Delft University of Technology

Cooperative multi-vessel systems in urban waterway networks

Chen, Linying; Huang, Yamin; Zheng, Huarong; Hopman, Hans; Negenborn, Rudy

Citation (APA)
Chen, L., Huang, Y., Zheng, H., Hopman, H., & Negenborn, R. (2020). Cooperative multi-vessel systems in urban waterway networks. *IEEE Transactions on Intelligent Transportation Systems*, *21*(8), 3294-3307. https://doi.org/10.1109/TITS.2019.2925536

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Cooperative Multi-Vessel Systems in Urban Waterway Networks

Linying Chen, Yamin Huang, Huarong Zheng, Hans Hopman, and Rudy Negenborn

Abstract—Urban waterways have great potential in cargo transport to relieve the congestion in the overloaded road networks. This paper explores the potential of applying cooperative multi-vessel systems (CMVSs) to improve the safety and efficiency of transport in urban waterway networks. A framework consisting of vessel train formation (VTF) and cooperative waterway intersection scheduling (CWIS) is proposed. Two types of controllers are introduced. Intersection controllers solve the CWIS problems and assign each vessel a desired time of arrival and vessel controllers are responsible for the VTF in waterway segments and the timely arrival at the intersections. An alternating direction method of multipliers (ADMM)-based negotiation framework is proposed for the cooperation among the controllers. The simulation experiments involving the scenarios in which up to 50 vessels sailing in the canal network in Amsterdam are carried out to illustrate the effectiveness of the proposed approach. In the simulation of an isolated intersection, rescheduling is triggered when some vessels cannot arrive on time. Although some ASVs arrive later, the time that is needed for all the ASVs to pass through is the same after rescheduling. Moreover, we compare the cooperative situation with the proposed CMVSs with a baseline situation. In the baseline situation, vessels avoid collisions using the generalized velocity obstacle (GVO) method and cross the intersection with a first in, first out rule. The CMVSs show better path following performance, while the GVO method needs fewer velocity changes. From the perspective of efficiency, the CMVSs help to reduce the total time to pass through the intersection.

Index Terms—Cooperative multi-vessel system, cooperative waterway intersection scheduling, waterway network, autonomous surface vessel, cooperative intelligent traffic system.

I. INTRODUCTION

IN DENSELY populated regions, like cities, road networks are often confronted with congestion and capacity problems. Many cities have considerable waterway resources, such as Amsterdam, Rotterdam, and Utrecht in The Netherlands, and cities in Jiangsu and Zhejiang Province, East China (Fig. 1). Waterway transport could offer an
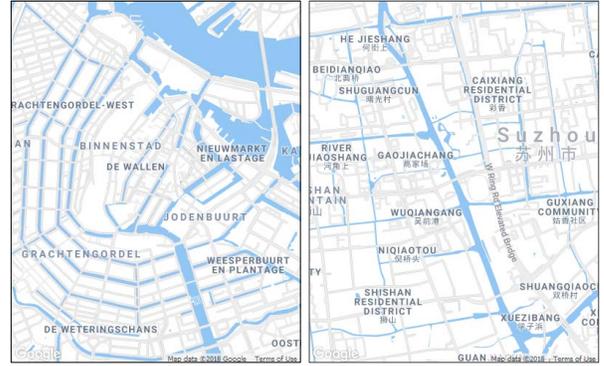
Fig. 1. Urban waterway networks in Amsterdam (left) and Jiangsu (right) [2].

environment-friendly alternative in terms of both energy consumption and noise emissions [1]. However, nowadays, the urban waterway networks are mostly used for leisure, tourism, and passenger public transportation. Urban waterways have great potential in cargo transport to relieve the congestion in the overloaded road networks.

The transport in urban waterway networks has the following characteristics. Firstly, the waterways are narrow and with low depth. Limited vertical clearance caused by non-removable bridges is also one of the characteristics. Secondly, the origins and destinations of the vessels are more dispersed compared with sea-going and inland shipping. Therefore, small dimension vessels are required for accessibility and flexibility. However, applying small vessels will increase the traffic density, which increases the trajectory conflicts between vessels. Moreover, as shown in Fig. 1, there are many intersections in waterway networks. Vessels in such networks have to frequently interact with vessels from different directions. Consequently, in waterway networks, cooperation among vessels becomes extremely important. Besides, applying autonomous vessels is seen as an innovation to improve the safety and efficiency of waterborne transport. The advantages, such as eliminating human error and better scheduling and control, make Autonomous Surface Vessels (ASVs) a good option to transport goods in urban areas. In this paper, we, therefore, explore the potential of applying fleets of cooperative ASVs.

We consider the situation in which cooperative ASV fleets sail in an urban waterway network to transport goods between specified origins and destinations. The cooperative ASVs are referred to as Cooperative Multi-Vessel Systems (CMVSs) [3]. We propose a framework consisting of Vessel Train Formation (VTF) and Cooperative Waterway Intersection Scheduling (CWIS) for the control of CMVSs. CWIS is used to determine

the arrival time of the ASVs to reduce conflicts at intersections. VTF is used for safe navigation in straight waterway segments, while helping the vessels arrive at a certain point at the desired time in sequence. Preliminary results of intersection crossing were presented in [4].

ASVs have received significant attention in recent years. Many works have been done for the control of ASVs, see [5]–[7]. However, those works mostly focus on an individual vessel. Few studies focused on the cooperation of ASVs, especially for the application of transport in waterway networks.

Regarding the similarity of vessels and vehicles, existing studies on cooperative driving of vehicles, such as platooning [8], can provide valuable references for the study of vessel coordination. Nevertheless, those methods cannot directly be applied to control vessels. Firstly, the main focus of cooperative driving has been on longitudinal control. However, in practice, steering (lateral control) is regarded as the ordinary practice of seamen. Secondly, sideway speed and Coriolis force are not considered when controlling vehicles, while those are important factors when controlling vessels [9]. Thirdly, the movement of vessels is significantly affected by the external environment, such as wind, wave, and current. Finally, waterways have intersections where the vessel trains need to interact with other vessels, often other vessel trains. Therefore, the cooperation of vessels at intersections needs to be emphasized. However, research on cooperation of vehicle platoons at interactions is still lacking.

This paper provides an extended version of [4] with more focus on waterway networks in which heterogeneous vessels and interconnected intersections are considered. Firstly, the maneuverability of vessels is considered. The vessels with different maneuverability can cooperate with each other by communicating their intentions. Secondly, in [4], the space blocks of an intersection are determined by the number of lanes of the connecting waterways. This may lead to a waste of space resource when the intersection is large. Therefore, in this paper, we introduced conflicting blocks to efficiently use space resources. Lastly, intersection scheduling is closed-loop. The feedback from vessels is an important input when the intersection controller making schedules. Furthermore, in this way, the interdependence of networked intersections can also be taken into account.

The remainder of this paper is organized as follows. Section II provides a broad literature review. Then, the problems that need to be solved and a negotiation framework for the cooperation of controllers in urban waterway networks are given in Section III. The CWIS problem and VTF problem are formulated and solved in Section IV and Section V, respectively. These two sections result in the cooperative control of vessels in urban waterway networks in Section VI. In Section VII, simulation experiments of vessels in an individual intersection and a canal network of Amsterdam are presented to assess the proposed framework.

II. LITERATURE REVIEW

In the literature, only a few studies focus on the vessels in waterway networks. In [10], the route choice behavior of vessels in an inland waterway network is investigated based on historical data. In [11], the authors carried out a study on the scheduling problems for locks in sequence, which shows the interdependence of infrastructures. However, to the best of our knowledge, no research has been carried out for the cooperative control of vessels in waterway networks. Nevertheless, the control of vessels in waterway networks can be divided into three parts, i.e., trajectory tracking, obstacle avoidance, and intersection crossing. Therefore, in this section, a brief review of the three control problems is presented.

A. Trajectory Tracking

When sailing in waterways, vessels usually follow geometric reference paths. Related topics are path following and trajectory tracking. The path following problem deals with the situation in which the reference path is independent of time, while the trajectory tracking control problem deals with the design of a controller that steers a vehicle to a time-parameterized reference path. Reviews on path following and trajectory tracking control can be found in [6], [12].

In existing research, widely used control design approaches fall into either one of the following methods or a combination thereof: Proportional-Integral-Derivative [13], Lyapunov-based control design [14], sliding mode control [15], Intelligent approaches, such as fuzzy logic, neural networks, and genetic algorithms [16], Optimization-based methods [17].

Among these methods, Model Predictive Control (MPC) gets much attention. The predictive property of MPC is beneficial as it enables the vessels detecting conflicts at an early stage, which alleviates the problems brought by the poor maneuverability of vessels. Moreover, MPC considers the latest available measurement of the state, which is particularly helpful to deal with uncertainties. Thus, an increasing number of researchers apply MPC for the control of vessels [18], [19].

B. Obstacle Avoidance

Many trajectory tracking methods also take obstacle avoidance into account. For example, optimization-based tracking controllers usually have obstacle avoidance constraints. Conventional obstacle avoidance methods, such as potential field [20] and velocity obstacles [21], usually do not consider cooperation between vessels. Vessels have to predict the actions that other vessels may take. A detailed review of collision avoidance technologies of ASVs that do not consider cooperation is provided in [5], [22].

Instead, in the methods for cooperative collision avoidance, vessels share their intentions. The actions of the involved vessels are determined by following a certain protocol like COLREGS [23]. However, rule-based methods usually suitable for encounter scenarios with a single vessel only. Encountering multiple vessels incorporates multiple rules, and to find the unique solution to the avoidance problem is difficult, if not impossible [5]. Another method to coordinate vessels is to achieve agreements through negotiation. According to the timing of information exchange, both serial [3] and parallel [24] iterative negotiation schemes have been proposed.
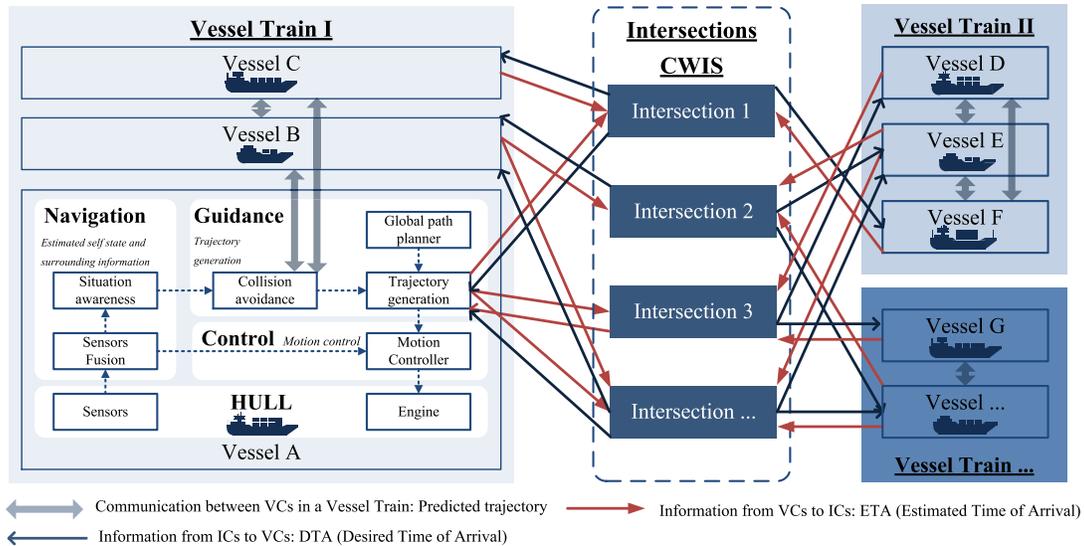
Fig. 2. Cooperative multi-vessel system in urban waterway networks.

