A model predictive scheduling strategy for coordinated inland vessel navigation and bridge operation

Segovia Castillo, P.; Puig, Vicenc; Reppa, Vasso

DOI
10.1109/CCTA54093.2023.10252942

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 2023 IEEE Conference on Control Technology and Applications, CCTA 2023

2023 IEEE Conference on Control Technology and Applications (CCTA)
August 16-18, 2023. Bridgetown, Barbados

A model predictive scheduling strategy for coordinated inland vessel navigation and bridge operation

Pablo Segovia[1], Vicenç Puig[2] and Vasso Reppa[1]

2023 IEEE Conference on Control Technology and Applications (CCTA) | 979-8-3503-3544-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/CCTA54093.2023.10252942

Abstract— **This paper presents the design of a model predictive scheduling strategy to address the inland waterborne transport (IWT) problem considering bridges that must open to enable vessel passage. The main contribution is the formulation of a control-oriented model of the problem, including propositional logic expressions that characterize system behavior and their conversion into (in)equality constraints. The resulting model is embedded into a predictive scheduling approach to determine bridge opening timetables and vessel passage times in a coordinated manner. The effectiveness of the strategy is demonstrated on a realistic case study based on the Rhine-Alpine corridor.**

I. INTRODUCTION

Freight transportation is an essential process within the supply chain, as it allows to transfer goods in an efficient manner and ensure their timely availability at the destination [1]. While several different transport modes may be used, inland waterborne transport (IWT) emerges as a cost-effective and environmentally-friendly alternative to move large amounts of cargo [2]. Despite the advantages it offers, IWT only represented 4% of the total goods transported in the EU-28 in 2016 [3]. This is mainly due to the fact that reliability of operations is negatively impacted by inaccurate information at service level and high system congestion, and thus calls for communication among interested parties [4].

IWT encompasses the simultaneous operation of vessels and infrastructure in cramped areas, which are moreover characterized by conflicting operational objectives, thus rendering IWT a challenging problem. While inadequate solutions may lead to suboptimal navigation and infrastructure utilization, the consideration of vessel-to-vessel (V2V), infrastructure-to-infrastructure (I2I) and vessel-to-infrastructure (V2I) communication in an agent-based framework has the potential to yield improved solutions, as intentions from one agent can be anticipated by others and prepared for in advance.

Zooming in on V2I communication, the most common pieces of infrastructure encountered in waterway networks are bridges—movable and fixed—and locks. Vessels must pass through infrastructure on their way towards destination, which they aim to do with minimal waiting times. Lock scheduling has been widely studied, considering both single-chamber [5], [6] and multiple-chamber [7], [8] serial lock configurations. Surprisingly enough, scheduling of movable bridges has not received the same degree of attention despite the fact that their operation has simultaneous implications for road, railway and waterborne transport, being [9] one of the few papers on the topic. However, bridges are considered to be characterized by predefined opening regimes, and therefore the approach does not utilize peak and valley passage demand to adjust openings.

This paper extends the preliminary work carried out in [9], where the operation timetables were fixed, by considering dynamic bridge operation. In other words, bridges can operate on demand and thus adapt their opening regimes to vessel passage needs, which are provided via V2I communication. Moreover, a control-oriented model of the process is designed, together with propositional logic expressions that govern system behavior. These are translated into (in)equality constraints and integrated into the design of a model predictive scheduling strategy, which determines passage times for every vessel-bridge pair and communicates these plans to each of the vessels. Moreover, the resulting opening schedules are shared with the bridges.

The rest of the paper is organized as follows: the dynamic bridge opening scheduling problem is described in Section II, and an approach to solve the problem is presented in Section III. A case study based on the Rhine-Alpine corridor serves to test the effectiveness of the approach in Section IV, allowing to draw conclusions and establish future research avenues in Section V.

II. PROBLEM STATEMENT

The dynamic bridge opening scheduling problem can be formulated as follows. A set of vessels $\mathcal{V}$ must pass a set of movable bridges $\mathcal{B}$ while sailing from origin to destination, with $|\mathcal{B}| = n$ and $|\mathcal{V}| = m$. A discrete problem setting is adopted, and thus time is divided into a set of time steps $\mathcal{K}$ of equal length. Furthermore:

- Bridge i, $i \in \{1, ..., n\}$, is characterized by its nominal width, $b^{(i)}$ [m], the maximum number of consecutive time steps it can stay open (so as to limit traffic disruption on bridge deck), $N_{up}^{(i)}$, and the minimum number of consecutive time steps it must remain closed immediately after an open-close switch, $N_{down}^{(i)}$, $\forall i \in \mathcal{B}$.

