Brodogradnja/Shipbilding/Open access Volume 69 Number 3, 2018

Hongbo Wang
Jingxian Liu
Jinfen Zhang
Kezhong Liu
Xugang Yang
Qing Yu

http://dx.doi.org/10.21278/brod69301 ISSN 0007-215X
eISSN 1845-5859

SELF-ORGANIZING COOPERATION MODEL FOR SHIPS NAVIGATING IN RESTRICTED ONE-WAY WATERWAY

UDC 629.5.05:629.5.077.1
Original scientific paper

Summary

An effective ship scheduling strategy is critical for the efficiency of waterway transportation, especially in restricted one-way waterways. By treating ship traffic as a distributed system, a novel Ship Self-Organizing Cooperation algorithm (SSOCA) is proposed to evaluate the effects of self-organizing cooperation between ships. An assumption is made that overtaking is not allowed under the given safety requirement. Observing that traffic efficiency is influenced by speed differences, two types of delay times, wait time and navigation time as increased by speed reductions, are applied to evaluate traffic efficiency. The mathematical model of delay time is inferred in different entry sequences subsequently. Taking advantage of the delay model, each ship makes a decision regarding its own sequence to find the local optimum iteratively. An Arena-based ship traffic model is constructed. The simulation results indicate that average delay time for ships is decreased in comparison with the First Come First Served (FCFS) model. That advantage exists for different combinations of traffic flow parameters. Moreover, a balance between efficiency and computation is also achieved by distributing the computational burden to each ship.

Key words: *self-organizing; cooperation mechanism; restricted one-way waterway; traffic efficiency; delay time; speed difference*

1. Introduction

Maritime transportation plays an important role in cargo shipping. As the numbers and sizes of ships increase [1], waterways are becoming one of the most important resources, especially in world-class ports and busy inland waterways. Some of them are becoming increasingly congestion. Congestion brings high navigational risks and also reduces overall traffic efficiency, with many delays. Waterway improvement projects are supposed to be among the most effective ways to mitigate such problems. However, huge infrastructure investments for waterways are not always available in practice. Another idea is to re-organize waterway traffic to improve efficiency by enhancing the overall utilization of the waterway resource and by reducing the overall delays of arriving ships. Ship scheduling [2] is among the most effective ways to do that. First Come First Serve (FCFS) is one of the most popular scheduling models for transportation. However, FCFS is not the most efficient because it prioritizes equality over efficiency.

In general, ship scheduling can be performed in centralized or distributed ways. A large amount of existing research has concentrated on centralized modes, including queuing models [3, 4], waterway-berth coordination [5, 6] and sequential scheduling models [7]. Among their premises, the information on all arriving ships should be available and a top-level agent assigns the arrival time for each ship. All the ships need to navigate under the requirements from the top-level agent. However, it should be noted that a ship is usually controlled by crewmembers who are responsible for route planning. Operations such as ship scheduling and collision avoidance are resolved individually and locally. Although Vessel Traffic Services (VTSs) are responsible for ship traffic from a macro-perspective [8], it is usually arguable how VTSs play an importance role in maritime supervision. In fact, VTSs are only in charge of information services and receiving reports on intentions and constraints from ships. That is to say, VTSs do not control ship traffic directly. Therefore, VTSs are not top-level agents. In addition, an assumption is often made in centralized models that all the ships are navigating under identical and stable speeds. Nevertheless, the speeds of ships usually vary by large amounts within certain ranges in many real cases. For instance, ship speed in the Pelagos Sanctuary is high and dependent on vessel type, with specific spatial distributions [9]. A similar result was also found for the Port of Rotterdam, in which a normal distribution was fit to a speed histogram with a mean value of 10.7 knots and a standard deviation of 1.2 knots [10]. Variances in speed exist not only at different locations at a given time but also at different times at a given location, which triggers changes in course or speed. In order to analyze such problems, several characteristic parameters [11] are proposed for traffic in restricted one-way waterways, and simulation results show that the interactions of speed differences are closely related to congestion degree. A sensitivity analysis [12] then shows that the interactions owing to speed variances also largely influence traffic efficiency. Overall, unstable traffic derived from those interactions reduces navigation efficiency. Because ships interact with each other in a parallelized way, it is difficult to predict the future state [13] of a ship traffic system in centralized scheduling mode.

Distributed scheduling provides an alternative that focuses on the interactions between individuals, which enables the ships to make decisions individually by negotiating with adjacent ships [14]. Distributed scheduling can be accomplished using a self-organization approach [15].

Individuals can respond to local stimulations, realizing the division of work and accomplishing complex goals [16]. The individuals are equals, manifest self-control and make decisions based on local information that conforms to the ship traffic situation. In this sense, ship traffic should be treated as a self-organizing system, and distributed scheduling based on self-organization is more suitable for ship traffic than central scheduling. Self-organization, which is a process in which some form of overall order arises from local interactions between parts of an initially disordered system, was introduced and formulated by W. Ross Ashby [17]. The main idea is to build the components of a system in such a way that they will find solutions to problems by themselves [15]. Because trends in traffic systems are far from balanced [18], self-organizing theory is suitable for studying the behaviours and characteristics of such systems. The self-organizing behaviours of drivers, which is commonly identified in vehicle traffic [19], decides nearly all the spatial-temporal behaviours of a traffic system [20], including route choice [21] and traffic light coordination [22]. From statistics of the Automatic Identification System (AIS), ships navigating in restricted waterways cluster, similar to results obtained in car-following models [23, 24]. However, there are some differences. First, vehicles often choose routes according to degree of congestion in road networks, whereas ships usually navigate over regular or recommended routes because waterway networks seldom exist. Even if a ship discovers a bottleneck emerging in the planned route, the ship's pilot would prefer to wait near the entrance rather than find an alternative. Second, vehicles are largely affected by traffic signal lights at intersections. Ships, however, act autonomously in restricted areas according to rules. In summary, the self-organizing models used to study vehicle traffic are not suitable for ship traffic due to the latter's particular characteristics.

