Inland waterway efficiency through skipper collaboration and joint speed optimization

European Journal of Operational Research 292 (2021) 276–285

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Innovative Applications of O.R.

Inland waterway efficiency through skipper collaboration and joint speed optimization☆

Christof Defryn[a], Julian Arthur Pawel Golak[a,*], Alexander Grigoriev[a], Veerle Timmermans[b]

[a] Maastricht University School of Business and Economics, P.O.Box 616, Maastricht 6200 MD, the Netherlands
[b] RWTH Aachen, Department of Management Science, Kackertstrasse 7, Aachen 52072, Germany

ARTICLE INFO

Article history:
Received 24 May 2019
Accepted 12 October 2020
Available online 24 October 2020

Keywords:
Scheduling
Congestion
Social welfare
Mechanism design
Online scheduling

ABSTRACT

We address the problem of minimizing the aggregated fuel consumption by the vessels in an inland waterway, e.g., a river, with a single lock. The fuel consumption of a vessel depends on its velocity and the slower it moves, the less fuel it consumes. Given entry times of the vessels into the waterway and the deadlines before which they need to leave the waterway, we start from the optimal velocities of the vessels that minimize their private fuel consumption, where we assume selfish behavior of the skippers. Presence of the lock and possible congestion on the waterway make the problem computationally challenging. First, we prove that in general, a Nash equilibrium might not exist, i.e., if there is no supervision on the vessels' velocities, there might not exist a strategy profile from which no vessel can unilaterally deviate to decrease its private fuel consumption. Next, we introduce simple supervision methods to guarantee the existence of a Nash equilibrium. Unfortunately, though a Nash equilibrium can be computed, the aggregated fuel consumption of such a stable solution can be high compared to the social optimum, where the total fuel consumption is minimized. Therefore, we propose a mechanism involving payments between vessels, guaranteeing a Nash equilibrium while minimizing the fuel consumption. This mechanism is studied for both the offline setting, where all information is known beforehand, and online setting, where we only know the entry time and deadline of a vessel when it enters the waterway.

1. Introduction

The high fuel prices, a congested road network and the increasing demand for transport due to globalization put a high pressure on the existing transportation network, especially road transport. The growing sense of resource scarcity and climate change motivates companies to rethink their logistical operations and, if possible, shift towards a more sustainable transport mode. In comparison to other transportation modes, the use of waterways is more sustainable (less greenhouse gas emission) and relatively cheap (due to economies of scale). Moreover, as a single vessel can replace over 100 trucks, increased use of the water network is likely to reduce congestion and the number of accidents on the road network. The Netherlands, located around the mouth of multiple important European rivers, has a dense network of over 4600km of navigable inland waterways, on which 36% of all freight transport (in tonne-kilometre) takes place (European Commission, 2017; Eurostat, 2019).

Besides longer travel times, mainly due to the relatively low density of the network, the high uncertainty in arrival time is one the major drawbacks of freight transport over inland waterways. This uncertainty is caused by the presence of many river obstacles, such as low bridges, narrow river segments, harbors and locks, which gives rise to unexpected congestion and waiting time[1]. This forces the skipper, the person in charge of the vessel, to increase the speed afterwards to guarantee an on-time arrival at the destination. However, the operational cost for the skipper is largely determined by the fuel consumption, which is related directly to the required power and, therefore, the speed of the vessel. The required speeding up results therefore in a direct increase of operational costs.

In this paper, we investigate how coordination and scheduling of all movements around these river obstacles can help to reduce

☆ A preliminary version of this work without essential models and proofs has been presented at the International conference on Mathematical Optimization Theory and Operations Research (MOTOR 2019), see Defryn, Golak, Grigoriev, and Timmermans (2019).

* Corresponding author.
E-mail address: j.golak@maastrichtuniversity.nl (J.A.P. Golak).

[1] Communicated to us by our industrial partner Trapps Wise. B.V. (https://trappswise.nl/).

https://doi.org/10.1016/j.ejor.2020.10.017
0377-2217/

congestion and waiting times, and therefore increase the efficiency of inland waterway transport. Moreover, by optimizing a recommended speed for each vessel between two consecutive obstacles, one can control the arrival times of the vessels at each obstacle, guaranteeing the minimal throughput time and at the same time the minimal total fuel consumption. Even for a single lock, the strategy of reducing the speed of the vessel to avoid waiting time has resulted in significant economic benefits (Ting & Schonfeld (1999)).

2. Literature review

2.1. Lock scheduling

Existing research on the optimization of river obstacles is mainly focused on lock scheduling. In a single lock scheduling problem, the operating times of a single lock are optimized for a set of vessels with given arrival times at the lock. By batching the vessels together and determining the optimal service time for each batch, the goal is to reduce overall waiting time at the lock.

Passchyn et al. (2016b) provide a polynomial time algorithm to optimally solve the single lock scheduling problem, given the arrival times of the vessels and the capacity of the lock. A complexity analysis together with a polynomial time algorithm that applies to special cases for the single lock scheduling problem with multiple parallel chambers is presented in Passchyn, Briskorn, and Spieksma (2019). The problem of physically placing vessels inside the chamber of the lock has been addressed in Verstichel, De Causmaecker, Spieksma, and Berghe (2014a) and Verstichel, De Causmaecker, Spieksma, and Berghe (2014b). The joint optimization of multiple sequential locks on the river is considered by Passchyn, Briskorn, and Spieksma (2016a) and Prandtstetter, Ritzinger, Schmidt, and Ruthmair (2015). Prandtstetter et al. (2015) propose a variable neighborhood search, whereas Passchyn et al. (2016a) use a MILP to find an exact solution. In the latter two contributions, the goal is to minimize the aggregated fuel cost or emission, while selfish behavior of skippers is not addressed.

There are also multiple case studies conducted for the lock scheduling problem, focused on specific lock sequences on important waterways in the world. Petersen and Taylor (1988) consider the *Welland Canal* in North America for which they provide a heuristic that employs optimal dynamic programming models for scheduling individual locks in order to determine operating schedules for the lock sequence. Smith, Sweeney, and Campbell (2009) present a simulation model to evaluate the quality of different heuristics on lock operations on the *Upper Mississippi River* in the US. This research has been extended by Smith et al. (2011). Here, the authors propose a MILP model to solve the lock scheduling problem with sequence-dependent setup and processing times. Using the same river segment, Nauss (2008) incorporate the malfunctioning of locks and study different responses to such a disruption so to minimize additional queue lengths. Also, a model for the lock scheduling problem with multiple parallel chambers for this river layout has been investigated by Ting and Schonfeld (2001). Finally, the *Kiel Canal* is considered by Lübbecke, Lübbecke, and Möhring (2019). In their paper, the authors incorporate collision of ships in the optimization model and provide a heuristic to determine a routing and scheduling for a fleet of ships in a collision-free manner.

To the best of our knowledge, only Passchyn et al. (2016a) take into account that skippers can choose the speed of their vessel, and hence influence the time at which they arrive at the lock. Their objective is to minimize overall CO_2 emissions by optimizing the speed at which vessels have to approach the locks using a MILP formulation. Although this approach is closely related to the problem addressed in this paper, Passchyn et al. (2016a) focus on minimizing the aggregated emissions without considering the fact that each skipper is mainly interested in minimizing his personal fuel cost and emissions (selfish behavior). As a consequence, skippers might deviate from the proposed solution and increase their individual utility. In this paper, we view this problem from a game-theoretic point of view, and propose a schedule in which no skipper can profitably deviate from the proposed solution.

2.2. Fuel reduction

Academic literature on fuel savings has been extensive in the context of ocean vessels. We refer to Psaraftis and Kontovas (2013) for a more detailed survey. Research on fuel consumption in inland waterways, on the other hand, is sparse. Inland waterways are significantly different compared to the ocean, as there are no ‘river’ obstacles in the ocean. Ting and Schonfeld (1999) found that the strategy of reducing vessels speed to avoid idle time has resulted in significant economic benefits for a single lock. This may been seen as a key observation for the motivation of the current work. The fact that fuel consumption grows non-linearly in the vehicle’s speed is corroborated by Bialystocki and Konovessis (2016).