This work was supported in part by the Researchlab Autonomous Shipping (RAS) of Delft University of Technology, in part by the project ”Novel inland waterway transport concepts for moving freight effectively (NOVIMOVE)” (this project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 858508), and in part by the Spanish State Research Agency (AEI) and the European Regional Development Fund (ERFD) through the project SaCoAV (ref. MINECO PID2020-114244RB-I00).

[1]Department of Maritime and Transport Technology, Delft University of Technology, Delft, the Netherlands (e-mail: {p.segoviacastillo, v.reppa}@tudelft.nl).

[2]Advanced Control Systems Group, Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain (e-mail: vicenc.puig@upc.edu).

N_{up} and N_{down} can also be referred to as maximum up-time and minimum down-time, respectively. Bridges are numbered such that $i=1$ and $i=n$ correspond to the first and last bridge to be passed through, respectively.
- Vessel j, $j \in \{1,...,m\}$, is characterized by its width, $v^{(j)}$ [m], which includes the safety distance between vessel j and the rest of vessels, and its voyage plans, represented by earliest and optimal passage instants through bridge i, $\tau_e^{(i,j)}, \tau_o^{(i,j)} \in \mathbb{Z}_+$, respectively, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}$, with $\mathbb{Z}_+$ the set of positive integers. Earliest and optimal passage instants can be computed considering the maximum speed and the speed that minimizes fuel consumption, respectively, and inter-bridge distances, and are known before the start of vessel journeys.

The assumptions of the problem are listed below:

- Vessels may enter the system at any time instant $k \in \mathcal{K}$. Once they have been scheduled through all bridges, they are no longer taken into consideration.
- Clearance under bridges (measured from water surface to bridge underside) is not sufficient for vessels to sail below bridges while these are closed.
- Multiple vessels may pass a bridge simultaneously provided that their combined width does not exceed the nominal width of the bridge.
- Choice of time step size is sufficiently large for vessel i to pass through bridge j in one time step with zero dwell time, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}$. This also allows to consider that bridges are either open or closed.

The objective of this paper is to compute a set of scheduling decisions $u_k^{(i,j)} \in \{0,1\}$, which are defined as follows:

$$u_k^{(i,j)} = \begin{cases} 1 & \text{if vessel } j \text{ is scheduled to} \\ & \text{pass bridge } i \text{ at instant } k, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Binary decisions $u_k^{(i,j)}$, which can also be referred to as manipulated variables or control inputs, appear naturally in scheduling problems of different nature, e.g., microgrid operation [10], production plant scheduling [11] and material allocation [12]. The objective of the dynamic bridge opening scheduling problem is to determine $u_k^{(i,j)}$ to satisfy optimal vessel passage times as much as possible. These decisions are then communicated to vessels, and are also used to elaborate bridge opening schedules, which are provided to bridges. Vessels and bridges are assumed to abide by the scheduling decisions. Note that decisions that deviate from optimal passage times may require vessels to adjust voyage settings to comply with the solution, but this is out of the scope of the paper.

III. Proposed approach

The proposed solution to the dynamic bridge opening scheduling problem consists of two parts. A control-oriented model of the IWT problem in the presence of movable bridges is derived first. Then, the original scheduling problem is recast as an optimization-based control problem, and makes use of the control-oriented model to determine optimal passage decisions.

A. Control-oriented model

The width occupancy evolution of bridge i can be described using the following discrete-time equation, $\forall i \in \mathcal{B}$:

$$x_{k+1}^{(i)} = x_k^{(i)} + \underbrace{\sum_{j \in \mathcal{V}_k^{(i)}} v^{(j)} u_k^{(i,j)}}_{\text{current resource booking}} - \underbrace{\sum_{j \in \mathcal{V}_{k-1}^{(i)}} v^{(j)} u_{k-1}^{(i,j)}}_{\text{delayed resource release}}, \quad (2)$$

where $x_k^{(i)} \in \mathbb{R}$ [m] represents the width occupancy of bridge i at time instant k, $\forall i \in \mathcal{B}, \forall k \in \mathcal{K}$. This occupancy can be defined as the amount of bridge width that is utilized by vessels to sail through at each time instant. Moreover, $v^{(j)}$ was defined as the width of vessel j in Section II.

Equation (2) can be viewed as a width occupancy balance, whereby the current occupancy of bridge i, i.e., $x_k^{(i)}$, increases at the next time instant, i.e, $x_{k+1}^{(i)}$, as a result of decisions $u_k^{(i,j)}=1, \forall j \in \mathcal{V}_k^{(i)}$. However, the last assumption in Section II stated that vessel passage through bridges is done in a single time step. Therefore, inclusion of delayed control actions $u_{k-1}^{(i,j)}$, which were determined at the previous time instant $k-1$, accounts for resource release to reset bridge width occupancy.