A novel self-organizing cooperation model for ships navigating in restricted waterways is proposed in this paper. The remainder of the paper is organized as follows. In Section 2, the reasons and preconditions of ship deceleration are ascertained by analyzing the process of ships navigating in restricted waterways. Considering safety distance, a mathematical model of delay time is inferred in Section 3. A ship self-organizing cooperation method is offered in Section 4. Using an Arena-based ship traffic simulation model, self-organizing cooperation is verified and discussed in Section 5. Finally, the conclusions are presented in Section 6.

2. Problem statement

The restricted waterway discussed in this paper is a one-way waterway with one entrance and one exit. Overtaking is prohibited to all ships navigating in the waterway. To avoid collisions in the waterway, ships are required to maintain safe distances between each other. Safe distance is usually described as ship domain [25]. In general, a ship will need a larger safe distance if it has a greater size or speed.

Assume that two ships are arriving sequentially at a restricted waterway of length of L_{wat}. The first arriving ship is *ship i*, and the later-arriving ship is *ship j*. As shown in Fig. 1, the whole process of the ships is divided into 4 phrases. In phrase 1 in Fig. 1(a), *ship i* arrives at the approach of the waterway at t_i^{arr} while *ship j* is traveling to the approach. If there are no ships ahead, *ship i* would immediately enter the waterway at t_i^{in}. Under that circumstance, t_i^{in} is equal to t_i^{arr}. *Ship j* then arrives at the approach of the waterway at t_j^{arr} and enters the waterway at t_j^{in}. In phrase 2 in Fig. 1(b), s*hip i* and *ship j* are sailing in the waterway. The original distance *d* between them exceeds safe distance d_j^s. *Ship i* and *ship j* could maintain their initial speeds v_i and v_j, respectively. If $v_j > v_i$, the distance between them would gradually decrease. At phase 3 in Fig. 1(c), *ship j* arrives a point *A* from the entrance point *O*, which are separated by a distance l_{ij}. *Ship j* would then slow to v_i to maintain a safe distance because $d = d_j^s$ and because overtaking is not allowed in the restricted waterway. At phrase 4 in Fig. 1(d), *ship i* and *ship j* depart the waterway at v_i. It can be seen that *ship j* is delayed owing to its speed reduction in the section of L_{wat} - l_{ij} (from *A* to *B* at the exit), and hence traffic efficiency is sacrificed to some degree.

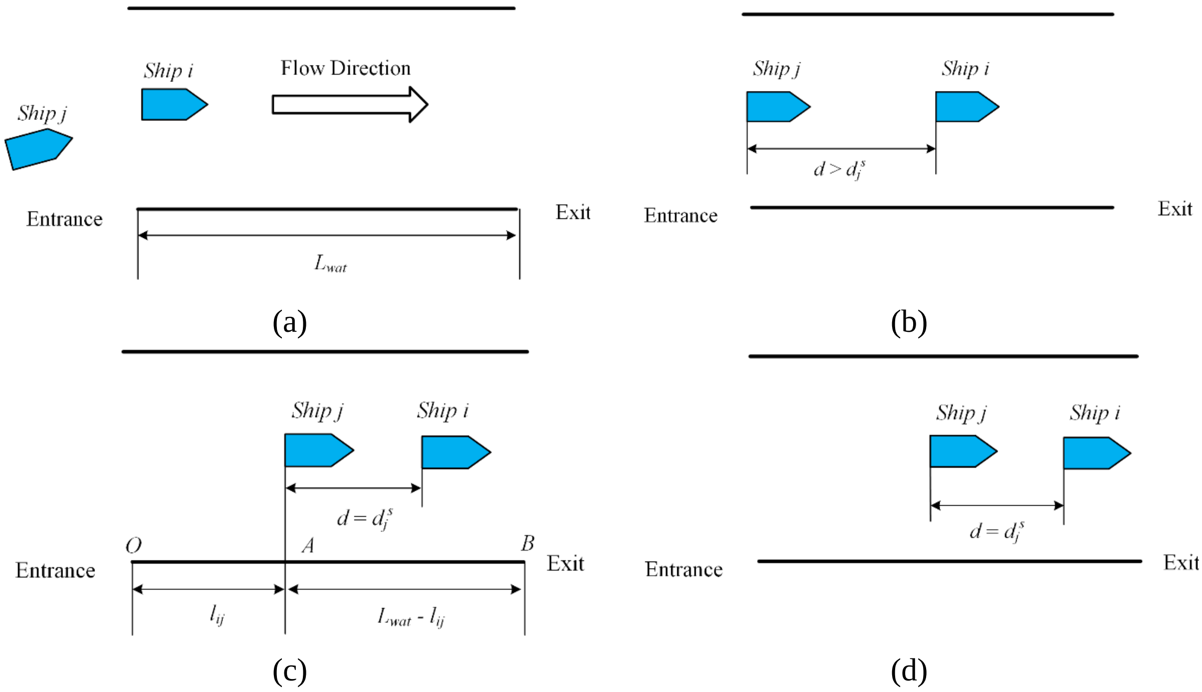


Fig. 1 Ship navigation process. (a) Arriving in phrase 1, (b) Maintaining speed in phrase 2, (c) Deceleration in phrase 3, (d) Departure in phrase 4.

