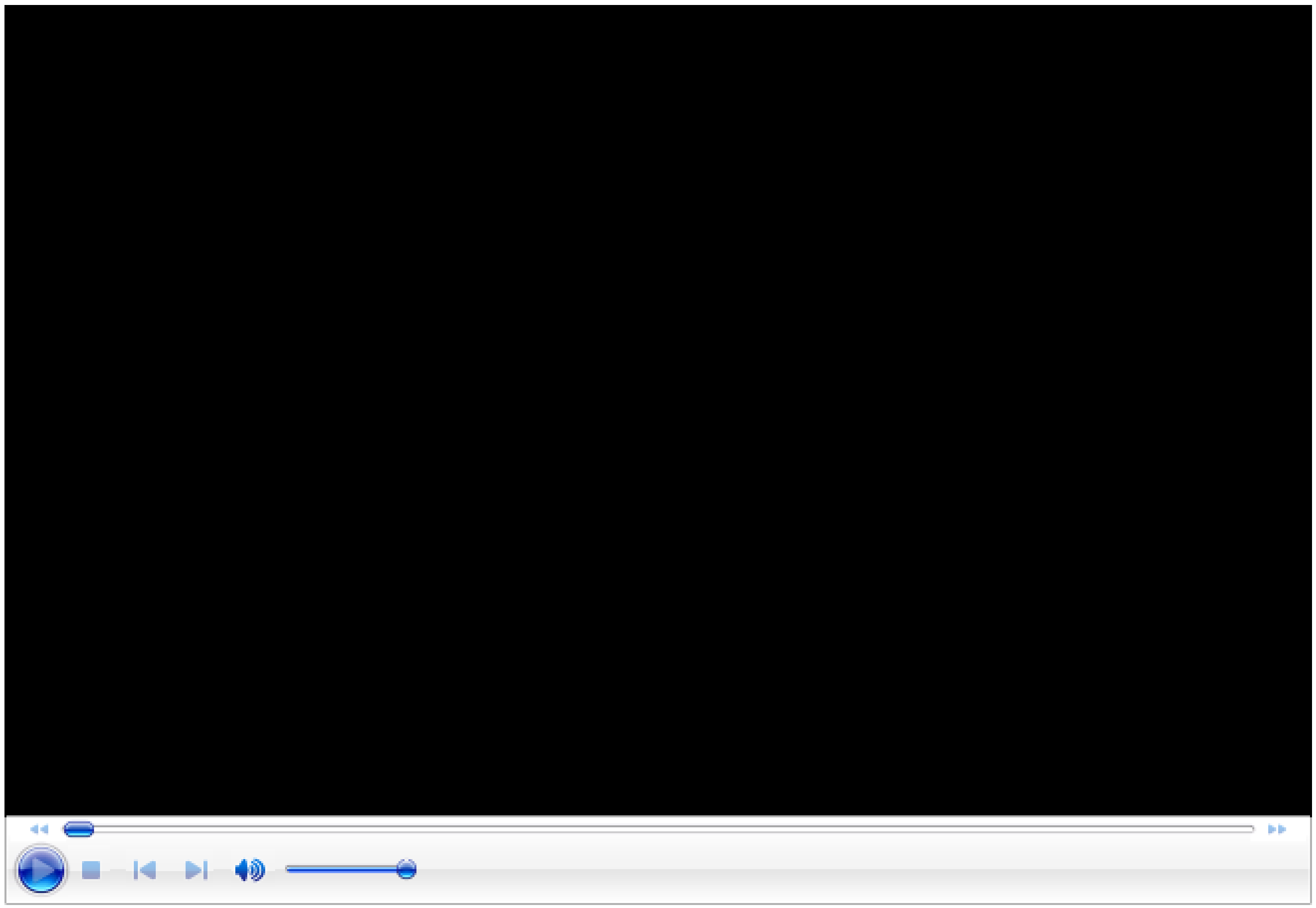
TWO PROCESSES: DRAYAGE AND LONG-HAUL TRANSPORTATION

“In an intermodal transportation chain, the initial and final trips represent 40% of total transport costs.”

Escudero, A.; Muñuzuri, J.; Guadix, J. & Arango, C. (2013) Dynamic approach to solve the daily drayage problem with transit time uncertainty. *Computers in Industry*



SYNCHROMODALITY
WHAT IS SYNCHROMODAL TRANSPORTATION?



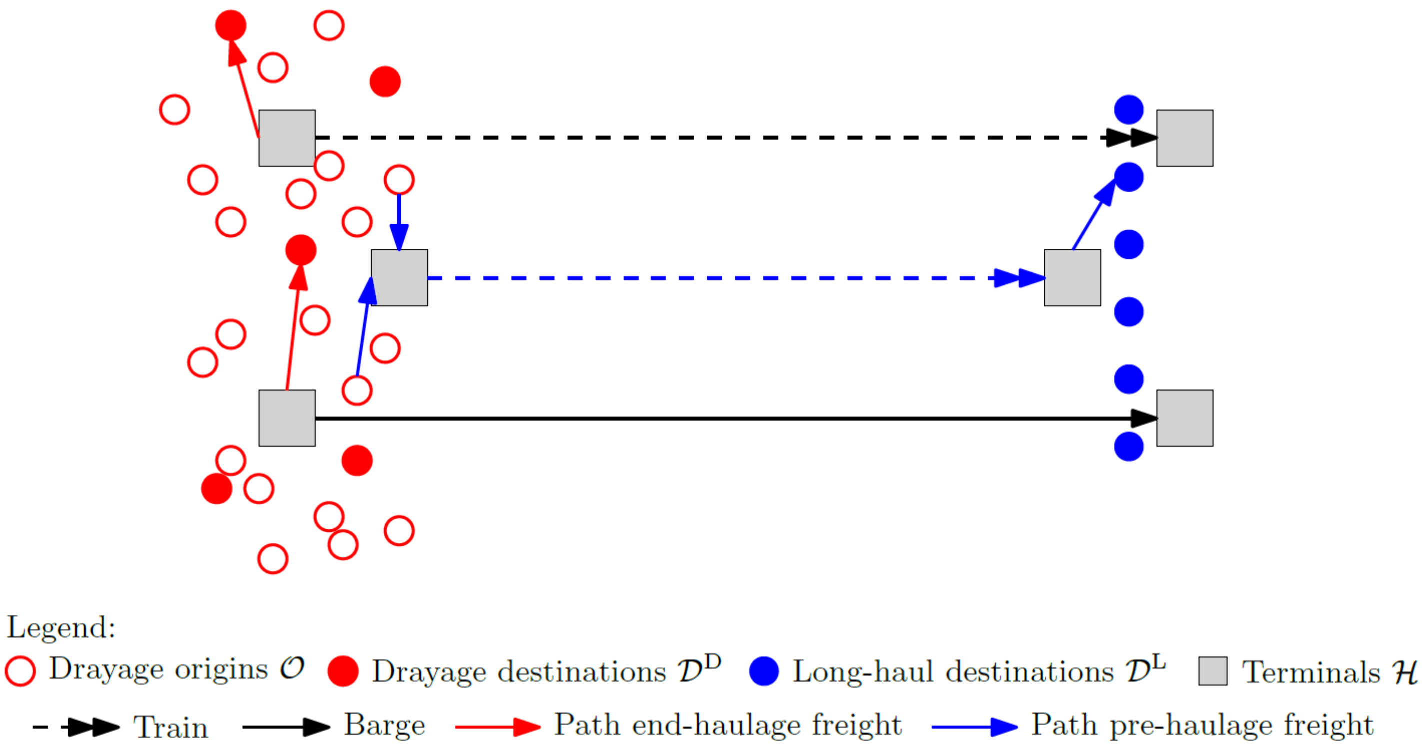
**Source of video: Dutch Institute for Advanced Logistics (DINALOG) www.dinalog.nl*