2.3. Our contributions

Previous research on lock scheduling is based on the assumption that lock operators have the full power to determine the operating schedule for the lock and operate under full information. In practice, this schedule is typically determined using the first come first serve (FIFO) principle based on the order at which vessels arrive at the lock. Skippers that know this have the incentive to speed up when approaching a lock in order to pass their predecessors and get served first. This action leads to overall longer waiting times before the locks, and increases the operational cost for these skippers due to the higher fuel consumption that is caused by maintaining a higher speed.

In this paper, we aim to minimize the aggregated fuel consumption by the vessels in the river, while keeping in mind that each skipper is a rational individual with the sole goal of minimizing his personal fuel cost or emissions. Our goal is to determine an optimal speed for each individual vessel and for each river segment. The positive relation between vessel speed and fuel consumption leads to the observation that maintaining the slowest speed — yet meeting the arrival deadline at the destination harbour — minimizes the total fuel consumption of a single vessel. Unfortunately, even a single lock on the river becomes a source of congestion and the speeds of the vessels have to be adjusted accordingly.

The paper is structured as follows. In Section 3, we model the problem as a non-cooperative game and discuss a variety of priority rules that can be used by the lock operators in case multiple vessels approach the lock (possibly in the opposite directions). Moreover, we discuss the existence of *Nash equilibria* — situations in which no skipper can unilaterally deviate from the proposed solution and decrease its individual cost. In Section 4, we introduce a *cooperative game* perspective on the traffic optimization problem at hand. We assume that binding contracts between different skippers are possible and propose a mechanism based on monetary payments. This situation will give rise to new Nash equilibria. We design an algorithm that computes these Nash equilibria while minimizing total fuel consumption on the river. A MILP formulation to solve the lock scheduling problem is included in Section 5. Finally, in Section 6, we extend our algorithm to comply with an online setting.

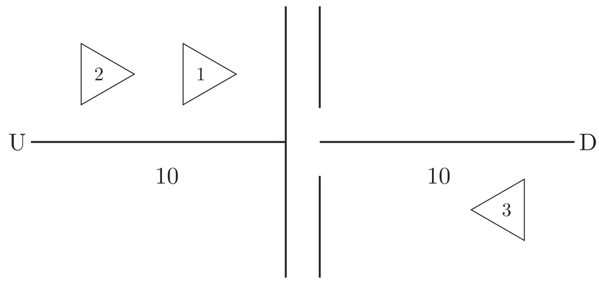

Fig. 1. The setup of locks and vessel for Example 1.

3. Non-cooperative game for traffic optimization at river obstacles

3.1. Mathematical notation of the system

Without loss of generality, we assume a waterway with a single lock. Let this lock be defined by its capacity C, i.e., the number of vessels that can be leveled up or down simultaneously, and its current state P, indicating whether the level of the water is high (equal to the upstream level) or low (equal to the downstream level). Let T be the time to change the lock state from high to low or vice versa. If a batch of vessels is processed, an additional T_i times units are required for each vessel i in the batch. That time represents the loading and unloading of vessels and varies across different types and sizes of vessels. The processing time of a batch of vessels is the sum of the lockage duration T and the individual processing times T_i for every vessel i in the batch. Moreover, let L_u and L_d be the distances between the upstream and downstream end points of the waterway respectively and the lock. From the moment that a vessel is within that distance from the lock, we consider it to be in the system. The complete system is, therefore, determined by the tuple $L = \{C, P, T, (T_i)_{i\in S}, L_u, L_d\}$.

Now, let U and D be sets of vessels that sail upstream or downstream respectively, and let $S = U \cup D$ be the set of all vessels. The size of the entire fleet is denoted by $n = |S|$. For each vessel $i \in U$, we are given an arrival time at the downstream end point of the river, denoted by a_i, and a deadline d_i, the latest time when the vessel has to reach the downstream end point of the waterway. Similarly, a_j and d_j are defined for each vessel $j \in D$, sailing in the opposite direction. Furthermore, we assume that vessels in set S are ordered according to their arrival times and that between any two sequential vessel arrivals at least $\varepsilon > 0$ time elapses. Finally, let $v_{i,p}$ denote the speed of vessel i along river segment $p \in \{u, d\}$, where u and d represent the upstream and downstream segments respectively. We assume the minimum and the maximum speed for any vessel is bounded by $v_{\min}$ and $v_{\max}$.

3.2. Model definition

In the game, each vessel $i \in S$ decides on $v_{i,d}$ and without loss of generality, we assume that all ships have equivalent speed limits, i.e. $v_{i,u} \in [v_{\min}, v_{\max}]$. Furthermore, define $v_i = (v_{i,d}, v_{i,u})$. Furthermore, let v_{-i} denote the strategy profile of every player in the game except for i and let $\mathbf{v}_S = (v_i)_{i\in S}$. Note that only constant speeds have been specified for both, upstream and downstream, waterway segments. Due to the convexity of the cost function defined below, skippers will have no incentive to alter their speed midway of the segments. The assumption of constant speeds is relaxed, when an online setting of the game is considered, in Section 6. To illustrate the game, consider the following example.

Example 3.1. Assume three vessels (see also Fig. 1): 1 and 2 sailing upstream and 3 sailing downstream. The waterway is 20 kilometers long, and the lock is placed in the middle of the waterway. As a result, $L_u = L_d = 10$. The entry/arrival times of the vessels are as follows: $a_1 = 0$, $a_2 = \varepsilon$ and $a_3 = 2\varepsilon$. Moreover, we know that $(v_{1,u}, v_{1,d}) = (5, 5)$, $(v_{2,u}, v_{2,d}) = (10, 5)$ and $(v_{3,u}, v_{3,d}) = (5, 10)$. Given the current speeds, vessel 1 arrives at the lock at time 2, vessel 2 at time $1 + \varepsilon$ and vessel 3 is expected to arrive at the lock at time $1 + 2\varepsilon$.

The total fuel consumption is given by the function $F(v)$, where v represents the speed of the vessel. The function is measured in tons per kilometer. We assume that fuel consumption is equal to zero if the vessel is not moving, i.e., its speed is equal to zero, and vessels are only standing still inside the lock. Following the conventions from the related literature, we assume convexity of $F(v)$, $v > 0$ (see Passchyn et al., 2016a).

To further simplify notations, and without loss of generalization, we consider the fuel consumption function to be the same for every vessel and equal to

$$F_i(v_i) = L_u F(v_{i,u}) + L_d F(v_{i,d}). \tag{1}$$

The fuel consumption of vessels in set S can therefore be written as

$$F_{tot}(\mathbf{v}_S) = \sum_{i\in S} F_i(v_i). \tag{2}$$

Each skipper i aims to minimize its total fuel consumption $F_i(v_i)$, given its deadline (denoted as d_i) on the arrival time at the destination. This deadline is considered a hard constraint. Arriving at the destination after the predefined deadline is considered infeasible, represented by an infinite penalty cost. In case the deadline is unrestrictive for the vessel, it will sail at the minimum speed $v_{\min}$. Therefore, we define the cost function for skipper $i \in S$ by

$$C_i(\mathbf{v}_S) = \begin{cases} F_i(v_i) & \text{if } a_i + L_u/v_{i,u} + L_d/v_{i,d} + q_i(\mathbf{v}_S) \le d_i; \\ \infty & \text{otherwise,} \end{cases} \tag{3}$$

where $q_i(\mathbf{v}_S)$ is the total processing time of vessel i at the lock, i.e., waiting time before entering the lock plus the lock re-level time T and the individual loading times. This waiting time depends on the congestion induced by the strategy profile, i.e., individual speeds of vessels in set S.

We now define the *social cost* $C(\mathbf{v}_S)$ of a strategy profile $\mathbf{v}_S$ as the aggregated cost of all players in S, defined as

$$C(\mathbf{v}_S) = \sum_{i\in S} C_i(\mathbf{v}_S). \tag{4}$$

The strategy profile $\mathbf{v}_S$ that minimizes the social cost is called the *social optimum*, and has a social cost of

$$C_{opt} = \min_{\mathbf{v}_S} C(\mathbf{v}_S). \tag{5}$$

3.3. Nash equilibrium and queuing discipline at the lock

In a *non-cooperative* game (without binding contracts between the skippers), we assume that skippers act selfishly and aim to minimize their individual costs. One of the most important tools that game theorists have at their disposal is the *Nash equilibrium*: a strategy profile v_S^* where no vessel can unilaterally deviate from its current strategy v_i^* and decrease its current cost. More formally, v_S^* is a Nash equilibrium if and only if

$$C_i(v_i^*, v_{-i}^*) \le C_i(v_i, v_{-i}^*), \forall v_i \in V_i. \tag{6}$$

The importance of the Nash equilibrium comes from the natural observation that agents/players/skippers are rather interested in selfishly minimizing their individual costs than reducing the social cost, i.e., the total cost of the entire fleet. The Nash equilibrium is calculated by minimizing the regret of the individual players, where regret is defined as the cost they could have saved by altering the strategy.