According to the description in Fig. 1, the deceleration has two preconditions. The first precondition is that v_j must be greater than v_i, as shown in Eq. (1); that is,

$$v_i < v_j \tag{1}$$

If the speeds of the two ships are same, their distance will be maintained until they depart the waterway. The distance between the two ships will gradually increase if $v_j < v_i$. *Ship j* will catch up with *ship i* only if $v_j > v_i$. The second precondition is that the difference in the arrival times is sufficiently small, which is expressed as Eq. (2). ΔT_{ij}^{arr} is the difference between t_i^{arr} and t_j^{arr}. The extreme is denoted as Δt_{ij}^{arr} such that *ship j* will not catch up with *ship i* until *ship i* leaves the waterway. Δt_{ij}^{arr} can be derived as Eq. (3).

$$\Delta T_{ij}^{arr} = t_j^{arr} - t_i^{arr} < \Delta t_{ij}^{arr} \tag{2}$$

$$\Delta t_{ij}{}^{arr} = \frac{L_{wat}}{v_i} - \frac{L_{wat} - d_j{}^s}{v_j} \tag{3}$$

If Eq. (2) is satisfied, *ship j* will slow down owing to the obstruction caused by *ship i*. If Eq. (2) is not satisfied, *ship j* will maintain its initial speed until leaving the waterway because *ship i* is sufficiently distant from *ship j* throughout the process. Thus, the precondition that *ship j* decelerates due to *ship i* is summarized in Eq. (4).

$$\begin{cases} t_i{}^{arr} < t_j{}^{arr} \\ v_i < v_j \\ \Delta T_{ij}{}^{arr} < \Delta t_{ij}{}^{arr} \end{cases} \tag{4}$$

3. Traffic efficiency for different sequences

To reflect the influence of deceleration, delay time is applied to evaluate traffic efficiency, being defined as the sum of wait time and the increased navigation time caused by deceleration, which is denoted deceleration time. Obviously, smaller delay times are more desirable. Therefore, the objective function for traffic efficiency in a restricted one-way waterway can be described as

$$Min \ \frac{1}{m}\sum_{i=1}^{m} T_i^{delay} = \frac{1}{m}\sum_{i=1}^{m}(T_i^{wait} + T_i^{dec}) \tag{5}$$

In Eq. (5), m is the number of arriving ships, T_i^{delay} is the delay time for *ship i*, T_i^{wait} is the wait time for *ship i*, and T_i^{dec} is the deceleration time.

The assumption is still that only two ships will pass through the waterway. The distance between the two ships always exceeds d_j^s when Eq. (4) is not satisfied. In that case, neither waiting nor deceleration occurs. Thus, the overall delay time T_{ij}^{delay} is zero, as expressed in Eq. (6), and traffic efficiency reaches its optimum.

$$T_{ij}{}^{delay} = T_i^{delay} + T_j{}^{delay} = 0 + 0 = 0 \tag{6}$$

However, when Eq. (4) is satisfied, *ship j* will slow down owing to the obstruction caused by *ship i*. If the sequence is changed, *ship j* will enter the waterway immediately, and the deceleration is eliminated. At the same time, the wait time for *ship i* is increased, although the deceleration time remains zero. Whether the sequence should be changed to improve traffic efficiency remains uncertain, and to answer that question, the mathematical model is analyzed in the following subsections.

3.1 Traffic efficiency for the original sequence

If *ship i* arrived and entered the waterway earlier, it passed through the waterway without restrictions throughout the entire process. That is to say, the wait time (T_i^{wait}) and deceleration time (T_i^{dec}) are both zero, as shown in Eq. (7) and Eq. (8). Hence, T_i^{delay} is described in Eq. (9).

$$T_i^{waiting} = 0 \tag{7}$$

$$T_i^{dec} = 0 \tag{8}$$

$$T_i^{delay} = T_i^{wait} + T_i^{dec} = 0 \tag{9}$$

Ship j enters the waterway after *ship i* and may be waiting at the entrance due to the safe distance constraint. When *ship j* arrives at the entrance, *ship i* has travelled some distance in the waterway. If the distance exceeds d_j^s, *ship j* can enter the waterway directly like *ship i*, and wait time T_j^{wait} is equal to zero, as shown in Eq. (10).

$$T_j^{wait} = 0 \tag{10}$$

Otherwise, *ship j* has to wait until the distance between *ship j* and *ship i* reaches d_j^s. In that case, T_j^{wait} can be expressed as Eq. (11).

$$T_j^{wait} = d_j^s / v_i - \left(t_j^{arr} - t_i^{arr}\right) \tag{11}$$

It can be determined that the result of Eq. (11) is negative when *ship j* does not wait at the entrance. T_j^0 as shown in Eq. (12), which is the wait time caused by the constraint of safe distance:

$$T_j^0 = Max\left(0, d_j^s / v_i - \left(t_j^{arr} - t_i^{arr}\right)\right). \tag{12}$$