The set of vessels $\mathcal{V}$ was originally defined as *static*, whereas $\mathcal{V}_k$ and $\mathcal{V}_{k-1}$ in Eq. (2) evince a *dynamic* nature. As vessels may enter and leave the system at any time instant $k \in \mathcal{K}$, the set of vessels to be scheduled is time-varying and is therefore denoted as $\mathcal{V}_k$, with $|\mathcal{V}_k| = m_k$. Moreover, $\mathcal{V}_k$ can be decomposed into non-overlapping subsets $\mathcal{V}_k^{(i)}$ such that $\mathcal{V}_k = \bigcup_{i=1}^{n} \mathcal{V}_k^{(i)}$, with $\mathcal{V}_k^{(i)} \triangleq \left\{ j : z_k^{(i,j)} = 1 \right\}, \forall i \in \mathcal{B}, \forall k \in \mathcal{K}$.

In order to track the vessel position in a qualitative manner, the term $z_k^{(i,j)}$ is introduced to indicate the next bridge to be passed by each vessel, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}$, and is defined as follows:

$$z_k^{(i,j)} = \begin{cases} 1 & \text{if bridge } i \text{ is the next bridge en} \\ & \text{route for vessel } j \text{ at instant } k, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Although $u_k^{(i,j)}$ and $z_k^{(i,j)}$ might appear somewhat similar, $z_k^{(i,j)} = 1$ indicates that vessel j *can* pass through bridge i at time instant k, while $u_k^{(i,j)} = 1$ indicates that vessel j *does* pass through bridge i at time instant k, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}$.

To capture bridge-vessel passage operations in more detail, two additional variables $\omega_k^{(j)}$ and $s_k^{(i)}$ are introduced. On the one hand, $\omega_k^{(j)} \in \{0,1\}$ denotes whether vessel j has been scheduled through the last bridge before reaching the destination, $\forall j \in \mathcal{V}_k$, and is defined as follows:

$$\omega_k^{(j)} = \begin{cases} 1 & \text{if vessel } j \text{ has been scheduled} \\ & \text{through last bridge at instant } k, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

On the other hand, $s_k^{(i)} \in \{0, 1\}$ indicates whether bridge i is open or closed at time instant k, and is defined as follows:

$$s_k^{(i)} = \begin{cases} 1 & \text{if bridge } i \text{ is open at instant } k, \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

Variable $s_k^{(i)}$ is introduced to simplify the design of certain constraints, but is in fact linked to $u_k^{(i,j)}$. Although this is discussed later on in this section, it is convenient to note here that bridge i should only be open if and only if at least one vessel is scheduled through bridge i at that time instant.

It should be apparent at this point that the scheduling problem should be designed in such way that logical incompatibilities are forbidden. To clarify this, suppose that $z_k^{(i,j)} = 1$ for a certain bridge i and vessel j at time instant k. Then, the scheduling problem should be endowed with a mechanism that necessarily sets $u_k^{(l,j)}$ equal to 0 for $l \neq i$. Note also that $u_k^{(i,j)}$ may or may not be set equal to 1 depending on other factors, e.g., $\tau_e^{(i,j)}$, $\tau_o^{(i,j)}$ and other operational constraints provided hereunder.

Logic rules involving the variables defined in Eqs. (1), (3)–(5) can be described by means of linear equations and (in)equalities [13]. Several propositional logic expressions that characterize the correct operation of the system are identified below for the dynamic bridge opening scheduling problem. Then, the systematic approach detailed in [14, Eqs. (5)–(8)] allows to transform a logic expression into its equivalent conjunctive normal form. Conversion of the resulting conjunction of clauses into linear (in)equalities is then straightforward, see [14, Table 1]. Then, the following logic rules can be stated for all bridges $i \in \mathcal{B}$, vessels $j \in \mathcal{V}_k$ and time instants $k \in \mathcal{K}$:

- If bridge i is not the next bridge en route for vessel j at time instant k, then vessel j cannot be scheduled through bridge i at time instant k. This can be formally stated as: $\left(z_k^{(i,j)} = 0\right) \rightarrow \left(u_k^{(i,j)} = 0\right)$. The equivalent constraint is

$$z_k^{(i,j)} - u_k^{(i,j)} \geq 0, \ \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (6)$$