C. Intersection Crossing

Few studies focuses on the problem of intersection crossing of vessels. In [4], the problem of letting vessels effectively pass through an intersection is formulated as a Job Shop Scheduling problem. Vessels passing through intersections is comparable to the situation of vehicles crossing non-signalized intersections. In the field of road transport, intersection crossing is one of the most challenging problems and attracts much attention. Related research can provide valuable references for the studies on intersection crossing of vessels.

An intersection is a shared resource that a limited number of vehicles want to utilize at the same time [25]. An intersection controller needs to solve a resource allocation problem to avoid conflicts. In the method cooperative resource reservation, the intersection is modeled as a collection of tiles. Vehicles need to reserve the tiles on their planned route for certain time slots and pass the intersection according to the reservation [26]. Another method is to modify the trajectories (velocity) to minimize overlap and evacuation time [27] or maximize the capacity [28]. A review of cooperative intersection management systems for road transport can be found in [25].

Existing methods are, however, for an isolated intersection. When looking into a transport network, the intersections are interconnected: an improvement of one intersection may lead to congestion at other intersections. At present, the research on cooperation of interconnected intersections is still lacking.

III. PROBLEM STATEMENT

In this paper, we focus on CMVSs in an urban waterway network. In this section, the assumptions are provided. Then, we propose a framework for the cooperative control of vessels in waterway networks. A negotiation framework is presented for the cooperation among multiple controllers.

A. Assumptions

The following assumptions are made throughout the paper:

1) A waterway network consists of waterway segments and intersections connecting the segments; other infrastructures, such as bridges and locks, are not considered;
2) All the vessels are autonomous, i.e., Autonomous Surface Vessels (ASVs), and their dynamics can be described by mathematical models;
3) All the controllers are able to communicate and cooperate with other controllers;
4) The communication is ideal: the bandwidth is sufficient, and there is no delay in communication among controllers;
5) The initial state of each ASV is feasible: the ASV is within navigable waters; there is no other vessels or obstacles within the stopping distance of the ASV.

The set of cooperative ASVs is referred to as a CMVS, a system in which vessels utilize Vessel-to-Vessel (V2V) and Vessel-to-Infrastructure (V2I) communication to negotiate and collaborate with each other for the aim of improving overall safety, efficiency, or for performing specific tasks [25].

B. Framework for the Cooperative Control of Vessels in Waterway Networks

A framework for the cooperative control of vessels in waterway networks is presented in Fig. 2. We introduce two types of controllers: a Vessel Controller (VC) for the control of an individual ASV, and an Intersection Controller (IC) for solving the conflicts of vessels at an intersection. A vessel controller uses sensors to get self-state information (e.g., position, speed, and heading), environmental information (e.g., wind speed and directions, current velocity) and information of obstacles. Based on the obtained information, the Navigation system creates pictures of the current situation and informs the Guidance system of collision risks. Combining with the predetermined global path, optimal trajectories with specified objectives and constraints can be determined. The commands are sent to actuators for autonomous navigation.

The cooperation of CMVS in the waterway network can be divided into two parts: segment sailing and intersection crossing. The ASVs in the same waterway segment form a vessel train by solving the VTF problem. They share the information about their predicted trajectories, which help them make better decisions on distance keeping with others and to benefit from sailing in groups at a closer distance. When approaching an intersection, VCs report their Estimated Time of Arrival (ETA) to the IC. Then, the IC makes conflict-free schedules and informs those vessels the Desired Time of Arrival (DTA) at the intersection by solving the Waterway Intersection Scheduling (WIS) problem. After passing through the intersection, vessels sailing in the same waterways then form new vessel trains for safe navigation. The communication and cooperation of vessels in different vessel trains are realized through ICs. Similarly, the ICs communicate and cooperate with each other by exchanging information with VCs.

C. Negotiation Framework

In a waterway network, the decisions a controller makes are influenced by the actions that other controllers take: a VC needs the information from other VCs to decide its collision avoidance actions, and it also needs the DTA from ICs to decide the reference trajectory; an IC is informed that the ASVs will arrive and their ETA when it makes schedules. All the controllers are closely connected. When a controller changes its schedule or trajectory, other controllers have to adjust theirs accordingly. To reach an agreement, a negotiation framework is needed.

In this cooperation problem, each controller makes decisions based on the information provided by other controllers. Therefore, an agreement is achieved when the actions each controller wants to take reach a consensus with the information it broadcasts. Thus, each controller has to handle its own objective and constraints, and the extra consensus and coupling constraints, i.e.,

Problem $\mathscr{A}$

$$\text{minimize } J\left(u_a\right) \tag{1}$$

subject to $\forall a, b \in \mathcal{N}, a \neq b$:

$$u_a \in \zeta_a, \tag{2}$$

$$u_a = z_a, \tag{3}$$

$$u_a \in \vartheta\left(h\left(x_a, u_a\right), g\left(x_b, z_b\right)\right), \tag{4}$$

where $J\left(u_a\right)$ and ζ_a indicate the objective and constraints of controller a; u_a and z_a are the control variable and broadcast variable, respectively; ϑ is the coupling constraint, which is related to the function $h\left(x_a, u_a\right)$ of the state and the control variables of controller a, and the function $g\left(x_b, z_b\right)$ of the state and the broadcast variables of coupling controller b.

In the literature, this cooperative problem can be solved with a non-iterative or iterative framework. Using the non-iterative methods, the controllers solve the optimization problem in a sequential order [29]. The controllers that perform computation later should calculate a solution according to the solutions that the other controllers computed earlier. Alternatively, in the iterative framework, the controllers obtain agreement through iterations. By exchanging and using the information about the controller's own decision and other controllers' preferences, the inputs should converge, and a set of actions for all controllers should be found. Thus, iterative frameworks have a larger potential to achieve overall optimal performance [30].

The Alternating Direction Method of Multipliers (ADMM) is one of the widely applied methods to solve the consensus problems iteratively [31]. In the basic framework introduced in [41], ADMM consists of three steps, minimization of the local variable, minimization of the interconnecting variable and a dual variable update. The updating of interconnecting variable are usually carried out by a coordinator who has the information of all controllers. In this paper, we propose a negotiation framework based on this basic framework, see Algorithm 1. The algorithm firstly solves the augmented Lagrangian of Problem $\mathscr{A}$. Then, the dual variables are updated to make the control variables and the broadcast variables converge. No coordinators are involved in this process. Each controller carries out the three steps itself according to the information others broadcast in each iteration.

This framework can be serial or parallel according to the timing of information exchange and computation. A comparison of parallel and serial control schemes has been presented in [32]. In the parallel scheme, all the controllers perform computations at the same time. Thus, when a controller performs computation at iteration s, it uses the information that the other controllers provide at the iteration $s-1$. An example of a parallel negotiation framework can be found in [33]. The scheme enjoys the advantage of parallel computation. However, because of the potential conflicts of objectives, the solutions may not converge. On the contrary, in the serial scheme, only one controller is performing computations at a time. The controller performs computation using the most up-to-date information of the controller broadcasts earlier during the same iteration. The serial scheme has preferable properties in terms of solution speed, by requiring fewer iterations, and solution quality. Details about the serial iterative scheme of the framework can be found in our earlier work [3].

In the following parts, the parallel framework is employed for the negotiation among ICs in the CWIS problem. Firstly, the sequence of intersections that a vessel passes through can be different. It is difficult to find out the right order of computation. Secondly, there are waterway segments connecting intersections, which can act as buffers to resolve the conflicts between the ICs. Thirdly, the number of ICs involved in CWIS is usually small. Therefore, conflicts are not serious. Moreover, in the case of non-convergence, if the ICs cannot make an agreement, a backup hierarchical architecture will be used: one of the ICs will work as a centralized controller to find the final solution. On the contrary, a serial framework is used for the negotiation among VCs in the VTF problem. Vessels in the same vessel train are within a close range. Moreover, due to limited navigable waters and the constraints on maneuverability, the VCs usually do not have many choices. Thus, the conflicts of their solutions are more serious than that of ICs. Making use of the most up-to-date information and the property of fast convergence make the serial scheme more suitable for the VTF problem.

Algorithm 1 ADMM-Based Negotiation Framework

1: **for** $s = 1 : S$ **do**
2:    **for** $a = 1 : N$ **do**
                    ▷ Each ASV solves a local problem
3:      $u_a^s := \underset{u_a}{\arg\min} \left(J_a(u_a) + (\lambda_a^{s-1})^T (u_a - z_a^{s-1}) + \right.$
         $\left. \rho_a/2 \left\| u_a - z_a^{s-1} \right\|_2^2 \right)$;
4:      **if** solution do not exist **then**
5:         $u_a^s := u_a^{s-1}$; $N_{\text{jump}} := N_{\text{jump}} + 1$;
6:      **end if**
             ▷ Update global variable and Lagrange multiplier
7:      $z_a^s := \varphi_a u_a^s + (1 - \varphi_a) z_a^{s-1} + \lambda_a^{s-1}/\rho_a$;
8:      $\lambda_a^s := \lambda_a^{s-1} + \rho_a \left(u_a - z_a^s \right)$;
9:      $ZX_a^s := h(x_a, z_a)$;
             ▷ Update primal and dual residual and tolerance
10:     $R_{\text{pri},a}^s := u_a^s - z_a^s$; $R_{\text{dual},a}^s := z_a^s - z_a^{s-1}$;
11:     $\varepsilon_{\text{pri},a}^s := \sqrt{Nn_u}\varepsilon^{\text{abs}} + \varepsilon^{\text{rel}} \max \left\{ \| u_a^s \|_2, \| z_a^s \|_2 \right\}$;
12:     $\varepsilon_{\text{dual},a}^s := \sqrt{Nn_u}\varepsilon^{\text{abs}} + \varepsilon^{\text{rel}} \| \lambda_a^s \|_2$;
                        ▷ Stopping check
13:     **if** $\left\| R_{\text{pri},a}^s \right\|_2 \leqslant \varepsilon_{pri,a}^s$ & $\left\| R_{\text{dual},a}^s \right\|_2 \leqslant \varepsilon_{dual,a}^s$ **then**
14:        $jdg := jdg + 1$;
15:     **end if**
                  ▷ Update the penalty parameter
16:     **if** $\left\| R_{\text{pri},a}^s \right\|_2 > 10 \left\| R_{\text{dual},a}^s \right\|_2$ **then**
17:        $\rho_a := 2\rho_a$;
18:     **else if** $\left\| R_{\text{dual},a}^s \right\|_2 > 10 \left\| R_{\text{pri},a}^s \right\|_2$ **then**
19:        $\rho_a := \rho_a / 2$;
20:     **end if**
             ▷ Send ZX_i^s, jdg and N_{jump} to others
21:    **end for**
22:    **if** $jdg = N$ and $N_{\text{jump}} = 0$ **then**
23:      Stop iteration;
24:    **end if**
25: **end for**

---

† Notation ρ_i is the penalty parameter; $\lambda_i(k)$ is the dual variable; $R_{\text{pri},a}$ and $R_{\text{dual},a}$ are primal and dual residual; $\varepsilon_{\text{pri},a}$ and $\varepsilon_{\text{dual},a}$ are primal and dual tolerance; $\cdot^s$ is the value of the corresponding variable at iteration s.

