Simultaneous optimization of speed and buffer times for robust transportation systems

Judith Mulder[1], Willem van Jaarsveld[2], and Rommert Dekker[1]

[1]Econometric Institute, Erasmus University Rotterdam, Rotterdam, the Netherlands, {mulder, rdekker}@ese.eur.nl

[2]School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, the Netherlands, W.L.v.Jaarsveld@tue.nl

Econometric Institute Report EI2016-36

Abstract

Transport companies often have a published timetable. To maintain timetable reliability despite delays, companies include buffer times during timetable development, and adjust the traveling speed during timetable execution. We develop an approach that can integrate decisions at different time scales (tactical and operational). We model execution of the timetable as a stochastic dynamic program (SDP). An SDP is a natural framework to model random events causing (additional) delay, propagation of delays, and real-time speed adjustments. However, SDPs alone cannot incorporate the buffer allocation, as buffer allocation requires to choose the same action in different states of the SDP. Our objective is finding the buffer allocation that yields the SDP which has minimal long run average costs. We derive several analytical insights into the model. We prove that costs are joint convex in the buffer times, and develop theory in order to compute subgradients. Our optimal algorithm for buffer time allocation is based on these results. Our case study considers container vessels sailing a round tour consisting of 14 ports based on Maersk data. Our algorithm finds the optimal timetable in less than 80 seconds. The optimal timetable yields cost reductions of about six to ten million USD per route per year in comparison to the current timetable.

1 Introduction

Timetables are used in container shipping, airlines and public transport to communicate planned arrival and departure times in advance to customers. However, delays are inevitable while executing the timetable, making the arrival times uncertain. Maintaining timetable reliability despite these delays is crucial: The timetable is relied upon by passengers and freight forwarders.

Transport companies combine two main methods to ensure a reliable schedule. Firstly, during timetable *development*, a more delay-resistant planning may be obtained by including buffer or slack time. In liner shipping, for example, the planned arrival at the port of Jeddah could be 9 days after the planned departure from Rotterdam, while the trip takes only 8 days on average when sailing at design speed. The 24 hours buffer time can capture (part of) a delay. But buffers increase the nominal travel time and therefore costs. So limited buffer time is available, and strategic allocation along a route is key. Secondly, during *execution* of the timetable, a ship may sail faster to recover from a delay with respect to the timetable. But increasing speed is very costly: Figure 1 shows that sailing at 28 knots instead of 14 knots increases fuel consumption per nautical mile by about 350% for a 8,000 TEU ship. For a trip from Rotterdam to Jeddah, this corresponds to over 1 million USD at a bunker price of 600 USD/ton, or over $6,000$ tons of CO_2 (Cariou 2011). Speed adjustments also have significant impact for other transport modes: For example doubling the average speed of a metro on a track roughly quadruples energy consumption (Binder and Albrecht 2012).

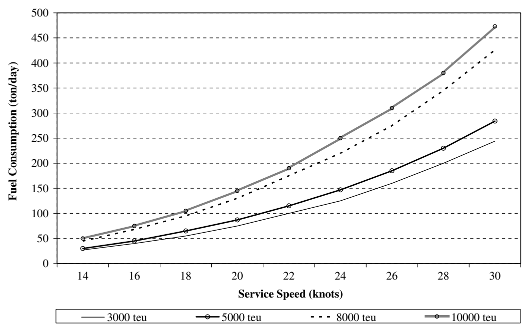

Figure 1: Fuel (bunker) consumption of several container ships at different travel speeds (from Notteboom and Vernimmen 2009).

We will focus on the liner shipping application throughout the paper. However, the methodology that we develop also applies to timetabling for metro's, and further research building onto this work could lead to useful applications in train and aircraft operations planning as well.

Our model consists of two levels. On the tactical level, construction of the timetable involves the allocation of buffers. We consider the situation in which routes have already been decided upon, so the available amount of total buffer time is known and given. On the operational level, the timetable is executed: random events cause (additional) delay, travel speed is optimized, and late arrivals and departures are penalized. We model the operational planning level as a Stochastic Dynamic Program (SDP). This SDP accurately models real-time recovery actions such as speed optimization, as well as propagation of delays from port call to port call. However, the buffer times are exogenous to this SDP: Different buffer time allocations yield SDPs that are structurally different. The optimal buffer allocation yields the SDP which has minimal long run average costs.

We contribute a theoretical analysis of the problem. E.g., we show that speed should increase as the delay with respect to the schedule increases, and provide a bound on the maximum speed increase that should result from additional delay. We then focus on optimizing the buffer time allocation. We develop theoretical results in order to optimally combine the buffer allocation decision on the tactical level, and speed optimization (as part of the SDP) on the operational level. We prove, under mild assumptions, that the minimum costs of operating the timetable are convex in the buffer time allocation. Additional theory is developed, leading to a simple and efficient approach for computing subgradients. Our algorithm for optimal buffer time allocation is based on these results. This algorithm finds the optimal solution in less than 80 seconds. Hence, our algorithm is not only exact, but also fast. We note that our algorithm is the first exact algorithm for this problem. We then report on a case study based on Maersk data for a round tour consisting of 14 ports.

The remainder of this paper is organized as follows. Section 2 reviews the existing literature. Section 3 provides a detailed description of the model. The theoretical analysis of the model is presented in Section 4. In Section 5 we develop the algorithm, and provide further theoretical results underlying the algorithm. Section 6 describes computational experiments. We conclude in Section 7.

2 Literature review

Timetables are often used in air, railway and maritime transport. Multiple studies have already been performed on managing, recovering and preventing delays in these transport modes. Wu and Caves (2003) and Wu (2005) show the importance of buffer time allocation on punctuality in air transport using a simulation approach. Clausen et al. (2010) give an overview on disruption management studies in the airline industry. They distinguish the reviewed studies in two different groups: *delay recovery* and *robust planning*. This distinction also mainly holds for railway and maritime transport.

Delay recovery aims to find a recovery policy such that delays in the existing timetables as a result of small disruptions are recovered from in order to minimize a certain objective (e.g. Wang and Meng (2012a), Brouer et al. (2013), Li et al. (2015a), Li et al. (2015b) in liner shipping, Corman et al. (2010), Binder and Albrecht (2012) in public transport and Rosenberger et al. (2003), Petersen et al. (2012), Arikan et al. (2016), Aktürk et al. (2014), Maher (2015) in air transport). In all these studies, the goal is to optimize recovery strategies after the occurrence of disruptions, but note that these studies encompass a wide range of modeling and solution approaches. Recovery strategies include travel time (or speed) adjustments and rerouting decisions. However, the influence of the available buffer time in the existing timetables is not considered in these studies. Visentini et al. (2014) review recovery actions in general transportation, while Psaraftis and Kontovas (2013) overview speed models for energy efficient maritime transportation.

Li et al. (2015a,b) propose a dynamic programming approach to determine the optimal recovery policy after a major disruption and under regular uncertainties and a major disruption respectively. Li et al. (2015a) consider problems with different recovery actions: with only speeding decisions, with speeding and port skipping decisions and with speeding, port skipping and port swapping decisions. They prove some structural results for the problem with only speeding decisions under one major disruption and no further uncertainty. Li et al. (2015b) extend the formulation to also include regular uncertainties, but they limit the recovery actions to only include speeding decisions. They prove some structural results for problems with only regular uncertainties and for problems with regular uncertainties and one major disruption and consider both problems with and without an earliest handling time constraint on the terminal operations.

Robust planning constructs timetables which perform well under uncertainty. Two different approaches are used to construct robust timetables. First, the total available buffer time in an existing schedule can be rearranged in order to obtain more robust networks (e.g. Kroon et al. (2007), Kroon et al. (2008), Fischetti et al. (2009), Hassannayebi et al. (2014), Wu et al. (2015) in public transport, Lan et al. (2006), AhmadBeygi et al. (2010), Chiraphadhanakul and Barnhart (2013) in air transport). All these studies only consider the allocation of buffer times (also framed as time supplements or slack time) in the schedule, but do not consider recovery strategies when disruptions occur. Second, schedules satisfying certain robustness concepts can be constructed. Du et al. (2015) and Norlund et al. (2015) describe methods to design robust schedules that minimize the fuel consumption in shipping taking into account uncertain weather conditions. However, only the fuel consumption of the planned schedule without recovery strategies is taken into account. Cucala et al. (2012) and Duran et al. (2015) consider similar problems for respectively public and air transport. These papers also determine an optimal speed policy together with the constructed timetable, but the speed is independent of incurred delays.

Delay-resistant timetables and real-time recovery actions are interrelated, and in recent years there has been increasing interest in approaches that incorporate both. Various approaches to incorporate wait-depart decisions in timetabling exist: A genetic algorithm (Engelhardt-Funke and Kolonko 2004), a light robustness concept for timetabling combined with scenario-based wait-depart decisions (Liebchen et al. 2010), and a recoverable robustness concept that aims to find timetables that are recoverable when disruptions occur (Cicerone et al. 2009, 2012). Furthermore, Gong et al. (2014) develop a two-stage approach based on a genetic algorithm to solve the integrated problem. The first stage considers the timetable optimization and the second stage the speed optimization. Two-stage stochastic programming (SP) with sample average approximation applies naturally to robust timetabling under stochastic delays (Kroon et al. 2007, 2008, Fischetti et al. 2009), and Qi and Song (2012) and Wang and Meng (2012b) have extended this approach to take into account speed adjustments in liner shipping.

2.1 Contribution

We consider timetabling and speed optimization under stochastic delays. We use a dynamic program to select the optimal speed dynamically, taking into account present fuel costs and future fuel and delay costs. Two papers have recently pioneered this approach for optimization of speed and other recovery actions in liner shipping. Li et al. (2015a) use a deterministic dynamic program to recover from a single larger disruption. Li et al. (2015b) present work that is more closely related to our operational problem because the ship in their model faces many small and large disruptions over time. In fact, the operational problem that we consider in this paper is the same as the problem with regular uncertainties under terminal operations with the earliest handling time constraints considered in Li et al. (2015b).

Our main contribution is simultaneously considering optimal dynamic speed adjustments and timetable optimization (in the form of the tactical buffer allocation). Optimization of the timetable is not considered in Li et al. (2015a,b). Because it is impossible to integrate the one-time buffer allocation decisions into the SDP framework, considerable new theory is developed in our paper to arrive at a tractable algorithm. E.g. we prove several additional properties for the operational problem (e.g. Conjecture 1 in Li et al. 2015b). Moreover, we derive many new insights with respect to tactical buffer allocations. These latter insights lead to the first exact solution approach for simultaneous optimization of buffer allocation and optimal dynamic speed selection.

3 The model

Consider a round tour with a fixed sequence of port calls and a total planned duration of T time units. A ship sails a *route* consisting of R round tours for a planned duration of RT. Eventually, we let $R \to \infty$ and focus on the long run average costs, which can be obtained by averaging the total route costs over time. Route costs consist in the costs of delayed port arrivals and departures and the costs of (optimally) performing recovery actions such as speed adjustments. The goal is to construct an optimal schedule by dividing the T time units over the round tour in such a way that the long run average costs are minimized.

The model combines a problem on the tactical planning level with a problem on the operational planning level. We will illustrate these two problems and their dependence with a small example of a ship sailing round tours. Suppose we have twenty hours available to complete each tour and it takes at least five hours to sail from port 1 to port 2, at least eight hours to sail from port 2 to port 3 and at least three hours to sail from port 3 back to port 1. In this case, the available buffer time is $20 - 5 - 8 - 3 = 4$ hours. In the tactical planning level, we need to decide to which sea legs these hours should be allocated. However, we do not know yet which delays the ship will incur while operating the route, since this information only becomes available at the operational level. The allocated buffer times hedge against this uncertainty. Furthermore, at the operational planning level, ships can adjust their sailing speed to recover from the incurred delays. Obviously, the amount of buffer allocated to each leg in the tactical planning level will influence the speed adjustments at the operational level; if we decide to allocate all four hours of buffer to the first sea leg, the ship will need less speeding up on this leg compared to the situation in which we only allocate one hour of buffer to this leg. In the remainder of this section, we formally model the tactical and operational level problems.

The tactical planning level

Denote the ports visited in the round tour by $P = \{1, \ldots, |P|\}$. Rounds start in port 1, visit ports $2, 3, \ldots, |P|-1, |P|$ and then return to port 1, after which a new round starts. The route consists of R round tours and $N = R|P| + 1$ port calls (including the final port call in port 1). Let $n \in \{1, \ldots, N\}$ index the port calls. The nth port call is made at port $p[n]$. Thus, $p[n] := p$ for $n = p, |P| + p, 2|P| + p, \ldots$, with $p \in P$.

Let t_n^{arr} and t_n^{dep} respectively denote the *planned* arrival and departure time of port call n. The planned arrival time of port call $n+1$ equals the planned departure time of port call n plus the planned sailing time. This planned sailing time consists of the fixed minimum sailing time needed between ports $p[n]$ and $p[n+1]$ (denoted by $t_{p[n]}^s$) and the buffer time included in the sea leg, which is a decision variable that will be denoted by $\vec{B}_{p[n]}$. Thus $t_{n+1}^{arr} = t_n^{dep} + t_{p[n]}^s + \vec{B}_{p[n]}$. The requirement of a cyclic schedule means that buffer time and minimum sailing time for a specific sea leg must be the same for each round. (The notation $t_{p[n]}^s$ and $\vec{B}_{p[n]}$ effectively enforces this requirement, see the definition of $p[n]$.) The planned departure time of the ship for port call n is simply the planned arrival time plus the fixed port time, which will be denoted by $t^p_{p[n]}$. Thus $t^{dep}_n = t^{arr}_n + t^p_{p[n]}$. The results in this paper can be extended to optimize buffers for the ports as well, but we do not include such variables because buffer times in ports are expensive and therefore uncommon.

We can set $t^{dep}_1 := 0$ without loss of generality. Then, all planned arrival and departure times for the remaining $R|P|$ port calls follow from the above recursive relations once we fix $\vec{B} := (\vec{B}_1, \ldots, \vec{B}_{|P|}) \in \mathbb{Z}^{|P|}_{\geq 0}$. So finding a schedule consists in fixing $\vec{B}$. Buffer times need to be integer valued, because the timetable is always communicated in integer time units to customers. The requirement that the total planned duration equals T implies that $\vec{B}$ should satisfy $\sum_{p\in P} \vec{B}_p = B$, where $B := T - \sum_{p\in P} t^p_p - \sum_{p\in P} t^s_p$. (We assume $T \geq \sum_{p\in P} t^p_p + \sum_{p\in P} t^s_p$, such that $B \geq 0$.) Since liner ships operate on weekly schedules, the total planned duration T will be an integer multiple of the amount of time units in one week.

The operational planning level

While the ship sails the route, unforeseen events cause the ship to be delayed with respect to the planned timetable, i.e. the *planned* arrival and departure times t^{arr}_n and t^{dep}_n. Discussions at a large liner carrier have revealed that both delays in the port and delays during the sea leg are important (cf. Wang and Meng 2012a, p. 616). Therefore, let $X^p_n \geq 0$ and $X^s_n \geq 0$ denote the random delay incurred during port call n, and in the sea leg after port call n, respectively. The random variables X^p_n and X^s_n are assumed to be independent of each other, and of all other random variables, in particular of $X^p_{n'}$ and $X^s_{n'}$ for $n \neq n'$. Distributions are arbitrary, but the random delay in a port in a specific position in the round trip is identically distributed in each round trip. Thus X^p_n and $X^p_{n'}$ are identically distributed if $p[n] = p[n']$. Similarly, X^s_n and $X^s_{n'}$ are identically distributed if $p[n] = p[n']$.

To reduce the delay with respect to the schedule, the liner company can perform two types of recovery actions. *Speed adjustments* during the sea leg are the preferred approach to deal with delays. But in case of excessive delays, *extreme (recovery) actions* in the port are sometimes taken in practice, such as cut-and-go. In cut-and-go, the vessel will stop (un)loading and will immediately leave the port. Let τ_n be the difference in the time used to sail from port $p[n]$ to port $p[n+1]$ (excluding unforeseen delays) and the minimum sailing time needed. We will refer to τ_n as the additional sailing time or the sailing time action. Let γ_n denote the time recovered by the extreme recovery action in the nth port, which is taken after the port delay is revealed. Note that τ_n and γ_n are online decision variables, these decisions are taken dynamically in each port and before each sea leg. In contrast, all buffer times $\vec{B}$ are decided upon before the ship starts sailing the route.

The following recursive relations for $1 \leq n < N$ govern the propagation of the delay during the trip:

$$d_{n+1}^{arr} = (d_n^{dep} + \tau_n - \vec{B}_{p[n]} + X_n^s)^+, \quad (1)$$

$$d_n^{dep} = (d_n^{arr} + X_n^p - \gamma_n)^+. \quad (2)$$

where $x^+ = \max\{x, 0\}$. Since ships have to adhere to the berthing plans made by terminal operators, we assume that ships cannot arrive early in a port. And a ship is not allowed to depart earlier than the schedule, because export containers may arrive just in time to be loaded according to the schedule.

Costs

For $p \in P$, let $\mathcal{D}_p^{arr}(d)$ and $\mathcal{D}_p^{dep}(d)$ be respectively the cost of arriving in and departing from port p of the round tour with a delay of d time units with respect to the schedule. We assume that both $\mathcal{D}_p^{arr}(d)$ and $\mathcal{D}_p^{dep}(d)$ are convex and increasing in d. Penalizing the average delay satisfies this assumption and is arguably the most intuitive approach for measuring delays. This latter approach is common (e.g. Kroon et al. 2008, Fischetti et al. 2009), but more general delay cost models have also been proposed (Wang and Meng 2012b).

Let $\mathcal{F}_p(\tau)$ denote the fuel cost incurred between port p and the next port when using a sailing time of $t_p^s + \tau$ time units. $\mathcal{F}_p(\tau)$ is decreasing and convex in τ. Indeed, for economic sailing speeds the bunker consumption rate can be accurately approximated by a constant times the third power of sailing speed (Notteboom and Vernimmen 2009, Brouer et al. 2014), which implies that $\mathcal{F}_p(\tau)$ is proportional to $\frac{1}{(t_p^s+\tau)^2}$, which is decreasing and convex in τ. (For details see Section 6.1.) Furthermore, let $\tau_p^u \geq 0$ be the upper bound on the sailing time action obtained from the minimum sailing speed. Then, $\mathcal{F}_p(\tau)$ is well-defined for all $0 \leq \tau \leq \tau_p^u$. Denote the costs of using the extreme recovery action to reduce the delay by one unit of time by $c^e > 0$.