As shown in Fig. 1(c), *ship j* navigates with speed v_j from *O* to *A* and reduces speed to v_i from *A* to *B*. Therefore, T_j^{dec} as expressed as Eq. (13), which is the deceleration time of *ship j* and is related to the speed change and deceleration distance (i.e. L_{wat} - l_{ij}).

$$T_j^{dec} = \frac{L_{wat} - l_{ij}}{v_i} - \frac{L_{wat} - l_{ij}}{v_j} \tag{13}$$

Here, l_{ij} is calculated based on Newton's first law, which is closely associated with arrival time and speed difference, as shown in Eq. (14).

$$l_{ij} = \frac{v_i \times (t_j^{arr} - t_i^{arr} + T_j^0) - d_j^s}{v_j - v_i} \times v_j \tag{14}$$

Therefore, the sum of wait time and deceleration time for *ship j* is T_j^{delay} in Eq. (15).

$$T_j^{delay} = T_j^{wait} + T_j^{dec} = T_j^0 + \frac{L_{wat} - l_{ij}}{v_i} - \frac{L_{wat} - l_{ij}}{v_j} \tag{15}$$

Furthermore, the overall delay time for the two ships, T_{ij}^{delay}, is expressed in Eq. (16) when the FCFS model is followed and Eq. (4) is satisfied.

$$T_{ij}^{delay} = T_i^{delay} + T_j^{delay} = T_j^{0} + \frac{L_{wat} - l_{ij}}{v_i} - \frac{L_{wat} - l_{ij}}{v_j} \tag{16}$$

3.2 Traffic efficiency for the changed sequence

The sequence of *ship i* and *ship j* is changed in this section, presuming that Eq. (4) is still satisfied. That is to say, *ship i* enters behind *ship j*, although *ship i* arrives the approach earlier. In general, anchorages lie near approaches. When *ship i* arrives, the ship would wait for *ship j* at the anchorage in Fig. 2(a) and then follow *ship j* as shown in Fig. 2(b).

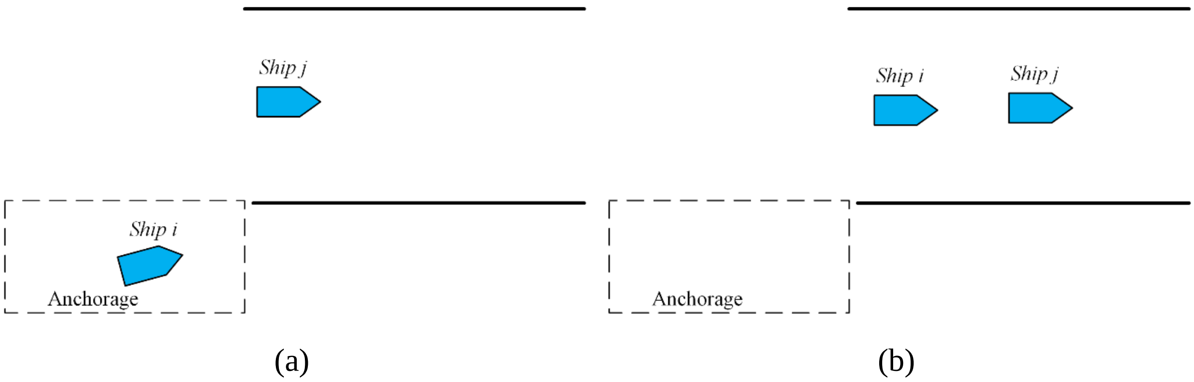


Fig. 2 *ship i* and *ship j* enter the waterway in a different sequence. (a) *ship j* is entering the waterway, and *ship i* is waiting at the anchorage, (b) *ship i* enters the waterway behind *ship j*.

As shown in Fig. 2(a), *ship i* does not enter the waterway until *ship j* arrives and steers for d_i^s under the constraint of safe distance. As a result, the wait time of *ship i* is the sum of the difference of arrival time and the period of *ship j* navigating for d_i^s as shown in Eq. (17). After entering the waterway, *ship i* does not slow down, as shown in Eq. (18), because of the increasing distance from *ship j*.

$$T_i^{wait} = t_j^{arr} - t_i^{arr} + d_i^{s} / v_j \tag{17}$$

$$T_i^{dec} = 0 \tag{18}$$

Therefore, the delay time of *ship i* is expressed as Eq. (19). In the meantime, *ship j* does not wait and has the deceleration time shown in Eq. (20).

$$T_i^{delay} = T_i^{wait} + T_i^{dec} = t_j^{arr} - t_i^{arr} + d_i^{s} / v_j \tag{19}$$

$$T_j^{delay} = T_j^{wait} + T_j^{dec} = 0 \tag{20}$$

Furthermore, T_{ji}^{delay}, which is the overall delay time when *ship j* enters the waterway first and *ship i* enters the waterway afterwards, is expressed in Eq. (21).

$$T_{ji}^{delay} = T_j^{delay} + T_i^{delay} = t_j^{arr} - t_i^{arr} + d_i^s / v_j \tag{21}$$

4. Ship self-organizing cooperation model

With regard to the transportation problem, it is usually time and space consuming [26] to calculate the solution using a deterministic algorithm. An algorithm that calculates all of the permutations is possible for traffic efficiency when many ships arrive in succession. However, the computations will grow exponentially according to the permutation combination formula, especially when it involves too many ships. Heuristic algorithms are usually used to solve such problems, as in Zhang et al. [5]. Nevertheless, it is usually challenging to obtain a mathematical expression when many ships are navigating with different speeds. In this section, a self-organizing cooperation model is proposed to solve the problem. The fundamental idea of this model is that a slower ship yields priority to another faster ship to improve traffic efficiency employing a cooperation mechanism [27]. Cooperation usually refers to the process of groups of organisms working or acting together for joint or mutual benefit [28]. In this paper, it mainly denotes that ships follow simple local rules to closely approach optimal global coordination [29] in a distributed mode.