IV. VESSEL TRAIN FORMATION

In this section, a 3 Degree of Freedom (DOF) dynamic model of an ASV is introduced. An MPC controller is designed to control the ASV with a linearized prediction model. The VTF problem is then solved with the proposed negotiation framework.

A. Dynamic Model of an Individual Vessel

1) 3 DOF Dynamic Model of an ASV: We consider n heterogeneous ASVs, whose dynamics are described with the 3 DOF model proposed in [9], with varying parameter values:

$$
\begin{aligned}
\dot{x}_i &= f_i(x_i, \tau_i) \\
&= \begin{bmatrix} \mathbf{0}^{3\times3} & R_i(\psi_i) \\ \mathbf{0}^{3\times3} & M_i^{-1}(-C_i(v_i) - D_i) \end{bmatrix} x_i + \begin{bmatrix} \mathbf{0}^{3\times3} \\ M_i^{-1} \end{bmatrix} \tau_i.
\end{aligned} \quad (5)
$$

where $x_i = \begin{bmatrix} \eta_i^T & v_i^T \end{bmatrix}^T$ and τ_i the system state and input, respectively; $\eta_i = [p_i, q_i, \psi_i]^T$ are coordinates p_i, q_i, and heading angle ψ_i in the North-East-Down coordinate system; $v_i = [u_i, v_i, r_i]^T$ are surge and sway velocities u_i, v_i, and yaw rate r_i in Body-fixed reference frame; $\tau_i = \begin{bmatrix} \tau_{u_i}, \tau_{v_i}, \tau_{r_i} \end{bmatrix}^T$ are forces τ_{u_i}, τ_{v_i}, and moment τ_{r_i} in Body-fixed reference frame. M_i is the system inertia matrix, including rigid-body and added mass matrices, $M_i = M_{RB,i} + M_{A,i}$; C_i is the Coriolis-centripetal matrix, including rigid-body and added mass Coriolis-centripetal matrices, $C_i = C_{RB,i} + C_{A,i}$; D_i is the damping force. In this paper, we consider a linear damping force; $R(\psi_i)$ is a rotation matrix.

The dynamic model is discretized with a sample time T_s:

$$
x_i(k+1 \,|\, k) = x_i(k) + \int_{kT_s}^{(k+1)T_s} f_i(x_i(t), \tau_i(t)) dt. \quad (6)
$$

2) Linearized Prediction Model: MPC has attracted increasing interests [18], [34] in the field of waterborne transport. Besides, distributed MPC has been used for cooperative control of networked vehicles [35]. Research indicates that MPC has many advantages for the control of large-scale networked systems [30]. Therefore, we consider MPC as a suitable approach for the control of vessels in the CMVS.

The basic concept of MPC is to use a dynamic model to forecast system behavior and optimize the forecast to produce the best decision [36]. Therefore, at each time step, a prediction is needed. The dynamics described in (5) are, however, highly nonlinear. If this nonlinear model is directly used to design the MPC controller, the MPC online predictions and optimizations would be too time-consuming for real-time control. Therefore, the successively linearized model presented in [18] is adopted in this paper.

At each time step, the controller calculates a sequence of control inputs for the whole predict horizon and the first control sample will be implemented. In the next step, as a start point, the control sequence is shifted one sample with an extensive of zeros at the end. Using this extended control sequence as seed input $\tau^e(k \,|\, k)$, we can obtain the seed state $x^e(k+1 \,|\, k)$ with (5). By applying Taylor's theorem and neglecting the higher order terms, Equation (6) becomes

$$
\begin{aligned}
x(k+1 \,|\, k) = x^e(k+1 \,|\, k) + A^d(k \,|\, k) \left(x(k \,|\, k) - x^e(k \,|\, k) \right) \\
+ B^d(k \,|\, k) \left(\tau(k \,|\, k) - \tau^e(k \,|\, k) \right), \quad (7)
\end{aligned}
$$

where A^d and B^d are Jacobian matrices.

B. Formulation of the VTF Problem

The main function of vessels is to transport goods from one place to another. Therefore, vessels usually have predetermined origins, destinations, and paths. In order to exchange information and enjoy the benefits of sailing together, vessels in a CMVS attempt to stay close to each other. At the same time, vessels should not collide with others. Thus, in the VTF problem, the following three rules are applied:

1) Trajectory following: attempt to follow the predetermined paths;

2) Aggregation: attempt to stay close to nearby vessels;
3) Collision avoidance: avoid collisions with nearby vessels.

According to the three rules, the objective of a single vessel in a CMVS can be described as

$$J_i\left(\tau_i(k)\right) = \sum_{l=1}^{H_{\mathrm p}} \sum_{j \in N_i} \Big(\alpha \|\eta_i(k+l\,|k) - w_i(k+l)\|_2^2$$
$$+ \beta \|d_{ij|i}\left(k+l\,|k\right) + \delta_{ij}(k+l\,|k)\|_2^2$$
$$+ \gamma \|\tau_i(k+l-1\,|k)\|_2^2\Big) \qquad (8)$$

where the three parts in the equation represent trajectory following, aggregation and control efforts, respectively: α, β and γ are the weights; $H_{\mathrm p}$ is the length of the prediction horizon; l is the lth time step in the prediction horizon; $\eta_i(k+l\,|k)$ is the prediction made at k about the position and heading of vessel i at $k+l$ according to the linearized dynamic model (7); $w_i(k+l)$ is the reference at $k+l$, including trajectory and heading; $d_{ij|i}(k+l\,|k)$ is the distance between ASV i and ASV j calculated by ASV i, $d_{ij|i}(k+l\,|k) = \|y_i(k+l\,|k) - y_{j|i}(k+l\,|k)\|_\infty$; $y_i(k+l\,|k)$ is the prediction made at k about the position of vessel i at $k+l$, and $y_{j|i}$ is the position of j that i received; δ_{ij} is introduced for aggregation, $d_{ij|i}(k+l\,|k) \leqslant \Upsilon$, $-\Upsilon \leqslant \delta_{ij|i}(k+l\,|k) \leqslant \Upsilon$; Υ is the aggregation range, $\Upsilon = \min(\Upsilon_1, \Upsilon_2, \cdots, \Upsilon_n)$, Υ_i is the communication range of i; $\tau_i(k)$ indicates control input over the prediction horizon.

Therefore, when applying the negotiation framework presented in Algorithm 1 for the cooperation of vessels. The local problem that each ASV needs to solve is as follows:

Problem $\mathscr{B}$:

minimize $J_i\left(\tau_i(k)\right)$ \qquad (9)

subject to $\forall i \in \mathcal{V}, \forall j \in N_i, \forall l \in H_{\mathrm p}$:

$$v_{i,\min} \leqslant v_i(k+l\,|k) \leqslant v_{i,\max} \qquad (10)$$
$$\tau_{i,\min} \leqslant \tau_i(k+l\,|k) \leqslant \tau_{i,\max} \qquad (11)$$
$$d_{ij|i}(k+l\,|k) \geqslant d_{\mathrm{ij,safe}} \qquad (12)$$
$$y_i \in \Xi$$
$$\text{Dynamics described by (7),} \qquad (13)$$

where N_i is the set of neighbors of vessel i, $N_i = \left\{ j \in \mathcal{V} : \|y_j - y_i\|_2 \leqslant \Upsilon_i \right\}$; $v_{i,\min}$, $v_{i,\max}$ and $\tau_{i,\min}$, $\tau_{i,\max}$ are the constraints on states and control inputs; Ξ indicates navigable waters.

The interconnecting variables that link the control problems of different vessels are the predicted trajectories of the ASVs. Thus the information being exchanged, ZX_a^s in Algorithm 1 (Line 9) consists of the predicted trajectories determined with the control inputs the vessels calculated in each iteration and the nonlinear dynamic model (5).

By adjusting the weights of the three parts in objective functions and constraints, Problem $\mathscr{B}$ can also be used to describe the control problem of ASVs under following situations:

- Path Following: if the vessel is the only vessel in the waterway, the vessel has an only objective, path following;

Algorithm 2 Vessel Train Formation†

1: VC $i \in \mathcal{VT}_\iota$ determines the control input $\tau_i^s(k)$ by solving the Augmented Lagrange form of Problem $\mathscr{B}$ with $y_{j|i}^s = \begin{bmatrix} \mathbf{I}^{2\times 2} & \mathbf{0}^{2\times 4} \end{bmatrix} ZX_i^s(k)$:

$$\tau_i^s(k) = \arg\min_{\tau_i(k)} \left(J_i\left(\tau_i(k)\right) + \left(\lambda_i^{s-1}\right)^{\mathrm T}\left(\tau_i(k) - z_i^{s-1}(k)\right)\right.$$
$$\left. + \rho_i/2 \, \left\|\tau_i(k) - z_i^{s-1}(k)\right\|_2^2\right)$$

2: VC i updates the global variable $z_i^s(k)$, Lagrange multipliers $\lambda_i^s(k)$, primal residual $R_{\mathrm{pri},i}^s$ and dual residual $R_{\mathrm{dual},i}^s$;
3: VC i updates interconnecting variable $ZX_i^s(k)$ according to Equations (5), and send it to other VCs;
4: The next VC j repeats Step 1-3 until all the VCs finish computation;
5: Each VC moves on to the next iteration $s+1$ and repeat Step 1-4 until the stopping criteria is met.

$\dagger$ Details about the VTF problem are addressed in [3].

- VTF: if more than one vessel is sailing in the waterway, both aggregation rule and collision avoidance constraint should be considered;
- Intersection Crossing: if more than one vessels are passing through an intersection, collision avoidance constraint is considered while the aggregation rule is ignored.

To summarize, the VTF control of vessels in a vessel train $\mathcal{VT}_\iota$ at each time step k consists of the steps in Algorithm 2.

V. COOPERATIVE WATERWAY INTERSECTION SCHEDULING

As mentioned, the scheduling of intersection crossings is, in fact, a resource allocation problem. In this section, an intersection is modeled with conflicting blocks. The problem of scheduling the order of the ASVs passing through an intersection is formulated. When looking into the waterway networks, the cooperation between intersection controllers is achieved through iterative negotiations.