We next give a stochastic dynamic programming (SDP) formulation of the operational planning level. Remember that the sailed route consists of $N = R|P| + 1$ port calls. Let $\mathcal{C}^{arr}_{n,N}(d;\vec{B})$ denote the total expected cost of completing the route when arriving for port call n with a delay of d time units. Let $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ denote these costs at the departure of port call n. The parameter $\vec{B}$ is added to emphasize that these costs depend on the timetable $\vec{B}$ that is used. The following SDP relation holds for $1 \leq n < N$:

$$\mathcal{C}^{dep}_{n,N}(d^{dep}_n;\vec{B}) \;=\; \mathcal{D}^{dep}_{p[n]}(d^{dep}_n) + \min_{0\leq\tau\leq\tau^u_{p[n]}} \left\{\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d^{dep}_n + \tau;\vec{B})\right\}, \quad (3)$$

$$\text{where} \quad \mathcal{K}_n(d^{dep}_n + \tau;\vec{B}) := \mathbb{E}_{X^s_n}\left[\mathcal{C}^{arr}_{n+1,N}\left((d^{dep}_n + \tau - \vec{B}_{p[n]} + X^s_n)^+;\vec{B}\right)\right]. \quad (4)$$

And the following SDP relation holds for $1 < n < N$:

$$\mathcal{C}^{arr}_{n,N}(d^{arr}_n;\vec{B}) = \mathcal{D}^{arr}_{p[n]}(d^{arr}_n) + \mathbb{E}_{X^p_n}\left[\min_{\gamma\geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d^{arr}_n + X^p_n - \gamma)^+;\vec{B}\right)\right\}\right]. \quad (5)$$

Note that delay propagates according to (1) and (2). Also, note that the extreme recovery action is taken after the port delay is incurred. For the final arrival in port 1, we have the following:

$$\mathcal{C}^{arr}_{N,N}(d^{arr}_N;\vec{B}) = \mathcal{D}^{arr}_{p[N]}(d^{arr}_N).$$

We introduce notation regarding the optimal sailing times and extreme recovery actions. Let $\mathcal{T}_n(d;\vec{B})$ denote the optimal sailing time after port call n (on the sea leg towards port call $n+1$) when the departure delay equals d:

$$\mathcal{T}_n(d;\vec{B}) := \min\left\{\tau' \,|\, \tau' \in \arg\min_{0\leq\tau\leq\tau^u_{p[n]}} \left\{\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d+\tau;\vec{B})\right\}\right\}. \quad (6)$$

Let $\mathcal{Y}_n(d + X^p_n;\vec{B})$ denote the optimal extreme recovery action in port call n when the delay (including port delay) equals $d + X^p_n$:

$$\mathcal{Y}_n(d + X^p_n;\vec{B}) := \max\left\{\gamma' \,|\, \gamma' \in \arg\min_{\gamma\geq 0} \left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d + X^p_n - \gamma)^+;\vec{B}\right)\right\}\right\}. \quad (7)$$

So as a tie-breaking rule, we use minimization of the delay in the next port.

The long run average costs

Because the buffers are transformed into a timetable, which is operated for many rounds, we adopt the long run average costs as performance criterion, which will be denoted by $\mathcal{C}^*(\vec{B})$ and is defined as follows:

$$\mathcal{C}^*(\vec{B}) := \lim_{R\to\infty} \frac{\mathcal{C}^{dep}_{1,R|P|+1}(d_1^{dep};\vec{B})}{R}, \qquad \forall \vec{B} \in \bar{\mathcal{B}}, \tag{8}$$

where $\bar{\mathcal{B}} = \left\{ \vec{B} \in \mathbb{R}^{|P|}_{\geq 0} \;\middle|\; \sum_{p\in P} \vec{B}_p = B \right\}$. For now, we assume that the limit on the RHS of (8) exists, and that it is independent of d_1^{dep}. Later, in Theorem 2, we will formally prove the existence of the limit, and that it is independent of d_1^{dep}, under mild conditions.

As detailed in Section 3.1, we will require that $\vec{B}_p \in \mathbb{Z}$, so we also introduce the set of feasible buffers $\mathcal{B} = \left\{ \vec{B} \in \mathbb{Z}^{|P|}_{\geq 0} \;\middle|\; \sum_{p\in P} \vec{B}_p = B \right\}$, where $\mathcal{B} \subseteq \bar{\mathcal{B}}$. Then, in this paper, we will consider the following optimization problem:

$$\mathcal{C}^* = \min_{\vec{B}\in\mathcal{B}} \mathcal{C}^*(\vec{B}). \tag{9}$$

This problem is non-standard. Each buffer allocation $\vec{B} \in \mathcal{B}$ yields a SDP whose optimal long term average costs equals $\mathcal{C}^*(\vec{B})$. But the buffer time variables themselves cannot be accommodated for in the SDP because they are one-time decisions that affect multiple states: after *each* departure from port j, the arrival delay in port $j+1$ is affected by $\vec{B}_j$. Note that the sailing speed decisions are part of the SDP, so the problem jointly optimizes the buffer allocation and the sailing speed decisions.

3.1 Assumptions for computational purposes

In general, solving for the optimal costs $\mathcal{C}^*(\vec{B})$ of the SDP that arises for *fixed* buffers $\vec{B}$ is already computationally intractable. This is because the SDP has a continuous state space because delay is continuous. (Apart from the current delay, the state consists of the current port $p[n]$ and whether we are arriving or departing.) To deal with this computational issue, we will assume discrete delays and piecewise linear fuel costs (see also Wang and Meng 2012a, who use a similar approach in their model). Specifically, after an appropriate basic time unit is chosen (for example the time unit in which timetables are communicated), we assume the following.

Assumption 1 (Discrete model primitives)**.** *The delays* X^s *and* X^p *take on values in* $\mathbb{Z}_{\geq 0}$*. The total buffer* B *and the maximum additional sailing time* τ_p^u *are in* $\mathbb{Z}_{\geq 0}$*. The functions* $\mathcal{D}_p^{arr}(\cdot)$*,* $\mathcal{D}_p^{dep}(\cdot)$ *and* $\mathcal{F}_p(\cdot)$ *are piecewise linear functions, with breakpoints on* $\mathbb{Z}_{\geq 0}$*. The initial delay is in* $\mathbb{Z}_{\geq 0}$*. Each allocated buffer* $\vec{B}_p$ *should be in* $\mathbb{Z}_{\geq 0}$*.*

We now discuss another computational issue. If we encounter large sea and port delays repeatedly, the delay with respect to the schedule may grow arbitrarily large. In practice, it seems reasonable to assume that when delay exceeds some (possibly large) threshold, it will be optimal to perform the extreme recovery action. We therefore make this assumption to simply and straightforwardly bound the maximum delay. For ease of exposition, we will also assume that the random sea and port delays are bounded by some arbitrary number. These assumptions will simplify the computation of the optimal costs associated with a buffer $\vec{B} \in \bar{\mathcal{B}}$.

Assumption 2 (Bounded delays)**.** *For each* $p \in P$*, there exists a delay* $d_p^{max} < \infty$ *such that* $\mathcal{D}_p^{dep}(d) - c^e d$ *is monotonically increasing* $\forall d > d_p^{max}$*. There exist* $X_{p[n]}^{s,max}, X_{p[n]}^{p,max} \in \mathbb{Z}_{\geq 0}$ *such that* $\forall n : \mathbb{P}\left(X_n^s > X_{p[n]}^{s,max}\right) = 0$*,* $\mathbb{P}\left(X_n^p > X_{p[n]}^{p,max}\right) = 0$*.*

These assumptions are not restrictive in practice as $d_p^{max}, X_p^{\mathrm{s,max}}$ and $X_p^{\mathrm{p,max}}$ can be taken to be large (e.g. one or more weeks when operating a weekly schedule).

4 Theoretical insights

In this section we will derive various theoretical insights into the problem. All proofs can be found in the appendix. Results in this section hold for the general model presented in Section 3. Assumptions 1 and 2, which are made for computational purposes only, are not needed to obtain the results in this section. The main result in this section is the joint convexity of $\mathcal{C}^*(\vec{B})$ in the decision variables $\vec{B}$. This result will provide the basis for our solution algorithm, because it implies the existence of subgradients. The last results in this section provide more insight in the optimal SDP solutions and can be used to bound the optimal recovery actions, which might be useful for an efficient implementation.

Our first result verifies that more delay is worse than less delay.

Lemma 1. *The functions* $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ *and* $\mathcal{C}^{arr}_{n+1,N}(d;\vec{B})$ *are nondecreasing in the amount of delay* d *for* $1 \leq n < N$ *and* $\vec{B} \in \bar{\mathcal{B}}$.

The following result is more surprising, since the costs are not separable because delays may propagate from port call to port call.

Lemma 2. *The functions* $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ *and* $\mathcal{C}^{arr}_{n+1,N}(d;\vec{B})$ *are joint convex in* d *and* $\vec{B} \in \bar{\mathcal{B}}$ *for* $1 \leq n < N$.

A direct result of Lemma 2 is that the average cost per period $\mathcal{C}^*(\vec{B})$ is also joint convex in $\vec{B}$.

Theorem 1. *The optimal long term average cost* $\mathcal{C}^*(\vec{B})$ *is joint convex in* $\vec{B} \in \bar{\mathcal{B}}$*, provided that* $\mathcal{C}^*(\vec{B})$ *exists for* $\vec{B} \in \bar{\mathcal{B}}$.

This result will be used later to find the optimal buffer $\vec{B}$, and thus the optimal schedule. (As for the condition: Theorem 2 proves the existence of the optimal long term average costs $\mathcal{C}^*(\vec{B})$ under sufficient conditions, namely Assumptions 1 and 2.)

The following results give some more insight into how the sailing times and extreme actions should depend on the current delay. The following lemma shows that the larger the delay, the more action should be taken. Hence, with this lemma we prove Conjecture 1 in Li et al. (2015b).

Lemma 3.

(a) The optimal sailing time action $\mathcal{T}_n(d^{dep}_n;\vec{B})$ *between two ports is nonincreasing in the departure delay* d^{dep}_n *for* $1 \leq n < N$ *and* $\vec{B} \in \bar{\mathcal{B}}$;

(b) the optimal extreme recovery action $\mathcal{Y}_n(d^{arr}_n + X^p_n;\vec{B})$ *in a port is nondecreasing in the amount of delay* $d^{arr}_n + X^p_n$ *before that action for* $1 < n < N$ *and* $\vec{B} \in \bar{\mathcal{B}}$.

Thus, a ship with larger departure delay should sail faster than a ship with smaller delay. We wonder whether it could even be optimal for the first ship to plan to "overtake" the latter ship. The following lemma answers this question, by proving that this can never be optimal.

Lemma 4.

(a) The optimal arrival delay $d_{n+1}^{arr} = (d_n^{dep} + \mathcal{T}_n(d_n^{dep}; \vec{B}) - \vec{B}_{p[n]} + X_n^s)^+$ *is stochastically nondecreasing in the departure delay* d_n^{dep}*, for* $1 \leq n < N$ *and* $\vec{B} \in \bar{\mathcal{B}}$*;*

(b) the optimal departure delay $d_n^{dep} = (d_n^{arr} + X_n^p - \mathcal{Y}_n(d_n^{arr} + X_n^p; \vec{B}))^+$ *is nondecreasing in the delay* $d_n^{arr} + X_n^p$ *after incurring port delay, for* $1 < n < N$ *and* $\vec{B} \in \bar{\mathcal{B}}$*.*

This lemma thus effectively bounds the maximal decrease in sailing time (and thus the increase in speed) that should result from being more delayed.

5 Solution Approach

Our objective is finding a $\vec{B} \in \mathcal{B}$ that minimizes $\mathcal{C}^*(\vec{B})$. Since $\mathcal{C}^*(\vec{B})$ is convex by Theorem 1, a range of optimal subgradient-based algorithms is at our disposal for this problem, provided we can compute *subgradients* of $\mathcal{C}^*(\vec{B})$ at arbitrary $\vec{B}$. We discuss this in-depth in Section 5.1, and provide a simple algorithm that works well computationally for our problem. The novelty of our algorithm lies in developing an approach for computing subgradients of $\mathcal{C}^*(\vec{B})$, which is discussed in Section 5.2.

5.1 Subgradient-based algorithms

Theorem 1 implies that $\mathcal{C}^*(\vec{B})$ is convex. Thus for each $\vec{B} \in \bar{\mathcal{B}}$, there exists a subgradient, i.e. a vector $g = (g_1, \ldots, g_{|P|})$, that satisfies the subgradient inequality:

$$\forall \vec{B}' \in \bar{\mathcal{B}} : \mathcal{C}^*(\vec{B}') \geq \mathcal{C}^*(\vec{B}) + \sum_{p \in P} g_p(\vec{B}'_p - \vec{B}_p). \tag{10}$$

We now first show how subgradients can be used in an efficient optimization algorithm.

Our algorithm iteratively generates subgradients using the method described in Section 5.2.2. In the ith iteration, the subgradient at $\vec{B}^i$ is computed. Denote it by $g^i = (g_1^i, \ldots, g_{|P|}^i)$, and denote $g_0^i = \mathcal{C}^*(\vec{B}^i) - \sum_{p \in P} g_p^i \vec{B}_p^i$. After iteration I, we have the following problem:

$$\begin{aligned}
\min \quad & z && && (11)\\
\text{s.t.} \quad & z \geq \sum_{p\in P} g_p^i \vec{B}_p + g_0^i && i \in \{1,\dots,I\} && (12)\\
& \sum_{p\in P} \vec{B}_p = B && && (13)\\
& \vec{B}_p \geq 0 && p \in P. && (14)
\end{aligned}$$

Here, (12) ensure that z satisfies the inequalities imposed by the subgradients, see (10). (13) and (14) ensure that $\vec{B} \in \bar{\mathcal{B}}$, i.e. integrality constraints are relxed. We will prove in the next section that our solution algorithm will in fact always return an integer solution, i.e. $\vec{B} \in \mathcal{B}$, justifying this relaxation. For any subgradients $g_1, \dots, g_I$, the optimal z^* of (11-14) satisfies $z^* \leq \mathcal{C}^* = \mathcal{C}^*(\vec{B}^*)$. Indeed, $\forall \vec{B}$, z^* must become $\max_{i\in\{1,\dots,I\}} \left\{ \sum_{p\in P} g_p^i \vec{B}_p + g_0^i \right\}$, which cannot exceed $\mathcal{C}^*(\vec{B}^*)$ by (10).

Algorithm 1: Solution algorithm

1. Initialize $i = 1$, $\vec{B}^1 = (\vec{B}_1^1, \dots, \vec{B}_{|P|}^1)$ with $\vec{B}_p^1 = \frac{B}{|P|}$, $UB = \infty$ and $LB = -\infty$.
2. Compute $\mathcal{C}^*(\vec{B}^i)$ and the gradient g^i at $\vec{B}^i$ (see Section 5.2.2).
3. If $\mathcal{C}^*(\vec{B}^i) < UB$, set $UB = \mathcal{C}^*(\vec{B}^i)$ and $\vec{B}^{UB} = \vec{B}^i$.
4. Let $(z^*, \vec{B}')$ denote the optimal solution of (11-14) for $I = i$. Set $LB = z^*$, $\vec{B}^{i+1} = \vec{B}'$.
5. If $UB - LB \leq \epsilon$, designate $\vec{B}^{UB}$ as ϵ-optimal and terminate. Otherwise, set $i \leftarrow i + 1$ and go to Step 2.

Algorithm 1 explains how we use this formulation in our optimization approach. In initial steps, the $\vec{B}^{i+1}$ from Step 4 may lie far away from the last search point $\vec{B}^i$, adversely impacting performance. Therefore, we limit the distance between $\vec{B}^i$ and $\vec{B}^{i+1} = \vec{B}$. Consider the constraints,

$$\sum_{p\in P} |\vec{B}_p - \vec{B}_p^i| \leq w^{max}, \qquad \forall p \in P : |\vec{B}_p - \vec{B}_p^i| \leq w_p^{max} \tag{15}$$

where w^{max} and $\forall p \in P : w_p^{max}$ are parameters. Then Step 4 is replaced by the following in the first 25 iterations of the algorithm:

4' Let $(z^*, \vec{B}')$ denote the optimal solution of (11-14)+(15) for $I = i$. Set $\vec{B}^{i+1} = \vec{B}'$. Let $\tilde{z}^*$ be the optimal solution value of (11-14). Set $LB = \tilde{z}^*$.

5.2 Subgradients

In general, computing subgradients involves analyzing the change of the objective function when the input changes. For our problem, changing $\vec{B}$ affects the *structure* of the SDP underlying $\mathcal{C}^*(\vec{B})$, which complicates the computation of the subgradient. In Section 5.2.1, we analyze this structure. This analysis involves a number of complex ideas and quite some additional notation, but it yields a relatively simple algorithm for computing subgradients that we present in Section 5.2.2.

5.2.1 Analysis of SDP structure

Throughout this section, we work with the specialized model that is obtained by imposing discrete model primitives (Assumption 1) and bounded delays (Assumption 2). In this section, we analyze the structure of SDP's for different underlying buffer allocations in order to develop a method to compute subgradients. As Algorithm 1 requires subgradients for general $\vec{B}$, we must also consider cases where $\vec{B} \in \bar{\mathcal{B}} \setminus \mathcal{B}$.

We first propose a transformation of the buffer allocation into the cumulative buffer allocation. Next, we will show that both the state and action space of the SDP are finite and can directly be obtained from the cumulative buffer allocation. As a result, a stationary deterministic policy exists that is average cost optimal.

Then, we propose an ordering of the ports based on the fractional values of the cumulative buffer allocation and define a set consisting of all buffer allocations with the same ordering. We show that the costs of each buffer allocation in this set can be expressed as a linear combination of the costs of the extreme points of this set. Hence, we can find a subgradient at each buffer allocation in this set by solving a system of $|P|$ linearly independent equations. In order to construct this system, we need to determine the costs of the extreme points of the set. We will show that the extreme points satisfy $\vec{B} \in \mathbb{Z}^{|P|}$. Hence, we can find a subgradient for arbitrary $\vec{B}$ by evaluating the costs of several integer buffer allocations. Moreover, this result indicates that the optimal buffer allocation will always be integer valued.

Buffer transformation

For notational convenience, we transform the buffers. Each $\vec{B} \in \mathcal{B}$ corresponds to a cumulative buffer allocation $\tilde{B}$, by setting $\tilde{B}_p := \sum_{p'=1}^{p-1} \vec{B}_{p'}$ and $\tilde{B} = (\tilde{B}_1, \ldots, \tilde{B}_{|P|})$. (Thus $\tilde{B}_1 := 0$.) Let $\tilde{\mathcal{B}}$ contain every $\tilde{B}$ that can be obtained in this fashion from a $\vec{B} \in \bar{\mathcal{B}}$. Thus $\forall \tilde{B} \in \tilde{\mathcal{B}} : \tilde{B}_{|P|} \leq B$, $\tilde{B}_1 = 0$ and $\forall \tilde{B} \in \tilde{\mathcal{B}}, \forall p \in P : \tilde{B}_{p+1} \geq \tilde{B}_p$. For $\tilde{B} \in \tilde{\mathcal{B}}$, define $\mathcal{C}^*(\tilde{B}) := \mathcal{C}^*(\vec{B})$, with $\vec{B}$ obtained by setting $\vec{B}_p = \tilde{B}_{p+1} - \tilde{B}_p$ for $p \in \{1, \ldots, |P|-1\}$, and $\vec{B}_{|P|} = B - \tilde{B}_{|P|}$.