4.1 Ship self-organizing cooperation algorithm

The assumption is that *ship i*, *ship j* and *ship k* will arrive at the entrance in rapid succession and navigate through the waterway as shown in Fig. 2. To reduce the overall delay time, the sequence between *ship i* and *ship j* should be changed when Eq. (22) is matched besides Eq. (4). Otherwise, the original sequence is retained for the two ships.

$$T_{ji}^{delay} < T_{ij}^{delay} \tag{22}$$

If *ship i* and *ship j* follow the initial sequence of entering the waterway when Eq. (22) is not satisfied, the same method employed in section 3 is applied to calculate the delay between *ship j* and *ship k*. In the case that *ship j* would enter the waterway before *ship i* after calculation, *ship k* is faced with whether rearranging *ship i* and *ship k* is reasonable. In that situation, it is determined that *ship i* will wait for *ship j* in anchorage so that $t_i^{in} = t_i^{arr}$ is no longer satisfied, and t_i^{in} should theoretically be reassigned as expressed in Eq. (23). Eq. (23) indicates that *ship i* does not enter the waterway until *ship j* navigates a distance d_i^s in the waterway, as shown in Fig. 2(a).

$$t_i^{in} = t_i^{arr} + (t_j^{arr} - t_i^{arr} + d_i^s / v_j) = t_j^{arr} + d_i^s / v_j \tag{23}$$

The deceleration criteria between *ship i* and *ship k* is shown in Eq. (24) for $t_i^{in} < t_k^{arr}$ and omitting the influence of *ship j*.

$$\begin{cases} t_i^{in} < t_k^{arr} \\ v_i < v_k \\ (t_k^{arr} - t_i^{in}) < (\dfrac{L_{wat}}{v_i} - \dfrac{L_{wat} - d_k^{\ s}}{v_k}) \end{cases} \tag{24}$$

Once Eq. (24) is satisfied, the criteria for a sequence change to improve traffic efficiency is presented in Eq. (25), which is similar to Eq. (22).

$$\begin{cases} T_{ki}^{\ delay} < T_{ik}^{\ delay} \\ T_{ik}^{\ delay} = T_k^{\ 0} + \dfrac{L_{wat} - l_{ik}}{v_i} - \dfrac{L_{wat} - l_{ik}}{v_k} \\ T_{ki}^{\ delay} = t_k^{arr} - t_i^{in} + d_i^{\ s} / v_k \\ l_{ik} = \dfrac{v_i \times (t_k^{arr} - t_i^{in} + T_k^{\ 0}) - d_k^{\ s}}{v_k - v_i} \times v_k \\ T_k^{\ 0} = Max\left(0, d_k^{\ s} / v_i - \left(t_k^{arr} - t_i^{in}\right)\right) \end{cases} \tag{25}$$

It is also possible that $t_i^{in} \geq t_k^{arr}$. Supposing the safe distances for *ship i* and *ship k* are same for simplicity, *ship i* would wait for at least t_i^{in} to satisfy the safe distance requirement, whereas the earliest time that *ship k* could enter the waterway is also t_i^{in}. That is to say, the two ships will wait at the anchorage, and the earliest times at which they could enter the waterway are both t_i^{in}. In that case, the faster ship should enter first to increase overall efficiency, as shown in Eq. (26).

$$\begin{cases} ship\ k \text{ should enter ahead of } ship\ i, \text{ when } t_k^{arr} \leq t_i^{in} \text{ and } v_i < v_k \\ ship\ i \text{ should enter ahead of } ship\ k, \text{ when } t_k^{arr} \leq t_i^{in} \text{ and } v_k \leq v_i \end{cases} \tag{26}$$

The problem here is to obtain details on arrivals of other ships. As a matter of fact, ship reporting systems [30] have been established in many restricted waterways and can be used to obtain position and dynamic information from passing ships. When a ship arrives at a report line, it should report its intention, arrival time and other information to the VTS centre for information sharing. Using an information inquiry and display system, all the ships could easily obtain other ships' information, including the arrival list.

According to local optimization, a ship only considers those ships that affect its navigation and puts others aside [31]. Under the assumption that every ship arrives at the approach at the time coinciding with its report, ships whose arrival times are earlier are the most likely obstacles in the waterway. Therefore, each ship only needs to find the target ship that will arrive at the waterway earlier for self-organization. Using the arrival list offered by VTS, every ship could seek out the target ship, even if the target ships are not detected by common instruments such as radar or AIS. Own ship could then ask for the target ship to wait at the anchorage temporarily so that own ship is not hampered in the waterway. Own ship should not make a request to give way for local interests if Eq. (22) or Eq. (25) is not satisfied.