A. Intersection Modeling

A vessel passing through the intersection along the path can be regarded as occupying space resources for a certain period. Fig. 3 gives an example of paths in an intersection. Two relations of overlapping paths are crossing and merging. Therefore, there are three types of conflicting blocks: the blocks in which paths cross each other, the blocks in which paths merge into one, and the blocks in which both crossing and merging occur.

B. Scheduling for an Isolated Intersection

One method to avoid conflicts is to set a rule that during the time slot that one vessel occupies a block, other vessels cannot enter the block. In this way, the WIS problem can be formulated as a job shop scheduling problem, in which several jobs need to be processed by a number of machines in a given order. The aim is to minimize the makespan, i.e., the
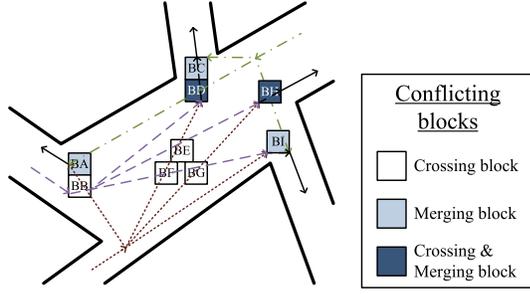
Fig. 3. Conflicting blocks at an intersection.

time required to let all vessels pass through the intersection, under the following conditions:

- Sequential constraint: a vessel passes through the blocks in a predetermined sequence;
- No-wait constraint: a vessel does not stop when it leaves a block;
- Disjunctive constraint: other vessels cannot enter a block until the vessel occupying the block has left it.

Therefore, the WIS problem can be formulated as follows:

Problem $\mathscr{C}$:

minimize $T_{\max}$ (14)

subject to $\forall i, j \in \mathcal{V}, i \neq j, \forall m, n \in \mathcal{B}, n = m + 1$:

$$T_{\max} \geqslant s_{im} + t_{im} \tag{15}$$

$$s_{im} \geqslant Ea_i; \tag{16}$$

$$\frac{d_{im}}{v_{i,\max}} \leqslant t_{im} \leqslant \frac{d_{im}}{v_{i,\min}} \tag{17}$$

$$s_{in} = s_{im} + t_{im} + T_{i,m \to n} \tag{18}$$

$$\frac{d_{i,m \to n}}{v_{\max}} \leqslant T_{i,m \to n} \leqslant \frac{d_{i,m \to n}}{v_{\min}} \tag{19}$$

$$s_{jm} \geqslant s_{im} + t_{im} \text{ OR } s_{im} \geqslant s_{jm} + t_{ja} \tag{20}$$

$$s_{jm} \geqslant s_{im} + t_{i,\text{safe}} \text{ OR } s_{im} \geqslant s_{jm} + t_{j,\text{safe}} \tag{21}$$

$$s_{jm} + t_{jm} \geqslant s_{im} + t_{im} + t_{j,\text{safe}} \text{ OR}$$
$$s_{im} + t_{im} \geqslant s_{jm} + t_{jm} + t_{i,\text{safe}}, \tag{22}$$

where $\mathcal{V}$ is the set of vessels that will pass through the intersection within a certain period; $\mathcal{B}$ is the set of conflicting blocks; block n is the block next to block m. In (14), $T_{\max}$ is the makespan, i.e., the total time needed for all vessels to pass through the intersection. Therefore, it is larger or equal to the passing time of each vessel at each block, i.e., the sum of the arrival time of vessel i at block m (s_{im}) and the time vessel i need to pass through block m (t_{im}) in (15). Equation (16) represents that, for each vessel i, there is a earliest arrival time Ea_i; t_{ia} is determined by (17), where d_{im} is the length of the path that vessel i need to pass through block m. Equation (18) is for the sequential and no-wait constraint. $T_{i,m \to n}$ is the time needed from block m to n, which also relates to the distance between block m and n ($d_{i,m \to n}$) and velocity limitations ($v_{i,\max}$ and $v_{i,\min}$), see (19). Equation (20) represents the disjunctive constraint. Equation (21) represents that the interval between the arrival time of the vessels at the same block should larger than a

predefined safe time interval $t_{i,\text{safe}}$; the same constraint holds for the situation when vessels leave the blocks, i.e. (22). $t_{i,\text{safe}}$ is calculated by safe distance $d_{ij,\text{safe}}$ and the velocity of the vessel, i.e., $t_{i,\text{safe}} = \dfrac{d_{ij,\text{safe}}}{d_{ia}/t_{ia}}$.

Job shop scheduling problems are usually formulated as Mixed Integer Programming (MIP) problems [37]. A small-size MIP problem can be solved within a reasonable amount of time. Thus, our WIS problem is formulated as an MIP problem with the constraints (20), (21) and (22) replaced by the following constraints:

$$\begin{cases} s_{im} + t_{im} \leqslant s_{jm} + \kappa(1 - \chi_{ij,m}) \\ s_{jm} + t_{jm} \leqslant s_{im} + \kappa \chi_{ij,m} \end{cases} \tag{23}$$

$$\begin{cases} s_{im} + t_{i,\text{safe}} \leqslant s_{jm} + \kappa(1 - \chi_{ij,m}) \\ s_{jm} + t_{j,\text{safe}} \leqslant s_{im} + \kappa \chi_{ij,m} \end{cases} \tag{24}$$

$$\begin{cases} s_{im} + t_{im} + t_{j,\text{safe}} \leqslant s_{jm} + t_{jm} + \kappa(1 - \chi_{ij,m}) \\ s_{jm} + t_{jm} + t_{i,\text{safe}} \leqslant s_{im} + t_{im} + \kappa \chi_{ij,m}, \end{cases} \tag{25}$$

where κ is an arbitrarily large number, $\kappa \gg \sum_{i \in \mathcal{V}} \sum_{m \in \mathcal{B}} t_{im}$, and $\chi_{ij,m}$ is a binary variable,

$$\chi_{ij,m} = \begin{cases} 1, & \text{if vessel } i \text{ passes block } m \text{ before } j, \\ 0, & \text{otherwise.} \end{cases} \tag{26}$$

C. Cooperation Among Interconnected Intersections

The coupling variables connecting the WIS problems of the intersection in a waterway network are the earliest arrival times of the vessels at the intersections. When a vessel has to pass through a sequence of intersections, the schedule that the IC make have impacts on the earliest arrival time at the subsequent intersection. The segments connecting the intersections can provide buffers where vessels can accelerate or decelerate to arrive at the DTA at the intersections.

In the CWIS problem for the intersections in a waterway network, the negotiation framework proposed in Section VI is used to obtain agreements among the ICs regarding coupling variables. The objective and constraints of each IC are formulated in Problem $\mathscr{C}$. The information being exchanged, ZX_a^s in Line 9 in Algorithm 1, consists of the earliest arrival time, which can be calculate as

$$\forall i \in \mathcal{V}, \forall p, q \in \mathcal{C}_i :$$
$$ETA_{i|q} = DTA_{i|p} + T_{i|p} + \frac{d_{i,p \to q}}{\widehat{v}_i} \tag{27}$$
$$Ea_{i|q} = ETA_{i|q} - BT_{i,p \to q}, \tag{28}$$

where $\mathcal{C}_i$ is the sequence of intersections that vessel i has to pass through; p and q are two adjacent intersections in the sequence, and vessel i passes through intersection p earlier than intersection q; $ETA_{i|q}$ is the ETA of vessel i at intersection q, it is the arrival time if vessel i keeps its planed velocity; $DTA_{i|p}$ is the DTA of vessel i that IC of intersection p calculated; $T_{i|p}$ is the total travel time of vessel i passing through intersection p; $d_{i,p \to q}$ is the distance from p to q; $\widehat{v}_i$ is the planned velocity of vessel i used to calculate initial reference path; $BT_{i,p \to q}$ is the buffer

---

Algorithm 3 Cooperative Waterway Intersection Scheduling

1: Each IC p determines the control input $u_p^s(k)$ by solving the Augmented Lagrange form of Problem $\mathscr{C}$ with $Ea_{i|q}^s = ZX_p^{s-1}(k)$:

$$u_p^s(k) = \underset{u_p(k)}{\arg\min} \quad (T_{p,\max}\left(u_p(k)\right)$$
$$+ (\lambda_p^{s-1})^T \left(u_p(k) - z_p^{s-1}(k)\right)$$
$$+ \rho_p/2 \left\| u_p(k) - z_p^{s-1}(k) \right\|_2^2)$$

where u_p is the control input of intersection p, $u_p = \left[[s_{11}\cdots s_{im}]^T, [t_{11}\cdots t_{im}]^T\right]^T$, $\forall m \in \mathcal{B}_p$, $\forall i \in \mathcal{V}_p$, $\mathcal{B}_p$ is the conflicting blocks in p, and $\mathcal{V}_p$ is the set of vessels passing p;

2: IC p update the global variable $z_p^s(k)$, Lagrange multipliers $\lambda_p^s(k)$, primal residual $R_{\mathrm{pri},p}^s$ and dual residual $R_{\mathrm{dual},p}^s$;

3: IC p update the interconnecting variable $ZX_p^s(k)$ according to (27) and (28), and send it to other ICs;

4: After all the ICs finish computation, move on to the next iteration $s + 1$ and repeat Step 1-3 until the stopping criteria are met.

---

time that the vessel accelerates ($BT_{i,p\to q} > 0$) or decelerates ($BT_{i,p\to q} < 0$).

To sum up, at each time step k, CWIS consists of the steps in Algorithm 3.

VI. CMVS IN URBAN WATERWAY NETWORKS

CWIS is from the perspective of ICs, while VTF is from the perspective of VCs. Assembling the two parts, the cooperation of ICs and VCs can be realized with Algorithm 4.

VII. SIMULATION EXPERIMENTS

In this section, simulation experiments are carried out to illustrate the potential of the proposed approach. We firstly consider a situation in which some vessels cannot meet the DTA and rescheduling is triggered. Then, a simulation of CMVSs crossing an intersection are presented to illustrate how WIS helps to improve the efficiency of waterborne transport. The results are compared with a baseline scenario in which vessels avoid collision using a revised version of Generalized Velocity Obstacle (GVO) that is proposed in [21], and pass through the intersection based on the First In First Out (FIFO) rule. In the end, simulation results of CMVSs in a canal network of Amsterdam are provided.