Existence of stationary deterministic optimal policies

It will be important in the analysis to know which delay values for port call n will result in a discrete delay in port p when only discrete actions are taken. Therefore, we introduce the set $Q_n(p; \tilde{B})$ that contains precisely the delays for port call n that result in a discrete delay in port calls to port p in the absence of fractional actions and/or waiting for departure or arrival. Then, $Q_n(p; \tilde{B})$ is defined as:

$$Q_n(p; \tilde{B}) := Q_{p[n]}(p; \tilde{B}) := \left\{ z + \tilde{B}_p - \tilde{B}_{p[n]} | z \in \mathbb{Z} \right\}, \quad \forall n \in \{1, \ldots, N\}. \tag{16}$$

Furthermore, we let $Q_n(\tilde{B}) := \bigcup_{p \in P} Q_n(p; \tilde{B})$. Hence, $Q_n(\tilde{B})$ contains delays that result in a discrete delay in some future port call. Recall that the initial delay is discrete by Assumption 1. Then, the following lemma shows, using the recursive relations (3) and (5), that $Q_n(\tilde{B})$ contains all delays that may occur for port call n.

Lemma 5. *Fix $\tilde{B} \in \tilde{\mathcal{B}}$, and choose τ_n and γ_n optimally using the rules implied by* (6) *and* (7) *to break ties. Then for each port call n: $d_n^{dep} \in Q_n(\tilde{B})$, $d_n^{dep} + \tau_n \in Q_n(\tilde{B})$, $d_n^{arr} + X_n^p \in Q_n(\tilde{B})$, $d_n^{arr} + X_n^p - \gamma_n \in Q_n(\tilde{B})$.*

Lemma 5 implies that only the values in $Q_n(\tilde{B})$ are relevant to consider in the SDP. We will denote a generic state of the SDP by s. Let $s_{p,\text{dep}}[z, p'; \tilde{B}]$ correspond to departing from port p with delay $d^{dep} = z + \tilde{B}_{p'} - \tilde{B}_p$. Let $s_{p,\text{port}}[z, p'; \tilde{B}]$ correspond to being in port

p with a delay of $d^{arr} + X^p = z + \tilde{B}_{p'} - \tilde{B}_p$, *after* incurring port delay. Let

$$S_{\tilde{B}} := \{s_{p,u}[z, p'; \tilde{B}] | z \in \mathbb{Z}, p \in P, p' \in P, u \in \{\text{dep}, \text{port}\}\}.$$

By Lemma 5, all combinations of states and delays that can occur for $\tilde{B} \in \tilde{\mathcal{B}}$ are in $S_{\tilde{B}}$, though $S_{\tilde{B}}$ also contains states that cannot occur because their associated delay is negative. It is immediate from Lemma 5 that the optimal actions τ and $-\gamma$ in any state $s_{p,u}[z, p'; \tilde{B}]$ must take their values in $Q_{p'}(p''; \tilde{B})$ for some p''. We will denote a generic action by a. In a state with $u = \text{dep}$, the action $a_{p'}[z, p''; \tilde{B}]$ will denote $\tau = z + \tilde{B}_{p''} - \tilde{B}_{p'}$, and in a state with $u = \text{port}$, action $a_{p'}[z, p''; \tilde{B}]$ will denote $-\gamma = z + \tilde{B}_{p''} - \tilde{B}_{p'}$. Let $A_{\tilde{B}} = \{a_{p'}[z, p''; \tilde{B}] | z \in \mathbb{Z}, p' \in P, p'' \in P\}$.

Since delay is non-negative, and bounded above by Assumption 2, for each $\tilde{B}$ a finite subset of $S_{\tilde{B}}$ and $A_{\tilde{B}}$ suffices for a complete description of the model. As a consequence, we have the following result.

Theorem 2. *For all $\vec{B} \in \bar{\mathcal{B}}$, the limit $\lim_{R\to\infty} \mathcal{C}^{dep}_{1,R|P|+1}(d_1^{dep}; \vec{B})/R$ exists and is independent of d_1^{dep}. There exists a stationary deterministic policy that is average cost optimal.*

A stationary deterministic policy for $\tilde{B} \in \tilde{\mathcal{B}}$ will be represented by a function $\Pi_{\tilde{B}} : S_{\tilde{B}} \to A_{\tilde{B}}$ and we denote the optimal stationary deterministic policy for $\tilde{B}$ by $\Pi^*_{\tilde{B}}$.

Subgradients

We will now investigate the change of $\mathcal{C}^*(\tilde{B})$ when $\tilde{B}$ changes. First some preliminaries. A cumulative buffer $\tilde{B} \in \tilde{\mathcal{B}}$ is *completely fractional* if for every $p, p' \in P$ with $p \neq p'$, the number $\tilde{B}_{p'} - \tilde{B}_p \notin \mathbb{Z}$. Note that every $Q_n(\tilde{B})$ contains at most $|P|$ different delay values in the interval $[0, 1)$. Completely fractional cumulative buffer allocations can be recognized by the fact that $Q_n(\tilde{B})$ contains exactly $|P|$ different delay values in this interval. Each of those $|P|$ values is by definition obtained from one of the sets $Q_n(p; \tilde{B})$ and hence associated to a port p. We can order the ports based on the value in the interval $[0, 1)$ in the set $Q_n(\tilde{B})$. For any $\tilde{B} \in \tilde{\mathcal{B}}$, it holds that $0 \leq \tilde{B}_p \leq B$. Thus, we can always write $\tilde{B}_p = z_p + x_p$, with $z_p \in \{0, 1, \ldots, B-1\}$ and $0 \leq x_p \leq 1$ by Assumption 1. For completely fractional $\tilde{B} \in \tilde{\mathcal{B}}$, this decomposition into z_p and x_p is unique. Let $f : P \to P$ be the unique permutation of P such that $p > p' \to x_{f(p)} > x_{f(p')}$. Uniqueness follows because

$\forall p, p' \in P$ with $p \neq p'$ it holds that $x_p \neq x_{p'}$, since $x_p = x_{p'}$ would contradict that $\tilde{B}$ is completely fractional. Note that the unique permutation f provides a formal definition of the above discussed ordering. Then, we can define for every completely fractional $\tilde{B} \in \tilde{\mathcal{B}}$ a new set $\Delta(\tilde{B})$ containing all cumulative buffer allocations $\tilde{B}'$ for which the ordering of $\tilde{B}$ is also a feasible ordering for $\tilde{B}'$.

$$\Delta(\tilde{B}) := \Big\{ \tilde{B}' \in \tilde{\mathcal{B}} \mid \forall p \in P, \tilde{B}'_p = \lfloor \tilde{B}_p \rfloor + x'_p, x'_p \in [0,1]; \tag{17}$$

$$0 = x'_{f(1)} \leq x'_{f(2)} \leq \ldots \leq x'_{f(|P|)} \leq 1 \Big\}. \tag{18}$$

Note that $\tilde{B}'_1 = 0$ by definition, such that $x'_1 = 0$ for any $\tilde{B}'$ and $\tilde{B} \in \Delta(\tilde{B})$. We are now ready to formulate the main result of this section.

Theorem 3. *Take any completely fractional $\tilde{B} \in \tilde{\mathcal{B}}$ and let $\Pi^*_{\tilde{B}}$ denote its average cost optimal policy. For all $\tilde{B}' \in \Delta(\tilde{B})$, define the policy $\hat{\Pi}_{\tilde{B}'}$ as follows:*

$$\hat{\Pi}_{\tilde{B}'}(s_{p,u}[z,p';\tilde{B}']) = a_{p'}[z',p'';\tilde{B}'] \quad \textit{iff} \quad \Pi^*_{\tilde{B}}(s_{p,u}[z,p';\tilde{B}]) = a_{p'}[z',p'';\tilde{B}].$$

Let $\hat{\mathcal{C}}(\tilde{B}')$ denote the long run average costs for $\tilde{B}'$ under $\hat{\Pi}_{\tilde{B}'}$. Then

$$\forall \tilde{B}' \in \Delta(\tilde{B}) : \hat{\mathcal{C}}(\tilde{B}') = \mathcal{C}^*(\tilde{B}') = \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p(\tilde{B}'_p - \tilde{B}_p),$$

where $g = (g_1, \ldots, g_{|P|})$ is a subgradient at $\tilde{B}$.

Theorem 3 indicates that there exists a special policy $\hat{\Pi}_{\tilde{B}'}$ for cumulative buffer allocations $\tilde{B}' \in \Delta(\tilde{B})$ such that the cost of $\tilde{B}'$ under $\hat{\Pi}_{\tilde{B}'}$ changes linearly in $\tilde{B}$. Furthermore, the coefficient vector of this change provides a subgradient, which directly implies that $\hat{\Pi}_{\tilde{B}'}$ is optimal for all allocations $\tilde{B}'$ in $\Delta(\tilde{B})$. It is surprising that $\hat{\Pi}_{\tilde{B}'}$ is optimal for $\tilde{B}'$, because $\hat{\Pi}_{\tilde{B}'}$ is rather different from $\Pi^*_{\tilde{B}}$: $s_{p,u}[z,p';\tilde{B}]$ and $s_{p,u}[z,p';\tilde{B}']$ represent different delays, and $a_p[z,p';\tilde{B}]$ and $a_p[z,p';\tilde{B}']$ represent different actions. The proof of Theorem 3 shows that, when states are expressed as $s_{p,u}[z,p';\tilde{B}']$, the transitions are independent of $\tilde{B}'$, and the theorem follows from that result and convexity.

Theorem 3 implies the following.

Corollary 1. *The subgradient g at $\tilde{B}$ from Theorem 3 is a subgradient for any $\tilde{B}' \in \Delta(\tilde{B})$.*

Integer optimal solution

To arrive at a simple algorithm to compute subgradients, we investigate $\Delta(\tilde{B})$. Note that $\Delta(\tilde{B})$ contains exactly $|P|$ integer buffer allocations, say $\tilde{B}^1, \ldots, \tilde{B}^{|P|}$. Then, $\tilde{B}^j := (\tilde{B}^j_1, \ldots, \tilde{B}^j_{|P|}) \in \tilde{\mathcal{B}}$ is given by

$$\tilde{B}^j_p = \begin{cases} z_p & \text{if } f^{-1}(p) \leq j \\ z_p + 1 & \text{if } f^{-1}(p) > j. \end{cases} \tag{19}$$

for $p \in P$. In other words, $\left\{\tilde{B}^j \,|j \in \{1, \ldots, |P|\}\right\}$ is the set of integer cumulative buffer allocations with the same ordering as $\tilde{B}$. We can show that $\Delta(\tilde{B})$ is the convex hull of this set.

Theorem 4. *Let $\tilde{B}^j$, $j \in \{1, \ldots, |P|\}$ be as defined in* (19). *Then, for any completely fractional $\tilde{B}$, $\Delta(\tilde{B})$ is the convex hull of $\left\{\tilde{B}^j \,|j \in \{1, \ldots, |P|\}\right\}$.*

A direct result of Theorems 3 and 4 is that $\mathcal{C}^*(\tilde{B})$ is a linear combination of $\mathcal{C}^*(\tilde{B}^j)$ for $j \in \{1, \ldots, |P|\}$ for all fractional $\tilde{B}$, implying that fractional $\tilde{B}$ can never be optimal. Hence, the optimal buffer allocation will always be integer valued.

5.2.2 Computing subgradients

The above analysis yields Algorithm 2 to compute a subgradient for any $\vec{B} \in \tilde{\mathcal{B}}$. The idea of the algorithm is to first find an ordering of the ports corresponding to their fractional delay value. Next, we can construct all $|P|$ integer cumulative buffer allocations with the same ordering and use them to find the costs of $\tilde{B}$ and the subgradient at $\tilde{B}$. We now explain some details. For Step 2, note that $\tilde{B}_1 := 0$ such that $x_1 = 0$, implying that $f(1) = 1$ never contradicts the other requirements on f. The $\tilde{B}'$ in Step 3 can be any completely fractional $\tilde{B}'$ with $\tilde{B}'_p = z'_p + x'_p$, such that $\forall p \in P : z'_p = z_p$ and $\forall p, p' \in P : p > p' \rightarrow x'_{f(p)} > x'_{f(p')}$. (If $\tilde{B}$ is completely fractional, it suffices to set $\tilde{B}' = \tilde{B}$.) Note that for our algorithm it is only important that a completely fractional $\tilde{B}'$ with those properties exists, we do not need to find one. It can then be verified that $\tilde{B} \in \Delta(\tilde{B}')$, and Theorem 4 shows that $\forall j \in \{1, \ldots, |P|\} : \tilde{B}^j \in \Delta(\tilde{B}')$. Note that Step 3 of Algorithm 1 can be improved by returning the best integer buffer allocation $\tilde{B}^j$ together with its cost in Algorithm 2.

Algorithm 2: Computing subgradients

1. Let $\tilde{B} \in \tilde{\mathcal{B}}$ be the cumulative buffer corresponding to $\vec{B} \in \bar{\mathcal{B}}$.

2. For all $p \in P$, write $\tilde{B}_p = z_p + x_p$, with $z_p \in \{0, 1, \ldots, B-1\}$ and $0 \leq x_p \leq 1$. Let $f : P \to P$ be any permutation of P such that $f(1) = 1$ and $\forall p, p' \in P : p > p' \to x_{f(p)} \geq x_{f(p')}$.

3. For each $j \in \{1, \ldots, |P|\}$ define $\tilde{B}^j = (\tilde{B}^j_1, \ldots, \tilde{B}^j_{|P|}) \in \tilde{\mathcal{B}}$,
$$\tilde{B}^j_p = \begin{cases} z_p & \text{if } f^{-1}(p) \leq j \\ z_p + 1 & \text{if } f^{-1}(p) > j. \end{cases}$$
for $p \in \{1, \ldots, |P|\}$. Then $\exists \tilde{B}' \in \tilde{\mathcal{B}}$ such that $\tilde{B} \in \Delta(\tilde{B}')$, and $\tilde{B}^j \in \Delta(\tilde{B}')$ for each $j \in \{1, \ldots, |P|\}$.

4. Compute $\mathcal{C}^*(\tilde{B}^j)$ for each $j \in \{1, \ldots, |P|\}$.

5. By Theorem 3 and Corollary 1, since $\tilde{B} \in \Delta(\tilde{B}')$, and $\tilde{B}^j \in \Delta(\tilde{B}')$ for each $j \in \{1, \ldots, |P|\}$, a subgradient g at $\tilde{B}$ satisfies the system of equations:
$$\mathcal{C}^*(\tilde{B}^j) = \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p(\tilde{B}^j_p - \tilde{B}_p), j \in \{1, \ldots, |P|\} \tag{20}$$
Solve the system to obtain $\mathcal{C}^*(\tilde{B})$ and a subgradient $g = (g_1, \ldots, g_{|P|})$ at $\tilde{B}$.

6. Use this subgradient to obtain a subgradient for $\mathcal{C}^*(\vec{B})$ at $\vec{B}$.

For Step 4, note that $\mathcal{C}^*(\tilde{B}^j)$ is the long run average cost of a finite state SDP, which can be solved efficiently using linear programming. The specific choice of $\tilde{B}^j$ reduces the complexity of finding $\mathcal{C}^*(\tilde{B}^j)$, because only integer buffers occur in all ports, reducing the size of the state and action space of the SDP (see Lemma 5). Equalities (20) in Step 5 are simply obtained by applying the equation in Theorem 3 to $\tilde{B}$ and $\tilde{B}^j$. Note that g_1 is free in (20), since $\tilde{B}_1 := 0$ for all $\tilde{B} \in \tilde{\mathcal{B}}$. For the same reason, the value of g_1 is inconsequential, so set it to 0. Since $\tilde{B}^j$ are linearly independent by construction, (20) have $|P|$ linearly independent equations, leading to a unique solution for the variables $g_2, \ldots, g_{|P|}$ and $\mathcal{C}^*(\tilde{B})$. Step 6 is straightforward, since $\tilde{B}$ is obtained from $\vec{B}$ using a linear transformation.

6 Case Study

6.1 Data

To test our method, we use the ME1 route in September 2012 of the Maersk Line network. Time is discretized in units of four hours. Table 1 shows the order in which the ports are visited in the route, the distances and sailing times between ports and the time needed in the port. The second column of Table 1 denotes the total time planned in the port to load and unload the ship. In the third column the distances between the ports in nautical miles are presented. Distances are obtained from SeaRates (2015). The distance shown for each port is the distance that the ship has to cover to sail from that port to the next port. The fourth column shows the sailing time in hours according to the schedule. The planned sailing time for Antwerp is 32 hours, which means that a ship might take 32 hours to sail from Antwerp to Bremerhaven before it will encounter a delay during its trip. The last column shows the buffer time in the current schedule assuming that the route is sailed at maximum speed. The time needed to make one full round tour is 1176 hours (7 weeks).

Port	Port time (hr)	Distance (nmi)	Sailing time (hr)	Buffer time (hr)
Jebel Ali	31	1329	72	12
Jawaharlal Nehru	33	443	24	4
Mundra	16	1122	56	4
Salalah	14	1553	68	0
Jeddah	11	778	36	0
Suez Canal	16	2283	100	0
Algeciras	18	1476	88	20
Felixstowe	24	156	16	8
Antwerp	16	366	32	16
Bremerhaven	24	283	24	8
Rotterdam	20	3829	192	24
Suez Canal	22	395	20	0
Aqaba	20	656	40	8
Jeddah	19	2648	124	8

Table 1: Characteristics of the route

We assume that the route is sailed using a post panamax ship with capacity 8,400 TEU, using data from Brouer et al. (2014). The minimum and maximum speed of this ship are 12 and 23 knots respectively. Bunker consumption per time unit can be accurately approximated as a constant times the third power of speed. Thus, the fuel cost function becomes:

$$\mathcal{F}_p(\tau) = \tilde{b}e\left(\frac{v}{\tilde{v}}\right)^3 \frac{(t_p^s+\tau)l}{24} = \tilde{b}e\frac{(t_p^s+\tau)l}{24}\left(\frac{\delta_p}{(t^s+\tau)l\tilde{v}}\right)^3,$$

where v is the sailing speed in knots (nmi/hour), δ_p is distance in nmi from port p to the next port and $l = 4$ denotes the number of hours in one time unit. The ship has a design speed of $\tilde{v} = 16.5$ knots, and bunker consumption at design speed is $\tilde{b} = 82.2$ ton per day. Bunker cost is assumed to be $e = 600$ USD per ton (Brouer et al. 2014).