The existence of the Nash Equilibrium depends on the waiting time of vessels in front of the locks. In turn, this waiting time is subject to the *queuing discipline* of the lock. This queuing discipline dictates the order in which vessels are served by the lock operator. As the waiting time impacts the optimal (required) speed after the lock, the queuing discipline directly affects the cost of each skipper. Therefore, different lock mechanisms yield different characteristics of the game. We consider the following three simple lock mechanisms:

Mechanism 1: Lock FIFO. For any $i, j \in U \cup D$, vessel i is served by the lock before vessel j if i arrives at the lock before j. If vessels i and j arrive at the lock at the same time, i will be served first if $a_i < a_j$.

Mechanism 2: System FIFO. For any $i, j \in U \cup D$, vessel i is served by the lock before vessel j if $a_i < a_j$.

Mechanism 3: System FIFO with filling idle time. Consider vessel $i \in U \cup D$. Assume that skippers choose strategies sequentially and all $(v_j)_{j=1,\dots,i-1}$ are given. For any $i, j \in U \cup D$ such that $j < i$, vessel i is served before j if it does not affect the time of departure of vessel j determined by the strategy profile $(v_j)_{j=1,\dots,i-1}$. Thus, given the lock schedule for earlier arriving ships, vessel i can either join a non-full lockage in schedule or join an empty lockage, as long as this does not affect the existing schedule.

The following example illustrates how these three mechanisms work and how they affect the payoff of a strategy profile.

Example 3.2. Consider again the setup of Example 3.1. Let us remind that the entry/arrival times of the vessels were $a_1 = 0$, $a_2 = \varepsilon$ and $a_3 = 2\varepsilon$. The lock has an infinite capacity and $T = T_1 = T_2 = T_3 = 0.5$.. Given the current speeds of the vessels, the arrival times at the locks are 2, $1 + \varepsilon$ and $1 + 2\varepsilon$, for vessels 1, 2 and 3, respectively.

First, if the lock operates under Mechanism 1, only the arrival times at the lock are relevant. Note that vessel 2 arrives at the lock first, vessel 3 second and vessel 1 is the last one. As vessels are processed in order of arrival time, the waiting times under the strategy profile are $2 + \varepsilon, 1, 2 - \varepsilon$ for vessel 1, 2 and 3 respectively.

Second, under mechanism 2, only the arrival times into the system are relevant. Note that vessel 1 arrives first in the system, vessel 2 second and vessel 3 last. The waiting times are $1, 2 - \varepsilon, 3 - 2\varepsilon$ for vessel 1, 2 and 3 respectively.

Lastly, when Mechanism 3 is applied, the arrival times into the system and at the locks are relevant. Note that if vessel 2 or 3 is served before vessel 1, the exit from the lock of vessel 1 would be delayed. Since vessel 1 arrives first into the system, it has priority and therefore it is processed first. Once vessel 1 is processed, the lock is open to the downstream side and vessels 2 and 3 are waiting on the upstream and downstream segments, respectively. Vessel 2 arrives first into the system, therefore it has priority. When vessel 1 has been processed, the lock is on the side of vessel 3. However, serving vessel 3 would increase the waiting time of vessel 2 by 0.5. Therefore, under this mechanism, vessel 2 is processed second and vessel 3 is processed last. The waiting times are $1, 2 - \varepsilon, 3 - 2\varepsilon$ for vessel 1, 2 and 3 respectively.

Since the choice of a lock mechanism influences the behavior of vessels, it also influences the existence of Nash equilibria. Under the assumption of Mechanism 1, where the priority of vessels is determined by the arrival of vessels at the lock, a Nash equilibrium might not exist, which is shown in the following example.

Table 1
Speed scenarios for Example 3.3.

Scenario	v_1	v_2	Improving move
1	10	$[5, v_{opt}]$	Player 1 should decrease v_1 to v_{opt}.
2	10	$(v_{opt}, 10]$	Player 2 should decrease v_2 to 5.
3	(5,10)	$v_2 \leq v_1$	Player 2 should increase v_2 to 10.
4	(5,10)	$v_2 \geq v_1$	Player 1 should increase v_1 to 10.
5	5	$(v_{opt}, 10]$	Player 2 should decrease v_2 to v_{opt}.
6	5	$[5, v_{opt}]$	Player 1 should increase v_1 to v_{opt}.

Example 3.3 (Mechanism 1)**.** Assume there are two vessels: vessel 1 sailing upstream and vessel 2 sailing downstream. The complete river segment is again 20 kilometers long, and the lock is placed in the middle of the waterway, hence, $L_u = L_d = 10$. The lock has capacity of 1 (though, any positive capacity will do) and its duration T and loading times T_1 and T_2 are set to 0.5. We assume that the fuel consumption function $F(v)$ is convex, non-negative and strictly increasing in speeds $v_{i,p} \in [5, 10]$, $p \in \{u, d\}$. We assume that the lock starts on the upstream side, but can switch to the downstream side in time whenever vessel 2 is the first one to arrive at the lock. We assume the arrival times in the system are given by $a_1 = 0$ and $a_2 = \epsilon$ and the deadlines are $d_1 = 4$ and $d_2 = 4 + \epsilon$. Note only the speed of a vessel before the lock affects the waiting time of other vessels. Therefore, for determining best responses, the strategy of the vessels can be expressed in their speed before the lock (denoted by v_1 for vessel 1, and v_2 for vessel 2). Note that for this example, $v_{opt} = 20/3$, i.e., the optimal speed for each vessel if it would be the only vessel on this waterway segment. We divide all possible speed scenarios into six cases, presented in Table 1.

1. Assume v_1 is equal to 10 and v_2 is any speed in the interval $[5, v_{opt}]$. Note that vessel 1 arrives first at the lock. However, since he arrives early into the system, he can reduce his speed to v_{opt} and still arrive first at the lock, which would reduce his costs.
2. Assume v_1 is equal to 10 and v_2 is any speed in the interval $(v_{opt}, 10]$. Given the speed of vessel 1, vessel 2 is unable to arrive earlier at the lock. Thus, his best response is to arrive after that the lock processed vessel 1, i.e. sailing at speed 5.
3. Assume v_1 is in the interval (5,10) and $v_2 \leq v_1$. In this scenario, both players exhibit a racing behavior. The bets response of vessel 2 is to increases its speed to $v_2' = v_1 + \varepsilon$. In response, vessel 1 increases its speed to $v_1' = v_2' + \varepsilon$ and vessel 2 increases its speed again to $v_2' = v_1' + \varepsilon$. This cycles until vessel 2 increase its speed to 10 and the game is in a different scenario
4. Assume v_2 is in the interval (5,10) and $v_1 \leq v_2$. This scenario is symmetric to the previous one.
5. Assume v_1 is equal to 5 and v_2 is in the interval $(v_{opt}, 10]$. In this scenario, vessel 2 arrives at the lock first. However, he would still arrive at the lock first if he decreases his speed to his optimal speed. Therefore, sailing at its optimal speed is the best move for vessel 2.
6. Assume v_1 is equal to 5 and v_2 is in the interval $[5, v_{opt}]$. In this scenario, vessel 1 can sail at his optimal speed and arrive the lock earlier than vessel 2. Thus, the optimal move for vessel 1 is to reduce his speed to its optimal.

We see that in every strategy profile, there is a skipper that can decrease its fuel consumption by changing its speed. Hence, there does not exist a Nash equilibrium.

Under lock operating mechanisms 2 and 3, however, the Nash equilibrium does exist as the order in which the vessels enter the lock is determined solely by the order in which they arrive into the system. Hence, it cannot occur that vessels race each other to the lock, which is the main idea behind our previous example. Under these two mechanisms, vessels cannot affect the costs of vessels that entered the river section earlier. This implies that vessels can sequentially choose a best response, taking into account the arrival times of the previous vessels. We prove this statement more formally in the next theorem.