EXAMPLE TRADE-OFF

TRANSPORTATION OF CONTAINERS FROM TWENTE TO ROTTERDAM

**Source of artwork: Combi Terminal Twente (CTT) www.ctt-twente.nl*

PROBLEM DESCRIPTION

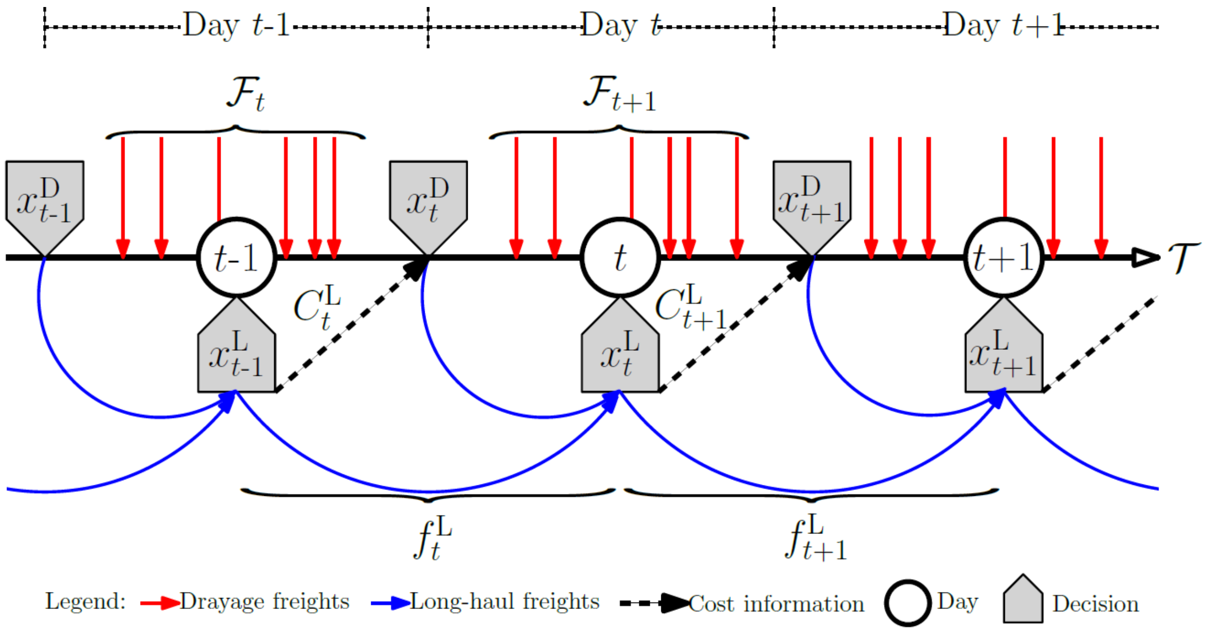


- ***Schedule when (and where) to transport each freight*** to achieve minimum costs over the network and over time.

PROBLEM DESCRIPTION

A stochastic optimization problem over a finite horizon where:

- ***Random freights arrive***
- ***Sequential schedules are made***



SCHEDULING DRAYAGE TRANSPORTATION

Full-Truckload Pickup-and-Delivery Problem with Time-Windows (FTPDPTW) to route trucks and assign terminals:

- ***Assignment of initial terminal for the long-haul of freights***

Scheduling Drayage Operations in Synchromodal Transport

Arturo E. Pérez Rivera and Martijn R.K. Mes

Department of Industrial Engineering and Business Information Systems
University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
{a.e.perezrivera,m.r.k.mes}@utwente.nl

Abstract. We study the problem of scheduling drayage operations in synchromodal transport. Besides the usual decisions to time the pick-up and delivery of containers, and to route the vehicles that transport them, synchromodal transport includes the assignment of terminals for empty and loaded containers. The challenge consists of simultaneously deciding on these three aspects while considering various resource and timing restrictions. We model the problem using mixed-integer linear programming (MILP) and design a matheuristic to solve it. Our algorithm iteratively confines the solution space of the MILP using several adaptations, and based on the incumbent solutions, guides the subsequent iterations and solutions. We test our algorithm under different problem configurations and provide insights into their relation to the three aspects of scheduling drayage operations in synchromodal transport.