6.2 Test instances

Given the fixed port times and the total duration of a round tour, 52 time units remain to allocate over the ports. By changing the additional delay distributions, different scenarios can be constructed. Since we do not know the actual delay distribution, we will gauge the outcomes under different delay distributions. We will assume that each $X_n^s \sim U\left(0, a + \left\lfloor \frac{\delta_{p[n]}}{b} \right\rfloor\right)$, where a and b are instance specific parameters and $\delta_{p[n]}$ is the distance between the current and the next port. For each test instance we can compute the minimum average time to complete one round tour of the route. This time is obtained by sailing at maximum speed and incurring the average delay in each port. The minimum average additional time to complete a round tour should not exceed the available time of 52 time units, since ships will not be able to recover from incurred delays in these scenarios. We will refer to the (positive) difference between the available time and the minimum average completion time as the expected net buffer time. Ten instances are constructed by varying the expected buffer time between 2.5 and 25 time units in steps of 2.5 time units. This is done using $a = \{3, 3, 2, 2, 2, 1, 1, 1, 0, 0\}$ and $b = \{1200, 1600, 800, 1300, 2000, 900, 1328, 2400, 1000, 1400\}$. The extreme expedite cost for the cut-and-go action is given by ten million USD per time unit and $\mathcal{D}_p(d) = 10,000d$ for $0 \leq d \leq d_p^{max} = 42$ time units (one week) for $p \in P$ and we assume that the unit costs are larger than ten million USD for $d > d_p^{max}$ for $p \in P$, such that delays are bounded by 42 time units. Finally, we set the parameters for our algorithm as $w_p^{max} = 0.25$ for $p \in P$, $w^{max} = 2$, and $\epsilon = 10^{-8}$ USD.

6.3 Results

For each test instance, we first calculate the cost of the schedule when we consider deterministic delays. That is, we assume that the delay incurred between each two ports is fixed and equal to the expected delay between those two ports. The optimal schedule is then found by allocating the available buffer time in such a way that a constant speed is used over the round tour. Furthermore, we calculate the costs of the initial schedule, the costs of the schedule in which the buffer time is uniformly distributed over the ports and the costs of the optimal schedule. All linear programming models are solved using CPLEX 12.6.

Available buffer (time units)	Expected delay (time units)	Expected buffer (time units)
28.0	25.5	2.5
28.0	23.0	5.0
28.0	20.5	7.5
28.0	18.0	10.0
28.0	15.5	12.5
28.0	13.0	15.0
28.0	10.5	17.5
28.0	8.0	20.0
28.0	5.5	22.5
28.0	3.0	25.0

Table 2: Characteristics of the ten instances

Table 2 shows the expected delay and expected buffer times in time units for the ten instances. The expected buffer times vary between 2.5 and 25 time units.

Table 3 shows the average expected round tour costs for the ten instances. Clearly, the costs of sailing a round tour decreases when the available buffer time increases. The deterministic schedule provides a lower bound on the optimal cost schedule. The difference between the cost of the deterministic and stochastic schedules is the effect of uncertainty on the cost, which is shown in the last three columns of the table. In these columns first the absolute cost of uncertainty is given and in between brackets the relative difference compared to the initial schedule is given. We observe that for high expected buffer times, a large part of the cost is already incurred in the deterministic case. Furthermore, the absolute difference in cost between the initial and the uniform schedule decreases when

Expected buffer (time units)	Deterministic schedule (million USD)	Cost of uncertainty		
		Initial (million USD)	Uniform (million USD)	Optimal (million USD)
2.5	3.831	0.904 (100%)	0.709 (78%)	0.702 (78%)
5.0	3.735	0.537 (100%)	0.381 (71%)	0.377 (70%)
7.5	3.644	0.446 (100%)	0.315 (71%)	0.292 (66%)
10.0	3.557	0.369 (100%)	0.257 (70%)	0.234 (63%)
12.5	3.471	0.317 (100%)	0.216 (68%)	0.193 (61%)
15.0	3.387	0.312 (100%)	0.225 (72%)	0.172 (55%)
17.5	3.306	0.272 (100%)	0.190 (70%)	0.137 (50%)
20.0	3.229	0.242 (100%)	0.173 (72%)	0.099 (41%)
22.5	3.153	0.256 (100%)	0.182 (71%)	0.068 (26%)
25.0	3.080	0.222 (100%)	0.158 (71%)	0.040 (18%)

Table 3: Total average round tour costs for the ten test instances

more buffer time is available, while the absolute difference in cost between the uniform and the optimal schedule increases when more time is available. From the relative costs, we can conclude that the uniform schedule always performs about 30% better than the initial schedule, while the optimal schedule has costs that are $22 - 82\%$ lower than the initial schedule. The relative performance of the optimal schedule increases when more buffer time is available. When only 2.5 time units of buffer time are available, the largest absolute cost reduction between the initial and the optimal schedule can be obtained, while the lowest absolute reduction is obtained for 12.5 time units of expected buffer. The lowest and largest reductions are respectively 123 and 202 thousand USD per round tour. Since liner companies usually provide weekly services, this would result in cost reductions of 6-10 million USD per year.

Expected buffer (time units)	Time (seconds)	Number subgradients
2.5	21	19
5.0	65	26
7.5	77	30
10.0	56	20
12.5	63	23
15.0	64	38
17.5	62	37
20.0	57	34
22.5	64	34
25.0	65	34

Table 4: Solution times for the ten test instances

Table 4 shows the solution times of the subgradient algorithm. Furthermore, the number of generated subgradients are shown. All instances can be solved to optimality within 80 seconds. In total, 19-38 subgradients have to be determined in the solution algorithm.

Port	On time prob	Avg arr delay (time units)	Distance (nmi)	Buffer time (time units)
Jebel Ali	0.49	0.78	1329	2
Jawaharlal Nehru	0.49	0.68	443	1
Mundra	0.41	0.68	1122	1
Salalah	0.30	1.09	1553	3
Jeddah	0.43	0.80	778	1
Suez Canal	0.38	0.73	2283	3
Algeciras	0.35	1.21	1476	3
Felixstowe	0.62	0.44	156	0
Antwerp	0.31	0.94	366	1
Bremerhaven	0.39	0.74	283	1
Rotterdam	0.63	0.44	3829	6
Suez Canal	0.50	1.00	395	1
Aqaba	0.33	1.00	656	1
Jeddah	0.37	0.84	2648	4

Table 5: Optimal buffer time on the next sea leg

Table 5 shows for each port the probability of arriving on time, the average arrival delay in time units and the optimal buffer allocation in time units for the instance with an expected buffer of 15 time units. In general, more buffer time is added to sea legs with larger distances, because on these legs larger additional delays are expected to be incurred.

Table 6 shows the sailing times in time units that will be used on the next sea leg given a certain amount of delay for the instance with 15 time units of expected buffer. The last columns show the range of feasible speeds and the planned sailing time for the given sea leg. The table shows that ships will not always speed up when a larger delay is incurred even when the maximum sailing speed limit is not reached yet (see for example a departure from Jebel Ali with 0 and 1 time units of delay). This confirms that the optimal sailing speed policy is not always to try to recover from all delays during the coming sea leg.Furthermore, the table shows that Lemmas 3 and 4 are indeed satisfied: ships will never slow down when they incur higher delays, but will also always arrive with at least the same amount of delay in the next port as when they would have incurred a lower delay. Moreover, ships might already speed up even when it sails according to schedule. This happens for example when

Port	Delay in time units 0	1	2	3	4	5	≥ 6	Feasible range	Planned sailing time
Jebel Ali	16	16	15	15	15	15	15	[15, 27]	17
Jawaharlal Nehru	6	5	5	5	5	5	5	[5, 9]	6
Mundra	14	13	13	13	13	13	13	[13, 23]	14
Salalah	19	19	18	17	17	17	17	[17, 32]	20
Jeddah	10	9	9	9	9	9	9	[9, 16]	10
Suez Canal	27	27	26	25	25	25	25	[25, 47]	28
Algeciras	19	18	17	17	17	17	17	[17, 30]	20
Felixstowe	2	2	2	2	2	2	2	[2, 3]	2
Antwerp	5	4	4	4	4	4	4	[4, 7]	5
Bremerhaven	4	4	4	4	4	4	4	[4, 5]	5
Rotterdam	46	45	44	44	43	42	42	[42, 79]	48
Suez Canal	6	5	5	5	5	5	5	[5, 8]	6
Aqaba	9	8	8	8	8	8	8	[8, 13]	9
Jeddah	32	31	30	30	29	29	29	[29, 55]	33

Table 6: Sailing time action in time units to be used on the next sea leg

the leaves the port of Rotterdam. The scheduled sailing time between Rotterdam and the Suez Canal is 48 time units, while the ship will only use 46 time units when leaving the port of Rotterdam without delay. Hence, an action of 2 time units is performed to hedge against expected delays incurred between Rotterdam and the Suez Canal.

Finally, when we consider the amount of extreme actions in the solutions, we observe that more extreme actions are taken when less buffer time is available. In the instances with 15 or more units of buffer, no extreme actions are taken in the optimal solutions. Furthermore, in the instance with 2.5 units of buffer most extreme actions are taken, namely in expectation 0.00038 time units per round tour, which corresponds to once every 353 years. This is in line with our desire to use the extreme actions as a device to limit the maximum delay and not as an economically feasible option.

7 Conclusion and Future Research

We developed a new approach for allocating buffers in timetables. Our model jointly optimizes decisions over two stages: buffer times during timetable development and speed optimization during timetable execution. We model the execution of the timetable as a stochastic dynamic program (SDP), allowing for accurate modelling of real-time recovery actions using the latest information, random events causing delays, and propagation of delays from port call to port call. Our theoretical analysis revealed that as the delay with respect to the timetable increases, so should our travelling speed.

Optimizing the buffer allocation decisions presented a challenge, because they must be exogenous to the SDP since they affect transitions in multiple states. In general, only enumeration techniques can optimize over variables exogenous to an SDP. But we were able to show, under relatively mild assumptions, that $\mathcal{C}^*(\vec{B})$ is convex in the buffer time variables. A detailed investigation of the cost function $\mathcal{C}^*(\vec{B})$ yielded a simple method to compute subgradients. We note that the form of this method may indicate a link with submodularity. In particular, similar results may perhaps be obtained using the so-called Lovász extension (Lovász 1983), but we believe this would mainly be of theoretical interest. Based on these results, we proposed a relatively simple algorithm.

In our experiments, the algorithm computes the optimal buffer time allocation in under 80 seconds. We compared the optimal schedule with the cost of the initial schedule as executed by Maersk Line and with the cost of a schedule in which buffer times are uniformly distributed over the ports. We observe that the uniform schedule provides very good solutions for schedules with low buffers, but that the optimal schedule generates costs that are six to ten million USD per year lower compared to the initial schedule. For schedules with high buffers, the optimal schedule also results in much lower costs than the uniform schedule.

Our experiments thus revealed that the proposed algorithm is very efficient. Its efficiency stems from the use of convexity of $\mathcal{C}^*(\vec{B})$, allowing us to take into account on-line speed optimization without severely reducing performance. Moreover, we directly extract subgradients from the SDP formulation, so we can take into account the stochasticity without sampling. These properties make the algorithm a good candidate for further research in timetable optimization, also in contexts other than container shipping. However, challenges need to be overcome to use the algorithm in settings where the timetable involves multiple trains/ships/metros that interact. Further research is needed to reveal whether the algorithm may be valuable in those settings as well.

Acknowledgement

The authors are thankful to Twan Dollevoet, Dennis Huisman, Leo Kroon, Harilaos Psaraftis and Marie Schmidt for their valuable comments on the manuscript. Furthermore, the authors want to thank Erasmus Smart Port Rotterdam for providing financial support for this research.

References

AhmadBeygi, S., A. Cohn, M. Lapp. 2010. Decreasing airline delay propagation by re-allocating scheduled slack. *IIE Transactions* **42**(7) 478–489.

Aktürk, M.S., A. Atamtürk, S. Gürel. 2014. Aircraft Rescheduling with Cruise Speed Control. *Operations Research* **62**(4) 829–845.

Arikan, U., S. Gürel, M.S. Aktürk. 2016. Integrated aircraft and passenger recovery with cruise time controllability. *Annals of Operations Research* **236**(3) 295–317.

Bertsekas, D.P. 2007. *Dynamic Programming and Optimal Control - Volume II*. 3rd ed. Athena Scientific, Belmont, Massachusetts.

Binder, A., T. Albrecht. 2012. Predictive energy-efficient running time control for metro lines. *CASPT12 - Conference on Advanced Systems for Public Transport*.

Brouer, B.D., J.F. Álvarez, C.E.M. Plum, D. Pisinger, M.M. Sigurd. 2014. A Base Integer Programming Model and Benchmark Suite for Liner-Shipping Network Design. *Transportation Science* **48**(2) 281–312.

Brouer, B.D., J. Dirksen, D. Pisinger, C.E.M. Plum, B. Vaaben. 2013. The Vessel Schedule Recovery Problem (VSRP) - a MIP model for handling disruptions in liner shipping. *European Journal of Operational Research* **224**(2) 362–374.

Cariou, P. 2011. Is slow steaming a sustainable means of reducing CO_2 emissions from container shipping? *Transportation Research Part D* **16**(3) 260–264.

Chiraphadhanakul, V., C. Barnhart. 2013. Robust flight schedules through slack re-allocation. *EURO Journal on Transportation and Logistics* **2** 277–306.

Cicerone, S., G. D'Angelo, G. Di Stefano, D. Frigiono, A. Navarra, M. Schachtebeck, A. Schöbel. 2009. Recoverable Robustness in Shunting and Timetabling. R.K. Ahuja, R.H. Möhring, C.D. Zaroliagis, eds., *Robust and Online Large-Scale Optimization*, *Lecture Notes in Computer Science*, vol. 5868. Springer, 28–60.

Cicerone, S., G. Di Stefano, M. Schachtebeck, A. Schöbel. 2012. Multi-stage recovery robustness for optimization problems: A new concept for planning under disturbances. *Information Sciences* **190** 107–126.

Clausen, J., A. Larsen, J. Larsen, N.J. Rezanova. 2010. Disruption management in the airline industry - Concepts, models and methods. *Computers & Operations Research* **37**(5) 809–821.

Corman, F., A. D'Ariano, D. Pacciarelli, M. Pranzo. 2010. A tabu search algorithm for rerouting trains during rail operations. *Transportation Research Part B* **44**(1) 175–192.

Cucala, A.P., A. Fernández, C. Sicre, M. Domínguez. 2012. Fuzzy optimal schedule of high speed train operation to minimize energy consumption with uncertain delays and driver's behavioral response. *Engineering Applications of Artificial Intelligence* **25** 1548–1557.

Du, Y., Q. Meng, Y. Wang. 2015. Budgeting the fuel consumption of a container ship over a round voyage via robust optimization. *Transportation Research Record: Journal of the Transportation Research Board* **2477** 68–75.

Duran, A.S., S. Gürel, M.S. Aktürk. 2015. Robust Airline Scheduling with Controllable Cruise Times and Chance Constraints. *IIE Transactions* **47**(1) 64–83.

Engelhardt-Funke, O., M. Kolonko. 2004. Analysing stability and investments in railway networks using advanced evolutionary algorithms. *International Transactions in Operational Research* **11**(4) 381–394.

Fischetti, M., D. Salvagnin, A. Zanette. 2009. Fast Approaches to Improve the Robustness of a Railway Timetable. *Transportation Science* **43**(3) 321–335.

Gong, C., S. Zhang, F. Zhang, J. Jiang, X. Wang. 2014. An Integrated Energy-Efficient Operation Methodology for Metro Systems Based on a Real Case of Shanghai Metro Line One. *Energies* **7** 7305–7329.

Hassannayebi, E., A. Sajedinejad, S. Mardani. 2014. Urban rail transit planning using a two-stage simulation-based optimization approach. *Simulation Modelling Practice and Theory* **49** 151–166.

Kroon, L.G., R. Dekker, M.C.J.M. Vromans. 2007. Cyclic Railway Timetabling: A Stochastic Optimization Approach. F. Geraets, L.G. Kroon, A. Schöbel, D. Wagner, C.D. Zaroliagis, eds., *Algorithmic Methods for Railway Optimization*, *Lecture Notes in Computer Science*, vol. 4359. Springer, 41–66.

Kroon, L.G., G. Maróti, M.J. Retel Helmrich, M.J.C.M. Vromans, R. Dekker. 2008. Stochastic improvement of cyclic railway timetables. *Transportation Research Part B* **42**(6) 553–570.

Lan, S., J.P. Clarke, C. Barnhart. 2006. Planning for Robust Airline Operations: Optimizing

Aircraft Routings and Flight Departure Times to Minimize Passenger Disruptions. *Transportation Science* **40**(1) 15–28.

Li, C., X. Qi, C.Y. Lee. 2015a. Disruption Recovery for a Vessel in Liner Shipping. *Transportation Science* **49**(4) 900–921.

Li, C., X. Qi, D. Song. 2015b. Real-time schedule recovery in liner shipping service with regular uncertainties and disruption events. *Transportation Research Part B* 1–27, In Press.

Liebchen, C., M. Schachtebeck, A. Schöbel, S. Stiller, A. Prigge. 2010. Computing delay resistant railway timetables. *Computers & Operations Research* **37**(5) 857–868.

Lovász, L. 1983. Submodular functions and convexity. A. Bachem, B. Korte, M. Grötschel, eds., *Mathematical Programming The State of the Art: Bonn 1982*. Springer Berlin Heidelberg, 235–257.

Maher, S.J. 2015. Solving the Integrated Airline Recovery Problem Using Column-and-Row Generation. *Transportation Science* **50**(1) 216–239.

Norlund, E.K., I. Gribkovskaia, G. Laporte. 2015. Supply vessel planning under cost, environment and robustness considerations **57** 271–281.

Notteboom, T.E., B. Vernimmen. 2009. The effect of high fuel costs on liner service configuration in container shipping. *Journal of Transport Geography* **17**(5) 325–337.

Petersen, J.D., G. Sölveling, J.P. Clarke, E.L. Johnson, S. Shebalov. 2012. An Optimization Approach to Airline Integrated Recovery. *Transportation Science* **46**(4) 482–500.

Psaraftis, H.N., C.A. Kontovas. 2013. Speed models for energy-efficient maritime transportation: A taxonomy and survey. *Transportation Research Part C* **26** 331–351.

Qi, X., D.P. Song. 2012. Minimizing fuel emissions by optimizing vessel schedules in liner shipping with uncertain port times. *Transportation Research Part E* **48**(4) 863–880.

Rosenberger, J.M., E.L. Johnson, G.L. Nemhauser. 2003. Rerouting Aircraft for Airline Recovery. *Transportation Science* **37**(4) 408–421.

SeaRates. 2015. Reference quide, port to port distance. `https://www.searates.com/reference/portdistance`. Accessed: 19 October 2015.

Visentini, M.S., D. Borenstein, J.Q. Li, P.B. Mirchandani. 2014. Review of real-time vehicle schedule recovery methods in transportation services. *Journal of Scheduling* **17**(6) 541–567.

Wang, S., Q. Meng. 2012a. Liner ship route schedule design with sea contingency time and port time uncertainty. *Transportation Research Part B* **46**(5) 615–633.

Wang, S., Q. Meng. 2012b. Robust schedule design for liner shipping services. *Transportation Research Part E* **48**(6) 1093–1106.

Wu, C.L. 2005. Inherent delays and operational reliability of airline schedules. *Journal of Air Transport Management* **11**(4) 273–282.

Wu, C.L., R.E. Caves. 2003. The punctuality performance of aircraft rotations in a network of airports. *Transportation Planning and Technology* **26**(5) 417–436.