A. Setup

1) ASVs: Two model vessels are used in the experiments, Delfia 1* and CyberShip 2. Delfia 1* is an ASV proto-type developed by TU Delft. Its shape is designed to make maneuvering applications in crowded environments easier than actual solutions allowing at the same time the possibility to cooperate with multiple ASVs. Delfia 1* has two 360° steering propellers, one at the bow and the other at the stern. CyberShip 2 is a scale replica of a supply ship [38]. It is

---

Algorithm 4 CMVS in Waterway Networks

1: ICs carry out CWIS to determine the DTA of the vessels at each intersection;

2: VCs generate the reference w_i. The reference trajectory y_i are calculated according to the DTA with a double integrator dyanamics: $y_i(k + 1) = y_i(k) + \hat{v_i}(k)$. The reference heading is determined according to y_i, and the changes between heading are within the range[$-\pi$, π];

3: In each time step k, for each vessel train $\mathcal{VT}_l$: (a)
   1) if there is no vessel, no actions need to be taken;
   2) if there is one vessel, the VC control the ASV for the aim of path following;
   3) if there is more than one vessel, the VCs set $\beta = 1$ if $\mathcal{VT}_l$ is in segments, and set $\beta = 0$ if $\mathcal{VT}_l$ is in intersections, then the VCs control the ASVs for VTF;

4: Each VC updates the state of the ASV with (5), and send the earliest arrival time to the ICs;

5: ICs check if the earliest arrival time of each vessel meets its DTA, if not, ICs carry out CWIS and inform the VCs the new DTA;

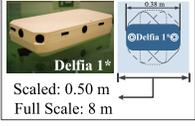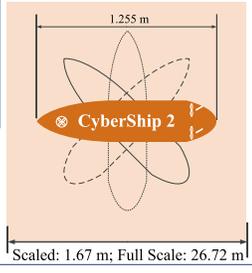6: ICs and VCs repeat Step 2-5 until all the vessels arrive their destinations.

---

fully actuated with two main propellers and two rudders aft, and one bow thruster. The models are scaled-up according to Froude scaling law with a scaling factor 1 : 16: the multiplication factors for length, force, moment and time are 16, 16^3, 16^4, and $\sqrt{16}$, respectively. The dimensions and control constraints of the two models are provided in Table I. The hydrodynamic parameters of the two model vessels are in Table II.

Each ASV is controlled by a MPC controller. The predictive horizon is $H_p = 7$. The weights in the objective function of the VTF problem in (8) are $\alpha = diag[10, 10, 30]$, $\beta = \begin{cases} 0 \text{ for VTF} \\ 1 \text{ for intersection crossing,} \end{cases}$ and $\gamma = 3$. In the VTF control, iterative updating sequence is adopt [3]. The absolute tolerance and relative tolerance in Algorithm 1 are $\varepsilon^{\mathrm{abs}} = 10^{-3}$ and $\varepsilon^{\mathrm{rel}} = 10^{-3}$. In CWIS, the velocity range is $\left[0.9\hat{v}_i, \hat{v}_i\right]$.

2) Research Area: A part of the canal network in Amsterdam is selected as the research area, see Fig. 4. There are four intersections in this network, including a general intersection (Intersection A), a large intersection (Intersection B), a dispersed intersection (Intersection C) and a small intersection (Intersection D). The conflicting blocks in each intersection are also provided in the figure.

For each waterway segment, a buffer zone is set to adjust the reference trajectories of the ASVs. The navigable waters are defined by the boundary of the waterways, which are described by straight lines. The safety distance between the boundary and an ASV is the width of the ASV. Some segments in the network are one-way, which are set as wide enough for the ASVs to overtake others.

TABLE I
INFORMATION ABOUT DELFIA 1* AND CYBERSHIP 2

	Delfia 1*		CyberShip 2	
	Scaled model	Full scale (1 : 16)	Scaled model	Full scale (1 : 16)
Width [m]	0.185	2.96	0.29	4.64
Length [m]	0.38	6.08	1.255	20.08
Safety distance[a]				
$\nu_{\max}$ [m/s; m/s; rad/s]	$\begin{bmatrix} 0.75 \\ 0.75 \\ 20\pi/180 \end{bmatrix}$	$\begin{bmatrix} 3 \\ 3 \\ 20\pi/180 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 15\pi/180 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \\ 15\pi/180 \end{bmatrix}$
$\nu_{\min}$ [m/s; m/s; rad/s]	$\begin{bmatrix} 0 \\ -0.75 \\ -20\pi/180 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -3 \\ -20\pi/180 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -1 \\ -15\pi/180 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -4 \\ -15\pi/180 \end{bmatrix}$
$\widehat{v_i}$ [m/s]	0.6	2.4	0.8	3.2
$\tau_{\max}$ / $-\tau_{\min}$ [N; N; Nm]	$\begin{bmatrix} 6 \\ 6 \\ 3 \end{bmatrix}$	$\begin{bmatrix} 24576 \\ 24576 \\ 196608 \end{bmatrix}$	$\begin{bmatrix} 6 \\ 6 \\ 3 \end{bmatrix}$	$\begin{bmatrix} 24576 \\ 24576 \\ 196608 \end{bmatrix}$

[a] When two ASVs encountered, the safety distance between ASV i and ASV j is $d_{ij,safe} = (d_{i,safe} + d_{j,safe})/2$

TABLE II
PARAMETERS FOR DELFIA 1* AND CYBERSHIP 2[a]

	Delfia 1*	CyberShip 2[b]		Delfia 1*	CyberShip 2
m	3.345	23.80	N_v	0.50439	0.1052
I_z	0.031	1.760	N_r	-0.22243	-1.900
x_g	0.0	0.046	$X_{\dot{u}}$	-0.2310	-2.0
X_u	-2.734	-0.7225	$Y_{\dot{v}}$	-1.334	-10.0
Y_v	-4.60250	-0.8612	$Y_{\dot{r}}$	0.0	0.0
Y_r	0.79546	-0.1079	$N_{\dot{r}}$	-0.110	-1.0

[a] The hydrodynamic derivatives follow the notations in [39];
[b] data from [38].

TABLE III
ORIGINS AND DESTINATIONS FOR THE ASVs

ASV No.		Destination			
		2	3	4	5
Origin	1	[18;26;8]	[14;6;1]	[25;7;9]	[4;19;22]
	2	-	[27;2;11]	[28;20;29]	[13;10;17]
	4	[3;30;21]	[23;24;5]	-	[15;12;16]

B. Intersection Crossing

In this part, simulation experiments of an intersection (Intersection B in Fig. 4) are carried out to illustrate the WIS problem. 9 conflicting blocks are set for the intersection. Firstly, we consider a delay scenario in which some ASVs can not arrive on time and the rescheduling is triggered. Then, we carry out a comparison between the scenario in which CMVSs cross intersection B and a baseline simulation. In the baseline simulation, cooperation is not considered and
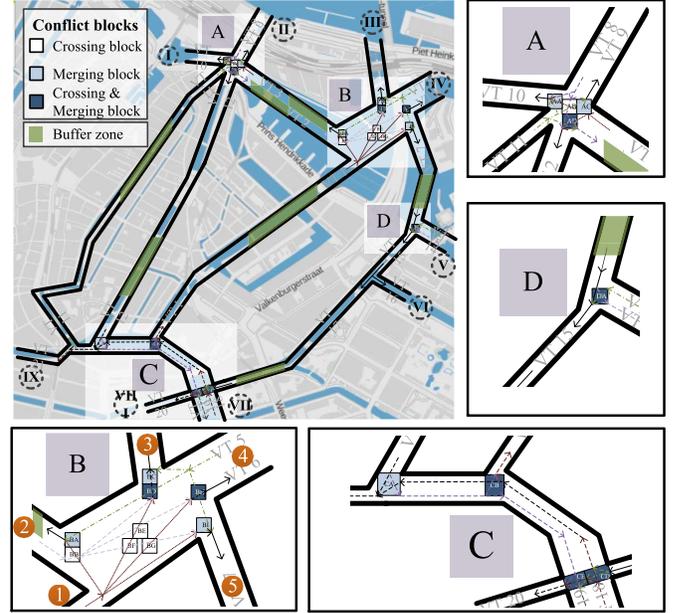

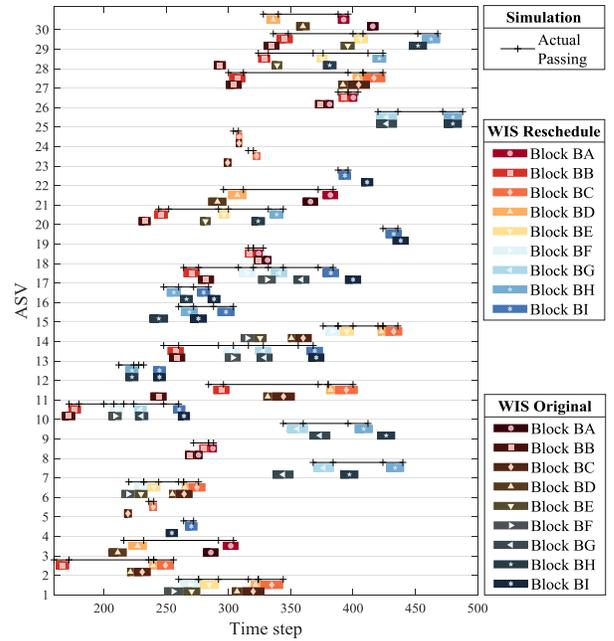Fig. 4. Waterway network in simulation (Map is from [40]).

Fig. 5. WIS results and actual passing time in the simulation.

the ASVs avoid collisions using the GVO method proposed in [21] and the FIFO rule for intersection crossing. GVO algorithms have been employed to various vehicles for collision prevention, such as wheeled robots, unmanned aerial vehicles, and ASVs. A brief introduction of the GVO method and related setting in the simulation is given in Appendix A.

1) Delay Scenario: We simulate the situation that 30 ASVs passing through Intersection B. The origins and destinations of the ASVs are given in Table III. The ASVs with odd numbers have the same setting with Delfia 1*, while the ASVs with even numbers have the same setting with CyberShip 2.
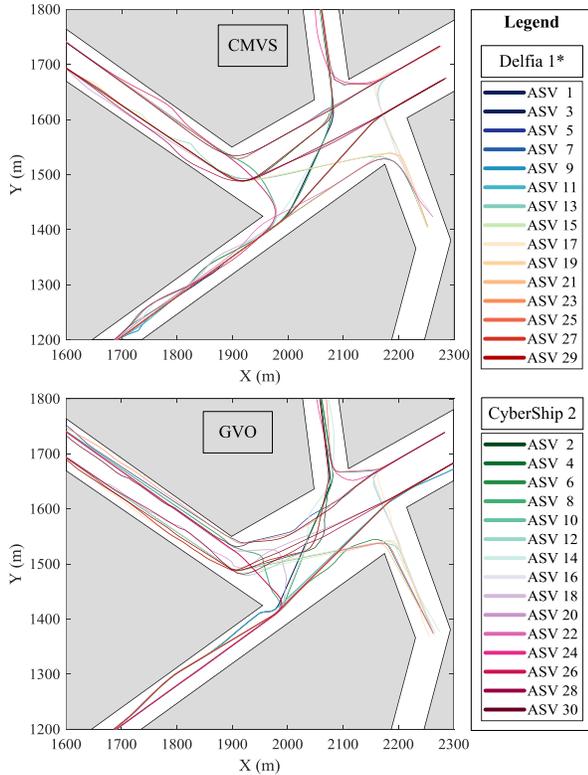
Fig. 6.   Comparison of trajectories (up: CMVSs; down: GVO method).