Theorem 3.4. *Consider the single lock scheduling problem, where the lock operates under Mechanism 2 or 3. Then, each game possesses at least one Nash equilibrium.*

Proof. We provide a generic construction of a strategy profile and show that this strategy profile constitutes a Nash equilibrium. Observe that under both Mechanism 2 and 3, for any speed v_i, the waiting time of vessel i only depends on the vessels arriving earlier in the system than vessel i. Consequently, knowing the strategies $v_1, \dots, v_{i-1}$ is sufficient to determine optimal strategy v_i.

By construction of the strategy profile, it is apparent that each vessel i chooses its best possible strategy with respect to the vessels arriving earlier. Also, strategies of vessels that arrive later cannot influence the costs experienced by vessel i. Hence, vessel i can not decrease its private cost and therefore the resulting strategy profile is a Nash equilibrium.

Note, that the difference between the two mechanisms occurs in the individual optimization of strategies: under Mechanism 3 the waiting times caused by profile $\mathbf{v}_S$ might be different from the waiting times under Mechanism 2 using the same vector $\mathbf{v}_S$. However, the implications and the arguments stay the same: the cost for vessel i is only affected by the strategies of the first $i-1$ vessels. □

A central authority could guarantee the existence of a Nash equilibrium by forcing the lock operators to use Mechanism 2 or Mechanism 3. However, the fact that a Nash equilibrium exists does not tell us anything about its cost efficiency. Selfish decision making may lead to a Nash equilibrium with a high social cost, which then leads to a waste of resources and high pollution on rivers. In Mechanism 2 and 3, individual costs highly depend on the strategies taken by the previous vessels. Therefore, selfish decision making may lead to the scenario in which later vessels are unable to cross the river segment before their deadline, resulting in a Nash equilibrium with an infinitely high social cost. Such scenario indicates that the *price of anarchy* of this game (the ratio between the highest social cost of any Nash equilibrium and the minimal social cost) is unbounded. This becomes apparent in the following example.

Example 3.5. We consider the same instance as in Example 3.3. However, this time we assume that the lock operates under Mechanism 2. We construct a Nash equilibrium with the procedure described in the proof of Theorem 3.4. This implies that $v_1^* = (20/3, 20/3)$. Note that there is no strategy in the strategy space of vessel 2, such that it passes the river segment before its deadline. Thus the social cost of this instance is infinitely high.

There exists a strategy profile such that both vessels cross the river before their deadlines. More precisely, $\mathbf{v} = ((5, 10), (10, 5))$ leads to a finite cost for both vessels. However, this strategy profile is not attained by Mechanism 2 nor by Mechanism 3. Because of this, the price of anarchy of the game at hand is unbounded. Note that the same results hold, when the lock is assumed to operate under Mechanism 3.

Note that the results in the section do not rely on the assumption on equal cost functions and speed limits. The goal of this section was to show that, though the concept of a Nash equilibrium seems appealing, in the non-cooperative setting it might not exist or it might be extremely inefficient compared to a socially optimal strategy profile. In the next section, we review the problem from a cooperative game point of view as we introduce the possibility to make binding contracts between the vessels.

4. Cooperative game for traffic optimization at river obstacles

We now assume that the vessels can make binding contracts and allow *payments* between skippers. As a result, the agents (skippers) can give an incentive to their counter-agents to adapt their speeds by reimbursing their extra costs. We aim to find a solution concept that is cost optimal while making sure that no player can profit from a unilateral deviation from the social optimum. More precisely, we introduce a payment system that fulfills two criteria:

1. By participating in the payment system, the cost of a player can never be higher than when he/she did not participate.
2. The payment system should give a vessel an incentive to behave as in the social optimum.

In this section, we consider full information about the lock, river segments and vessels that will enter the system to be known in advance. An online variant of this problem is presented in Section 6, in which only the information about the river segment and the lock are publicly known while information about the vessels becomes only available when a vessel physically enters the waterway. Furthermore, we assume that the lock operates under mechanism 2 or 3. First, we propose an algorithm that returns for each vessel a speed v_i, and the payment scheme $P_{i,j}$ indicating payment of skipper i to skipper j for the requested velocity adjustment. Second, we prove that the solution proposed by the algorithm satisfies the two criteria mentioned above.

4.1. Iterative payment scheme algorithm

The algorithm sequentially determines optimal speeds and payments in the order of vessels arrival by considering all vessels 1 through i, denoted by the set $\bar{S}_i$. In the first iteration, only vessel 1 is considered and its optimal speed is determined. Let ζ_1 be the operating cost associated with this strategy such that $\zeta_1 = C_1(v_1)$. During future iterations, it will be ensured that the cost for this skipper will not go above the cost of this benchmark situation. To do this, other skippers should fully reimburse any cost increase that results from changing the strategy for the skipper.

Now, let $P'_{j,j'}$ be the payment scheme for all $j' < j < i$ at iteration i. Moreover, all guaranteed costs ζ_j are considered to be known for all $j < i$. To determine the speeds v_j for all $j \in \bar{S}_i$ and payments $P_{i,j}$ for all $j < i$, we solve the following optimization problem: determine new velocities of the vessels from $\bar{S}_i$ such that the sum of the costs and payments for vessel i is minimized, while the total cost of each vessel $j < i$ is at most ζ_j. Then, we compute the value of the guaranteed cost ζ_i of player i. More formally, we define following relations.

$$P'_{i,k} := C_k\left(v'_{\bar{S}_i}\right) - \zeta_k - \sum_{j \in \bar{S}_{i-1} : j > k} P'_{jk} \quad \forall\, k \in \bar{S}_{i-1}, \tag{7}$$

$$\zeta_i := C_i\left(v'_{\bar{S}_i}\right), \tag{8}$$

where v'_S and P' are the solutions to the following optimization problem. For a given vessel $i > 1$, having computed all optimal values P' for all $i' < i$, the mathematical program reads

$$\min_{\mathbf{v}_{\bar{S}_i}; P_{i,j}} \left(C_i\left(\mathbf{v}_{\bar{S}_i}\right) + \sum_{k \in \bar{S}_{i-1}} P_{i,k} \right) \tag{9}$$

subject to

$$C_k\left(\mathbf{v}_{\bar{S}_i}\right) - P_{i,k} - \sum_{\substack{j \in \bar{S}_{i-1} \\ j > k}} P'_{j,k} \le \zeta_k, \qquad \forall\, k \in \bar{S}_{i-1}. \tag{10}$$

Algorithm 1: Payment mechanism.

Input: $(L := (C, T, P, L_u, L_d), U, D, (a_i, d_i, v_{\min}, v_{\max})_{i\in U\cup D})$
Output: Optimal set of speeds and payments.
$\bar{S}_1 = \{1\}$;
$\zeta_1 = C_{opt}(S_1)$;
for *i from 2 to n* **do**
  $\bar{S}_i = \bar{S}_{i-1} \cup \{i\}$;
  Compute $C_{opt}(\bar{S}_i)$ and let $v^*_{\bar{S}_i}$ be the optimal parameters;
  $C_{opt,k}(\bar{S}_i) := C_k\big(v^*_{\bar{S}_i}\big) \quad \forall k \in \bar{S}_i$;
  $P'_{i,k} := C_{opt,k}(\bar{S}_i) - \zeta_k - \sum_{j\in\bar{S}_{i-1}:j>k} P'_{jk} \quad \forall\ k \in \bar{S}_{i-1}$;
  $\zeta_i := C_{opt,i}(\bar{S}_i)$;
end
return $(v^*_S, (P'_{ij})_{i,j\in S})$

Algorithm 1 represents the payment system which outputs both speeds and payments for all skippers. Note that the optimization problem has been replaced by a computation of the social optimal speeds. This is a valid substitution due to Theorem 4.3 below.

The subroutine computing $C_{opt}(\bar{S}_i)$ can be implemented in various ways. In the next section, we provide a MILP-formulation to solve the lock scheduling problem to optimality. This formulation is based on the model in Passchyn et al. (2016a) and has been adjusted to comply with our problem statement. Moreover, we show that the problem is NP-complete in the strong sense, this way motivating design of MILP-formulations and approximation algorithms for the problem. Note that the MILP can be extended to allow for non equivalent speed limits and cost functions and therefore all the results generalise. Regarding existence of good approximation algorithms, we leave this question open. We stress that any α-approximation algorithm directly leads to an α-approximate Nash equilibrium. This follows from Theorem 4.3 and Theorem 4.4 below, in which we show that the social optimum and Nash Equilibrium coincide.