- At time instant k, vessel j either has a single bridge immediately en route or has already been assigned to all bridges. This can be formally stated as: $\left(\sum_{i=1}^{n} z_k^{(i,j)}\right) \oplus \omega^{(j)}$, where $\oplus$ denotes the logical XOR operation. The equivalent constraint is

$$\sum_{i=1}^{n} z_k^{(i,j)} + \omega_k^{(j)} = 1, \ \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (7)$$

- If bridge i is the next bridge en route for vessel j at time instant k and vessel j is not scheduled through bridge i at time instant k, then bridge i will be the next bridge en route for vessel j at time instant $k+1$. This can be formally stated as: $\left(\left(z_k^{(i,j)} = 1\right) \wedge \left(u_k^{(i,j)} = 0\right)\right) \rightarrow \left(z_{k+1}^{(i,j)} = 1\right)$. The equivalent constraint is

$$-z_k^{(i,j)} + u_k^{(i,j)} + z_{k+1}^{(i,j)} \geq 0, \ \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (8)$$

- If vessel j is scheduled through bridge i at time instant k and bridge i is not the last bridge, then bridge $i+1$ will be the next bridge en route at time instant $k+1$. This can be formally stated as: $\left(u_k^{(i,j)} = 1\right) \rightarrow \left(z_{k+1}^{(i+1,j)} = 1\right)$. The equivalent constraint is

$$z_{k+1}^{(i+1,j)} - u_k^{(i,j)} \geq 0, \ \forall i \in \mathcal{B} \setminus \{n\}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (9)$$

- If vessel j is scheduled through bridge i at time instant k and bridge i is the last bridge, then vessel j has been completely scheduled at time instant $k+1$. This can be formally stated as: $\left(u_k^{(i,j)} = 1\right) \rightarrow \left(\omega_{k+1}^{(j)} = 1\right)$. The equivalent constraint is

$$\omega_{k+1}^{(j)} - u_k^{(i,j)} \geq 0, \ i = n, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (10)$$

- If earliest passage time of vessel j through bridge i is greater than time instant k, then vessel j cannot be scheduled through bridge i at time instant k. This can be formally stated as: $\left(k \leq \tau_e^{(i,j)} - 1\right) \rightarrow \left(u_k^{(i,j)} = 0\right)$. The equivalent constraint is

$$k \geq u_k^{(i,j)} \left(\tau_e^{(i,j)} - 1\right) + 1, \ \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}. \quad (11)$$

- Bridge i should only be open at time instant k if and only if at least one vessel is scheduled through bridge i at time instant k. This can be formally stated as: $\left(s_k^{(i)} = 1\right) \leftrightarrow \left(\sum_{j \in \mathcal{V}_k} u_k^{(i,j)} \geq 1\right)$. The equivalent constraint is

$$s_k^{(i)} \leq \sum_{j=1}^{m_k} u_k^{(i,j)} \leq m_k s_k^{(i)}, \ \forall i \in \mathcal{B}, \forall k \in \mathcal{K}, \quad (12)$$

  and m_k is the number of vessels to be scheduled at time instant k.
- If bridge i was open at instant $k-1$ and closes at instant k, then bridge i must remain closed during at least $N_{down}^{(i)}$ consecutive time instants. This can be formally stated as: $\left(s_{k-1}^{(i)} - s_k^{(i)} = 1\right) \rightarrow \left(s_l^{(i)} = 0\right)$. The equivalent constraint is

$$s_{k-1}^{(i)} - s_k^{(i)} \leq 1 - s_l^{(i)}, \ \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}, \quad (13)$$

  and $l = k, ..., \min\left(k + N_{down}^{(i)} - 1, T\right)$, where T is the scheduling horizon. In a receding horizon control approach such as the one considered in Section III-B, T equals the prediction horizon, denoted as H_p.

Equations (11) and (12) are the result of implications between a variable and an inequality. This requires to introduce a tolerance ε and a lower (upper) bound c (C): ε can be set equal to 1 should the coefficients and variables be integers [15, p. 170], and c (C) can be computed as the lower (upper) inequality bound [15, p. 171].

In addition to the previous constraints, the following physical and operational constraints must also be observed:

- Maximum bridge width capacity must be respected:

$$0 \leq x_k^{(i)} \leq b^{(i)},\ \forall i \in \mathcal{B}, \forall k \in \mathcal{K}. \tag{14}$$

- Bridge i can remain open during at most $N_{up}^{(i)}$ consecutive time instants:

$$\sum_{l=k}^{\min\left(k+N_{up}^{(i)},T\right)} s_l^{(i)} \leq N_{up}^{(i)},\ \forall i \in \mathcal{B}, \forall k \in \mathcal{K}. \tag{15}$$

B. Scheduling strategy: design and implementation

The scheduling strategy is designed as an optimization-based control problem. Therefore, an appropriate performance function is required so that its value can be optimized while fulfilling constraints (2), (6)–(15), yielding optimal scheduling decisions.