The top level of the ship self-organizing cooperation algorithm (SSOCA) is presented in Tab. 1. According to Tab. 1, every ship reports the details of its navigational information to the VTS. The VTS then stores the information to *ArrivingList*. From *ArrivingList*, a ship can acquire the information of the last arriving ships before it. Once a ship has determined an entry sequence after calculating the delay times under different sequences, the decided sequence is reported and stored in *SequenceList*. The next ship behind in *ArrivingList* then undertakes further calculations. The procedure will proceed until all the ships are cleared from *ArrivingList*, which means that all the ships have determined their own sequences, and the self-organization is realized.

Tab. 1 Pseudo code of SSOCA

Input: *ArrivingList*
SequenceList ← the first two ships from *ArrivingList*;
Remove the first two ships from *ArrivingList*;
while *ArrivingList* is not empty **do**
ShipTemp ← Collect the last two ships (*ship i* and *ship j*, $t_i^{arr} < t_j^{arr}$) in *SequenceList* and the first ship (*ship k*) in *ArrivingList*;
if the entry sequence between *ship i* and *ship j* has not been changed in *SequenceList* **then**
T_{jk}^{delay} ← Calculate the delay time of *ship j* and *ship k* when *ship j* enters waterway before *ship k*;
T_{kj}^{delay} ← Calculate the delay time of *ship k* and *ship j* when *ship k* enters waterway before *Ship j*;
if $T_{jk}^{delay} <= T_{kj}^{delay}$ **then**
ShipTemp ←*ship i*, *j* and *k* in sequence;
else
ShipTemp ←*ship i*, *k and j* in sequence;
endif
else
T_{ik}^{delay} ← Calculate the delay time of *ship i* and *ship k* when *ship i* enters waterway before *ship k*;
T_{ki}^{delay} ← Calculate the delay time of *ship k* and *ship i* when *ship k* enters waterway before *Ship i*;
if $T_{ik}^{delay} <= T_{ki}^{delay}$ **then**
ShipTemp ←*ship j*, *i* and *k* in sequence;
else
ShipTemp ←*ship j*, *k* and *i* in sequence;
endif
endif
SequenceList ←*ShipTemp*;
Remove *ship k* from *ArrivingList*;
Clear T_{jk}^{delay}, T_{kj}^{delay}, T_{ik}^{delay}, T_{ki}^{delay}, *ShipTemp*;
endwhile
return *SequenceList*

The delay times of ships in different sequences are calculated according to the ships' arrival times and speeds, which can be obtained from *ArrivingList*. Tab. 2 presents pseudo code for ship self-organizing cooperation. Unlike previous algorithms such as in [5] and [7], the main purpose of the algorithm is to decide the entry sequence in *SequenceList* rather than the entry time. In consideration of computation efficiency, every ship can obtain the sequence in *SequenceList* as long as it undertakes computation. Meanwhile, in order to be fair to all, the ships arriving first will be served first as much as possible. It also should be noted that it is assumed that all the ships are willing to make a sacrifice to promote local traffic efficiency. That is to say, there is a trade-off between equity and efficiency in the scheduling model. Furthermore, deceleration is deemed to accomplished instantly, and the safe distances for all the ships are the same for simplification.