Definition 4.1 (α-approximation algorithm)**.** An algorithm is considered an α-approximation algorithm for a problem if and only if for every instance of the problem it can find a solution within a factor α of the optimum solution. Let *ALG* be the value of a solution provided by the algorithm, and *OPT* the optimal solution of the minimization problem, then

$$ALG \le \alpha\ OPT. \tag{11}$$

Definition 4.2 (α-approximate equilibrium)**.** For any $\alpha \ge 1$, we define strategy $\mathbb{v}$ to be an α-approximate equilibrium when for every player i, and every alternative strategy $v'_i \in V_i$:

$$C_i(v_i, \mathbb{v}_{-i}) \le \alpha C_i(v'_i, \mathbb{v}_{-i}). \tag{12}$$

Given a solution to the optimization problem above, in Theorem 4.3, we show that the optimal speeds in that problem are equivalent to the speeds in the social optimum computed on vessels in the set $\bar{S}_i$.

Theorem 4.3. $v'_{\bar{S}_i} = \arg\min_{\mathbf{v}_{\bar{S}_i}} \sum_{k\in\bar{S}_i} C_k(\mathbf{v}_{\bar{S}_i})$*. or equivalently* $\mathbf{v}'_{\bar{S}_i} = \mathbf{v}^*_{\bar{S}_i}$

Proof. Consider the mathematical program given in Eqs. 10 and (11). Note, that the constraint can be defined for all $k \in \bar{S}_{i-1}$:

$$C_k\big(\mathbf{v}_{\bar{S}_i}\big) - P_{i,k} - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{j,k} \le \zeta_k \tag{13}$$

which can be rewritten as:

$$P_{i,k} \ge C_k\big(\mathbf{v}_{\bar{S}_i}\big) - \zeta_k - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{j,k} \tag{14}$$

Note the objective is to minimize and the objective function is increasing in $P_{i,k}$ for all k. Thus, the $P_{i,k}$ is as low as possible and the inequality is always binding. The program can furthermore be rewritten as follows:

$$P_{i,k} = C_k\big(\mathbf{v}_{\bar{S}_i}\big) - \zeta_k - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{j,k} \tag{15}$$

Then we can reformulate the objective function in the following way:

$$v'_{\bar{S}_i} = \arg\min_{\mathbf{v}_{\bar{S}_i}} \Big(C_i\big(\mathbf{v}_{\bar{S}_i}\big) + \sum_{k\in\bar{S}_{i-1}} P_{i,k}\Big) \tag{16}$$

$$= \arg\min_{\mathbf{v}_{\bar{S}_i}} \Big(C_i\big(\mathbf{v}_{\bar{S}_i}\big) + \sum_{k\in\bar{S}_{i-1}} \Big(C_k\big(\mathbf{v}_{\bar{S}_i}\big) - \zeta_k - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{j,k}\Big)\Big) \tag{17}$$

$$= \arg\min_{\mathbf{v}_{\bar{S}_i}} \Big(\sum_{k\in\bar{S}_i} C_k\big(\mathbf{v}_{\bar{S}_i}\big) - \sum_{k\in\bar{S}_{i-1}} \Big(\zeta_k + \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{j,k}\Big)\Big) \tag{18}$$

$$= \arg\min_{\mathbf{v}_{\bar{S}_i}} \Big(\sum_{k\in\bar{S}_i} C_k\big(\mathbf{v}_{\bar{S}_i}\big)\Big) \tag{19}$$

$$= v^*_{\bar{S}_i}, \tag{20}$$

□

Lastly, in Theorem 4.4, we show that in the ith iteration of the algorithm the best response for skipper i is to obey the payment mechanism. This means that the guaranteed cost of vessel i plus the payments this skipper has to pay to all other skippers is lower than or equal to the cost of any strategy not involving the payments.

Theorem 4.4. *In Algorithm 1 for each* $\bar{S}_i$*, it holds that*

$$\zeta_i + \sum_{k\in\bar{S}_{i-1}} P_{i,k} \le C_i(v_i, \mathbf{v}^*_{\bar{S}_{i-1}}) \text{ for all } v_i \in V_i. \tag{21}$$

Proof. Note that by (13) to (15), we know that after every iteration i, it holds for every $k \in \bar{S}_{i-1}$

$$P'_{i,k} = C_{opt,k}(\bar{S}_i) - \zeta_k - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{jk} \tag{22}$$

which can be rewritten as

$$C_{opt,k}(\bar{S}_i) = \zeta_k + P'_{i,k} + \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{jk} \tag{23}$$

$$= \zeta_k + \sum_{\substack{j\in\bar{S}_i\\ j>k}} P'_{jk} \tag{24}$$

This leads to the following equality.

$$\zeta_i + \sum_{k\in\bar{S}_{i-1}} P'_{ik} = C_{opt,i}(\bar{S}_i) + \sum_{k\in\bar{S}_{i-1}} \Big(C_{opt,k}(\bar{S}_i) - \zeta_k - \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{jk}\Big) \tag{25}$$

$$= \sum_{k\in\bar{S}_i} C_{opt,k}(\bar{S}_i) - \sum_{k\in\bar{S}_{i-1}} \Big(\zeta_k + \sum_{\substack{j\in\bar{S}_{i-1}\\ j>k}} P'_{jk}\Big) \tag{26}$$

$$= \sum_{k \in \bar{S}_i} C_{opt,k}(\bar{S}_i) - \sum_{k \in \bar{S}_{i-1}} C_{opt,k}(\bar{S}_{i-1}) \quad (27)$$

$$= C_{opt}(\bar{S}_i) - C_{opt}(\bar{S}_{i-1}) \quad (28)$$

Furthermore, we know that, by definition, the social optimum of the set $\bar{S}_i$ cannot have a higher cost than the sum of the social optimum of the set $\bar{S}_{i-1}$ plus any individual strategy of vessel i. Therefore, we can formulate the following inequality.

$$C_{opt}(\bar{S}_i) \leq C_i(v_i, \mathbf{v}^*_{\bar{S}_{i-1}}) + C_{opt}(\bar{S}_{i-1}) \quad \text{for all } v_i \in V_i \quad (29)$$

Then by rewritten the inequality and using the result established in equality (28).

$$C_{opt}(\bar{S}_i) - C_{opt}(\bar{S}_{i-1}) \leq C_i(v_i, \mathbf{v}^*_{\bar{S}_{i-1}}) \quad \text{for all } v_i \in V_i \quad (30)$$

$$\zeta_i + \sum_{k \in \bar{S}_{i-1}} P'_{ik} \leq C_i(v_i, \mathbf{v}^*_{\bar{S}_{i-1}}) \quad \text{for all } v_i \in V_i \quad (31)$$

□

From Theorems 4.3 and 4.4, it follows that the stated criteria for an efficient payment mechanism are fulfilled by Algorithm 1.

4.2. Truthfulness of the Payment Scheme

Theorem 4.5. *Under payment scheme skippers report deadline truthfully*

Proof. Recall from (25) to (28) that

$$\zeta_i + \sum_{k \in \bar{S}_{i-1}} P'_{ik} = C_{opt}(\bar{S}_i) - C_{opt}(\bar{S}_{i-1}) \quad (32)$$

Assume that vessels $1\ldots, i-1$ have reported their deadline truthfully and their payments $P_{j,k}$ and guaranteed costs ζ_j have been computed for all $j, k = 1\ldots, i-1$. Furthermore, despite the fact that the true deadline of vessel i is d_i, he reports $\tilde{d}_i \neq d_i$. We do, however, assume that ship i is following the speed advice. In what follows, we show that a truthful reporting of deadline of vessel i is a best response.

Consider the case, in which vessel i extends his deadline, i.e. $\tilde{d}_i > d_i$. Due to the convexity of the the cost function, the optimal speeds are selected such that each vessel arrives exactly at its deadline at the end of the waterway. Therefore, by following the speeds, which result by the extended deadline, vessel i will arrive at the end of the waterway after his true deadline. Therefore, vessel i has infinite cost in this scenario.