Wu, Y., J. Tang, Y. Yu, Z. Pan. 2015. A stochastic optimization model for transit network timetable design to mitigate the randomness of traveling time by adding slack time. *Transportation Research Part C* **52** 15–31.

A Proofs of theoretical results

To simplify notation, define

$$\mathcal{L}_n(d;\vec{B}) := \min_{0\le\tau\le\tau^u_{p[n]}} \left\{\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d+\tau;\vec{B})\right\}.$$

Then $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ can be written as

$$\mathcal{C}^{dep}_{n,N}(d;\vec{B}) = \mathcal{D}^{dep}_{p[n]}(d) + \mathcal{L}_n(d;\vec{B}). \tag{21}$$

Proof of Lemma 1. By backward induction in n, starting at N. Let $\vec{B} \in \vec{\mathcal{B}}$ be arbitrary. For $n = N$, $\mathcal{C}^{arr}_{N,N}(d;\vec{B}) = \mathcal{D}^{arr}_{p[N]}(d)$, which is a nondecreasing function in d by assumption.

Assume now that $\mathcal{C}^{arr}_{n+1,N}(d;\vec{B})$ is nondecreasing in d for some $1 < n < N$. We will prove that $\mathcal{C}^{arr}_{n,N}(d;\vec{B})$ is also nondecreasing in d. Let $d, d' \in \mathbb{R}_{\ge 0}$ be arbitrary such that $d' \ge d$. Then,

$$\begin{aligned}\mathcal{K}_n(d;\vec{B}) &= \mathbb{E}_{X^s_n}\left[\mathcal{C}^{arr}_{n+1,N}\left(\left(d - \vec{B}_{p[n]} + X^s_n\right)^+;\vec{B}\right)\right]\\ &\le \mathbb{E}_{X^s_n}\left[\mathcal{C}^{arr}_{n+1,N}\left(\left(d' - \vec{B}_{p[n]} + X^s_n\right)^+;\vec{B}\right)\right]\\ &= \mathcal{K}_n(d';\vec{B}),\end{aligned}$$

where the inequality follows from the induction hypothesis and because X^s_n does not depend on the current delay, since by assumption X^s_n is independent of all other random variables.

This proves that $\mathcal{K}_n(d;\vec{B})$ is nondecreasing in d. Then,

$$\begin{aligned}\mathcal{L}_n(d;\vec{B}) &= \min_{0\le\tau\le\tau^u_{p[n]}}\left\{\mathcal{F}_{p[n]}(\tau)+\mathcal{K}_n\left(d+\tau;\vec{B}\right)\right\}\\ &\le \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)+\mathcal{K}_n\left(d+\mathcal{T}_n(d';\vec{B});\vec{B}\right)\\ &\le \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)+\mathcal{K}_n\left(d'+\mathcal{T}_n(d';\vec{B});\vec{B}\right)\\ &= \mathcal{L}_n(d';\vec{B}),\end{aligned}$$

where the first inequality holds because $0\le\mathcal{T}_n(d';\vec{B})\le\tau^u_{p[n]}$ and the second because $\mathcal{K}_n(d;\vec{B})$ is nondecreasing in d. Hence, $\mathcal{L}_n(d;\vec{B})$ is nondecreasing in d. By (21) we know that $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ is the sum of two nondecreasing functions, namely $\mathcal{D}^{dep}_{p[n]}(d)$ and $\mathcal{L}_n(d;\vec{B})$, which proves that $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ is also nondecreasing in d.

Further, $\mathcal{C}^{arr}_{n,N}(d;\vec{B})$ is the sum of $\mathcal{D}^{arr}_{p[n]}(d)$, which is nondecreasing in d by assumption, and $\mathbb{E}_{X^p_n}\left[\min_{\gamma\ge0}\left\{c^e\gamma+\mathcal{C}^{dep}_{n,N}\left((d+X^p_n-\gamma)^+;\vec{B}\right)\right\}\right]$, for which we find:

$$\begin{aligned}&\mathbb{E}_{X^p_n}\left[\min_{\gamma\ge0}\left\{c^e\gamma+\mathcal{C}^{dep}_{n,N}\left((d+X^p_n-\gamma)^+;\vec{B}\right)\right\}\right]\\ &\le \mathbb{E}_{X^p_n}\left[c^e\mathcal{Y}_n(d'+X^p_n;\vec{B})+\mathcal{C}^{dep}_{n,N}\left(\left(d+X^p_n-\mathcal{Y}_n(d'+X^p_n;\vec{B})\right)^+;\vec{B}\right)\right]\\ &\le \mathbb{E}_{X^p_n}\left[c^e\mathcal{Y}_n(d'+X^p_n;\vec{B})+\mathcal{C}^{dep}_{n,N}\left(\left(d'+X^p_n-\mathcal{Y}_n(d'+X^p_n;\vec{B})\right)^+;\vec{B}\right)\right]\\ &= \mathbb{E}_{X^p_n}\left[\min_{\gamma\ge0}\left\{c^e\gamma+\mathcal{C}^{dep}_{n,N}\left((d'+X^p_n-\gamma)^+;\vec{B}\right)\right\}\right]\end{aligned}$$

where the second inequality holds because $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ is nondecreasing in d. The last equality holds because the additional delay incurred is independent of the current delay, because by assumptions it is independent of all other random variables. Hence, $\mathcal{C}^{arr}_{n,N}(d;\vec{B})$ is nondecreasing in d, which completes the induction argument. □

Proof of Lemma 2. By backward induction in n, starting at N. By assumption $\mathcal{C}^{arr}_{N,N}(d;\vec{B})=\mathcal{D}^{arr}_{p[N]}(d)$ is joint convex in d and $\vec{B}$. Now suppose $\mathcal{C}^{arr}_{n+1,N}(d;\vec{B})$ is joint convex in d and $\vec{B}$ for some $1<n<N$. Let $d,d'\in\mathbb{R}_{\ge0}$ be arbitrary nonnegative real numbers and let

$\vec{B}, \vec{B}' \in \bar{\mathcal{B}}$ and $\lambda \in [0,1]$ be arbitrary. Then,

$$
\begin{aligned}
&\lambda \mathcal{K}_n(d;\vec{B}) + (1-\lambda)\mathcal{K}_n(d';\vec{B}') \\
&\quad = \lambda \mathbb{E}_{X_n^s}\left[\mathcal{C}^{arr}_{n+1,N}\left(\left(d - \vec{B}_{p[n]} + X_n^s\right)^+; \vec{B}\right)\right] + (1-\lambda)\mathbb{E}_{X_n^s}\left[\mathcal{C}^{arr}_{n+1,N}\left(\left(d' - \vec{B}'_{p[n]} + X_n^s\right)^+; \vec{B}'\right)\right] \\
&\quad \geq \mathbb{E}_{X_n^s}\left[\mathcal{C}^{arr}_{n+1,N}\left(\lambda\left(d - \vec{B}_{p[n]} + X_n^s\right)^+ + (1-\lambda)\left(d' - \vec{B}'_{p[n]} + X_n^s\right)^+; \lambda\vec{B} + (1-\lambda)\vec{B}'\right)\right] \\
&\quad \geq \mathbb{E}_{X_n^s}\left[\mathcal{C}^{arr}_{n+1,N}\left(\left(\lambda(d - \vec{B}_{p[n]} + X_n^s) + (1-\lambda)(d' - \vec{B}'_{p[n]} + X_n^s)\right)^+; \lambda\vec{B} + (1-\lambda)\vec{B}'\right)\right] \\
&\quad = \mathcal{K}_n(\lambda d + (1-\lambda)d'; \lambda\vec{B} + (1-\lambda)\vec{B}'),
\end{aligned}
$$

where the first inequality holds by the induction hypothesis and because X_n^s is independent of the current delay, since by assumption it is independent of all other random variables. The second inequality follows because $\mathcal{C}^{arr}_{n+1,N}(d,\vec{B})$ nondecreasing in d. It follows that $\mathcal{K}_n(d;\vec{B})$ is also joint convex in d and $\vec{B}$. Next,

$$
\begin{aligned}
&\lambda \mathcal{L}_n(d;\vec{B}) + (1-\lambda)\mathcal{L}_n(d';\vec{B}') \\
&\quad = \lambda\left(\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right) + \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right)\right) + \\
&\qquad (1-\lambda)\left(\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B}')\right) + \mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B}');\vec{B}'\right)\right) \\
&\quad = \lambda\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right) + (1-\lambda)\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B}')\right) + \\
&\qquad \lambda\mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right) + (1-\lambda)\mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B}');\vec{B}'\right) \\
&\quad \geq \mathcal{F}_{p[n]}\left(\lambda\mathcal{T}_n(d;\vec{B}) + (1-\lambda)\mathcal{T}_n(d';\vec{B}')\right) + \\
&\qquad \mathcal{K}_n\left(\lambda\left(d + \mathcal{T}_n(d;\vec{B})\right) + (1-\lambda)\left(d' + \mathcal{T}_n(d';\vec{B}')\right); \lambda\vec{B} + (1-\lambda)\vec{B}'\right) \\
&\quad \geq \min_{0 \leq \tau \leq \tau^u_{p[n]}}\left\{\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n\left(\tau + \lambda d + (1-\lambda)d'; \lambda\vec{B} + (1-\lambda)\vec{B}'\right)\right\} \\
&\quad = \mathcal{L}_n\left(\lambda d + (1-\lambda)d'; \lambda\vec{B} + (1-\lambda)\vec{B}'\right)
\end{aligned}
$$

where the first inequality follows because $\mathcal{F}_{p[n]}$ is convex and $\mathcal{K}_n$ is joint convex. The second inequality holds because $0 \leq \mathcal{T}_n(d;\vec{B}), \mathcal{T}_n(d';\vec{B}') \leq \tau^u_{p[n]}$. Hence, $\mathcal{L}_n(d;\vec{B})$ is joint convex in d and $\vec{B}$. Then, $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ is the sum of two (joint) convex functions, so $\mathcal{C}^{dep}_{n,N}(d;\vec{B})$ is joint convex in d and $\vec{B}$.

Further, $\mathcal{C}^{arr}_{n,N}(d;\vec{B})$ is the sum of $\mathcal{D}^{arr}_{p[n]}(d)$, which is convex in d and hence joint convex in $(d,\vec{B})$ by assumption, and $\mathbb{E}_{X_n^p}\left[\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}((d_n^{arr} + X_n^p - \gamma)^+;\vec{B})\right\}\right]$, for which we find:

$$
\begin{aligned}
&\lambda \mathbb{E}_{X_n^p}\left[\left(c^e \mathcal{Y}_n(d+X_n^p;\vec{B}) + \mathcal{C}_{n,N}^{dep}\left(\left(d+X_n^p-\mathcal{Y}_n(d+X_n^p;\vec{B})\right)^+;\vec{B}\right)\right)\right] + \\
&(1-\lambda)\mathbb{E}_{X_n^p}\left[\left(c^e \mathcal{Y}_n(d'+X_n^p;\vec{B}') + \mathcal{C}_{n,N}^{dep}\left(\left(d'+X_n^p-\mathcal{Y}_n(d'+X_n^p;\vec{B}')\right)^+;\vec{B}'\right)\right)\right] \\
\geq\ &\mathbb{E}_{X_n^p}\left[c^e\left(\lambda\mathcal{Y}_n(d+X_n^p;\vec{B}) + (1-\lambda)\mathcal{Y}_n(d'+X_n^p;\vec{B}')\right)\right] + \\
&\mathbb{E}_{X_n^p}\Big[\mathcal{C}_{n,N}^{dep}\Big(\lambda\left(d+X_n^p-\mathcal{Y}_n(d+X_n^p;\vec{B})\right)^+ + (1-\lambda)\left(d'+X_n^p-\mathcal{Y}_n(d'+X_n^p;\vec{B}')\right)^+; \\
&\lambda\vec{B}+(1-\lambda)\vec{B}'\Big)\Big] \\
\geq\ &\mathbb{E}_{X_n^p}\left[c^e\left(\lambda\mathcal{Y}_n(d+X_n^p;\vec{B}) + (1-\lambda)\mathcal{Y}_n(d'+X_n^p;\vec{B}')\right)\right] + \\
&\mathbb{E}_{X_n^p}\Big[\mathcal{C}_{n,N}^{dep}\Big(\left(\lambda d+(1-\lambda)d'+X_n^p-\left(\lambda\mathcal{Y}_n(d+X_n^p;\vec{B})+(1-\lambda)\mathcal{Y}_n(d'+X_n^p;\vec{B}')\right)\right)^+; \\
&\lambda\vec{B}+(1-\lambda)\vec{B}'\Big)\Big] \\
\geq\ &\mathbb{E}_{X_n^p}\left[\min_{\gamma\geq 0}\left\{c^e\gamma + \mathcal{C}_{n,N}^{dep}\left(\left(\lambda d+(1-\lambda)d'+X_n^p-\gamma\right)^+;\lambda\vec{B}+(1-\lambda)\vec{B}'\right)\right\}\right]
\end{aligned}
$$

where the first inequality holds by the induction hypothesis and the second inequality holds because $\mathcal{C}_{n,N}^{dep}(d;\vec{B})$ is nondecreasing in d. Hence, $\mathcal{C}_{n,N}^{arr}(d;\vec{B})$ is joint convex in d and $\vec{B}$, which proves the lemma. □

Proof of Theorem 1. The $\lim_{R\to\infty}\frac{\mathcal{C}_{1,R|P|+1}^{dep}(d_1^{dep};\vec{B})}{R}$ exists by assumption, and since convexity is preserved when taking limits, this limit is joint convex in $\vec{B}$ and d_1^{dep} by Lemma 2. By assumption, the limit is independent of d_1^{dep}, which implies the desired result. □

Proof of Lemma 3. Take any $n\in[1,N-1]$ and $\vec{B}\in\bar{\mathcal{B}}$ and let $d,d'\in\mathbb{R}_{\geq 0}$ such that $d'\geq d$. We will prove that $\mathcal{T}_n(d;\vec{B})\geq\mathcal{T}_n(d';\vec{B})$ by contradiction. Suppose $\mathcal{T}_n(d;\vec{B})<\mathcal{T}_n(d';\vec{B})$. By (6) it follows that

$$
\begin{aligned}
\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right)+\mathcal{K}_n\left(d+\mathcal{T}_n(d;\vec{B});\vec{B}\right) &\leq \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)+\mathcal{K}_n\left(d+\mathcal{T}_n(d';\vec{B});\vec{B}\right) \\
\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right)-\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right) &\leq \mathcal{K}_n\left(d+\mathcal{T}_n(d';\vec{B});\vec{B}\right)-\mathcal{K}_n\left(d+\mathcal{T}_n(d;\vec{B});\vec{B}\right).
\end{aligned}
$$

Furthermore, by (6) it follows that:

$$
\begin{aligned}
\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)+\mathcal{K}_n\left(d'+\mathcal{T}_n(d';\vec{B});\vec{B}\right) &< \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right)+\mathcal{K}_n\left(d'+\mathcal{T}_n(d;\vec{B});\vec{B}\right) \\
\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right)-\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right) &> \mathcal{K}_n\left(d'+\mathcal{T}_n(d';\vec{B});\vec{B}\right)-\mathcal{K}_n\left(d'+\mathcal{T}_n(d;\vec{B});\vec{B}\right).
\end{aligned}
$$

Hence,

$$\mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B});\vec{B}\right) - \mathcal{K}_n\left(d' + \mathcal{T}_n(d;\vec{B});\vec{B}\right) < \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right) - \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)$$
$$\leq \mathcal{K}_n\left(d + \mathcal{T}_n(d';\vec{B});\vec{B}\right) - \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right),$$

thus $\mathcal{K}_n(d;\vec{B})$ has decreasing increments, which contradicts the convexity of $\mathcal{K}_n(d;\vec{B})$. Hence, $\mathcal{T}_n(d;\vec{B}) \geq \mathcal{T}_n(d';\vec{B})$. The second part of the lemma can be proven analogously. □

Proof of Lemma 4. Let $n \in [1, N-1]$ be arbitrary and let $d, d' \in \mathbb{R}_{\geq 0}$ be arbitrary such that $d' \geq d$. We need to prove that $d + \mathcal{T}_n(d;\vec{B}) \leq d' + \mathcal{T}_n(d';\vec{B})$, because, since $\vec{B}_{p[n]}$ is fixed and X_n^s is independent of d_n^{dep} by assumption, this implies the desired result.

Define $\tau' := d + \mathcal{T}_n(d;\vec{B}) - d'$ and $\tau := d' + \mathcal{T}_n(d';\vec{B}) - d$. Assume now (by contradiction) that $d + \mathcal{T}_n(d;\vec{B}) > d' + \mathcal{T}_n(d';\vec{B})$. Then $\tau' > \mathcal{T}_n(d';\vec{B})$ and $\tau < \mathcal{T}_n(d;\vec{B})$. By (6)

$$\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right) + \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right) < \mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d + \tau;\vec{B}).$$

The inequality is strict because $\mathcal{T}_n(d;\vec{B})$ is by definition the smallest minimizer, see (6). By rearranging terms, we obtain

$$\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right) - \mathcal{F}_{p[n]}(\tau) < \mathcal{K}_n(d + \tau;\vec{B}) - \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right)$$
$$= \mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B});\vec{B}\right) - \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right).$$

where the equality is due to the definition of τ. For $\mathcal{T}_n(d';\vec{B})$, the definition (6) implies

$$\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right) + \mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B});\vec{B}\right) \leq \mathcal{F}_{p[n]}(\tau') + \mathcal{K}_n(d' + \tau';\vec{B}).$$

which implies

$$\mathcal{F}_{p[n]}(\tau') - \mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right) \geq \mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B});\vec{B}\right) - \mathcal{K}_n(d' + \tau';\vec{B})$$
$$= \mathcal{K}_n\left(d' + \mathcal{T}_n(d';\vec{B});\vec{B}\right) - \mathcal{K}_n\left(d + \mathcal{T}_n(d;\vec{B});\vec{B}\right).$$

Combining these, we obtain:

$$\begin{aligned}\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})\right)-\mathcal{F}_{p[n]}(\tau)&<\mathcal{K}_n\left(d'+\mathcal{T}_n(d';\vec{B});\vec{B}\right)-\mathcal{K}_n\left(d+\mathcal{T}_n(d;\vec{B});\vec{B}\right)\\&\le\mathcal{F}_{p[n]}(\tau')-\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d';\vec{B})\right)\\&=\mathcal{F}_{p[n]}\left(\mathcal{T}_n(d;\vec{B})-(d'-d)\right)-\mathcal{F}_{p[n]}(\tau-(d'-d)).\end{aligned}$$

Thus, $\mathcal{F}_{p[n]}(d)$ has decreasing increments which contradicts convexity. Thus, $d+\mathcal{T}_n(d;\vec{B})\le d'+\mathcal{T}_n(d';\vec{B})$. The second part of the lemma can be proven analogously. □

To prove Lemma 5, we first prove that the value functions are continuous and piecewise linear with specific breakpoints. Define, for a given $\vec{B}$ and corresponding $\tilde{B}$, $\Psi(Q_n;\tilde{B})$ as the set of functions that are piecewise linear, with breakpoints only on $Q_n(\tilde{B})$. (For convenience, we will write $\Psi(Q_n)$ for $\Psi(Q_n;\tilde{B})$.) More precisely: for any $f(\cdot)\in\Psi(Q_n)$ and any open interval $(\underline{d},\bar{d})$ that does not intersect $Q_n(\tilde{B})$ (thus $(\underline{d},\bar{d})\subseteq\mathbb{R}\setminus Q_n(\tilde{B})$), there exist a slope $a\in\mathbb{R}$ and an offset $b\in\mathbb{R}$ such that $\forall d\in(\underline{d},\bar{d}):f(d)=ad+b$.