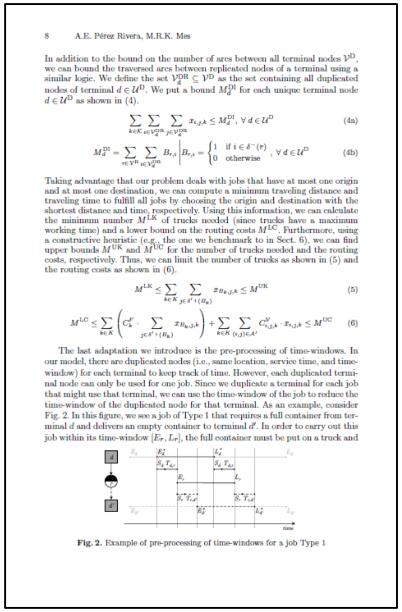
Keywords: Drayage operations, synchromodal transport, matheuristic

1 **Introduction**

During the last years, intermodal transport has received increased attention from academic, industrial, and governmental stakeholders due to potential reductions in cost and environmental impact [10]. To achieve such benefits, these stakeholders have proposed new forms of organizing intermodal transport. One of these new initiatives is synchromodality, which aims to improve the efficiency and sustainability of intermodal transport through flexibility in the choice of mode and in the design of transport plans [12]. However, the potential benefits of any new form of intermodal transport depend to a great extent on the proper planning of drayage operations, also known as pre- and end-haulage or first and last-mile trucking. Drayage operations, which account for 40% of the total transport costs in an intermodal transport chain [5], are the first step where the synchromodal flexibility in transport mode can be taken advantage of. In this paper, we study the scheduling of drayage operations of intermodal transport considering terminal assignment (i.e., long-haul mode) decisions.

Drayage operations in intermodal transport include delivery and pick-up requests of either empty or loaded containers, to and from a terminal where long-haul modes arrive and depart. These operations occur, for example, at a Logistic

T. Bektaş et al. (Eds.), ICCL 2017, LNCS 10572, pp. 404–419, 2017.
https://doi.org/10.1007/978-3-319-68496-3_27



In addition to the bound on the number of arcs between all terminal nodes $\mathcal{V}^{D}$, we can bound the traversed arcs between replicated nodes of a terminal using a similar logic. We define the set $\mathcal{V}_d^{DR} \subseteq \mathcal{V}^{D}$ as the set containing all duplicated nodes of terminal $d \in \mathcal{U}^{D}$. We put a bound M_d^{DI} for each unique terminal node $d \in \mathcal{U}^{D}$ as shown in (4).

$$\sum_{k\in K}\sum_{i\in\mathcal{V}_d^{DR}}\sum_{j\in\mathcal{V}_d^{DR}} x_{i,j,k} \leq M_d^{DI},\ \forall\, d \in \mathcal{U}^{D} \quad (4a)$$

$$M_d^{DI} = \sum_{r\in\mathcal{V}^{R}}\sum_{i\in\mathcal{V}_d^{DR}} B_{r,i} \,\Big|\, B_{r,i} = \begin{cases}1 & \text{if } i \in \delta^{-}(r)\\ 0 & \text{otherwise}\end{cases},\ \forall\, d \in \mathcal{U}^{D} \quad (4b)$$

Taking advantage that our problem deals with jobs that have at most one origin and at most one destination, we can compute a minimum traveling distance and traveling time to fulfill all jobs by choosing the origin and destination with the shortest distance and time, respectively. Using this information, we can calculate the minimum number M^{LK} of trucks needed (since trucks have a maximum working time) and a lower bound on the routing costs M^{LC}. Furthermore, using a constructive heuristic (e.g., the one we benchmark to in Sect. 6), we can find upper bounds M^{UK} and M^{UC} for the number of trucks needed and the routing costs, respectively. Thus, we can limit the number of trucks as shown in (5) and the routing costs as shown in (6).

$$M^{LK} \leq \sum_{k\in K}\sum_{j\in\delta'+(B_k)} x_{B_k,j,k} \leq M^{UK} \quad (5)$$

$$M^{LC} \leq \sum_{k\in K}\left(C_k^{\mathcal{V}}\cdot\sum_{j\in\delta'+(B_k)} x_{B_k,j,k}\right) + \sum_{k\in K}\sum_{(i,j)\in\mathcal{A}'} C_{i,j,k}^{\mathcal{V}}\cdot x_{i,j,k} \leq M^{UC} \quad (6)$$

The last adaptation we introduce is the pre-processing of time-windows. In our model, there are duplicated nodes (i.e., same location, service time, and time-window) for each terminal to keep track of time. However, each duplicated terminal node can only be used for one job. Since we duplicate a terminal for each job that might use that terminal, we can use the time-window of the job to reduce the time-window of the duplicated node for that terminal. As an example, consider Fig. 2. In this figure, we see a job of Type 1 that requires a full container from terminal d and delivers an empty container to terminal d'. In order to carry out this job within its time-window $[E_r, L_r]$, the full container must be put on a truck and

Fig. 2. Example of pre-processing of time-windows for a job Type 1

Pérez Rivera, A.E., Mes, M.R.K. (2017). **Scheduling Drayage Operations in Synchromodal Transport.** *Lecture Notes in Computer Science,* Volume 10572, pp. 404-419. Springer. DOI 10.1007/978-3-319-68496-3_27

SCHEDULING LONG-HAUL TRANSPORTATION

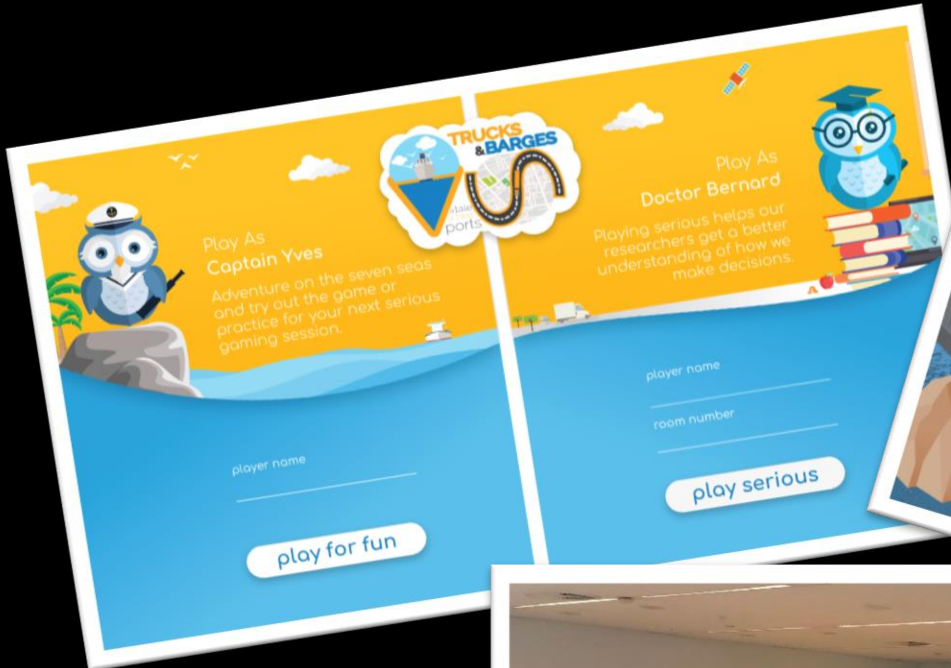
Markov Decision Process (MDP) to consolidate freights in daily barges or postpone their transport:

- ***Arrival of freight is stochastic <u>and</u> dependent on drayage decisions***

Pérez Rivera, A.E., Mes, M.R.K. (2016). **Anticipatory Freight Selection in Intermodal Long-haul Round-trips.** *Transportation Research Part E: Logistics and Transportation Review.* Volume 105: pp. 176-194. Elsevier. DOI 10.1016/j.tre.2016.09.002

INTERMEZZO – SOME PUBLICITY

WWW.TRUCKSANDBARGES.NL



INTEGRATED SCHEDULING

UNIFIED CONTROL OF DRAYAGE AND LONG-HAUL TRANSPORTATION

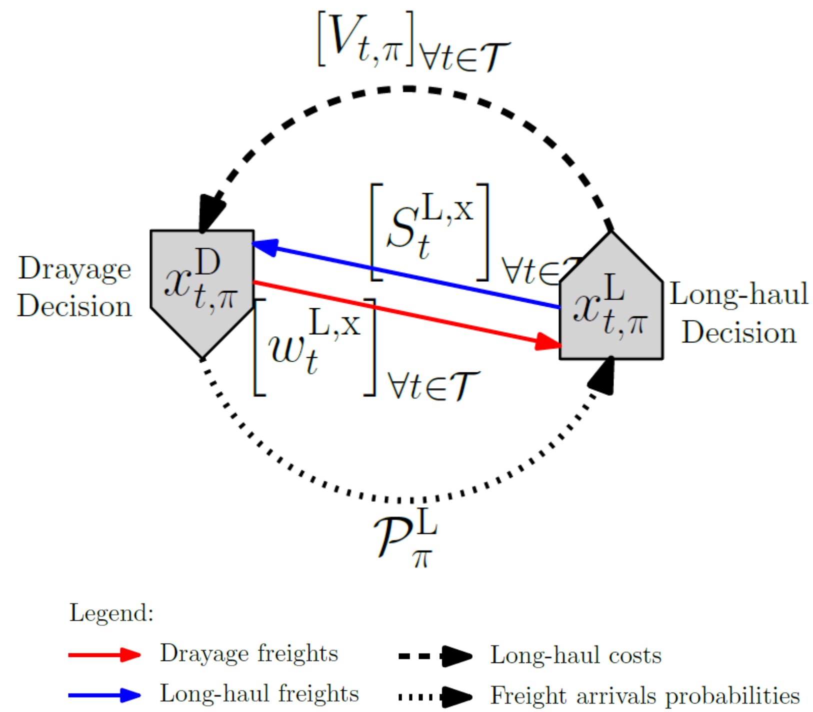
The goal is to ***minimize the total expected network-wide costs***, where the *drayage schedule depends on the long-haul policy*, and where the *long-haul policy depends on the arrivals from the drayage schedule*.

$$\min_{\pi \in \Pi} \mathbb{E}\left[\sum_{t \in \mathcal{T}} \left(z_t^{\mathrm{D}}\left(x_{t,\pi}^{\mathrm{D}}\right) + z_t^{\mathrm{L}}\left(x_{t,\pi}^{\mathrm{L}}\right)\right) \middle| s_0^{\mathrm{L}}, \mathcal{P}^{\mathrm{D}}, \Gamma \right]$$

where

$$x_{t,\pi}^{\mathrm{D}} = \underset{x_t^{\mathrm{D}} \in \mathcal{X}_t^{\mathrm{D}}}{\operatorname{argmin}} \left[\tilde{z}_{t,\pi}^{\mathrm{D}}\left(x_t^{\mathrm{D}}\right)\right]$$

$$\Gamma\left(\mathcal{P}^{\mathrm{D}}, \left[x_{t,\pi}^{\mathrm{D}}\right]_{\forall t \in \mathcal{T}}\right) = \mathcal{P}_\pi^{\mathrm{L}}$$

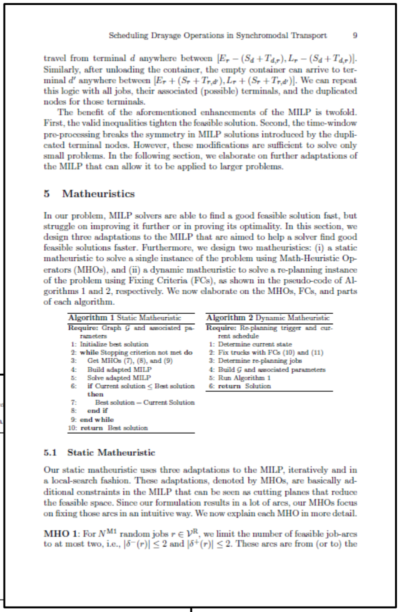


HEURISTIC APPROACH

HEURISTICS FOR THE DRAYAGE SCHEDULE AND LONG-HAUL POLICY

- We use a ***Matheuristic (MH)*** for scheduling drayage transportation, which uses various cuts based on the 'terminal assignment cost' resulting from the long-haul policy.

- We use an ***Approximate Dynamic Programming (ADP)*** algorithm for learning a long-haul policy, i.e., Value Function Approximation (VFA), based on the observed distributions from a simulation of the MH.



Scheduling Drayage Operations in Synchromodal Transport 9

travel from terminal d anywhere between $[E_r - (S_d + T_{d,r}), L_r - (S_d + T_{d,r})]$. Similarly, after unloading the container, the empty container can arrive to terminal d' anywhere between $[E_r + (S_r + T_{r,d'}), L_r + (S_r + T_{r,d'})]$. We can repeat this logic with all jobs, their associated (possible) terminals, and the duplicated nodes for those terminals.