The ASVs from the same origin are sorted from numbers low to high and set off with a time interval of 16s. The first ASVs from Origin 1, 2 and 3 start at $k = 0$, and the first ASV from Origin 4 start at $k = 120$. Then, ASV 7, 11, 14 and 28 arrive at the intersection 60 s later than their initial ETA.

The original and rescheduling WIS results are shown in Fig. 5. The changes of the ETA of ASV 7, 11, 14 and 28 not only change the DTA of themselves but also other ASVs. The rescheduling that the IC carries out contributes to improve the utilization rate of the intersection. Although some ASVs arrive later, the time that is needed for all the ASVs to pass through is the same: the last ASV leaves at 488 s. The actual passing time of each ASV is given, as well. Mostly, the ASVs can arrive at a position at the desired time. However, there are still small differences between the actual passing time and the scheduled time. The main reason is that WIS is continuous while the VTF and simulation are discrete.

2) Comparison With the GVO Method: Fig. 6 – Fig. 8 show the comparison of the simulation results of CMVS and GVO. In Fig. 6, ASVs in CMVSs overtake others at the segments to change their orders in the vessel trains. Then, the ASVs pass through the intersection smoothly. On the contrary, ASVs using the GVO method take collision avoidance actions both in the segments and in the intersection. The collision avoidance actions also lead to larger deviations from the predetermined path, especially when the ASVs are crossing the intersection, see Fig. 7. In the figure, path following error refers to the distance between the position of an ASV and the straight line joining two adjacent waypoints. Fig. 7 also provides a
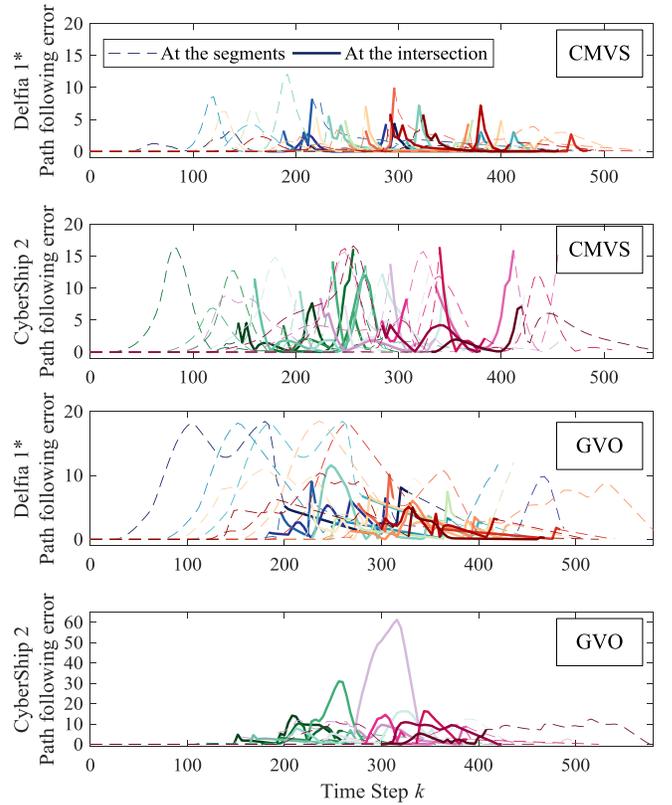

Fig. 7.   Comparison of path following errors (up: CMVSs; down: GVO method). For the legend, see Fig 6.

comparison of tracking performance of Delfia 1* and CyberShip 2. With either CMVS or GVO, Delfia 1* have smaller path following errors.

Fig. 8 shows the linear and angular velocities of each vessel. For a better trajectory tracking performance, vessels in CMVSs adjust their velocities more frequently. Moreover, as the GVO method aims at keeping current velocity, the changes in velocities are smaller. However, adding yaw rate makes the choice set smaller and sometimes VCs cannot find a solution using GVO. Therefore, the constraint on the yaw rate is not considered in the simulation. Thus, at some time step, the yaw rate larger than the limitation. Moreover, as GVO uses target velocity as the control input, the constraints on force and moment are not considered, either.

Table IV provides the intersection passing time and total travel time of each vessel. In general, the efficiency is improved for both individual vessel and the waterway network when applying CMVSs. For most of the vessels, sailing in CMVSs not only saves time to pass through the intersection, but also shorten the total travel time. Some ASVs spend the same time to pass through the intersection, but they have a shorter total travel time, such as ASV 3 and ASV 27. Some ASVs make sacrifices, such as ASV 7, ASV 9, ASV 15 and ASV 22.

C. CMVS in a Waterway Network

In the experiment, we simulate the situation that 50 ASVs sailing in the waterway network shown in Fig. 4. The origin and destination of each ASV are provided in Table V.
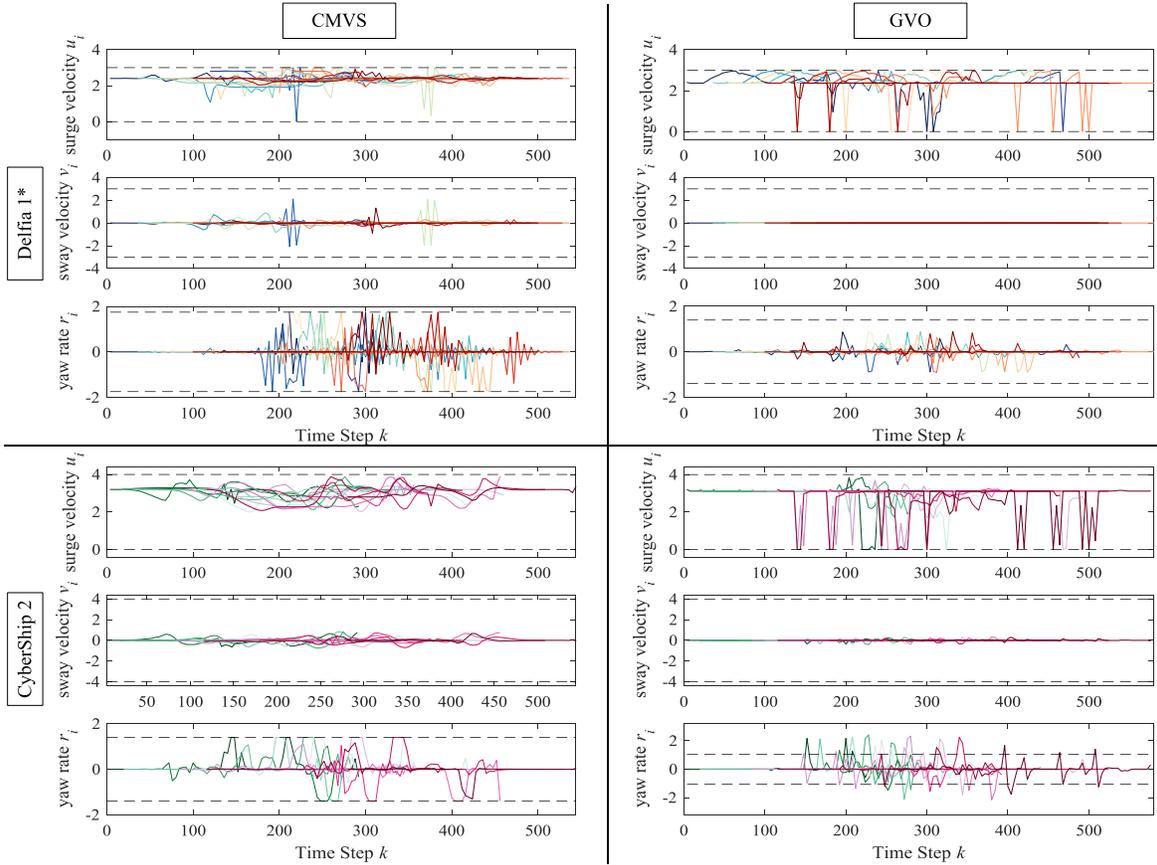
Fig. 8. Linear and angular velocities of the ASVs using the proposed method and the GVO method. For the legend, see Fig 6.

The four ICs achieve an agreement after 11 iterations. Fig. 9 shows the differences between the earliest arrival time and DTA of each vessel through iterations. Considering the buffer time that the segments can provide, the DTA of each ASV at each intersection is later than its earliest arrival time when the iteration stops, i.e., $DTA - ETA - BT \geqslant 0$.

Fig. 10 provides the trajectory of each ASV and screen-shots at certain time steps. After passing through an intersection, the ASVs form new vessel trains, see subfigure (a). The ASVs overtake others to change their orders in a vessel train to meet the DTA of next intersection, see subfigure (b). The WIS helps the ASVs to efficiently use the space between two adjacent ASV. For example, in subfigure (c), a Delfia 1* merges into the flow using the gap between two vessels. Due to the speed difference, most of the time, vessels prefer to form vessel train with the ASVs that have the same dynamics, such as vessels in subfigure (d) and (e).

VIII. CONCLUSION

A. Conclusions

In this paper, we explore the potential of applying fleets of cooperative ASVs to improve the safety and efficiency of transport in urban waterway networks. We propose a framework consisting of Vessel Train Formation (VTF) and Cooperative Waterway Intersection Scheduling (CWIS) for the cooperative control of ASVs in waterway networks. Two types of controllers are introduced. Intersection Controllers (ICs)
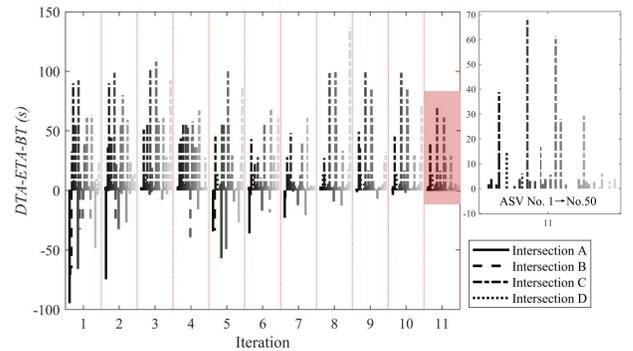

Fig. 9. Difference between DTA and ETA in each iteration.

solve CWIS problems and assign each vessel the desired time of arrival. Vessel Controllers (VCs) control the vessels in the same waterway segment to form a vessel train. The agreements of VCs and ICs are achieved with an ADMM-based negotiation framework. Simulation experiments of vessels sailing in the canal network in Amsterdam are carried out to illustrate the effectiveness of the proposed framework.