Lemma 6 (Auxiliary towards Lemma 5)**.** *For every n with $1\le n<N$: $\mathcal{K}_n(\,\cdot\,;\vec{B})\in\Psi(Q_n)$, $\mathcal{C}^{dep}_{n,N}(\,\cdot\,;\vec{B})\in\Psi(Q_n)$, $\mathcal{C}^{arr}_{n+1,N}(\,\cdot\,;\vec{B})\in\Psi(Q_{n+1})$. This yields additional results for the actions (with $\tilde{B}$ the cumulative buffer allocation corresponding to $\vec{B}$):*

1. *Optimal sailing time: for every n with $1\le n<N$ and every $d\ge0$ that $\mathcal{T}_n(d;\vec{B})\in\mathbb{Z}_{\ge0}$ and/or $d+\mathcal{T}_n(d;\vec{B})\in Q_n(\tilde{B})$.*
2. *Extreme actions: for every n with $1<n<N$ and every $d\ge0$ that $\mathcal{Y}_n(d;\vec{B})=0$ and/or $d-\mathcal{Y}_n(d;\vec{B})\in Q_n(\tilde{B})$.*

Proof of Lemma 6. We will prove the lemma by induction. For the base case, note that $\mathcal{C}^{arr}_{N,N}(d^{arr}_N;\vec{B})=\mathcal{D}^{arr}_{p[N]}(d^{arr}_N)\in\Psi(Q_N)$, since $\mathcal{D}^{arr}_{p[N]}(d^{arr}_N)$ is piecewise linear with breakpoints on $\mathbb{Z}_{\ge0}$ by Assumption 1, and $\mathbb{Z}_{\ge0}\subseteq Q_N(\tilde{B})$ because $\mathbb{Z}_{\ge0}=Q_N(p[N];\tilde{B})$.

Thus, for some n with $1\le n<N$, the induction hypothesis is $\mathcal{C}^{arr}_{n+1,N}(d;\vec{B})\in\Psi(Q_{n+1})$, and we will show that $\mathcal{K}_n(d;\vec{B})\in\Psi(Q_n)$, $\mathcal{C}^{dep}_{n,N}(d;\vec{B})\in\Psi(Q_n)$ and $\mathcal{C}^{arr}_{n,N}(d;\vec{B})\in\Psi(Q_n)$.

We first show that $\mathcal{K}_n(d;\vec{B})=\mathbb{E}_{X^s_n}\left[\mathcal{C}^{arr}_{n+1,N}\left((d-\vec{B}_{p[n]}+X^s_n)^+;\vec{B}\right)\right]$ is in $\Psi(Q_n)$. Conditioned on X^s_n, by Assumption 1 there exists $z^s\in\mathbb{Z}_{\ge0}$ such that $X^s_n=z^s$. By definition of the cumulative buffers $\tilde{B}'$ it holds that $\tilde{B}'_{p[n+1]}-\vec{B}'_{p[n]}=\tilde{B}'_{p[n]}+z_n$, with $z_n=B\in\mathbb{Z}$

if $p[n+1] = 1$ and $z_n = 0$ otherwise. Fix any open interval $(\underline{d}, \bar{d}) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$. For any $d \in (\underline{d}, \bar{d})$ suppose $d - \vec{B}_{p[n]} + z^s \in Q_{n+1}(\tilde{B})$. That would imply $\exists z \in \mathbb{Z}, p \in P$ such that $d - \vec{B}_{p[n]} + z^s = z + \tilde{B}_p - \tilde{B}_{p[n+1]}$, and thus $d = (z - z_s) + \tilde{B}_p - (\tilde{B}_{p[n+1]} - \vec{B}_{p[n]}) = (z - z^s - z_n) + \tilde{B}_p - \tilde{B}_{p[n]} \in Q_n(\tilde{B})$, a contradiction with $(\underline{d}, \bar{d}) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$. Hence, it holds that $d - \vec{B}_{p[n]} + z^s \notin Q_{n+1}(\tilde{B})$. Thus $(\underline{d} - \vec{B}_{p[n]} + z^s, \bar{d} - \vec{B}_{p[n]} + z^s) \subseteq \mathbb{R} \setminus Q_{n+1}(\tilde{B})$. Since $0 \in Q_{n+1}(\tilde{B})$, the following two cases are exhaustive: 1) $\forall d \in (\underline{d}, \bar{d}) : d - \vec{B}_{p[n]} + z^s \geq 0$ and 2) $\forall d \in (\underline{d}, \bar{d}) : d - \vec{B}_{p[n]} + z^s \leq 0$. For the first case, by induction hypothesis and since $(\underline{d} - \vec{B}_{p[n]} + z^s, \bar{d} - \vec{B}_{p[n]} + z^s) \subseteq \mathbb{R} \setminus Q_{n+1}(\tilde{B})$, we know that $\exists a, b \in \mathbb{R}$ such that $\forall d \in (\underline{d}, \bar{d})$:

$$\mathcal{C}^{arr}_{n+1,N}\left((d - \vec{B}_{p[n]} + z^s)^+; \vec{B}\right) = \mathcal{C}^{arr}_{n+1,N}\left(d - \vec{B}_{p[n]} + z^s; \vec{B}\right) = a(d - \vec{B}_{p[n]} + z^s) + b.$$

Note that the RHS is affine in d. For the second case, we find

$$\forall d \in (\underline{d}, \bar{d}) : \mathcal{C}^{arr}_{n+1,N}\left((d - \vec{B}_{p[n]} + z^s)^+; \vec{B}\right) = \mathcal{C}^{arr}_{n+1,N}\left(0; \vec{B}\right) = a'd + b',$$

with $a' = 0$ and $b' = \mathcal{C}^{arr}_{n+1,N}\left(0; \vec{B}\right)$. Now, since

$$\mathcal{K}_n(d; \vec{B}) = \sum_{z^s \in \mathbb{Z}_{\geq 0}} \mathbb{P}(X^s_n = z^s) \mathcal{C}^{arr}_{n+1,N}\left((d - \vec{B}_{p[n]} + z^s)^+; \vec{B}\right),$$

and since each of the functions on the RHS is affine in d for all $d \in (\underline{d}, \bar{d})$, $\mathcal{K}_n(d; \vec{B})$ is affine in d for $d \in (\underline{d}, \bar{d})$. This proves $\mathcal{K}_n(d; \vec{B}) \in \Psi(Q_n)$.

We next show that $\mathcal{C}^{dep}_{n,N}(d; \vec{B}) = \mathcal{D}^{dep}_{p[n]}(d) + \mathcal{L}_n(d; \vec{B}) \in \Psi(Q_n)$, where $\mathcal{L}_n(d; \vec{B}) = \min_{0 \leq \tau \leq \tau^u_{p[n]}} \left\{\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d + \tau; \vec{B})\right\}$. Since $\mathcal{D}^{dep}_{p[n]}(d)$ is piecewise linear with breakpoints on $\mathbb{Z}_{\geq 0} \subseteq Q_n(\tilde{B})$ by Assumption 1, it remains to show that $\mathcal{L}_n(d; \vec{B}) \in \Psi(Q_n)$. Fix an interval $(\underline{d}, \bar{d}) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$, and let $d \in (\underline{d}, \bar{d})$. For brevity, let $\tau^* = \mathcal{T}(d; \vec{B})$ denote the optimal sailing time for d. By (6), τ^* is the smallest minimizer of $\mathcal{F}_{p[n]}(\tau) + \mathcal{K}_n(d + \tau; \vec{B})$, and therefore τ^* must be one of the breakpoints of $\mathcal{F}_{p[n]}(\,\cdot\,)$ (which occur at $\mathbb{Z}_{\geq 0}$) and/or $d + \tau^*$ must be one of the breakpoints of $\mathcal{K}_n(\,\cdot\,; \vec{B})$ (which occur at $Q_n(\tilde{B})$ by $\mathcal{K}_n(\,\cdot\,; \vec{B}) \in \Psi(Q_n)$). It thus suffices to consider the following two cases: 1) $\tau^* \in \mathbb{Z}_{\geq 0}$ and 2) $\tau^* + d \in Q_n(\tilde{B})$. (This is additional result 1.)

For Case 1, note that for every $d' \in (\underline{d}, \bar{d})$ we have $d' \notin Q_n(\tilde{B})$ and $\tau^* \in \mathbb{Z}_{\geq 0}$ and thus $d' + \tau^* \notin Q_n(\tilde{B})$. This implies $(\underline{d} + \tau^*, \bar{d} + \tau^*) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$. Thus, by $\mathcal{K}_n(d; \vec{B}) \in \Psi(Q_n)$ there exist $a, b \in \mathbb{R}$ such that for every $d' \in (\underline{d}, \bar{d})$:

$$\mathcal{L}_n(d'; \vec{B}) \leq \mathcal{F}_{p[n]}(\tau^*) + \mathcal{K}_n(d' + \tau^*; \vec{B}) = \mathcal{F}_{p[n]}(\tau^*) + a(d' + \tau^*) + b = a'd' + b' \tag{22}$$

with $a' = a$ and $b' = b + a\tau^* + \mathcal{F}_{p[n]}(\tau^*)$. Write $d' = d + x$. We now show that $\mathcal{L}_n(d + x; \vec{B}) = a'(d + x) + b'$. This is immediate for $x = 0$, so suppose $x \neq 0$. Let $\epsilon > 0$ be such that $d - \epsilon x \in (\underline{d}, \bar{d})$. The proof of Lemma 2 shows that $\mathcal{L}_n(d; \vec{B})$ is joint convex in $(d, \vec{B})$, and therefore convex in d, which implies $\lambda \mathcal{L}_n(d + x; \vec{B}) + (1 - \lambda)\mathcal{L}_n(d - \epsilon x; \vec{B}) \geq \mathcal{L}_n\big(\lambda(d + x) + (1 - \lambda)(d - \epsilon x); \vec{B}\big)$ for any $\lambda \in [0, 1]$. Setting $\lambda = \epsilon/(1 + \epsilon)$ and multiplying by $(1 + \epsilon)$ yields:

$$\begin{aligned}
\epsilon \mathcal{L}_n(d + x; \vec{B}) &\geq (1 + \epsilon)\mathcal{L}_n(d; \vec{B}) - \mathcal{L}_n(d - \epsilon x; \vec{B}) \\
&\geq (1 + \epsilon)[a'd + b'] - [a'(d - \epsilon x) + b'] \\
&= \epsilon[a'(d + x) + b'] \\
&= \epsilon[\mathcal{F}_{p[n]}(\tau^*) + \mathcal{K}_n(d + x + \tau^*; \vec{B})] \geq \epsilon \mathcal{L}_n(d + x; \vec{B})
\end{aligned}$$

where the second inequality results from (22) and optimality of τ^* for d, the equality at the third line rearranges terms, and the final (in)equalities result from (22). This shows $\mathcal{L}_n(d'; \vec{B}) = a'd' + b'$ (which implies that τ^* is optimal for every $d' \in (\underline{d}, \bar{d})$). Thus for Case 1 we have established that $\mathcal{L}_n(d'; \vec{B})$ is affine in d' for all $d' \in (\underline{d}, \bar{d})$.

Now Case 2: $\tau^* + d \in Q_n(\tilde{B})$. For any $d' \in (\underline{d}, \bar{d})$, we will show that the action $\tau' = \tau^* - d' + d$ is optimal. Because $\tau' + d' = \tau^* + d \in Q_n(\tilde{B})$ and $d' \notin Q_n(\tilde{B})$, we know that $\tau' \notin \mathbb{Z}_{\geq 0}$. This implies $(\tau^* - \bar{d} + d, \tau^* - \underline{d} + d) \subseteq \mathbb{R} \setminus \mathbb{Z}_{\geq 0}$. Because $\mathcal{F}_{p[n]}(\tau)$ is piecewise linear with breakpoints on $\mathbb{Z}_{\geq 0}$ by Assumption 1, we now know that there exist $a, b \in \mathbb{R}$ such that for every $\tau' \in (\tau^* - \bar{d} + d, \tau^* - \underline{d} + d)$ it holds that $\mathcal{F}_{p[n]}(\tau') = a\tau' + b$. This yields:

$$\mathcal{L}_n(d'; \vec{B}) \leq a\tau' + b + \mathcal{K}_n(d' + \tau'; \vec{B}) = a(\tau^* - d' + d) + b + \mathcal{K}_n(d + \tau^*; \vec{B}) = a'd' + b' \tag{23}$$

with $a' = -a$ and $b' = a\tau^* + ad + b + \mathcal{K}_n(d + \tau^*; \vec{B})$. This allows us to show that $\mathcal{L}_n(d'; \vec{B}) = a'd' + b'$, exactly in the same fashion as for Case 1, using (23) and convexity of $\mathcal{L}_n(d'; \vec{B})$. Thus also for Case 2, we have established that $\mathcal{L}_n(d'; \vec{B})$ is affine in d' for all $d' \in (\underline{d}, \bar{d})$.

Since the two cases are exhaustive, we have shown that $\mathcal{L}_n(d'; \vec{B})$ is affine in d' for all $d' \in (\underline{d}, \bar{d})$. This shows that $\mathcal{L}_n(d'; \vec{B}) \in \Psi(Q_n)$, and thus $\mathcal{C}^{dep}_{n,N}(d; \vec{B}) = \mathcal{D}^{dep}_{p[n]}(d) + \mathcal{L}_n(d; \vec{B}) \in \Psi(Q_n)$.

Finally, we show that $\mathcal{C}^{arr}_{n,N}(d; \vec{B}) = \mathcal{D}^{arr}_{p[n]}(d) + \mathbb{E}_{X^p_n}\left[\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d + X^p_n - \gamma)^+; \vec{B}\right)\right\}\right]$ is in $\Psi(Q_n)$. We condition on X^p_n, and write $X^p_n = z^p$, with $z^p \in \mathbb{Z}_{\geq 0}$ by Assumption 1. We first show that $\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d + z^p - \gamma)^+; \vec{B}\right)\right\} \in \Psi(Q_n)$. Fix an interval $(\underline{d}, \bar{d}) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$, and let $d \in (\underline{d}, \bar{d})$. Denote $\gamma^* = \mathcal{Y}(d + z^p, \vec{B})$ for brevity. Since $c^e > 0$ and $d + z^p \geq 0$, optimality of γ^* implies that $\gamma^* \leq d + z^p$, and thus $(d + z^p - \gamma^*)^+ = d + z^p - \gamma^*$. Also, γ^* is the largest minimizer of $c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d + z^p - \gamma)^+; \vec{B}\right)$, and the following two cases are thus exhaustive: Case 1) $\gamma^* = 0$ and Case 2) $(d + z^p - \gamma^*)^+ = d + z^p - \gamma^*$ is a breakpoint of $\mathcal{C}^{dep}_{n,N}(\,\cdot\,; \vec{B})$, and thus $d + z^p - \gamma^* \in Q_n(\tilde{B})$. (This is additional result 2.)

For Case 1, since $\forall d' \in (\underline{d}, \bar{d}) : d' \notin Q_n(\tilde{B})$, we know that $d' + z^p - \gamma^* \notin Q_n(\tilde{B})$, and thus $(\underline{d} + z^p - \gamma^*, \bar{d} + z^p - \gamma^*) \subseteq \mathbb{R} \setminus Q_n(\tilde{B})$. Therefore, by $\mathcal{C}^{dep}_{n,N}(\cdot; \vec{B}) \in \Psi(Q_n)$, there exist $a, b \in \mathbb{R}$ such that $\forall d' \in (\underline{d}, \bar{d})$:

$$\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d' + z^p - \gamma)^+; \vec{B}\right)\right\} \leq c^e\gamma^* + \mathcal{C}^{dep}_{n,N}(d' + z^p - \gamma^*; \vec{B}) = ad' + b \tag{24}$$

Because $\gamma^* = 0$ is optimal for $d' = d$ by definition, we can proceed in the same way as before to show that the inequality in (24) can be strengthened to an equality. For Case 2, we note that $\gamma' = \gamma^* + d' - d > 0$ for $d' \in (\underline{d}, \bar{d})$, because $\gamma' = 0$ would contradict $d' + z^p - \gamma' = d + z^p - \gamma^* \in Q_n(\tilde{B})$, since $d' + z^p \notin Q_n(\tilde{B})$ by $d' \in (\underline{d}, \bar{d})$. We obtain:

$$\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d' + z^p - \gamma)^+; \vec{B}\right)\right\} \leq c^e(\gamma^* + d' - d) + \mathcal{C}^{dep}_{n,N}(d + z^p - \gamma^*; \vec{B}) = ad' + b, \tag{25}$$

with $a = c^e$ and $b = c^e(\gamma^* - d) + \mathcal{C}^{dep}_{n,N}(d + z^p - \gamma^*; \vec{B})$. Since γ^* is optimal for d by assumption, we can proceed in the same way as before to show that the inequality in (25) can be strengthened to equality. This yields $\min_{\gamma \geq 0}\left\{c^e\gamma + \mathcal{C}^{dep}_{n,N}\left((d' + z^p - \gamma)^+; \vec{B}\right)\right\} \in \Psi(Q_n)$ and since $\mathcal{D}^{arr}_{p[n]}(d)$ is piecewise linear with breakpoints on $\mathbb{Z}_{\geq 0}$, we find that $\mathcal{C}^{arr}_{n,N}(d;\vec{B}) \in \Psi(Q_n)$, which completes the proof. $\square$

With this lemma, we are now ready to prove Lemma 5.

Proof of Lemma 5. The proof is by induction, starting at $n = 1$. Let $\vec{B}$ be the buffer allocation corresponding to $\tilde{B}$. Note that $d^{dep}_1 \in \mathbb{Z}_{\geq 0} \subseteq Q_1(\tilde{B})$ by Assumption 1 and by definition of $Q_1(\tilde{B})$. We will now assume that $d^{dep}_n \in Q_n(\tilde{B})$ holds for some n with $1 \leq n < N$.