Scheduling error minimization is the operational objective considered in this work, and can be defined as the sum of differences between optimal passage times and scheduling decisions. The quadratic error is chosen to be penalized in this paper, which can be mathematically expressed as

$$J_k = \sum_{i=1}^{n} \sum_{j=1}^{m_k} \left(k u_k^{(i,j)} - \tau_o^{(i,j)} \right)^2,\ \forall k \in \mathcal{K}. \tag{16}$$

Given the fact that $u_k^{(i,j)}$ is dimensionless, it cannot be directly compared to $\tau_o^{(i,j)}$, which has discrete time units. Therefore, $u_k^{(i,j)}$ is multiplied by the discrete time instant k. Then, J_k is minimized when vessel j is scheduled through bridge i at time instant $k = \tau_o^{(i,j)}$, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}$.

The model predictive scheduling problem can then be formulated as

$$\min_{\left\{u_{l|k}^{(i,j)}\right\}_{l=k}^{k+H_p-1}} J\left(u_{l|k}^{(i,j)}\right) \tag{17}$$

subject to

$$\begin{aligned} &\text{constraints (2), (6)–(15)}, && \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}, \\ &x_{k|k}^{(i)} = x_k^{(i)}, && \forall i \in \mathcal{B}, \\ &z_{k|k}^{(i,j)} = z_k^{(i,j)}, && \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \\ &\omega_{k|k}^{(j)} = \omega_k^{(j)}, && \forall j \in \mathcal{V}_k, \end{aligned}$$

with $\left\{u_{l|k}^{(i,j)}\right\}_{l=k}^{k+H_p-1} \triangleq \left\{u_{k|k}^{(i,j)}, u_{k+1|k}^{(i,j)}, \cdots, u_{k+H_p-1|k}^{(i,j)}\right\}$, where k, l and $k+l|k$ represent the current time instant, the time instant along the prediction horizon, and the predicted value of the variable at instant $k+l$ using information available at instant k, respectively. According to the receding horizon philosophy, only $u_{k|k}^{(i,j)}$ is applied to the system. Problem (17) is solved again at the next time instant to utilize updated information, thus transforming the original open-loop approach into a closed-loop one [16].

Algorithm 1 sketches the main implementation details to solve the dynamic bridge opening scheduling problem. Problem initialization is such that all bridges are assumed to be completely available, and all vessels must initially pass through the first bridge. As mentioned before, the number of vessels to be scheduled varies over time. Therefore, a new problem must be created at every time instant for the vessels present in the system. This process is repeated until all vessels have been scheduled through all bridges and there are no new vessels to be scheduled. Execution of Algorithm 1 concludes when this condition is met.

Algorithm 1 Model predictive scheduling implementation

Input: $b^{(i)}, N_{up}^{(i)}, N_{down}^{(i)}, v^{(j)}, \tau_e^{(i,j)}, \tau_o^{(i,j)}, \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k$
Output: $u_k^{(i,j)},\ \forall i \in \mathcal{B}, \forall j \in \mathcal{V}_k, \forall k \in \mathcal{K}$

1: Set $k = 1$ and define $x_1^{(i)} = 0$, $z_1^{(1,j)} = 1$ and $\omega_1^{(j)} = 0$, $\forall i \in \mathcal{B}, \forall j \in \mathcal{V}_1$
2: **while** $m_k > 0$ **do**
3: Design and solve problem (17) considering $\mathcal{V}_k$
4: Extract $u_{k|k}^{(i,j)}$ and determine $x_{k+1}^{(i,j)}$, $z_{k+1}^{(i,j)}$, $\omega_{k+1}^{(j)}$ and $s_k^{(i)}$ using Eqs. (2), (6)–(15)
5: **if** $\omega_{k+1}^{(j)} = 1$ **then**
6: Vessel j has been scheduled through last bridge: delete from the list
7: **else**
8: Vessel j has not been scheduled through last bridge: keep in the list
9: **end if**
10: $k \leftarrow k + 1$
11: Add vessels entering the system at time instant $k+1$ to the list of vessels and initialize as in Step 1
12: Define $\mathcal{V}_{k+1}^{(i)}$ using the result of Steps 8 and 11
13: **end while**

IV. Case Study

The case study presented in [9] is used to test the scheduling approach presented in Section III. The waterway is described first, together with the main features of vessels and bridges. Then, the scheduling solution is discussed.