In the simulation of an individual intersection, rescheduling is triggered when some vessels cannot arrive on time. The rescheduling contributes to using time and space resources efficiently. Consequently, the total time that is needed for all the vessels to pass through the intersection does not increase. A comparison of the proposed approach and a GVO method is provided. The results show that: the proposed method has

TABLE IV
COMPARISON OF THE TRAVEL TIME OF THE TWO METHODS

No.	CMVS		GVO		Difference	
	Intersection [s]	Total [s]	Intersection [s]	Total [s]	Intersection [s]	Total [s]
ASV 1	116	408	128	424	-12	-16
ASV 2	88	292	120	340	-32	-48
ASV 3	116	340	124	368	-8	-28
ASV 4	80	284	84	296	-4	-12
ASV 5	28	168	36	188	-8	-20
ASV 6	84	300	92	324	-8	-24
ASV 7	116	392	124	372	-8	20
ASV 8	44	336	56	356	-12	-20
ASV 9	120	388	124	376	-4	12
ASV 10	108	292	112	312	-4	-20
ASV 11	116	400	124	416	-8	-16
ASV 12	40	140	48	152	-8	-12
ASV 13	124	380	132	388	-8	-8
ASV 14	80	328	104	336	-24	-8
ASV 15	60	176	56	176	4	0
ASV 16	44	140	52	164	-8	-24
ASV 17	132	396	136	424	-4	-28
ASV 18	40	344	72	384	-32	-40
ASV 19	100	368	104	376	-4	-8
ASV 20	100	276	108	344	-8	-68
ASV 21	116	340	132	384	-16	-44
ASV 22	76	312	84	296	-8	16
ASV 23	28	168	36	184	-8	-16
ASV 24	24	140	40	160	-16	-20
ASV 25	116	368	124	380	-8	-12
ASV 26	40	328	52	348	-12	-20
ASV 27	116	392	120	416	-4	-24
ASV 28	96	296	116	352	-20	-56
ASV 29	132	368	140	396	-8	-28
ASV 30	88	268	108	328	-20	-60
Average	85.6	304.27	96.27	325.33	-10.67	-21.07
Makespan	340	544	348	588	-8	-44

[a] Difference=CMVS-GVO.

TABLE V
ORIGIN AND DESTINATION OF EACH ASV IN A WATERWAY NETWORK

	Origin	II	IV	V	VII
Destination	I	32	[42;28]	-	[44;15;1]
	II	-	[17;41]	-	[36;23;2]
	III	[40;22]	47	-	[4;18]
	IV	[34;35]	-	-	[24;39]
	V	6	14	-	13
	VI	3	[46;8]	49	-
	VII	[16;11;30]	[5;48]	27	-
	VIII	[45;33]	[29;21]	[50;26]	9
	IX	[7;38]	[25;37;31]	[43;19]	[20;10;12]

Fig. 10.   Trajectories of the ASVs in an urban waterway network.

better path following performance; the GVO method has fewer velocity changes; CMVSs helps to reduce the makespan and total travel time. In the end, a simulation of vessels sailing in the canal network in Amsterdam is presented to show the cooperation among ICs.

B. Future Research

Further research will consider the following directions:

Firstly, in this paper, a successive linearized dynamics model is used for trajectory prediction. However, errors between the linearized and nonlinear dynamic models may lead to violations of safety constraints. This paper deal with this problem by adding a margin of $10\% \times d_{ij,safe}$. Although the safety constraint is not violated in the simulations, more efforts are needed to deal with the errors introduced by linearization.
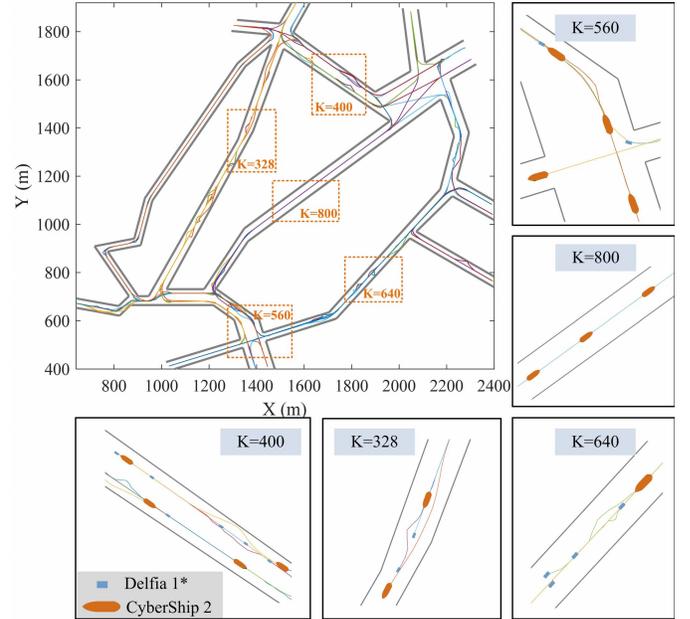
Secondly, communication constraints, such as packet loss, delays, and operation time should be considered in future research when applying CMVSs in reality.

Thirdly, due to the networked structure, vessels in waterway networks can choose different routes to avoid congestions. Therefore, the proposed method can be combined with vehicle routing problem for the transport in waterway networks.

Moreover, the proposed framework can be extended to include other infrastructures, such as locks and bridges. Similarly, vessels passing through movable bridges and locks can be regarded as the occupation of space and time. Thus, they can also be formulated as resource allocation problems.

Furthermore, in this paper, we assume all the participators are autonomous and cooperative. However, future waterborne transport system will be a system in which both human-operated and autonomous vessels exist. Besides, the recreation vessels in urban waterway networks might not be cooperative. When ASVs encounter those non-cooperative participators, collision avoidance actions can be determined by assuming they will keep their own state or predicting the trajectory of the targets according to historical data, such as AIS data [41].

APPENDIX A
THE GVO METHOD USED IN THE SIMULATION

Generalized velocity obstacle (GVO) is a reactive collision avoidance algorithm incorporating vehicle dynamics. Using GVO algorithm in collision avoidance contains three fundamental modules, namely design of controller, construction of UO set, and optimization. In the experiments, the first two modules follow the framework introduced in [21] with some slight modifications. A new optimization module and a communication module are introduced for the application of transport in waterway networks.

The changes made in controller design and UO set construction are addressed as follows. Firstly, the feedback gains of
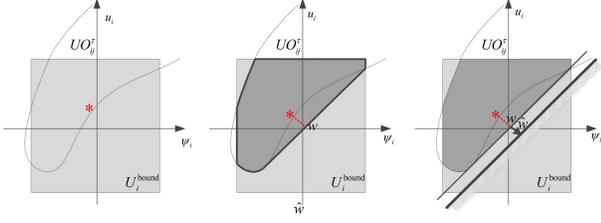
Fig. 11.   Approximation of feasible space.

Proportional-Derivative (PD) controller are changed to $K_p = diag([200, 200, 10])$ and $K_d = diag([5, 5, 5])$. Secondly, the velocity $\mathbf{u}$ inputted to controller consists of u, v, and ψ, where v is always set to 0. Thus, the $\mathbf{u}$ reduces to two degrees of freedom. Thirdly, UO set is denoted as the velocity leading to collision, i.e. $\mathbf{u} \in UO_{ij}^\tau$. In particular, UO_{ij}^τ collects $\mathbf{u}$ leading the ship i to collide with ship j in $[0, \tau]$. Hence, the complement of UO_{ij}^τ is a set of collision-free velocities, which is noted as $\overline{UO}_{ij}^\tau = \mathbb{R}^2 \backslash UO_{ij}^\tau$. In this simulation, $\tau = 6$. Fourthly, the reference velocity is formulated as $\mathbf{r}^* = [\bar{u}^*, 0, \bar{\psi}^*]$, where $\bar{\psi}^*$ is the angle between $wp_i - \mathbf{P}_i(t)$ and the x-axis; $\bar{u}^*$ is economic speed. In the simulation, $\bar{u}^* = 0.8$ [m/s] for CSII ship and $\bar{u}^* = 0.6$ [m/s] for Delfia 1*. Reference velocity $\mathbf{r}^*$ is shown as * in Fig. 11.

In the simulation experiments, we applied a new optimization module. The variable of this optimization is $\mathbf{u}$ and the objective is approaching to $\mathbf{r}^*$. Moreover, the optimal solution needs to be collision-free and reachable. That implies the final solution falls in $\overline{UO}_{ij}^\tau$ and satisfies the kinematic constraints, i.e. $u \in [u_{\min}, u_{\max}]$ and $\psi \in [-\psi_{\max}, \psi_{\max}]$, noted as U_i^{bound}. Accordingly, the feasible space for ship i to prevent collision with ship j is $U_{ij}^{\text{fea}} = \overline{UO}_{ij}^\tau \cap U_i^{\text{bound}}$. Since U_{ij}^{fea} is non-convex, an approximation is necessary, see Fig. 11. We employ a convex hull to approximate the intersection of UO_{ij}^τ and U_i^{bound}, noted as $\text{CH}\left(UO_{ij}^\tau \cap U_i^{\text{bound}}\right)$; Then, the point which is closest to the reference velocity $\mathbf{r}^*$ on the boundary of $\text{CH}\left(UO_{ij}^\tau \cap U_i^{\text{bound}}\right)$ can be found:

$$w = \underset{\mathbf{u} \in \partial\text{CH}\left(UO_{ij}^\tau \cap U_i^{\text{bound}}\right)}{\arg\min} \| \mathbf{u} - \mathbf{r}^* \| \tag{29}$$

$$U_{ij}^{\text{fea}} = \begin{cases} \{\mathbf{u}|\, (\mathbf{u} - w) \cdot (w - \mathbf{r}^*) \geqslant 0\}, \\ \qquad \text{if } \mathbf{r}^* \in \text{CH}\left(UO_{ij}^\tau \cap U_i^{\text{bound}}\right) \\ \{\mathbf{u}|\, (\mathbf{u} - w) \cdot (w - \mathbf{r}^*) \leqslant 0\} \text{ otherwise} \end{cases} \tag{30}$$

where ∂ refers to the boundary of a set. When the optimal velocity is on the boundary of UO set, two vehicles will approach to each other infinitely close [42]. Thus, in the simulation, we adopt a repulsive term $\hat{w}$, $w := w + \dfrac{w}{\|w\|}\hat{w}$ and $\hat{w} = 0.02$. Equation (30) allows us to use a linear constraint to approximate UO_{ij}^τ. Then, the feasible space becomes convex.

Then, the collision avoidance actions can be determined by solving following optimization problem:

Problem $\mathscr{D}$:

$$\text{minimize } J_{vo}(\mathbf{u}_i) = (\mathbf{u}_i - \mathbf{r}_i^*)^T \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix} (\mathbf{u}_i - \mathbf{r}_i^*) \tag{31}$$

$$\text{subject to } \mathbf{u}_i \in \bigcap_{j \neq i} U_{ij}^{\text{fea}}, \mathbf{u}_i \in U_i^{\text{bound}} \tag{32}$$

Moreover, communication among controllers is introduced. In the original GVO algorithm, each controller assumes perfect knowledge of other participators and is responsible for collision avoidance. However, this assumption is strict for ASVs sailing in narrow waterways. The controllers usually cannot find a feasible solution without others cooperation. Thus, here, each ship updates and broadcasts its planned trajectory to other ships in the same waterway. The order the ships broadcast information is the order they enter the waterway. In each decision loop, a ship forecast its 6-second trajectory. If the desired velocity $\mathbf{r}^*$ falls in any UO_{ij}^τ, the controller solves Problem $\mathscr{D}$ to find another solution. If there is no feasible solution, we will try the opposite update order. If the controller still cannot find a collision-free solution, the ship will stop.