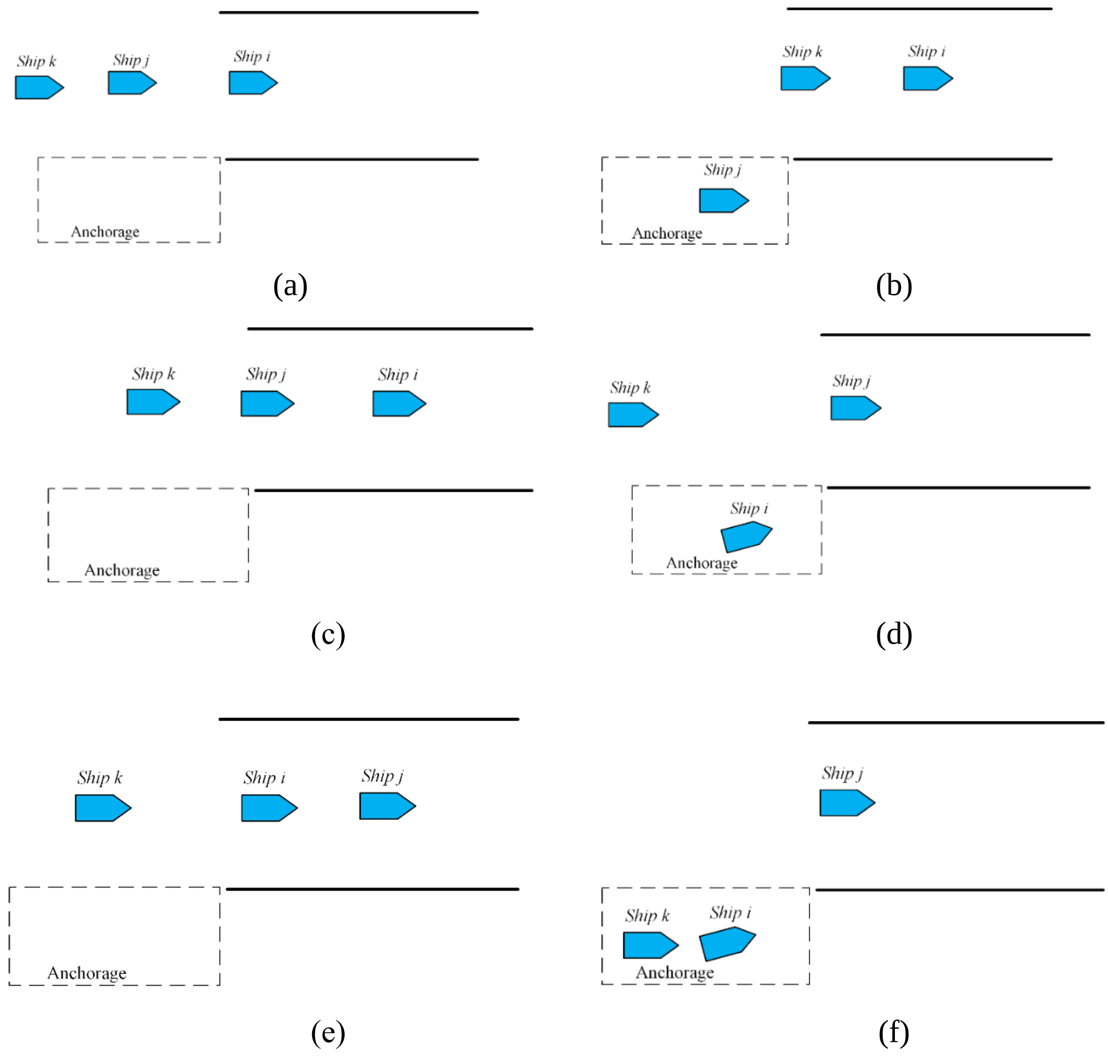


Fig. 3 *ship i*, *ship j* and *ship k* in different sequences of entering the waterway. (a) *ship i* is entering the waterway, *ship j* and *k* are approaching the entrance, (b) *ship i* is navigating in the waterway, *ship k* is entering the waterway, and *ship j* is waiting at the anchorage, (c) *ship i* is navigating in the waterway, *ship j* is entering the waterway, and *ship k* is approaching the entrance, (d) *ship j* is entering the waterway, *ship i* is waiting at the anchorage, and *ship k* is approaching the entrance, (e) *ship j* and *ship i* are navigating in the waterway, and *ship k* is approaching the entrance, (f) *ship j* is entering the waterway, and *ship i* and *ship k* are waiting at the anchorage.

The ship self-organizing cooperation in each round is described as follows. First, own ship looks for the two front ships in *ArrivingList* and then judges whether the two ships change the sequence in *SequenceList*. If the sequence is unchanged, own ship (for example, *ship k* in Fig. 3(a)) makes a further judgement of whether own ship will be influenced by the target ship ahead (for example, *ship j* in Fig. 3(a)) according to Eq. (4), omitting the other hampers. If so, Eq. (22) is the second criterion for changing the sequence. In other words, the better strategy is that the target ship awaits own ship when Eq. (4) and Eq. (22) are both satisfied, as displayed in Fig. 3(b). Otherwise, when either of Eq. (4) or Eq. (22) is not satisfied, it is more effective that own ship enters the waterway afterwards, as indicated in Fig. 3(c).

Comparing the arrival times of two ships ahead in *ArrivingList*, own ship may find that they have already changed the sequence in *SequenceList*. Eq. (25) and Eq. (26) are applied in that circumstance. When $t_i^{in} < t_k^{arr}$, own ship (for example, *ship k* in Fig. 3(d)) confirms whether it will switch the order with the waiting target ship (for example, *ship i* in Fig. 3(d)) according to Eq. (24) and Eq. (25), leaving the others (for example, *ship j* in Fig. 3(d) and Fig. 3(e)) out. Otherwise, the more efficient sequence could be decided in accordance with Eq. (26), as indicated in Fig. 3(f).

When all the ships are cleared from *ArrivingList* and confirm the sequence in *SequenceList*, the final entry sequence of all the ships can be obtained.

Tab. 2 Pseudo code for ship self-organizing cooperation in each round

Function Round Self-organization();	
Input: L_{wat};	% length of waterway
Input: d^s;	% safe distance of ship
Input: *ShipTemp*;	% *ship i* and *j* ($t_i^{arr} < t_j^{arr}$) are the last two ships in succession in *SequenceList* and *ship k* is the first *ship* in *ArrivingList*;
Input: t_i^{arr};	% arrival time of *ship i*
Input: v_i;	% speed of *ship i*
Input: t_j^{arr};	% arrival time of *ship j*
Input: v_j;	% speed of *ship j*
Input: t_k^{arr};	% arrival time of *ship k*
Input: v_k;	% speed of *ship k*
for each *ship k* in *ShipTemp* **do**	
if the sequence between *ship i* and *j* in *SequenceList* is *ship i* and *ship j* in sequence **then**	% the sequence is not updated
if $v_j < v_k$ and $t_k^{arr} - t_j^{arr} < L_{wat} / v_j - (L_{wat} - d^s) / v_k$ **then**	
$T_k^0 \leftarrow Max(0, d^s / v_j - (t_k^{arr} - t_j^{arr}))$;	
$l_{jk} \leftarrow (v_j \times (t_k^{arr} - t_j^{arr} + T_k^0) - d^s) \times v_k / (v_k - v_j)$;	
$T_{jk}^{delay} \leftarrow T_k^0 + (L_{wat} - l_{jk}) / v_j - (L_{wat} - l_{jk}) / v_k$;	
$T_{kj}^{delay} \leftarrow t_k^{arr} - t_j^{arr} + d^s / v_k$;	
if $T_{kj}^{delay} < T_{jk}^{delay}$ **then**	
ShipTemp ← *ship i, k and j* in sequence;	
else	
ShipTemp ← *ship i, j* and *k* in sequence;	