A. System description

The Rhine-Alpine corridor connects major economic centers such as Brussels and Antwerp, the Randstad region, the Rhine-Ruhr and Rhine-Neckar regions, and Milan and Genoa. It constitutes one of the busiest European freight routes, joining the Rotterdam and Antwerp ports to the Mediterranean basin. Furthermore, its throughput represents 19% of EU's total GDP [17].

The Beneden Merwede is a river stretch within the Rhine-Alpine corridor that runs between Dordrecht and Hardinxveld-Giessendam (the Netherlands). A schematic representation is provided in Figure 1. Data regarding road and railway movable bridges are provided in Table I. Given the small inter-bridge distance between the first and second bridge, these are scheduled as a single bridge, and thus the results will be identical.

Fifty vessels sail from Dordrecht to Hardinxveld-Giessendam, passing through the four bridges during nav-

TABLE I

MOVABLE BRIDGES IN THE BENEDEN MERWEDE

Bridge (number and name)	Width [m]	Maximum up-time [min]	Minimum down-time [min]	Approx. distance from previous bridge [m]
(1) Traffic bridge Dordrecht	44	10	15	–
(2) Railway bridge Grotebrug	44	10	15	50
(3) Traffic bridge Papendrecht	30	10	10	4500
(4) Railway bridge Baanhoek	30	15	5	2500

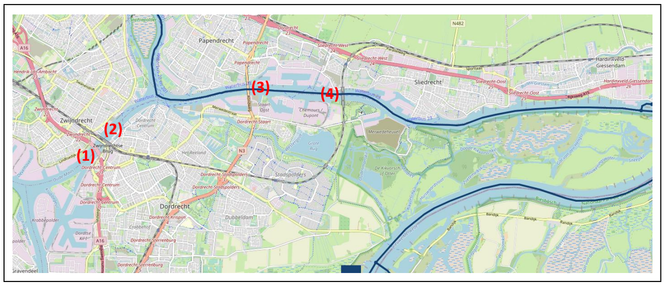

Fig. 1. Schematic representation of the Beneden Merwede (source: https://vaarweginformatie.nl/)

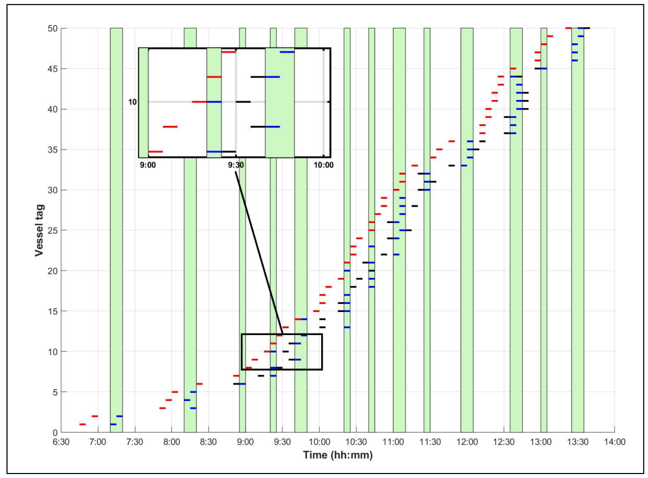

Fig. 2. First and second bridges: $\tau_e^{(i,j)}$ (red), $\tau_o^{(i,j)}$ (black), $u_k^{(i,j)}$ (blue) and opening slots (green vertical bars)

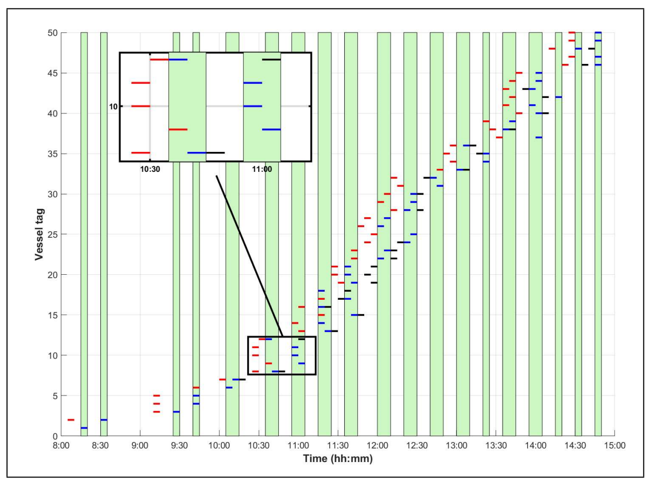

Fig. 3. Third bridge: $\tau_e^{(i,j)}$ (red), $\tau_o^{(i,j)}$ (black), $u_k^{(i,j)}$ (blue) and opening slots (green vertical bars)

igation. Values of vessel widths are aligned with the CEMT class of the waterway, and earliest and optimal bridge passage times are generated according to inter-bridge distances.