The benefit of the aforementioned enhancements of the MILP is twofold. First, the valid inequalities tighten the feasible solution. Second, the time-window pre-processing breaks the symmetry in MILP solutions introduced by the duplicated terminal nodes. However, these modifications are sufficient to solve only small problems. In the following section, we elaborate on further adaptations of the MILP that can allow it to be applied to larger problems.

5 Matheuristics

In our problem, MILP solvers are able to find a good feasible solution fast, but struggle on improving it further or in proving its optimality. In this section, we design three adaptations to the MILP that are aimed to help a solver find good feasible solutions faster. Furthermore, we design two matheuristics: (i) a static matheuristic to solve a single instance of the problem using Math-Heuristic Operators (MHOs), and (ii) a dynamic matheuristic to solve a re-planning instance of the problem using Fixing Criteria (FCs), as shown in the pseudo-code of Algorithms 1 and 2, respectively. We now elaborate on the MHOs, FCs, and parts of each algorithm.

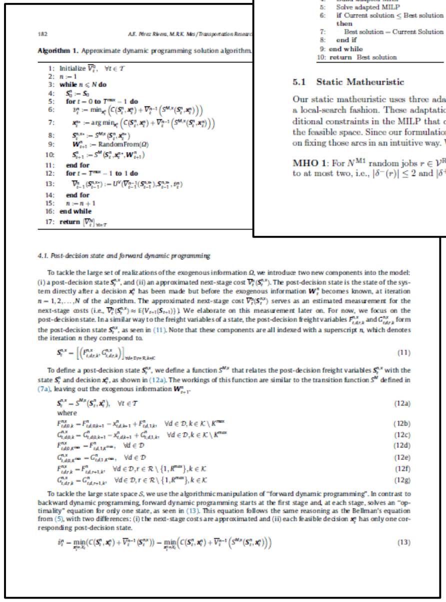
```
Algorithm 1 Static Matheuristic
Require: Graph G and associated parameters
 1: Initialize best solution
 2: while Stopping criterion not met do
 3:     Get MHOs (7), (8), and (9)
 4:     Build adapted MILP
 5:     Solve adapted MILP
 6:     if Current solution ≤ Best solution then
 7:         Best solution = Current Solution
 8:     end if
 9: end while
10: return Best solution
```

```
Algorithm 2 Dynamic Matheuristic
Require: Re-planning trigger and current schedule
 1: Determine current state
 2: Fix trucks with FCs (10) and (11)
 3: Determine re-planning jobs
 4: Build G and associated parameters
 5: Run Algorithm 1
 6: return Solution
```

5.1 Static Matheuristic

Our static matheuristic uses three adaptations to the MILP, iteratively and in a local-search fashion. These adaptations, denoted by MHOs, are basically additional constraints in the MILP that can be seen as cutting planes that reduce the feasible space. Since our formulation results in a lot of arcs, our MHOs focus on fixing those arcs in an intuitive way. We now explain each MHO in more detail.

MHO 1: For N^{M1} random jobs $r \in \mathcal{V}^R$, we limit the number of feasible job-arcs to at most two, i.e., $|\delta^-(r)| \leq 2$ and $|\delta^+(r)| \leq 2$. These arcs are from (or to) the



182 A.E. Pérez Rivera, M.R.K. Mes / Transportation Researc[h]

Algorithm 1. Approximate dynamic programming solution algorithm.

```
 1: Initialize V̄_t^0, ∀t ∈ T
 2: n := 1
 3: while n ≤ N do
 4:   S_0^n := S_0
 5:   for t = 0 to T^max − 1 do
 6:     v̂_t^n := min_{x_t^n} ( C(S_t^n, x_t^n) + V̄_t^{n−1}( S^{M,x}(S_t^n, x_t^n) ) )
 7:     x_t^{n*} := arg min_{x_t^n} ( C(S_t^n, x_t^n) + V̄_t^{n−1}( S^{M,x}(S_t^n, x_t^n) ) )
 8:     S_t^{n,x*} := S^{M,x}(S_t^n, x_t^{n*})
 9:     W_{t+1}^n := RandomFrom(Ω)
10:     S_{t+1}^n := S^M(S_t^n, x_t^{n*}, W_{t+1}^n)
11:   end for
12:   for t = T^max − 1 to 1 do
13:     V̄_{t−1}^n(S_{t−1}^{n,x*}) := U^V(V̄_{t−1}^{n−1}(S_{t−1}^{n,x*}), S_{t−1}^{n,x*}, v̂_t^n)
14:   end for
15:   n := n + 1
16: end while
17: return [V̄_t^N]_{∀t∈T}
```

4.1. Post-decision state and forward dynamic programming

To tackle the large set of realizations of the exogenous information Ω, we introduce two new components into the model: (i) a post-decision state $S_t^{n,x}$, and (ii) an approximated next-stage cost $\overline{V}_t^n(S_t^{n,x})$. The post-decision state is the state of the system directly after a decision x_t^n has been made but before the exogenous information W_t^n becomes known, at iteration $n = 1, 2, \ldots, N$ of the algorithm. The approximated next-stage cost $\overline{V}_t^n(S_t^{n,x})$ serves as an estimated measurement for the next-stage costs (i.e., $\overline{V}_t^n(S_t^{n,x}) \approx \mathbb{E}\{V_{t+1}(S_{t+1})\}$). We elaborate on this measurement later on. For now, we focus on the post-decision state. 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 $S_t^{n,x}$, as seen in (11). Note that these components are all indexed with a superscript n, which denotes the iteration n they correspond to.

$$S_t^{n,x} = \left[\left(F_{t,d,r,k}^{n,x}, G_{t,d,r,k}^{n,x}\right)\right]_{\forall d\in\mathcal{D}, r\in\mathcal{R}, k\in\mathcal{K}} \quad (11)$$

To define a post-decision state $S_t^{n,x}$, we define a function $S^{M,x}$ that relates the post-decision freight variables $S_t^{n,x}$ with the state S_t^n and decision x_t^n, as shown in (12a). The workings of this function are similar to the transition function S^M defined in (7a), leaving out the exogenous information W_{t+1}^n.