References

[1] J. Rohács and G. Simongáti, "The role of inland waterway navigation in a sustainable transport system," *Transport*, vol. 22, no. 3, pp. 148–153, 2007.

[2] (2018). *Google Maps*. [Online]. Available: https://www.google.com/maps/@31.2953034,120.5609974,14.38z

[3] L. Chen, J. J. Hopman, and R. R. Negenborn, "Distributed model predictive control for vessel train formations of cooperative multi-vessel systems," *Transp. Res. C, Emerg. Technol.*, vol. 92, pp. 101–118, Jul. 2018.

[4] L. Chen, R. R. Negenborn, and J. J. Hopman, "Intersection crossing of cooperative multi-vessel systems," in *Proc. 15th IFAC Symp. Control Transp. Syst.*, Savona, Italy, Jun. 2018, vol. 51, no. 9, pp. 379–385.

[5] S. Campbell, W. Naeem, and G. W. Irwin, "A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres," *Annu. Rev. Control*, vol. 36, no. 2, pp. 267–283, 2012.

[6] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annu. Rev. Control*, vol. 41, pp. 71–93, Jan. 2016.

[7] C. Tam, R. Bucknall, and A. Greig, "Review of collision avoidance and path planning methods for ships in close range encounters," *J. Navigat.*, vol. 62, no. 3, pp. 455–476, 2009.

[8] S. E. Li, Y. Zheng, K. Li, L.-Y. Wang, and H. Zhang, "Platoon control of connected vehicles from a networked control perspective: Literature review, component modeling, and controller synthesis," *IEEE Trans. Veh. Technol.*, to be published.

[9] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. London, U.K.: Wiley, 2011.

[10] L. Chen, J. Mou, and H. Ligteringen, "Route choice model in inland waterway network: A case study in China," in *Proc. Transp. Res. Board 97th Annu. Meeting*, Washington, DC, USA, Jan. 2018. [Online]. Available: https://trid.trb.org/view/1495184

[11] C. J. Ting and P. Schonfeld, "Integrated control for series of waterway locks," *J. Waterway, Port, Coastal, Ocean Eng.*, vol. 124, no. 4, pp. 199–206, 1998.

[12] H. Ashrafiuon, K. R. Muske, and L. C. McNinch, "Review of nonlinear tracking and setpoint control approaches for autonomous underactuated marine vehicles," in *Proc. Amer. Control Conf.*, 2010, pp. 5203–5211.

[13] K. S. Tsakalis and S. Dash, *Identification for PID Control*. London, U.K.: Springer, 2012, pp. 283–317.

[14] K. P. Tee and S. S. Ge, "Control of fully actuated ocean surface vessels using a class of feedforward approximators," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 4, pp. 750–756, Jul. 2006.

[15] H. Ashrafiuon, K. R. Muske, L. C. McNinch, and R. A. Soltan, "Sliding-mode tracking control of surface vessels," *IEEE Trans. Ind. Electron.*, vol. 55, no. 11, pp. 4004–4012, Nov. 2008.

[16] G. N. Roberts, R. Sutton, A. Zirilli, and A. Tiano, "Intelligent ship autopilots—A historical perspective," *Mechatronics*, vol. 13, no. 10, pp. 1091–1103, 2003.

[17] G. Wen, S. S. Ge, C. L. P. Chen, F. Tu, and S. Wang, "Adaptive tracking control of surface vessel using optimized backstepping technique," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3420–3431, Sep. 2019.

[18] H. Zheng, R. R. Negenborn, and G. Lodewijks, "Predictive path following with arrival time awareness for waterborne AGVs," *Transp. Res. C, Emerg. Technol.*, vol. 70, pp. 214–237, Sep. 2016.

[19] M. Abdelaal, M. Fränzle, and A. Hahn, "Nonlinear model predictive control for trajectory tracking and collision avoidance of underactuated vessels with disturbances," *Ocean Eng.*, vol. 160, pp. 168–180, Jul. 2018.

[20] R. Daily and D. M. Bevly, "Harmonic potential field path planning for high speed vehicles," in *Proc. Amer. Control Conf.*, Seattle, WA, USA, Jun. 2008, pp. 4609–4614.

[21] Y. Huang, L. Chen, and P. van Gelder, "Generalized velocity obstacle algorithm for preventing ship collisions at sea," *Ocean Eng.*, vol. 173, pp. 142–156, Feb. 2019.

[22] T. Statheros, G. Howells, and K. M. Maier, "Autonomous ship collision avoidance navigation concepts, technologies and techniques," *J. Navigat.*, vol. 61, no. 01, pp. 129–142, 2008.

[23] C. Tam and R. Bucknall, "Cooperative path planning algorithm for marine surface vessels," *Ocean Eng.*, vol. 57, pp. 25–33, Jan. 2013.

[24] H. Zheng, R. R. Negenborn, and G. Lodewijks, "Robust distributed predictive control of waterborne AGVs—A cooperative and cost-effective approach," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2449–2461, Aug. 2018.

[25] L. Chen and C. Englund, "Cooperative intersection management: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 570–586, Feb. 2016.

[26] M. Ahmane *et al.*, "Modeling and controlling an isolated urban intersection based on cooperative vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 28, pp. 44–62, Mar. 2013.

[27] J. Gregoire, S. Bonnabel, and A. de La Fortelle, "Optimal cooperative motion planning for vehicles at intersections," in *Proc. IEEE Intell. Vehicles Symp.*, Oct. 2012, pp. 1–6.

[28] H. Ghaffarian, M. Fathy, and M. Soryani, "Vehicular ad hoc networks enabled traffic controller for removing traffic lights in isolated intersections based on integer linear programming," *IET Intell. Transport Syst.*, vol. 6, no. 2, pp. 115–123, Jun. 2012.

[29] Y. Kuwata and J. P. How, "Cooperative distributed robust trajectory optimization using receding horizon milp," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 2, pp. 423–431, Mar. 2011.

[30] R. R. Negenborn and J. M. Maestre, "Distributed model predictive control: An overview and roadmap of future research opportunities," *IEEE Control Syst.*, vol. 34, no. 4, pp. 87–97, Aug. 2014.

[31] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[32] R. R. Negenborn, B. De Schutter, and J. Hellendoorn, "Multi-agent model predictive control for transportation networks: Serial versus parallel schemes," *Eng. Appl. Artif. Intell.*, vol. 21, no. 3, pp. 353–366, Apr. 2008.

[33] H. Zheng, R. R. Negenborn, and G. Lodewijks, "Fast ADMM for distributed model predictive control of cooperative waterborne AGVs," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1406–1413, Jul. 2017.

[34] M. Abdelaal, M. Fräzle, and A. Hahn, "NMPC-based trajectory tracking and collison avoidance of underactuated vessels with elliptical ship domain," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 22–27, 2016.

[35] T. Keviczky, F. Borrelli, K. Fregene, D. Godbole, and G. J. Balas, "Decentralized receding horizon control and coordination of autonomous vehicle formations," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 1, pp. 19–33, Jan. 2008.

[36] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control Theory and Design*. Madison, WI, USA: Nob Hill Publishing, 2015.

[37] W.-Y. Ku and J. C. Beck, "Mixed integer programming models for job shop scheduling: A computational analysis," *Comput. Oper. Res.*, vol. 73, pp. 165–173, Sep. 2016.

[38] R. Skjetne, T. I. Fossen, and P. V. Kokotović, "Adaptive maneuvering, with experiments, for a model ship in a marine control laboratory," *Automatica*, vol. 41, no. 2, pp. 289–298, 2005.

[39] The Society of Naval Architects and Marine Engineers, "Nomenclature for treating the motion of a submerged body through a fluid," Soc. Naval Architects Mar. Eng., New Jersey, NJ, USA, Tech. Rep. 1-5, 1952.

[40] Rijkswaterstaat. (Aug. 2018). *Map of Waterways*. [Online]. Available: https://waterinfo.rws.nl/#!/kaart/waterhoogte-t-o-v-nap/

[41] S. Hexeberg, A. L. Flåten, B.-O. H. Eriksen, and E. F. Brekke, "Ais-based vessel trajectory prediction," in *Proc. 20th Int. Conf. Inf. Fusion*, Xi'an, China, Jul. 2017, pp. 1–8.

[42] J. Alonso-Mora, P. Beardsley, and R. Siegwart, "Cooperative collision avoidance for nonholonomic robots," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 404–420, Apr. 2018.
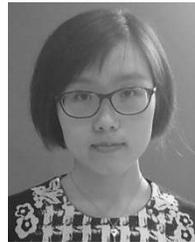
Linying Chen received the B.Sc. degree in international economics and trade from Wuhan University, China, in 2011, and the B.Sc. degree in maritime administration and the M.Sc. degree in transport information engineering and control from the Wuhan University of Technology in 2011 and 2014, respectively. She is currently pursuing the Ph.D. degree with the Department of Maritime and Transport Technology, Delft University of Technology, Delft, The Netherlands.

Her current research interests include autonomous vessels and cooperative control of multi-vessel systems.

Yamin Huang received the B.Sc. degree in maritime administration and the M.Sc. degree in transport information engineering and control from the Wuhan University of Technology, Wuhan, China, in 2011 and 2014, respectively. He is currently pursuing the Ph.D. degree with the Safety and Security Science Group, Department of Values, Technology and Innovation, Delft University of Technology, Delft, The Netherlands.

His research interests focus on marine traffic risk assessment and collision avoidance.

Huarong Zheng received the B.Sc. degree in logistics engineering and the M.Sc. degree in intelligent transportation engineering from the Wuhan University of Technology in 2011 and 2013, respectively, and the Ph.D. degree from the Delft University of Technology, Delft, The Netherlands, in 2016.

She is currently an Assistant Researcher with the College of Control Science and Engineering, Zhejiang University, Hangzhou, China. Her research interests include model predictive control and distributed control with applications in transport engineering and logistics.

Hans Hopman is currently a Full Professor (Ship Design, Production and Operations) with the Department of Maritime and Transport Technology, Delft University of Technology, Delft, The Netherlands. His research interests include design, engineering, production, repair, and operation of ships and other floating marine objects, including their machinery and electric equipment, and also includes research on supporting topics, such as logistics, finance, marketing, sustainability, and shipping management.

Rudy Negenborn received the M.Sc. degree in computer science/intelligent systems from Utrecht University in 1998 and the Ph.D. degree in distributed control for networked systems from the Delft University of Technology, Delft, The Netherlands, in 2007.

He is currently a Full Professor (Multi-Machine Operations and Logistics) with the Department of Maritime and Transport Technology, Delft University of Technology. His research interests are in the areas of distributed control, multi-agent systems, model predictive control, and optimization. He applies the developed theories to address control problems in large-scale transportation and logistic systems.