B. Results

Passage times of the fifty vessels through the four bridges are determined by applying Algorithm 1. Results are obtained in Matlab R2020b using Gurobi Optimization 9.1.2 and YALMIP [18]. A time step size of five minutes and a prediction horizon $H_p = 1$ hour are selected. Although discrete times are denoted with integers, these values are translated into corresponding five-minute time intervals to simplify result visualization and analysis.

Figures 2, 3 and 4 depict the scheduling results for the first and second bridges, third, and fourth bridge, respectively. Note that information provided to vessels and the corresponding bridge is shown in the same figure. On the one hand, vertical green bars represent time slots during which bridges are open. On the other hand, earliest, optimal and scheduled vessel passage times are depicted as red, black and blue horizontal bars, respectively, and their width equals one time step, i.e., five minutes. In the event that the scheduled passage matches optimal vessel plans, the overlap is resolved by plotting the scheduled passage time.

Analysis of the results shows that vessels are scheduled as close to optimal passage times as possible while guaranteeing constraint fulfillment. All vessels are scheduled after their earliest passage times. No vessel is scheduled outside bridge opening timetables, and bridges are only open during the time steps vessels pass bridges. Maximum up-times and minimum down-times specified in Table I are respected, which leads to uneven bar widths in contrast to [9]. Maximum bridge width occupancy is respected, as shown in Figure 5. Furthermore, a delay of one sample between scheduling decisions and width occupancy of bridges can be noticed upon inspection of Figures 2–5, in accordance with Eq. (2).

A quantitative result analysis is carried out on the basis of the following key performance indicators (KPIs): percentage of vessels scheduled at their optimal passage time and relative bridge width occupancy during opening (minimum, maximum and average). The values are summarized in Table II. On the one hand, it is interesting to note that satisfaction of optimal passage plans for the largest bridges, i.e., bridges 1 and 2, are the lowest. This can be explained—at least partially—by the fact that these two bridges are characterized by the strictest maximum up-times and minimum down-times. On the other hand, relative bridge width occupancy shows both that no bridge is overcapacitated and that vessel

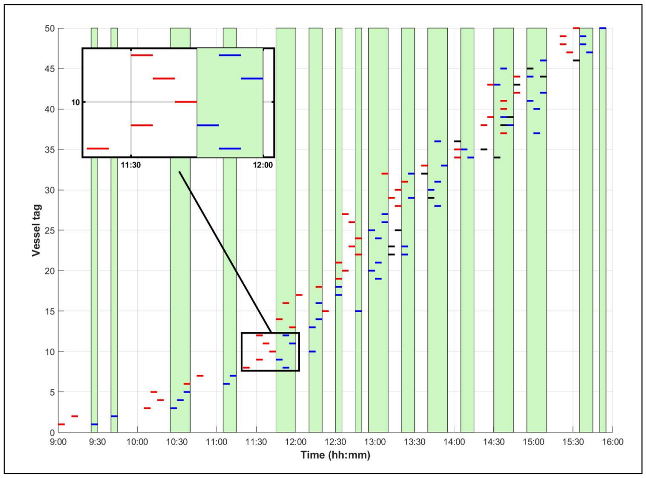

Fig. 4. Fourth bridge: $\tau_e^{(i,j)}$ (red), $\tau_o^{(i,j)}$ (black), $u_k^{(i,j)}$ (blue) and opening slots (green vertical bars)

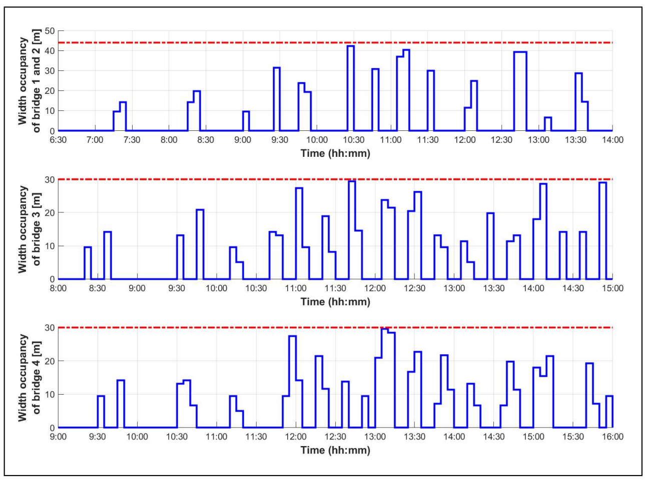

Fig. 5. Width occupancy of all bridges

passage is carried out in a similar manner for all bridges.