By additional result 1 of Lemma 6, we must either have $\mathcal{T}_n(d^{dep}_n;\vec{B}) \in \mathbb{Z}_{\geq 0}$ or $d^{dep}_n + \mathcal{T}_n(d^{dep}_n;\vec{B}) \in Q_n(\tilde{B})$. Because by assumption $d^{dep}_n \in Q_n(\tilde{B})$, in both cases we obtain $d^{dep}_n + \mathcal{T}_n(d^{dep}_n;\vec{B}) \in Q_n(\tilde{B})$, and thus $\exists z \in \mathbb{Z}, p \in P$ such that $d^{dep}_n + \mathcal{T}_n(d^{dep}_n;\vec{B}) = z + \tilde{B}_p - \tilde{B}_{p[n]}$. Since X^s_n takes on integer values, write $X^s_n = z^s$ with $z^s \in \mathbb{Z}_{\geq 0}$. By definition of the cumulative buffers $\tilde{B}'$ it holds that $\tilde{B}'_{p[n]} + \vec{B}'_{p[n]} = \tilde{B}'_{p[n+1]} + z_n$, with $z_n = B \in \mathbb{Z}$ if $p[n+1] = 1$ and $z_n = 0$ otherwise. Thus $d^{arr}_{n+1} = (d^{dep}_n + \mathcal{T}_n(d^{dep}_n;\vec{B}) + X^s_n - \vec{B}_{p[n]})^+ = \big((z + z^s) + \tilde{B}_p - \tilde{B}_{p[n]} - \vec{B}_{p[n]}\big)^+ = \big((z + z^s - z_n) + \tilde{B}_p - \tilde{B}_{p[n+1]}\big)^+$. Now consider the cases $d^{arr}_{n+1} = 0$ and $d^{arr}_{n+1} > 0$. In the former case, $d^{arr}_{n+1} \in Q_{p[n+1]}(p[n+1];\tilde{B}) \subseteq Q_{p[n+1]}(\tilde{B})$, and in the latter case we find $d^{arr}_{n+1} = (z + z^s - z_n) + \tilde{B}_p - \tilde{B}_{p[n+1]} \in Q_{p[n+1]}(p;\tilde{B}) \subseteq Q_{p[n+1]}(\tilde{B})$. Thus $d^{arr}_{n+1} \in Q_{n+1}(\tilde{B})$. Since X^p_{n+1} takes on values in $\mathbb{Z}_{\geq 0}$, $d^{arr}_{n+1} + X^p_{n+1} \in Q_{n+1}(\tilde{B})$ follows immediately.

Now, by additional result 2 of Lemma 6, we must either have $\mathcal{Y}_{n+1}(d^{arr}_{n+1} + X^p_{n+1};\vec{B}) = 0$ or $d^{arr}_{n+1} + X^p_{n+1} - \mathcal{Y}_{n+1}(d^{arr}_{n+1} + X^p_{n+1};\vec{B}) \in Q_{n+1}(\tilde{B})$. Since $d^{arr}_{n+1} + X^p_{n+1} \in Q_{n+1}(\tilde{B})$, in both cases we obtain $d^{arr}_{n+1} + X^p_{n+1} - \mathcal{Y}_{n+1}(d^{arr}_{n+1} + X^p_{n+1};\vec{B}) \in Q_{n+1}(\tilde{B})$. Thus $d^{dep}_{n+1} = (d^{arr}_{n+1} + X^p_{n+1} - \mathcal{Y}_{n+1}(d^{arr}_{n+1} + X^p_{n+1};\vec{B}))^+ \in Q_{n+1}(\tilde{B})$. This completes the proof by induction. $\square$

Proof of Theorem 2. We first show that the limit in the corollary corresponds to the long-term average costs of a finite-state, finite-action SDP. This is straightforward, but a but tedious. We distinguish between departure states and port states: Departure states are identified by the delay d^{dep}_n and the port $p[n]$ and correspond to the moment of departure. Port states are identified by the delay $d^{arr}_n + X^p_n$ after incurring port delay and the port $p[n]$. (States depend only on $p[n]$, and not on n.)

Let $\tilde{B}$ be the cumulative buffer allocation corresponding to $\vec{B}$. For the extreme action, we impose the additional restriction that $\gamma_n \geq d^{arr}_n + X^p_n - d^{max}_{p[n]}$. This does not affect

$\mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})$, because $\gamma_n < d^{arr}_n + X^p_n - d^{max}_{p[n]}$ cannot be optimal since $\mathcal{D}_{p[n]}(d) - c^e d$ is monotonically increasing for all $d > d^{max}_{p[n]}$ by Assumption 2. We thus have $0 \leq d^{dep}_n \leq d^{max}_{p[n]} < \infty$, since early departure is not allowed. Also, $d^{arr}_n + X^p_n \geq d^{arr}_n \geq 0$ since early arrival is not allowed and $d^{arr}_{n+1} + X^p_{n+1} \leq (d^{dep}_n + \tau_n - \vec{B}_{p[n]} + X^s_n)^+ + X^p_{n+1} \leq d^{max}_{p[n]} + \tau^u_{p[n]} + X^{\mathrm{s,max}}_{p[n]} + X^{\mathrm{p,max}}_{p[n+1]} < \infty$. Thus, delays are bounded below and above. In addition, only delays in $Q_{p[n]}(\tilde{B})$ occur by Lemma 5, and we will thus restrict the delays to this set without affecting $\mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})$.

For the actions, we have $0 \leq \tau_n \leq \tau^u_{p[n]}$ by assumption. Since $c^e > 0$, it can never be optimal for γ_n to exceed $d^{arr}_n + X^p_n$ (for which we already found an upper bound), and $\gamma_n \geq 0$ by assumption. Thus actions can be bounded above and below. As a consequence of Lemma 5, we may impose that actions are in $Q_p(\tilde{B})$ for some $p \in P$ without affecting $\mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})$. This, together with boundedness of the actions, implies that only a finite number of actions need to be considered for each state.

Furthermore, $\mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})$ corresponds to the optimal expected costs incurred over R rounds, when starting with departure in port 1 and ending with arrival in port 1. During these R rounds, a total of $2R|P|$ states are visited, and thus $\mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})/(2R|P|)$ corresponds to the optimal average costs per state over the next $2R|P|$ states in a Markov Decision Problem (MDP), when starting with departure in port 1. This MDP has finitely many states and actions, by the above discussion. This implies that there exists a stationary deterministic policy that is average cost optimal (Bertsekas 2007, Prop. 4.1.3, Prop 4.1.7), proving the second claim of the corollary. Moreover, this implies that $\lim_{R\to\infty} \mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})$ $/(2R|P|)$ exists for all $d^{dep}_1 \in \mathbb{Z}_{\geq 0}$ (Bertsekas 2007, Prop. 4.1.2, Prop. 4.1.3). Thus, the limit $\lim_{R\to\infty} \mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})/R$ also exists for all $d^{dep}_1 \in \mathbb{Z}_{\geq 0}$. (Note that $d^{dep}_1 \in \mathbb{Z}_{\geq 0}$ by Assumption 1.)

We next prove that $\lim_{R\to\infty} \mathcal{C}^{dep}_{1,R|P|+1}(d^{dep}_1;\vec{B})/R$ is independent of $d^{dep}_1 \in \mathbb{Z}_{\geq 0}$. Thereto, we will prove that the *weak accessibility* condition holds, which states that the set of states can be partitioned into two subsets S_1 and S_2 such that the following holds: 1) States $s \in S_1$ are transient under every stationary policy. 2) For every two states $s, s' \in S_2$, state s' is *accessible* from state s (Bertsekas 2007, p199). A state s' is accessible from state s if there exists a stationary policy such that the probability of entering state s' in a finite number of transitions starting from state s is strictly positive (Bertsekas 2007, p199).

Note that if s' is accessible from s and s'' is accessible from s', then s'' is accessible from s. Indeed, there must exist a sequence of states starting at s, going to s' and finally to s'', and if we take the right actions in all these states, there is a positive probability that this sequence occurs when we start at s. Should this sequence contain multiple visits to the same state (with different prescribed actions), then removing the loops yields a sequence from s to s'' that visits all states only once. For this latter sequence, a policy exists such that the sequence happens with positive probability when starting at s, showing that s'' is accessible from s.

Note that the state s_0 representing $d^{dep}_{p[n]} = 0$ is accessible from all states, by any policy that sets $\gamma_{p[n]} = d^{arr}_{p[n]} + X^p_n$ for $p[n] \in P$. Hence, it remains to show that if a state s is recurrent under some policy, then s is accessible from s_0.

Recall that $X^{\text{s,max}}_{p[n]}$ and $X^{\text{p,max}}_{p[n]}$ are chosen such that $\mathbb{P}(X^s_n = X^{\text{s,max}}_{p[n]}) > 0$ and $\mathbb{P}(X^p_n = X^{\text{p,max}}_{p[n]}) > 0$ by Assumption 2. Moreover, assume that $B < \sum_{p \in P}(\tau^u_p + X^{\text{p,max}}_p + X^{\text{s,max}}_p)$. (The *degenerate* alternative $B \geq \sum_{p \in P}(\tau^u_p + X^{\text{p,max}}_p + X^{\text{s,max}}_p)$ is ignored because trivially optimal solutions $\vec{B}$ with $\vec{B}_p \geq X^{\text{p,max}}_{p-1} + \tau^u_p + X^{\text{s,max}}_p$, for $p \in P$ are feasible for this case.)

We show that there exists a port p for which the arrival delay $d^{arr}_p + X^p_n > d^{max}_p$ is accessible from s_0. Take a stationary policy Π which takes minimal action, i.e. it sets $\forall n : \tau_n = \tau^u_n$ and $\gamma_n = 0$ if $d^{arr}_n + X^p_n \leq d^{max}_{p[n]}$. Because $\gamma = 0$, there is a strictly positive probability that the additional delay incurred in a round tour equals $\sum_{p \in P}(\tau^u_p + X^{\text{p,max}}_p + X^{\text{s,max}}_p) - B > 0$ (namely, when we incur the maximum possible delay in each port and sea leg). Thus, possibly after multiple rounds, with positive probability we reach a port call n' for which $d^{arr}_{p[n']} + X^p_{n'} \geq d^{max}_{p[n']}$. From this state, *any* delay state for $d^{dep}_{p[n']}$ is accessible, by choosing $\gamma_{n'}$ appropriately.

Take a state $s_2 \in S_2$ that is recurrent under a certain policy Π'. A state corresponding to $d^{dep}_{p[n']} = d$ for some d is visited every round, and if s_2 is not accessible from such a state, then s_2 cannot be recurrent. Thus s_2 is accessible from a state $d^{dep}_{p[n']} = d$ for some d, say state s'. But we just showed that s' is accessible from s_0, and thus s_2 is also accessible from s_0. So, states that are recurrent under a policy communicate with s_0. This proves weak accessibility for our model, and thus that the long run average costs are independent of the starting state d^{dep}_0 (Bertsekas 2007, p199, Prop 4.2.3). □

Lemma 7 (Auxiliary towards Theorem 3)**.** *The set $\Delta(\tilde{B})$ can also be represented as:*

$$\Delta(\tilde{B}) := \left\{\tilde{B}' \in \tilde{\mathcal{B}} \,\middle|\, \forall p, p' \in P : \left\lfloor \tilde{B}_{p'} - \tilde{B}_p \right\rfloor \le \tilde{B}'_{p'} - \tilde{B}'_p \le \left\lfloor \tilde{B}_{p'} - \tilde{B}_p \right\rfloor + 1 \right\}. \tag{26}$$

Proof of Lemma 7. Let $\tilde{B} = (\tilde{B}_1, \ldots, \tilde{B}_{|P|})$ be completely fractional, and $\tilde{B}' \in \Delta(\tilde{B})$. Write $\tilde{B}_p = z_p + x_p$, with $z_p \in \{0, 1, \ldots, B-1\}$ and $0 \le x_p < 1$. (Setting $x_p = 1$ is never required because $0 \le \tilde{B}_p < B$ since $\tilde{B}$ is completely fractional.) Note that by definition $\tilde{B}_1 = \tilde{B}'_1 = 0$, and that $\lfloor \tilde{B}_p \rfloor = z_p$. Hence, the constraints in (26) concerning $p = 1$ simplify to: $\forall p' \in P : z_{p'} \le \tilde{B}'_{p'} \le z_{p'} + 1$. In other words, these constraints require that for $p \in P$ there exists a $x'_p \in [0, 1]$ such that $\tilde{B}'_p = z_p + x'_p = \lfloor \tilde{B}_p \rfloor + x'_p$, which is equivalent to the first condition in (17).

Let $f : P \to P$ be the unique permutation of P such that $p' > p \to x_{f(p')} > x_{f(p)}$, and let f^{-1} denote its inverse. Uniqueness follows because $\forall p, p' \in P$ with $p \ne p'$ it holds that $x_p \ne x_{p'}$, since $x_p = x_{p'}$ would contradict that $\tilde{B}$ is completely fractional. Thus $f^{-1}(p') > f^{-1}(p)$ if and only if $x_{p'} > x_p$. We find:

$$\begin{aligned}
\left\lfloor \tilde{B}_{p'} - \tilde{B}_p \right\rfloor &= \lfloor z_{p'} + x_{p'} - (z_p + x_p) \rfloor \\
&= \lfloor z_{p'} - z_p + x_{p'} - x_p \rfloor \\
&= \begin{cases} (z_{p'} - z_p) & \text{if } f^{-1}(p') > f^{-1}(p) \\ (z_{p'} - z_p) - 1 & \text{if } f^{-1}(p') < f^{-1}(p). \end{cases}
\end{aligned}$$

Note that $\tilde{B}'_{p'} - \tilde{B}'_p = z_{p'} - z_p + x'_{p'} - x'_p$. Thus the condition $\lfloor \tilde{B}_{p'} - \tilde{B}_p \rfloor \le \tilde{B}'_{p'} - \tilde{B}'_p \le \lfloor \tilde{B}_{p'} - \tilde{B}_p \rfloor + 1$ for $p, p' \in P$ that is part of the definition of $\Delta(\tilde{B})$ in (26) is equivalent to the following condition on $x'_p, x'_{p'}$:

$$\begin{aligned}
0 \le x'_{p'} - x'_p \le 1 \qquad & \text{if } f^{-1}(p') > f^{-1}(p) \\
-1 \le x'_{p'} - x'_p \le 0 \qquad & \text{if } f^{-1}(p') < f^{-1}(p).
\end{aligned} \tag{27}$$

Thus, for any $i, i' \in \{1, \ldots, |P|\}$ with $i' > i$, substituting $p' = f(i')$ and $p = f(i)$ in (27) yields $x'_{f(i')} - x'_{f(i)} \ge 0$ since $f^{-1}(f(i')) > f^{-1}(f(i))$. Thus $x'_{f(1)} \le x'_{f(2)} \le \ldots \le x'_{f(|P|)}$.

Conversely, assume $0 = x'_{f(1)} \le x'_{f(2)} \le \ldots \le x'_{f(|P|)} \le 1$. Then (27) is satisfied, since $\forall p \in P : 0 \le x_p \le 1$ implies that $-1 \le x_{p'} - x_p \le 1$, and the other inequalities of (27) follow by reversing the above argument. □

Lemma 8 (Auxiliary towards Theorem 3)**.** *Take any completely fractional $\tilde{B} \in \tilde{\mathcal{B}}$ and let $\Pi^*_{\tilde{B}}$ denote its average cost optimal policy. For all $\tilde{B}' \in \Delta(\tilde{B})$, let $\hat{\Pi}_{\tilde{B}'}$ be the policy as defined in Theorem 3. Then, for each random sequence $X = (X_1, \dots, X_{2N-2})$ and for all $n \in \{1, \dots, N\}$, there exist $z, z', z'', z''' \in \mathbb{Z}, p, p', p'', p''' \in P$ such that for all $\tilde{B}' \in \Delta(\tilde{B})$ under policy $\hat{\Pi}_{\tilde{B}'}$ we have:*

$$d_n^{dep} = z + \tilde{B}'_p - \tilde{B}'_{p[n]}, \qquad \tau_n = z' + \tilde{B}'_{p'} - \tilde{B}'_p$$
$$d_{n+1}^{arr} + X_{n+1}^p = z'' + \tilde{B}'_{p''} - \tilde{B}'_{p[n+1]}, \quad \gamma_{n+1} = z''' + \tilde{B}'_{p'''} - \tilde{B}'_{p''},$$

with $z, z', z'', z''', p, p', p''$ and p''' all independent of $\tilde{B}'$, provided policy $\hat{\Pi}_{\tilde{B}'}$ is used.

Proof of Lemma 8. In a finite horizon, the sequence of random variables

$$X = (X_1, \dots, X_{2N-2}) = (X_1^s, X_2^p, X_2^s, \dots, X_{N-1}^p, X_{N-1}^s, 0)$$

yields a sequence of states and a sequence of actions:

$$(s_1, \dots, s_{2N-2}) := (d_1^{dep}, d_2^{arr} + X_2^p, d_2^{dep}, \dots, d_{N-1}^{arr} + X_{N-1}^p, d_{N-1}^{dep}, d_N^{arr}),$$

$$(a_1, \dots, a_{2N-2}) := (\tau_1, \gamma_2, \tau_2, \dots, \gamma_{N-1}, \tau_{N-1}, 0).$$

These latter sequences may depend on $\tilde{B}'$, the stationary deterministic policy $\hat{\Pi}_{\tilde{B}'}$, and the random sequence X.

By Lemma 5 and the notation following that lemma, $\forall i \in \{1, \dots, 2N-2\}$, the state in period i can be expressed as $s_i = s_{p[i],u[i]}[z_i, p'[i]; \tilde{B}']$. Since the port sequence is fixed, and arrivals and departures alternate, $p[i]$ and $u[i]$ are independent of $\tilde{B}'$ and the policy. We will show that under $\hat{\Pi}_{\tilde{B}'}$, the variables $z_i \in \mathbb{Z}$ and $p'[i] \in P$ are *independent* of $\tilde{B}'$, as long as $\tilde{B}' \in \Delta(\tilde{B})$. That means that for a fixed random sequence X, the delay in state i can be expressed as $d = z_i + \tilde{B}'_{p'[i]} - \tilde{B}'_{p[i]}$, and that $z_i, p'[i]$ and $p[i]$ are independent of $\tilde{B}'$, as long as $\tilde{B}' \in \Delta(\tilde{B})$ and as long as we use the policy $\hat{\Pi}_{\tilde{B}'}$. Note that this will imply the claims.

The proof is by induction in i. For $i = 1$, $d_1^{dep} = z \in \mathbb{Z}_{\geq 0}$ by Assumption 1. Setting

$s_1 = s_{p[1],\text{dep}}[z, p[1]; \tilde{B}']$ yields this delay for all $\tilde{B}' \in \Delta(\tilde{B})$, which implies independence for $i = 1$. The induction step will be proved separately for odd, and for even i.