$$S_t^{n,x} = S^{M,x}(S_t^n, x_t^n), \quad \forall t \in \mathcal{T} \quad (12a)$$

where

$$F_{t,d,0,k}^{n,x} = F_{t,d,0,k+1}^n - x_{t,d,k+1}^n + F_{t,d,1,k}^n, \quad \forall d \in \mathcal{D}, k \in \mathcal{K} \setminus K^{max} \quad (12b)$$

$$G_{t,d,0,k}^{n,x} = G_{t,d,0,k+1}^n - x_{t,d,k+1}^n + G_{t,d,1,k}^n, \quad \forall d \in \mathcal{D}, k \in \mathcal{K} \setminus K^{max} \quad (12c)$$

$$F_{t,d,0,K^{max}}^{n,x} = F_{t,d,1,K^{max}}^n, \quad \forall d \in \mathcal{D} \quad (12d)$$

$$G_{t,d,0,K^{max}}^{n,x} = G_{t,d,1,K^{max}}^n, \quad \forall d \in \mathcal{D} \quad (12e)$$

$$F_{t,d,r,k}^{n,x} = F_{t,d,r+1,k}^n, \quad \forall d \in \mathcal{D}, r \in \mathcal{R} \setminus \{1, R^{max}\}, k \in \mathcal{K} \quad (12f)$$

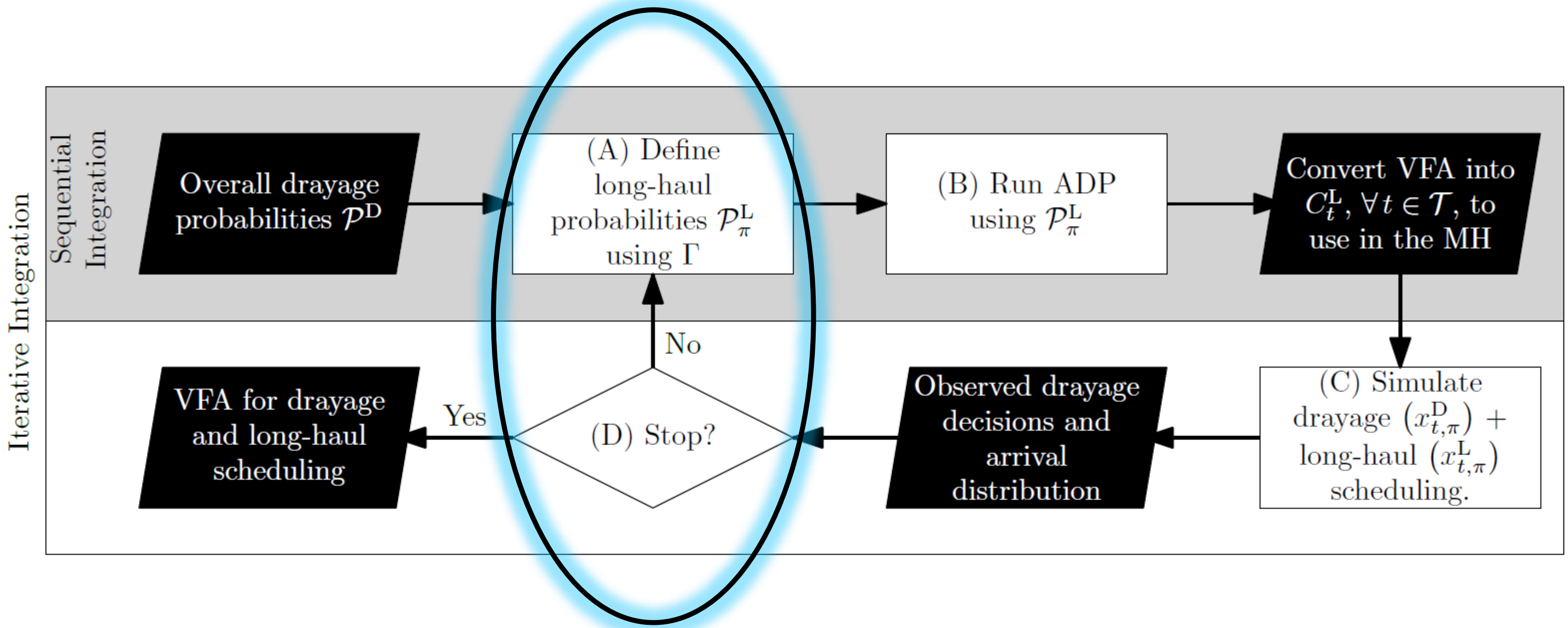
$$G_{t,d,r,k}^{n,x} = G_{t,d,r+1,k}^n, \quad \forall d \in \mathcal{D}, r \in \mathcal{R} \setminus \{1, R^{max}\}, k \in \mathcal{K} \quad (12g)$$

To tackle the large state space $\mathcal{S}$, we use the algorithmic manipulation of "forward dynamic programming". In contrast to backward dynamic programming, forward dynamic programming starts at the first stage and, at each stage, solves an "optimality" equation for only one state, as seen in (13). This 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 x_t^n has only one corresponding post-decision state.

$$\hat{v}_t^n = \min_{x_t^n \in \mathcal{X}_t}\left(C(S_t^n, x_t^n) + \overline{V}_t^{n-1}(S_t^{n,x})\right) = \min_{x_t^n \in \mathcal{X}_t}\left(C(S_t^n, x_t^n) + \overline{V}_t^{n-1}\left(S^{M,x}(S_t^n, x_t^n)\right)\right) \quad (13)$$