V. Conclusions and Future Research

This paper presented the design of a model predictive scheduling strategy to coordinate inland vessel navigation and movable bridge operation to render waterborne transport more competitive. A control-oriented model of the problem was formulated, paying special attention to logic expressions that govern system behavior. A systematic approach to convert the expressions to mathematical (in)equalities was employed, and the resulting model was used to create a predictive scheduling strategy that determined vessel passage times and bridge operation timetables ensuring coordination.

Several research avenues can be explored on the basis of the results presented in this paper. On the one hand, vessel voyage plans are characterized by a certain degree of uncertainty, which may be aggravated by the presence of vessels that do not perform V2I communication, e.g., recreational boats. Robust and stochastic control approaches will be considered to mitigate the uncertainty. On the other hand, operational objectives from the standpoint of bridges will be included in the cost function, and the use of Pareto optimization will be explored to determine satisfactory trade-off solutions.

TABLE II

Values of key performance indicators (KPIs)

KPI		Value (in percentage)		
		Bridges 1 and 2	Bridge 3	Bridge 4
Percentage of vessels scheduled at their optimal passage time		34	48	72
Relative bridge width occupancy	Minimum	15	16.83	16.83
	Maximum	96.14	98	98.83
	Average	55.41	49.25	49.16

References

[1] T. G. Crainic, "Long-haul freight transportation," *Handbook of transportation science*, pp. 451–516, 2003.

[2] B. Ji, X. Yuan, Y. Yuan, X. Lei, T. Fernando, and H. H. Iu, "Exact and heuristic methods for optimizing lock-quay system in inland waterway," *European Journal of Operational Research*, vol. 277, no. 2, pp. 740–755, 2019.

[3] European Commission and Directorate-General for Mobility and Transport, *EU transport in figures: statistical pocketbook 2018*. Publications Office, 2018.

[4] P. Shobayo and E. van Hassel, "Container barge congestion and handling in large seaports: a theoretical agent-based modeling approach," *Journal of Shipping and Trade*, vol. 4, no. 1, Jun. 2019.

[5] W. Passchyn, D. Briskorn, and F. C. Spieksma, "Mathematical programming models for lock scheduling with an emission objective," *European Journal of Operational Research*, vol. 248, no. 3, pp. 802–814, 2016.

[6] P. Segovia, M. Pesselse, T. Van Den Boom, and V. Reppa, "Scheduling inland waterway transport vessels and locks using a switching max-plus-linear systems approach," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 748–762, 2022.

[7] B. Ji, X. Yuan, and Y. Yuan, "A hybrid intelligent approach for co-scheduling of cascaded locks with multiple chambers," *IEEE Transactions on Cybernetics*, vol. 49, no. 4, pp. 1236–1248, 2019.

[8] B. Ji, D. Zhang, S. S. Yu, and C. Kang, "Mathematical programming models for scheduling multiple cascaded waterway locks," *Computers & Industrial Engineering*, vol. 156, p. 107289, 2021.

[9] P. Segovia, R. R. Negenborn, and V. Reppa, "Vessel passage scheduling through cascaded bridges using mixed-integer programming," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 248–253, 2022, 18th IFAC Workshop on Control Applications of Optimization CAO 2022.

[10] A. Parisio, E. Rikos, and L. Glielmo, "A model predictive control approach to microgrid operation optimization," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 5, pp. 1813–1827, 2014.

[11] A. Cataldo, A. Perizzato, and R. Scattolini, "Production scheduling of parallel machines with model predictive control," *Control Engineering Practice*, vol. 42, pp. 28–40, Sep. 2015.

[12] J. Xin, R. R. Negenborn, and T. van Vianen, "A hybrid dynamical approach for allocating materials in a dry bulk terminal," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1326–1336, 2018.

[13] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[14] R. Raman and I. E. Grossmann, "Relation between MILP modelling and logical inference for chemical process synthesis," *Computers & Chemical Engineering*, vol. 15, no. 2, pp. 73–84, 1991.

[15] H. P. Williams, *Model building in mathematical programming*. John Wiley & Sons, 2013.

[16] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer Science & Business Media, 2013.

[17] European Commission and Innovation and Networks Executive Agency, *CEF support to Rhine - Alpine Corridor*. Publications Office, 2018.

[18] J. Löfberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *IEEE International Symposium on Computer Aided Control Systems Design*, 2004.