Consider the case, in which vessel i shortens his deadline, i.e. $\tilde{d}_i < d_i$. Let $\tilde{\zeta}_i$, $\tilde{P}'_{ik}$ and $\tilde{C}_{opt}(\bar{S}_i)$ be the individual cost, payments of i to each vessel k and optimal cost, each under the assumption that vessel i reported $\tilde{d}_i$. In the proof of Theorem 4.4, we know that that the individual costs for the true deadline and the shortened deadline are

$$\zeta_i + \sum_{k \in \bar{S}_{i-1}} P'_{ik} = C_{opt}(\bar{S}_i) - C_{opt}(\bar{S}_{i-1}) \quad (33)$$

and

$$\tilde{\zeta}_i + \sum_{k \in \bar{S}_{i-1}} \tilde{P}'_{ik} = \tilde{C}_{opt}(\bar{S}_i) - C_{opt}(\bar{S}_{i-1}) \quad (34)$$

respectively. However, since a shorter deadline of vessel i is more restrictive in the optimization, it has to hold that $C_{opt}(\bar{S}_i) \leq \tilde{C}_{opt}(\bar{S}_i)$. Thus, the total cost of the system can only decrease and therefore it follows

$$\zeta_i + \sum_{k \in \bar{S}_{i-1}} P'_{ik} \leq \tilde{\zeta}_i + \sum_{k \in \bar{S}_{i-1}} \tilde{P}'_{ik}. \quad (35)$$

Therefore, we have shown that vessel i cannot reduce its cost by reporting a non-truthful deadline. □

A similar result does not hold for reporting of non-truthful cost functions. If a vessel reports a cost function, which is steeper than his true cost function, it receives a higher payment for deviating his speed and therefore he can reduce his cost by doing so. However, cost functions are defined in the technical manual of a vessel and therefore it is physically impossible for skippers to lie about them.

5. A MILP-formulation for finding a social optimum

First, we show that even deciding on existence of a feasible solution to the optimization problem (FEASIBILITY CHECK) is strongly NP-complete. Consider an instance of the classic machine scheduling problem SEQUENCING WITH RELEASE TIMES AND DEADLINES. The problem is known to be strongly NP-complete, see e.g., Garey and Johnson (1979). Given a set J of n tasks and, for each task $j \in J$, a length $p_j \in Z^+$, a release time $r_j \in Z^+_0$, and a deadline $d_j \in Z^+$, the question is whether there exists a one-processor schedule for J that satisfies the release time constraints and meets all the deadlines. We reduce SEQUENCING WITH RELEASE TIMES AND DEADLINES to FEASIBILITY CHECK. Given an instance of SEQUENCING WITH RELEASE TIMES AND DEADLINES, we create an instance of FEASIBILITY CHECK as follows. Let each job be represented by a vessel with $a_j = r_j$, $d_j = d_j$ and $T_j = p_j$. Furthermore, let set U contain all vessels, set minimum speed to 0 and let the maximum speed be unbounded. Next, set capacity to 1 and lockage duration equal to 0. Clearly, an instance of SEQUENCING WITH RELEASE TIMES AND DEADLINES is a yes-instance if and only if the corresponding instance of FEASIBILITY CHECK is a yes-instance.

Next, we describe a MILP-formulation that can be used to compute a social optimum. This formulation is an adjustment of the model from Passchyn et al. (2016a). In their paper, the authors propose a model to minimize emission on a waterway with multiple locks. Our program differs from that model in a few ways. First of all, it is a bit simpler as we solve a single lock scheduling problem. On the other hand, our model takes into account a need for compensation for the lost time in case of a slow speed towards the lock, and/or for a high speed towards the lock. We give a brief overview of the model. For more details, we refer the interested reader to Passchyn et al. (2016a).

We introduce the variables $\bar{v}_{i,p} = \frac{1}{v_{i,p}}$ and let $\bar{E}(\bar{v})$ express the emissions as a function of the reciprocal of vessel speed, $\bar{v}_{i,p}$. To enable the usage of MILP, we use a piece-wise linear approximation of $\bar{E}(\bar{v})$. The decision variables in the model are based on possible lockages. It is clear that for each lock, the number of lockages in the optimal solution does not need to be greater than double the number of ships. Thus, the upper bound for the number of lockages is defined as $K = 2|S|$. The set of possible lockages is defined as $K = \{1, \ldots, K\}$. In addition to variables $\bar{v}_i$, we introduce for all $k \in K$ t_k as the starting time of the k'th lockage and for each $i \in S$ and $k \in K$ we define:

$$z_{i,k} = \begin{cases} 1, & \text{if vessel} i \text{is handled by the} k\text{th lock movement.} \\ 0, & \text{otherwise} \end{cases}$$

The mathematical programming formulation, presented below, returns an optimal strategy profile for a set of vessels.

$$\min \sum_{i \in S} \bar{E}(\bar{v}_i) \quad (36)$$

Subject to

$$A_i \leq t_k - l_d \bar{v}_{i,d} + M_i^{A,u} \left(1 - \sum_{\kappa=0}^{k} z_{i,\kappa}\right) \qquad \forall i \in U, k \in K \quad (37)$$

$$A_i \le t_k - l_d\bar{v}_{i,u} + M_i^{A,d}(1-\sum_{\kappa=0}^{k} z_{i,\kappa}) \qquad \forall i\in D, k\in K \tag{38}$$

$$D_i \ge t_k + T + \sum_{j\in S} z_{j,k}T_j + l_d\bar{v}_{i,u} - M_i^{D,u}(1-\sum_{\kappa=k}^{K} z_{i,\kappa}) \quad \forall i\in U, k\in K \tag{39}$$

$$D_i \ge t_k + T + \sum_{j\in S} z_{j,k}T_j + l_u\bar{v}_{i,d} - M_i^{D,d}(1-\sum_{\kappa=k}^{K} z_{i,\kappa}) \quad \forall i\in D, k\in K \tag{40}$$

$$\sum_{k\in K} z_{i,k} = 1 \qquad \forall i\in S \tag{41}$$

$$t_k \ge t_{k-1} + T + \sum_{j\in S} z_{j,k-1}T_j \qquad \forall k\in K\setminus\{1\} \tag{42}$$

$$z_{i,k} + z_{j,k} \le 1 \qquad \forall i\in U, j\in D, k\in K \tag{43}$$

$$z_{i,k-1} + z_{j,k} \le 1 \qquad \forall i,j\in U, k\in K\setminus\{1\} \tag{44}$$

$$z_{i,k-1} + z_{j,k} \le 1 \qquad \forall i,j\in D, k\in K\setminus\{1\} \tag{45}$$

$$\sum_{i\in S} z_{i,k} \le C \qquad \forall k\in K \tag{46}$$

$$1/v^{max} \le \bar{v}_{i,p} \le 1/v^{min} \qquad \forall i\in S, p\in\{u,p\} \tag{47}$$

$$z_{i,k}\in\{0,1\} \qquad \forall i\in S, k\in K \tag{48}$$

$$t_k \in R_+ \qquad \forall k\in K \tag{49}$$

The objective function (36) minimizes the aggregated costs of the individual vessels. Constraints (37) and (38) ensure that vessels arrive at the lock before their respective lockage time has started and constraints (39) and (40) ensure that vessels are leaving the system before their deadline. Constraints (39) - (42) are based on values $M^{A,u}, M^{A,d}, M^{D,u}$ and $M^{D,d}$, which are large constants specified in Passchyn et al. (2016a)

Constraints (41) to (46) are used to model the working of the lockages. Thus, (41) ensures that each vessel is scheduled on exactly one lockage, while (42) ensures that difference between any two starting times of lockages is at least the lockage duration, $T+\sum_{i\in S}T_i$. Constraint (43) guarantees that a lockage does not contain vessels coming from opposing sides of the river. Furthermore, the requirement that lockages are alternating between opposing sides is ensured by constraints (44) and (45). Finally, constraint (46) restricts lockages to carry only as many vessels as specified by the capacity, C.

6. Online setting

The assumption of perfect information on arrival times is likely to be violated in real-life. That is, there is no information prior to the arrival of the vessels at the boundaries of the system. Each time a vessel enters, the optimal speed and payments are recomputed taking into account the location of the vessels already present on the waterway. Note that the definition of a social optimum and a best response of a player are dependent on the information setting of the game. Therefore, we have to dynamically redefine/adjust these quantities in an online setting.