HEURISTIC APPROACH

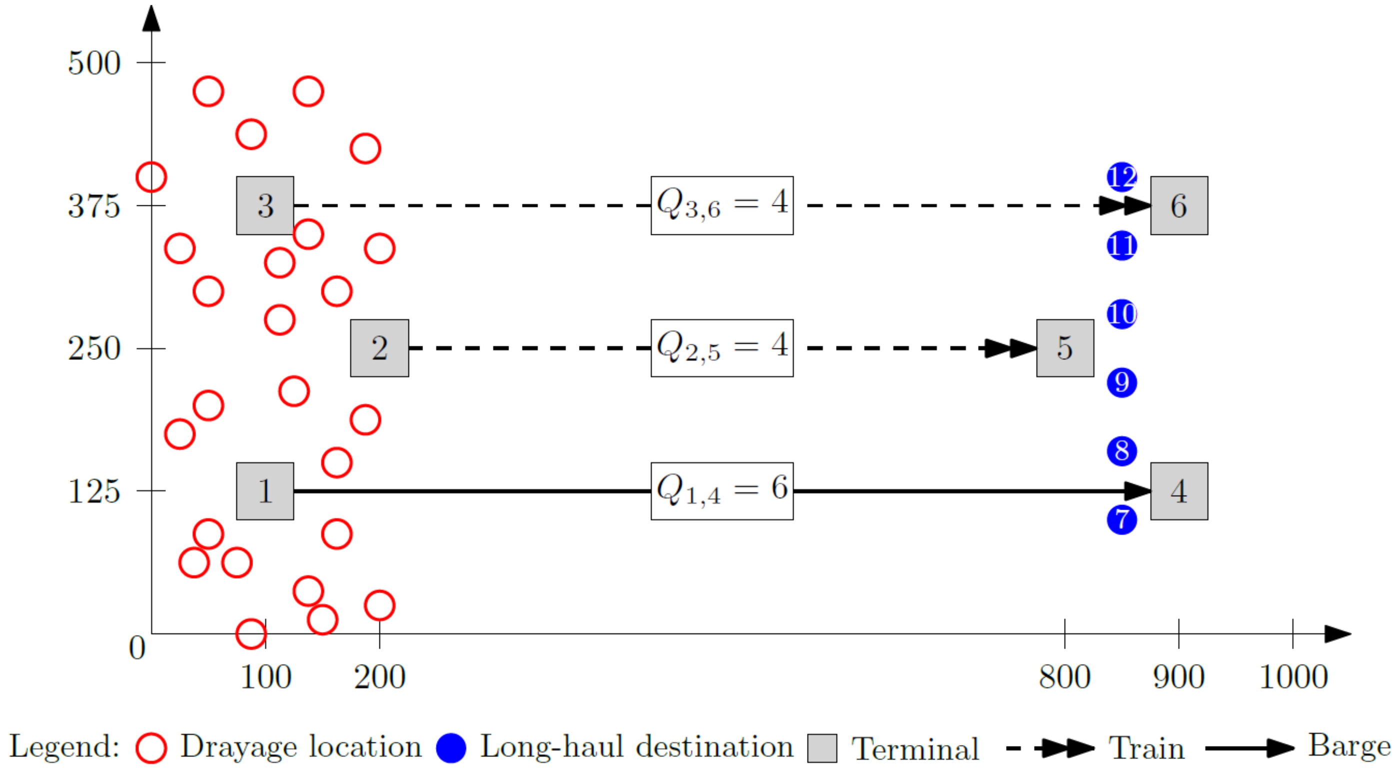
INTEGRATION OF THE TWO HEURISTICS



UNIVERSITY OF TWENTE.

NUMERICAL EXPERIMENTS

INSTANCES SETUP



Freight demand:
20 freights per day (≈Poisson dist.)

Drayage location:
Random (R) or Clustered (C).

Drayage type:
Pre-haulage (P) or End-haulage (E).

Long-haul Destinations:
Balanced (B) or Unbalanced (U).

NUMERICAL EXPERIMENTS

EXPERIMENTAL PHASES

We divide the experiments in two phases:

1. ***Calibration phase:***
   - Settings for heuristic parameters.
   - Influence in drayage and long-haul schedules.
2. ***Evaluation phase:***
   - Savings with respect to a benchmark approach commonly found in practice.
   - Sensitivity to different cost setups.