First assume the statement holds for some odd i (which corresponds to a departure delay for some port call n). Let $s_i = s_{p[n],\text{dep}}[z, p; \tilde{B}']$, which corresponds to departing from port $p[n]$ with delay $d_n^{dep} = z + \tilde{B}'_p - \tilde{B}'_{p[n]}$. Assume $\Pi^*_{\tilde{B}}(s_{p[n],\text{dep}}[z, p; \tilde{B}]) = a_p[z', p'; \tilde{B}]$. Note that an action of this form must be optimal since $d_n^{dep} + \mathcal{T}_n(d_n^{dep}; \vec{B}) \in Q_n(\tilde{B})$ by Lemma 5. Thus $a_i = \hat{\Pi}_{\tilde{B}'}(s_i) = a_p[z', p'; \tilde{B}']$ is the action taken under $\hat{\Pi}_{\tilde{B}'}$. Note that z' and p' are independent of $\tilde{B}'$ by induction hypothesis, and by construction of $\hat{\Pi}_{\tilde{B}'}$. Write $X_n^s = z^s$ and $X_{n+1}^p = z^p$, where $z^s, z^p \in \mathbb{Z}_{\geq 0}$ by Assumption 1. Note that a_i denotes $\tau_n = z' + \tilde{B}'_{p'} - \tilde{B}'_p$. Thus action a_i in state s_i yields $d_n^{dep} + \tau_n = (z + z') + \tilde{B}'_{p'} - \tilde{B}'_{p[n]}$. By definition of the cumulative buffers $\tilde{B}'$ it holds that $\tilde{B}'_{p[n]} + \vec{B}'_{p[n]} = \tilde{B}'_{p[n+1]} + z_n$, with $z_n := B \in \mathbb{Z}$ if $p[n+1] = 1$ and $z_n = 0$ otherwise. Thus, $d_n^{dep} + \tau_n - \vec{B}'_{p[n]} + X_n^s = (z + z' + z^s) + \tilde{B}'_{p'} - (\tilde{B}'_{p[n]} + \vec{B}'_{p[n]}) = (z + z' + z^s - z_n) + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]} = \tilde{z} + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]}$, for some $\tilde{z} \in \mathbb{Z}$. Thus $d_{n+1}^{arr} = (\tilde{z} + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]})^+$.

We now distinguish two cases: 1) $\tilde{z} + \tilde{B}_{p'} - \tilde{B}_{p[n+1]} \leq 0$ and 2) $\tilde{z} + \tilde{B}_{p'} - \tilde{B}_{p[n+1]} > 0$. For Case 1, we have $\tilde{B}_{p[n+1]} - \tilde{B}_{p'} \geq \tilde{z}$, which implies that $\left\lfloor \tilde{B}_{p[n+1]} - \tilde{B}_{p'} \right\rfloor \geq \tilde{z}$, and thus, by Lemma 7, that $\tilde{B}'_{p[n+1]} - \tilde{B}'_{p'} \geq \tilde{z}$. Hence, $d_{n+1}^{arr} + X_{n+1}^p = (\tilde{z} + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]})^+ + z^p = z^p = z^p + \tilde{B}'_{p[n+1]} - \tilde{B}'_{p[n+1]}$ which implies $s_{i+1} = s_{p[n+1],\text{port}}[z'', p''; \tilde{B}']$ with $z'' = z^p$ and $p'' = p[n+1]$. For Case 2, we have $\tilde{B}_{p[n+1]} - \tilde{B}_{p'} < \tilde{z}$, and thus $\left\lceil \tilde{B}_{p[n+1]} - \tilde{B}_{p'} \right\rceil \leq \tilde{z}$ which implies (Lemma 7) that $\tilde{B}'_{p[n+1]} - \tilde{B}'_{p'} \leq \tilde{z}$. Thus $d_{n+1}^{arr} + X_{n+1}^p = \tilde{z} + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]} + z^p = (\tilde{z} + z^p) + \tilde{B}'_{p'} - \tilde{B}'_{p[n+1]}$, implying $s_{i+1} = s_{p[n+1],\text{port}}[z'', p''; \tilde{B}']$ with $z'' = \tilde{z} + z^p$ and $p'' = p'$. Note that $\tilde{z} = z + z' + z^s + z_n$ is independent of $\tilde{B}'$ by induction hypothesis, and thus case checking is independent of $\tilde{B}'$. (It is thus essential that $\tilde{B}$ can be used for case checking.) Hence, z'' and p'' are independent of $\tilde{B}'$, which proves the result for $i + 1$.

Now assume the result holds for some even i' (which corresponds to a port delay for some port call n). Let $s_{i'} = s_{p[n],\text{port}}[z'', p''; \tilde{B}']$, which corresponds to being in port $p[n]$ with delay $d_n^{arr} + X_n^p = z'' + \tilde{B}'_{p''} - \tilde{B}'_{p[n]}$. Assume $\Pi^*_{\tilde{B}}(s_{p[n],\text{port}}[z'', p''; \tilde{B}]) = a_{p''}[z''', p'''; \tilde{B}]$. Note that an action of this form must be optimal by Lemma 5. Let $a_{i'} = \hat{\Pi}_{\tilde{B}'}(s'_i) = a_{p''}[z''', p'''; \tilde{B}']$, and note that z''' and p''' are independent of $\tilde{B}'$ by induction hypothesis. We find $d_n^{dep} = (d_n^{arr} + X_n^p - \gamma_n)^+$. We have $d_n^{arr} + X_n^p - \gamma_n = (z'' + \tilde{B}'_{p''} - \tilde{B}'_{p[n]}) + (z''' + \tilde{B}'_{p'''} - \tilde{B}'_{p''}) = \tilde{z}' + \tilde{B}'_{p'''} - \tilde{B}'_{p[n]}$, where $\tilde{z}' = z'' + z''' \in \mathbb{Z}$.

Distinguish between two cases: 1) $\tilde{z}' + \tilde{B}_{p'''} - \tilde{B}_{p[n]} \leq 0$ and 2) $\tilde{z}' + \tilde{B}_{p'''} - \tilde{B}_{p[n]} > 0$. For the first case, we have $\tilde{B}_{p[n]} - \tilde{B}_{p'''} \geq \tilde{z}'$, which implies that $\left\lfloor \tilde{B}_{p[n]} - \tilde{B}_{p'''} \right\rfloor \geq \tilde{z}'$, and thus, by Lemma 7, that $\tilde{B}'_{p[n]} - \tilde{B}'_{p'''} \geq \tilde{z}'$. Hence, $d_n^{dep} = (d_n^{arr} + X_n^p - \gamma_n)^+ = 0 = \tilde{B}'_{p[n]} - \tilde{B}'_{p[n]}$ which implies $s_{i'+1} = s_{p[n],\text{dep}}[z'''', p''''; \tilde{B}']$ with $z'''' = 0$ and $p'''' = p[n]$. For Case 2, we have $\tilde{B}_{p[n]} - \tilde{B}_{p'''} < \tilde{z}'$, and thus $\left\lceil \tilde{B}_{p[n]} - \tilde{B}_{p'''} \right\rceil \leq \tilde{z}'$ which implies (Lemma 7) that $\tilde{B}'_{p[n]} - \tilde{B}'_{p'''} \leq \tilde{z}'$. Thus $d_n^{dep} = (d_n^{arr} + X_n^p - \gamma_n)^+ = \tilde{z}' + \tilde{B}'_{p'''} - \tilde{B}'_{p[n]}$, implying $s_{i'+1} = s_{p[n],\text{dep}}[z'''', p''''; \tilde{B}']$ with $z'''' = \tilde{z}'$ and $p'''' = p'''$. Note that $\tilde{z}'$ is independent of $\tilde{B}'$, thus so is the case checking. Thus p'''' and z'''' are independent of $\tilde{B}'$, which proves the statement for $i' + 1$. This completes the proof by induction.

□

Lemma 9 (Auxiliary towards Theorem 3)**.** *Take any completely fractional $\tilde{B} \in \tilde{\mathcal{B}}$ and let $\Pi^*_{\tilde{B}}$ denote its average cost optimal policy. For all $\tilde{B}' \in \Delta(\tilde{B})$, let $\hat{\Pi}_{\tilde{B}'}$ be the policy as defined in Theorem 3. Then, $\hat{\mathcal{C}}(\tilde{B}')$ is affine in $\tilde{B}'$.*

Proof of Lemma 9. Let $X = (X_1, \ldots, X_{2N-2}) = (X_1^s, X_2^p, X_2^s, \ldots, X_{N-1}^p, X_{N-1}^s, 0)$ be a sequence of random variables. It follows from Lemma 8 that $d_n^{dep} = z + \tilde{B}'_p - \tilde{B}'_{p[n]}$ with $z \in \mathbb{Z}$ and $p \in P$ both independent of $\tilde{B}'$, provided policy $\hat{\Pi}_{\tilde{B}'}$ is used. Hence, $\mathcal{D}^{dep}_{p[n]}(d_n^{dep})$ is affine in $\tilde{B}'$ for the sequence X under $\hat{\Pi}_{\tilde{B}'}$. Indeed, $\left\lfloor \tilde{B}_p - \tilde{B}_{p[n]} \right\rfloor \leq \tilde{B}'_p - \tilde{B}'_{p[n]} \leq \left\lfloor \tilde{B}_p - \tilde{B}_{p[n]} \right\rfloor + 1$ for all $\tilde{B}' \in \Delta(\tilde{B})$ by Lemma 7. Since the delay costs $\mathcal{D}^{dep}_{p[n]}(d_n^{dep})$ are piecewise linear with breakpoints at $\mathbb{Z}_{\geq 0}$ by Assumption 1, there exist constants c_1 and c_2 such that $\mathcal{D}^{dep}_{p[n]}(z + \tilde{B}'_p - \tilde{B}'_{p[n]}) = c_1 + c_2(z + \tilde{B}'_p - \tilde{B}'_{p[n]})$ for all $\tilde{B}' \in \Delta(\tilde{B})$, which is affine in $\tilde{B}'$. For the sailing costs $\mathcal{F}_{p[n]}(\tau_n)$, we again have $\left\lfloor \tilde{B}_{p'} - \tilde{B}_p \right\rfloor \leq \tilde{B}'_{p'} - \tilde{B}'_p \leq \left\lfloor \tilde{B}_{p'} - \tilde{B}_p \right\rfloor + 1$ by Lemma 7. Thus, since $\tau_n = z' + \tilde{B}_{p'} - \tilde{B}_p$ is feasible for $\tilde{B}$, $\tau'_n = z' + \tilde{B}'_{p'} - \tilde{B}'_p$ is feasible for $\tilde{B}'$. We know from Lemma 8 that $z' \in \mathbb{Z}$ and $p, p' \in P$ are all independent of $\tilde{B}'$ for all $\tilde{B}' \in \Delta(\tilde{B})$. Since $\mathcal{F}_{p[n]}(\tau_n)$ is a piecewise linear function with breakpoints at $\mathbb{Z}_{\geq 0}$ by Assumption 1, $\mathcal{F}_{p[n]}(z' + \tilde{B}'_{p'} - \tilde{B}'_p)$ is affine in $\tilde{B}'$ for all $\tilde{B}' \in \Delta(\tilde{B})$. In a very similar fashion it can be shown that, for the sequence X under $\hat{\Pi}_{\tilde{B}'}$, the arrival delay $\mathcal{D}^{arr}_{p[n]}(d_n^{arr})$, and the costs of the extreme action $c^e \gamma_n$, are affine in $\tilde{B}'$. (Note that since by Lemma 8 we can write $d_{n+1}^{arr} + X_{n+1}^p = z'' + \tilde{B}'_{p''} - \tilde{B}'_{p[n+1]}$, and since $X_{n+1}^p \in \mathbb{Z}$ by Assumption 1, we can also write $d_{n+1}^{arr} = \tilde{z} + \tilde{B}'_{p''} - \tilde{B}'_{p[n+1]}$.) Because this holds for all n, the total costs incurred over the periods $\{1, \ldots, N = R|P| + 1\}$ for the sequence X under $\hat{\Pi}_{\tilde{B}'}$ are affine in $\tilde{B}' \in \Delta(\tilde{B})$.

The total *expected* costs under $\hat{\Pi}_{\tilde{B}'}$ over the periods $\{1, \ldots, N = R|P| + 1\}$ are the expectation of the costs for each sequence over all sequences, and they are affine in $\tilde{B}'$ because taking a linear combination over affine functions yields an affine function. The average expected costs $\hat{\mathcal{C}}(\tilde{B}')$ under $\hat{\Pi}_{\tilde{B}'}$ are obtained by dividing the total costs incurred over $\{1, \ldots, N = R|P|+1\}$ by R, taking the limit $R \to \infty$. This limit exists for all $\tilde{B}'$ since it corresponds to the average costs of a stationary policy in a finite-state Markov Process (see the proof of Theorem 2). Moreover, $\hat{\Pi}_{\tilde{B}'}$ is affine in $\tilde{B}'$ for $\tilde{B}' \in \tilde{\mathcal{B}}$, since the limit of functions that are affine in $\tilde{B}'$ is affine in $\tilde{B}'$, provided the limit exists for each $\tilde{B}'$. □

Proof of Theorem 3. Lemma 9 establishes the existence of g_0 and $g = (g_1, \ldots, g_{|P|})$ such that $\forall \tilde{B}' \in \Delta(\tilde{B}) : \hat{\mathcal{C}}(\tilde{B}') = g_0 + \sum_{p \in P} g_p \tilde{B}'_p$.

We now show that g must be a subgradient at $\tilde{B}$. For some arbitrary $\tilde{B}'' \in \tilde{\mathcal{B}}$, let $\tilde{B}(x) = \tilde{B} + x(\tilde{B}'' - \tilde{B})$. We have $\mathcal{C}^*(\tilde{B}(0)) = \mathcal{C}^*(\tilde{B})$, and $\mathcal{C}^*(\tilde{B}) = \hat{\mathcal{C}}(\tilde{B}) = g_0 + \sum_{p \in P} g_p \tilde{B}_p$ because $\hat{\Pi}_{\tilde{B}}$ is optimal for $\tilde{B}$ by construction. Because $\tilde{B}$ is completely fractional, there is some $\epsilon > 0$ such that $\tilde{B}(-\epsilon) = \tilde{B} - \epsilon(\tilde{B}'' - \tilde{B}) \in \Delta(\tilde{B})$. Thus $\hat{\Pi}_{\tilde{B}(-\epsilon)}$ is feasible for $\tilde{B}(-\epsilon)$, and we obtain $\mathcal{C}^*(\tilde{B}(-\epsilon)) \leq \hat{\mathcal{C}}(\tilde{B}(-\epsilon)) = g_0 + \sum_{p \in P} g_p \tilde{B}(-\epsilon)_p = g_0 + \sum_{p \in P} g_p [\tilde{B}_p - \epsilon(\tilde{B}''_p - \tilde{B}_p)] = \mathcal{C}^*(\tilde{B}) - \epsilon \sum_{p \in P} g_p (\tilde{B}''_p - \tilde{B}_p)$. Now, by Theorem 1, $\mathcal{C}^*(\vec{B})$ is convex in $\vec{B} \in \mathcal{B}$, and since $\tilde{B}$ is obtained by an affine transformation of $\vec{B}$, $\mathcal{C}^*(\tilde{B})$ is convex in $\tilde{B} \in \tilde{\mathcal{B}}$. Thus $\mathcal{C}^*(\tilde{B}(x))$ is convex in x, which implies that $(1 + \epsilon)\mathcal{C}^*(\tilde{B}(0)) \leq \mathcal{C}^*(\tilde{B}(-\epsilon)) + \epsilon\mathcal{C}^*(\tilde{B}(1))$. Thus $\epsilon\mathcal{C}^*(\tilde{B}'') = \epsilon\mathcal{C}^*(\tilde{B}(1)) \geq (1 + \epsilon)\mathcal{C}^*(\tilde{B}(0)) - \mathcal{C}^*(\tilde{B}(-\epsilon)) \geq (1 + \epsilon)\mathcal{C}^*(\tilde{B}) - \left[\mathcal{C}^*(\tilde{B}) - \epsilon \sum_{p \in P} g_p (\tilde{B}''_p - \tilde{B}_p)\right] = \epsilon \left[\mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p (\tilde{B}''_p - \tilde{B}_p)\right]$. Thus we find $\forall \tilde{B}'' \in \tilde{\mathcal{B}} : \mathcal{C}^*(\tilde{B}'') \geq \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p (\tilde{B}''_p - \tilde{B}_p)$, which is precisely the subgradient inequality. Thus g is a subgradient at $\tilde{B}$.

Additionally, for any $\tilde{B}' \in \Delta(\tilde{B})$, we know that $\mathcal{C}^*(\tilde{B}') \leq \hat{\mathcal{C}}(\tilde{B}')$ since $\hat{\Pi}_{\tilde{B}'}$ is a feasible policy for $\tilde{B}'$. But $\hat{\mathcal{C}}(\tilde{B}') = g_0 + \sum_{p \in P} g_p \tilde{B}'_p = \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p (\tilde{B}'_p - \tilde{B}_p) \leq \mathcal{C}^*(\tilde{B}')$, where the second equality follows because $\mathcal{C}^*(\tilde{B}) = g_0 + \sum_{p \in P} g_p \tilde{B}_p$, and the inequality is the subgradient inequality at $\tilde{B}$ that we just proved. Combining these inequalities yields $\forall \tilde{B}' \in \Delta(\tilde{B}) : \mathcal{C}^*(\tilde{B}') = \hat{\mathcal{C}}(\tilde{B}')$, completing the proof. □

Proof of Corollary 1. Let $\tilde{B}$ be completely fractional and let g denote the subgradient from Theorem 3. For arbitrary $\tilde{B}'' \in \tilde{\mathcal{B}}$, we obtain the subgradient inequality at $\tilde{B}' \in \Delta(\tilde{B})$:

$$\mathcal{C}^*(\tilde{B}'') \geq \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p(\tilde{B}''_p - \tilde{B}_p) = \mathcal{C}^*(\tilde{B}') + \sum_{p \in P} g_p(\tilde{B}''_p - \tilde{B}'_p)$$

Here, the inequality holds because g is a subgradient at $\tilde{B}$, and the equality because $\mathcal{C}^*(\tilde{B}') = \hat{\mathcal{C}}(\tilde{B}') = \mathcal{C}^*(\tilde{B}) + \sum_{p \in P} g_p(\tilde{B}'_p - \tilde{B}_p)$ for $\tilde{B}' \in \Delta(\tilde{B})$ by Theorem 3. But the subgradient inequality for g at $\tilde{B}'$ shows that g is a subgradient at $\tilde{B}'$. □

Proof of Theorem 4. The extreme points induced by $0 \leq x_{f(1)} \leq x_{f(2)} \leq \ldots \leq x_{f(|P|)} \leq 1$ are the extreme points of $\Delta(\tilde{B})$. It is easy to verify that the extreme points are exactly given by $\tilde{B}^j$ for $j \in \{1, \ldots, |P|\}$. Namely, each extreme point will have $x_p = 0$ or $x_p = 1$ for $1 \leq p \leq |P|$. Furthermore, the ordering induced by the bijection f ensures that $x_p = 1$ can only be valid if $x_{p'} = 1$ for all p' such that $f(p') > f(p)$. Since $0 \leq x_{f(1)} \leq x_{f(2)} \leq \ldots \leq x_{f(|P|)} \leq 1$ only contains linear inequalities, the feasible region is a polyhedron. This polyhedron is clearly bounded. Combining this with the fact that the extreme points of the polyhedron are $\tilde{B}^j$, $j \in \{1, \ldots, |P|\}$, the convex hull of $\{\tilde{B}^j | j \in \{1, \ldots, |P|\}$ must equal $\Delta(\tilde{B})$. □