Let the distance between vessel i and the exit of the waterway at time t be denoted by h_i^t. Furthermore, define $\mathbf{h}_S = (h_i^t)_{i\in S}$. The best response of vessel i, given the strategies of the other vessels, is defined as the strategy that minimizes the cost of vessel i conditional on the position of the other vessels at time t. The cost of vessel i under strategy profile $\mathbf{v}$ conditional on the position of all vessels in set $\bar{S}$ at time t is denoted as $C_i\left(\mathbf{v}_{\bar{S}}|\mathbf{h}_{\bar{S}}^t\right)$. The social optimum is defined as a strategy profile, which provides the lowest possible cost given the positions of vessels in $\bar{S}$ at time t.

Similar to the offline setting, the algorithm sequentially determines optimal speeds and payments at the arrival of each vessel. In each iteration, a set $\bar{S}_i$ is constructed containing all vessels currently in the system. Assume that vessel i arrives and vessels in $\bar{S}$ have not left the waterway yet. Moreover, the payments $P'_{j,j'}$ for all $k\le j'<j<i$ and the guaranteed costs ζ_j for all $k\le j<i$ are assumed to be given. Since each vessel is at a different position, payments and costs are normalized to units per kilometers. Therefore, we solve the following optimization problem: determine new velocities of the vessels from $\bar{S}_i$ such that the sum of the costs and payments for vessel i is minimized, while the normalized total cost of each vessel $k\le j<i$ is at most the normalized guaranteed cost. Given the following relations

$$C_{opt,k}^{a_i}(\bar{S}_i) = C_k\left(\mathbf{v}'_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right) \quad \forall\ k\in\bar{S}_i, \tag{50}$$

$$P'_{i,k} := C_{opt,k}^{a_i}(\bar{S}_i) - h_k^{a_i}\left(\sum_{\substack{j\in\bar{S}_i\setminus\{i\}\\ j>k}}\left(\frac{P'_{j,k}}{h_k^{a_j}}\right) + \frac{\zeta_k}{l_d+l_u}\right) \quad \forall\ k\in\bar{S}_i\setminus\{i\}, \tag{51}$$

$$\zeta_i := C_{opt,i}^{a_i}(\bar{S}_i) \tag{52}$$

we define the online optimization problem as

$$min_{\mathbf{v}_{\bar{S}_i}:P_{i,j}}\left(C_i\left(\mathbf{v}_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right) + \sum_{k\in\bar{S}_i\setminus\{i\}} P_{i,k}\right) \tag{53}$$

subject to

$$\frac{C_k\left(\mathbf{v}_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right)}{h_k^{a_i}} - \frac{P_{i,k}}{h_k^{a_i}} - \sum_{\substack{j\in\bar{S}_i\setminus\{i\}\\ j>k}}\frac{P'_{j,k}}{h_k^{a_j}} \le \frac{\zeta_k}{l_d+l_u} \quad \forall\ k\in\bar{S}_i\setminus\{i\}. \tag{54}$$

Again, it can be shown that the two conditions for an efficient payment mechanism are fulfilled in the online setting. The proof is similar to the one discussed in the offline case. The resulting algorithm is given in Algorithm 2.

Theorem 6.1 below shows that the strategy profile is again equivalent to the social optimum, conditionally on the position of players at arrival time of vessel i. Thus, the optimal strategy profile is an output of Algorithm 2.

Theorem 6.1. *$v'_{\bar{S}_i} = \arg\min_{(\mathbf{v}_{\bar{S}_i}}\sum_{k\in\bar{S}_i}C_k\left(\mathbf{v}_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right)$ or equivalently $\mathbf{v}'_{\bar{S}_i} = \mathbf{v}^*_{\bar{S}_i}$*

Proof. Consider the mathematical program given in equations (53) and (54). Note, that the constraint can be defined for all $k\in\bar{S}_{\setminus}\{i\}$:

$$\frac{C_k\left(\mathbf{v}_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right)}{h_k^{a_i}} - \frac{P_{i,k}}{h_k^{a_i}} - \sum_{\substack{j\in\bar{S}_i\setminus\{i\}\\ j>k}}\frac{P'_{j,k}}{h_k^{a_j}} \le \frac{\zeta_k}{l_d+l_u} \tag{55}$$

which can be rewritten as:

$$P_{i,k} \ge C_k\left(\mathbf{v}_{\bar{S}_i}|\mathbf{h}_{\bar{S}_i}^{a_i}\right) - h_k^{a_i}\left(\sum_{\substack{j\in\bar{S}_i\setminus\{i\}\\ j>k}}\left(\frac{P^*_{j,k}}{h_k^{a_j}}\right) + \frac{\zeta_k}{l_d+l_u}\right) \tag{56}$$

Algorithm 2: Payment mechanism Online Setting.

Input: $(L := (C, T, P, L_u, L_d), U, D, (a_i, d_i, v_{\min}, v_{\max})_{i \in U \cup D})$
Output: Optimal set of speeds and payments.
Vessel i arrives in the system at time a_i:
For each vessel currently present in the waterway, update the distance to the destination as follows:;
$h_i^{a_i} = l_d$ if $i \in U$;
$h_i^{a_i} = l_u$ if $i \in D$;
$h_j^{a_i} = h_j^{a_{i-1}} - v_j^*(a_i - a_{i-1})$ for $j \in S_{i-1}$;
Let $\bar{S}_i$ be the set of vessels in the waterway at time a_i as follows:;
$\bar{S}_i = \emptyset$ if $i = 0$;
$\bar{S}_i = \bar{S}_{i-1} \setminus \left\{ j \in \bar{S}_{i-1} | h_j^{a_i} \le 0 \right\}$;
if $\bar{S}_i \neq \emptyset$ **then**
  Compute $C_{opt}^{a_i}(\bar{S}_i)$ and let $\mathbf{v}_{\bar{S}_i}^*$ be the optimal parameters;

$$C_{opt,k}^{a_i}(\bar{S}_i) = C_k\left(v_{\bar{S}_i}^* | \mathbf{h}_{\bar{S}_i}^{a_i}\right) \quad \forall\ k \in \bar{S}_i;$$

$$P_{i,k}^* := C_{opt,k}^{a_i}(\bar{S}_i) - h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P_{j,k}^*}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \ \forall\ k \in \bar{S}_i \setminus \{i\};$$

$$\zeta_i := C_{opt,i}^{a_i}(\bar{S}_i);$$

else

$$\zeta_i = \min_{v_i} C_i(v_i);$$

end
$\bar{S}_i = \bar{S}_{i-1} \cup \{i\}$;

Note that with the same argument as in Theorem 4.4, the constraint must be binding:

$$P_{i,k} = C_k\left(\mathbf{v}_{\bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) - h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P_{j,k}^*}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \tag{57}$$

Then we can reformulate the objective function in the following way.

$$v'_{\bar{S}_i} \in \arg\min_{\mathbf{v}_{\bar{S}_i}: P_{i,j}} \left(C_i\left(\mathbf{v}_{\bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) + \sum_{k \in \bar{S}_i \setminus \{i\}} P_{i,k} \right) \tag{58}$$

$$\in \arg\min_{\mathbf{v}_{\bar{S}_i}: P_{i,j}} \left(C_i\left(\mathbf{v}_{\bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) + \textstyle\sum_{k \in \bar{S}_i \setminus \{i\}} C_k\left(\mathbf{v}_{\bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) - h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \right) \tag{59}$$

$$\in \arg\min_{\mathbf{v}_{\bar{S}_i}: P_{i,j}} \left(\textstyle\sum_{k \in \bar{S}_i} \left(C_k\left(v_{j \in \bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) \right) - \sum_{k \in \bar{S}_i \setminus \{i\}} h_k^{a_i} \left(\sum_{\substack{j \in \bar{S}_{i-1} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \right) \tag{60}$$

$$\in \arg\min_{\mathbf{v}_{\bar{S}_i}: P_{i,j}} \left(\sum_{k \in \bar{S}_i} \left(C_k\left(\mathbf{v}_{\bar{S}_i} | \mathbf{h}_{\bar{S}_i}^{a_i}\right) \right) \right) \tag{61}$$

$$= \mathbf{v}_{\bar{S}_i}^* \tag{62}$$

□

Likewise, Theorem 6.2 shows that it is the best response for vessel i to obey the payment mechanism.