NUMERICAL EXPERIMENTS

CALIBRATION PHASE – PARAMETERS FOCUS ON DRAYAGE OR LONG-HAUL

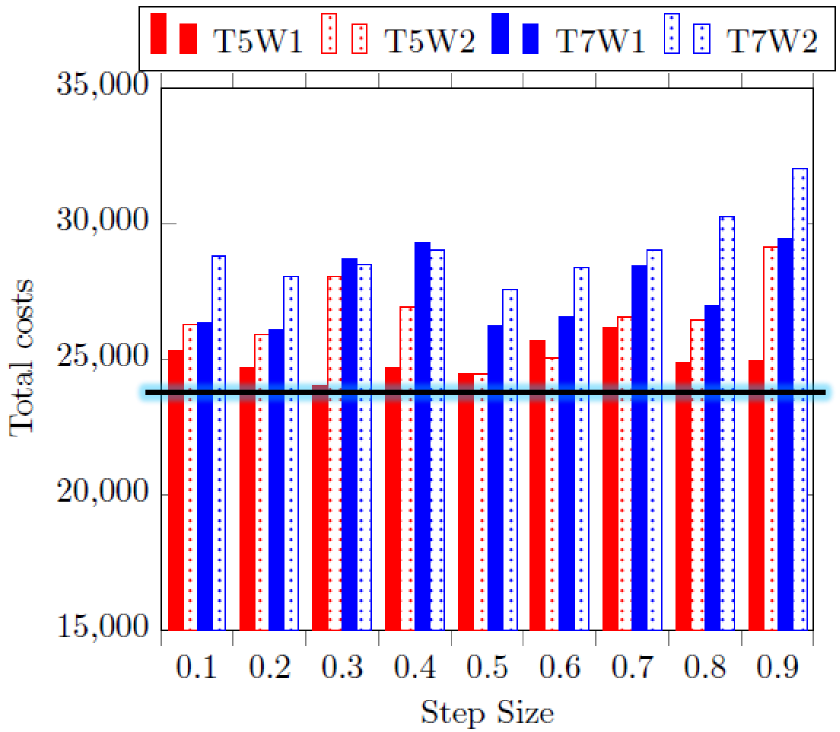


Figure 6.10: *Total Costs C-P-U*

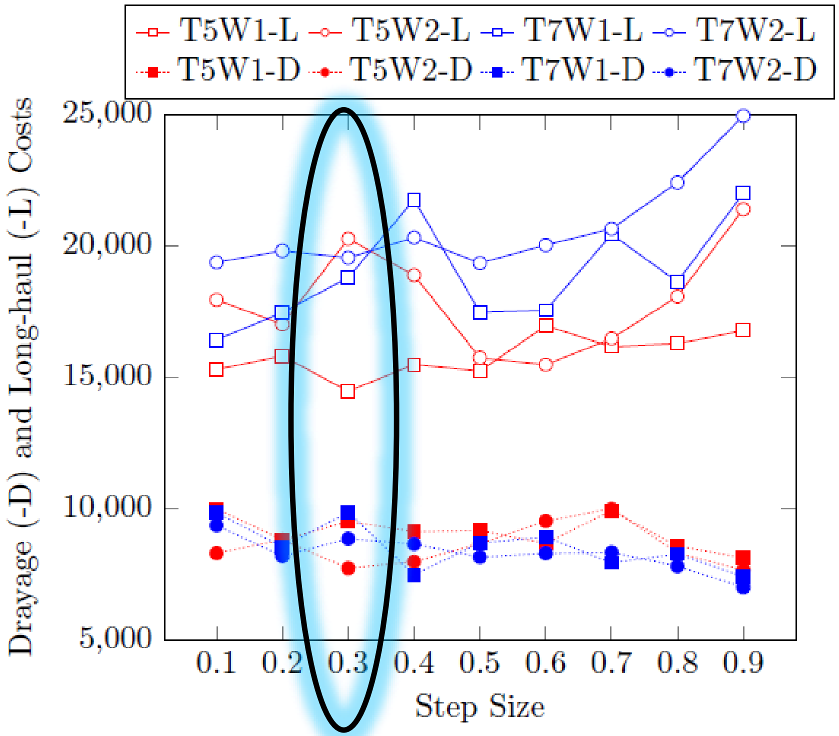


Figure 6.11: *Individual Costs C-P-U*

NUMERICAL EXPERIMENTS

EVALUATION PHASE – NORMAL COST SETUP

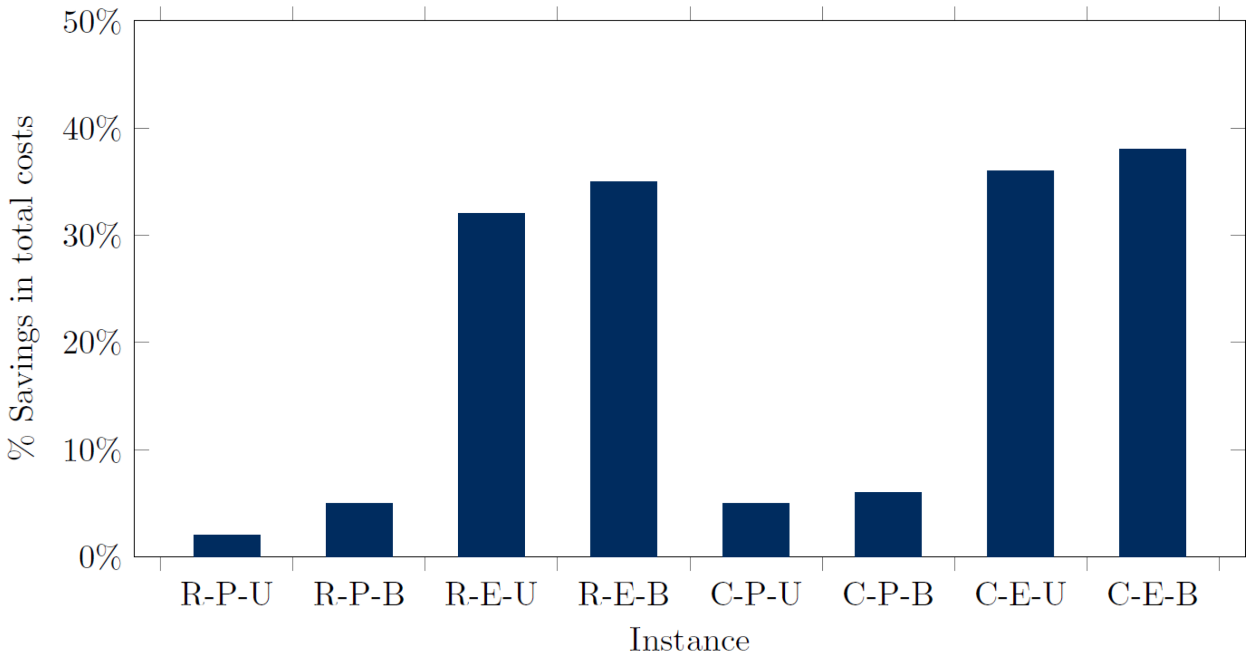


Table 1: Percentage difference with the benchmark in normal drayage-cost setup

Instance	R-P-U	R-P-B	R-E-U	R-E-B	C-P-U	C-P-B	C-E-U	C-E-B
Long-haulCosts	-10%	-14%	-63%	-65%	-14%	-13%	-63%	-65%
DrayageCosts	17%	18%	33%	32%	16%	12%	21%	22%
Long-haulUtilization	4%	1%	-55%	-55%	5%	0%	-56%	-55%
Pre-haulageClosest	-21%	-27%	-82%	-81%	-37%	-35%	-81%	-82%

NUMERICAL EXPERIMENTS
EVALUATION PHASE – COST SENSITIVITY

Table 6.5: *Percentage difference with the benchmark in high drayage-cost setup*

Instance	Costs			Long-haul	Pre-haulage to
	Total	Long-haul	Drayage	Utilization	closest terminal
R-P-U	3%	-12%	6%	4%	5%
R-P-B	5%	-5%	7%	0%	4%
R-E-U	13%	-62%	29%	-55%	-72%
R-E-B	12%	-63%	30%	-55%	-74%
C-P-U	-9%	50%	-20%	-30%	18%
C-P-B	-12%	38%	-23%	-27%	21%
C-E-U	4%	-64%	19%	-55%	-71%
C-E-B	3%	-64%	18%	-55%	-73%

CONCLUSIONS

- We proposed the ***integration of a MH for drayage scheduling and an ADP for long-haul scheduling*** through (i) the inclusion of long-haul assignment costs in drayage decisions, and (ii) an improved VFA in the long-haul decisions.

- Numerical experiments show that ***our integrated scheduling approach performs up to 38% better than separated scheduling*** in terms of total network costs, with larger drayage costs.

- Further ***research on the integration mechanisms of the MH and ADP, and their calibration,*** is necessary to achieve the most of integrated scheduling in synchromodal transport.

UNIVERSITY OF TWENTE. DINALOG Dutch Institute for Advanced Logistics

THANKS FOR YOUR ATTENTION!

ARTURO E. PÉREZ RIVERA

PhD Candidate

Department of Industrial Engineering and Business Information Systems

University of Twente, The Netherlands

https://www.utwente.nl/bms/iebis/staff/perezrivera/

a.e.perezrivera@utwente.nl