Theorem 6.2. *In Algorithm 2 for each $\bar{S}_i$, it holds that*

$$\zeta_i + \sum_{k \in \bar{S} \setminus \{i\}} P_{i,k} \le C_i\left(\mathbf{v}_i, v_{\bar{S} \setminus \{i\}}^* | \mathbf{h}_{\bar{S}_i}^{a_i}\right) \text{ for all } v_i \in V_i. \tag{63}$$

Proof. Note that by (55) to (57), we know that after every iteration i, it holds for every $k \in \bar{S}_i \setminus \{i\}$

$$P'_{i,k} = C_{opt,k}^{a_i}(\bar{S}_i) - h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \tag{64}$$

which can be rewritten as

$$C_{opt,k}^{a_i}(\bar{S}_i) = P'_{i,k} + h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \tag{65}$$

$$= h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \\ j > k}} \left(\frac{P_{j,k}^*}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \tag{66}$$

This leads to the following equality.

$$\zeta_i + \sum_{k \in \bar{S} \setminus \{i\}} P'_{i,k}$$

$$= C_{opt,i}^{a_i}(\bar{S}_i) + \sum_{k \in \bar{S} \setminus \{i\}} \left(C_{opt,k}^{a_i}(\bar{S}_i) - h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \right) \tag{67}$$

$$= \sum_{k \in \bar{S}} C_{opt,k}^{a_i}(\bar{S}_i) - \sum_{k \in \bar{S} \setminus \{i\}} \left(h_k^{a_i}\left(\sum_{\substack{j \in \bar{S}_i \setminus \{i\} \\ j > k}} \left(\frac{P'_{j,k}}{h_k^{a_j}} \right) + \frac{\zeta_k}{l_d + l_u} \right) \right) \tag{68}$$

$$= \sum_{k \in \bar{S}} C_{opt,k}^{a_i}(\bar{S}_i) - \sum_{k \in \bar{S} \setminus \{i\}} C_{opt,k}^{a_i}(\bar{S}_i \setminus \{i\}) \tag{69}$$

$$= C_{opt}^{a_i}(\bar{S}_i) - C_{opt}^{a_i}(\bar{S}_i \setminus \{i\}) \tag{70}$$

Furthermore, we know that, by definition, the social optimum of the set $\bar{S}_i$ cannot have a higher cost than the sum of the social optimum of the set $\bar{S}_i \setminus \{i\}$ plus any individual strategy of vessel i. Therefore, we can formulate the following inequality.

$$C_{opt}^{a_i}(\bar{S}_i) \le C_i(v_i, v_{\bar{S}_i \setminus \{i\}}^*) + C_{opt}^{a_i}(\bar{S}_i \setminus \{i\}) \quad \text{for all } v_i \in V_i \tag{71}$$

Then by rewriting the inequality and using the result established in (67) - (70)

$$C_{opt}^{a_i}(\bar{S}_i) - C_{opt}^{a_i}(\bar{S}_i \setminus \{i\}) \le C_i(v_i, \mathbf{v}_{\bar{S}_{i-1}}^*) \quad \text{for all } v_i \in V_i \tag{72}$$

$$\zeta_i + \sum_{k \in \bar{S}_{i-1}} P_{ik}^* \le C_i(v_i, \mathbf{v}_{\bar{S}_{i-1}}^*) \quad \text{for all } v_i \in V_i \tag{73}$$

□

7. Future work

Note that under iterative payment scheme, whenever a vessel enters the system, its total fuel cost and payments to the other vessels become known, and will not change anymore. Hence, the lock operator can also serve as a bank: whenever a vessel crosses the lock, it pays (or receives) the payments. This implies that the lock operator needs a cash reserve, as it is likely that the first vessels entering the lock receive money from the vessels that did not arrive at the lock yet. Clearly, this cash reserve needs to be at most the cost of an optimal profile minus the minimum fuel cost of all earlier vessels. An interesting open question arises: what is the minimum amount of cash reserves needed to cover all payments without any risk of bankruptcy.

Furthermore, we assume that missing the deadline results in a infinite cost for skippers. A reasonable assumption is to assume that a delay results in a specified cost. Therefore, an interesting - but probably challenging - question would be to know if similar results apply if we the assumptions on delayed skippers changes.

Acknowledgements

We are grateful to the cooperating partner "Trapps Wise B.V." for providing us insight into problems, which the inland-waterway transportation companies face.

Furthermore, we would like to thank the reviewers for their thorough reading and providing numerous suggestions for improvement of the article.

References

Bialystocki, N., & Konovessis, D. (2016). On the estimation of ship's fuel consumption and speed curve: a statistical approach. *Journal of Ocean Engineering and Science, 1*(2), 157–166.

Defryn, C., Golak, J. A., Grigoriev, A., & Timmermans, V. (2019). Inland waterway efficiency through skipper collaboration and joint speed optimization. In M. Khachay, Y. Kochetov, & P. M. Pardalos (Eds.), *Mathematical optimization theory and operations research, motor 2019, Ekaterinburg, Russia, July 8–12, 2019, proceedings*. In *Lecture Notes in Computer Science*. Springer. to appear

European Commission (2017). Inland navigation in europe, market observation. Central commission for the navigation of the rhine, annual report 2017. https://www.inland-navigation-market.org/wp-content/uploads/2017/09/CCNR_annual_report_EN_Q2_2017_BD_-1.pdf. Accessed 21 May 2019.

Eurostat (2019). Navigable inland waterways, by horizontal dimensions of vessels and pushed convoys. http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=iww_if_hordim&lang=en. Accessed 21 May 2019.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. New York, NY, USA: W. H. Freeman & Co..

Lübbecke, E., Lübbecke, M. E., & Möhring, R. H. (2019). Ship traffic optimization for the kiel canal. *Operations Research, note*.

Nauss, R. M. (2008). Optimal sequencing in the presence of setup times for tow/barge traffic through a river lock. *European Journal of Operational Research, 187*(3), 1268–1281.

Passchyn, W., Briskorn, D., & Spieksma, F. C. R. (2016a). Mathematical programming models for lock scheduling with an emission objective. *European Journal of Operational Research, 248*(3), 802–814.

Passchyn, W., Briskorn, D., & Spieksma, F. C. R. (2019). No-wait scheduling for locks. *INFORMS Journal on Computing*. to appear

Passchyn, W., Coene, S., Briskorn, D., Hurink, J. L., Spieksma, F. C. R., & Berghe, G. V. (2016b). The lockmasters problem. *European Journal of Operational Research, 251*(2), 432–441.

Petersen, E. R., & Taylor, A. J. (1988). An optimal scheduling system for the welland canal. *Transportation Science, 22*(3), 173–185.

Prandtstetter, M., Ritzinger, U., Schmidt, P., & Ruthmair, M. (2015). A variable neighborhood search approach for the interdependent lock scheduling problem. In G. Ochoa, & F. Chicano (Eds.), *Evolutionary computation in combinatorial optimization –15th european conference, EVOCOP 2015, Copenhagen, Denmark, April 8–10, 2015, proceedings*. In *Lecture Notes in Computer Science: 9026* (pp. 36–47). Springer.

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

Smith, L. D., Nauss, R. M., Mattfeld, D. C., Li, J., Ehmke, J. F., & Reindl, M. (2011). Scheduling operations at system choke points with sequence-dependent delays and processing times. *Transportation Research Part E: Logistics and Transportation Review, 47*(5), 669–680.

Smith, L. D., Sweeney, D. C., & Campbell, J. F. (2009). Simulation of alternative approaches to relieving congestion at locks in a river transportion system. *Journal of the Operational Research Society, 60*(4), 519–533.

Ting, C.-J., & Schonfeld, P. (1999). Effects of speed control on tow travel costs. *Journal of Waterway, Port, Coastal, and Ocean Engineering, 125*(4), 203–206.

Ting, C.-J., & Schonfeld, P. (2001). Control alternatives at a waterway lock. *Journal of Waterway, Port, Coastal, and Ocean Engineering, 127*(2), 89–96.

Verstichel, J., De Causmaecker, P., Spieksma, F. C. R., & Berghe, G. V. (2014a). Exact and heuristic methods for placing ships in locks. *European Journal of Operational Research, 235*(2), 387–398.

Verstichel, J., De Causmaecker, P., Spieksma, F. C. R., & Berghe, G. V. (2014b). The generalized lock scheduling problem: An exact approach. *Transportation Research Part E: Logistics and Transportation Review, 65*, 16–34.