endif
else
ShipTemp ← *ship i, j* and *k* in sequence;
endif
else % sequence has been changed
$t_i^{in} \leftarrow t_j^{arr} + d^s / v_j$;
if $t_i^{in} < t_k^{arr}$ **then**
if $v_i < v_k$ and $t_k^{arr} - t_i^{in} < L_{wat} / v_i - (L_{wat} - d^s) / v_k$ **then**
$T_k^0 \leftarrow Max(0, d^s / v_i - (t_k^{arr} - t_i^{in}))$;
$l_{ik} \leftarrow (v_i \times (t_k^{arr} - t_i^{in} + T_k^0) - d^s) \times v_k / (v_k - v_i)$;
$T_{ik}^{delay} \leftarrow T_k^0 + (L_{wat} - l_{ik}) / v_i - (L_{wat} - l_{ik}) / v_k$;
$T_{ki}^{delay} \leftarrow t_k^{arr} - t_i^{in} + d^s / v_k$;
if $T_{ki}^{delay} < T_{ik}^{delay}$ **then**
ShipTemp ← *ship j, k and i* in sequence;
else
ShipTemp ← *ship j, i* and *k* in sequence;
endif
else
ShipTemp ← *ship j, i* and *k* in sequence;
endif
else % *ship i* and *k* are both waiting at the anchorage;
if $v_i < v_k$ **then**
ShipTemp ← *ship j, k* and *i* in sequence;
else
ShipTemp ← *ship j, i* and *k* in sequence;
endif
endif
endif
endfor
return *ShipTemp*

4.2 Simple SSOCA and secondary SSOCA

A simple SSOCA and secondary SSOCA are offered in addition to SSOCA. In view of the weaker computational capabilities of individual ships, the simple SSOCA model is proposed for easy calculation in Tab. 3. This model only refers to the arrival times of ships as expressed in Eq. (4) and ignores the calculation when the ship ahead could enter the waterway earliest, even if the ship ahead has changed the sequence, such as t_i^{in} in Eq. (23). Hence, the computational process is simplified to a large degree.

Tab. 3 Pseudo code for simple SSOCA

Input: *ArrivingList*
SequenceList ← the first ships in *ArrivingList*;
Remove the first ships from *ArrivingList*;
while *ArrivingList* is not empty **do**
ShipTemp ← Collect the last ships (*ship i*) in *SequenceList* and the first *ship* (*j*) in *ArrivingList*;
if $v_i < v_j$ and $t_j^{arr} - t_i^{arr} < L_{wat} / v_i - (L_{wat} - d^s) / v_j$ **then**
$T_j^0 \leftarrow Max(0, d^s / v_i - (t_j^{arr} - t_i^{arr}))$;
$L_{ij} \leftarrow (v_i \times (t_j^{arr} - t_i^{arr} + T_j^0) - d^s) \times v_j / (v_j - v_i)$;
$T_{ij}^{delay} \leftarrow T_j^0 + (L_{wat} - l_{ij}) / v_i - (L_{wat} - l_{ij}) / v_j$;
$T_{ji}^{delay} \leftarrow t_j^{arr} - t_i^{arr} + d^s / v_j$;
if $T_{ji}^{delay} < T_{ij}^{delay}$ **then**
ShipTemp ← *ship j* and *i* in sequence;
else
ShipTemp ←*ship i* and *j* in sequence;
endif
else
ShipTemp ← *ship i* and *j* in sequence;
endif
SequenceList ←*ShipTemp*;
Remove *ship j* from *ArrivingList*;
Clear T_{ij}^{delay}, T_{ji}^{delay}, *ShipTemp*;
endwhile
return *SequenceList*

In addition, it is noted that the sequence of a ship is just moved one step forward in the SSOCA model. Nevertheless, it is possible that the ship ahead still matches the requirements of both Eq. (4) and Eq. (22) after own ship has moved forward once in *SequenceList*. The further experiment is very necessary to inspect if own ship could continue to move forward for better results. Hence, an algorithm herein named the secondary SSOCA model, the pseudocode for which is shown in Tab. 4, is introduced to test that idea. The algorithm is performed with SSOCA twice, taking the *SequenceList* as the new *ArrivingList* in the secondary SSOCA model.

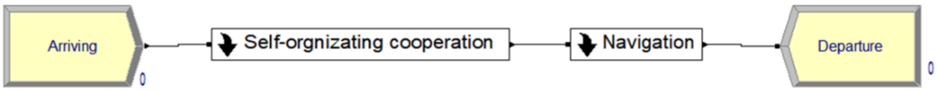
Tab. 4 Pseudo code for secondary SSOCA

Input: *ArrivingList*
Obtain *SequenceList* according to SSOCA;
ArrivingList ← *SequenceList*;
Obtain the second *SequenceList* according to SSOCA;
return the second *SequenceList*

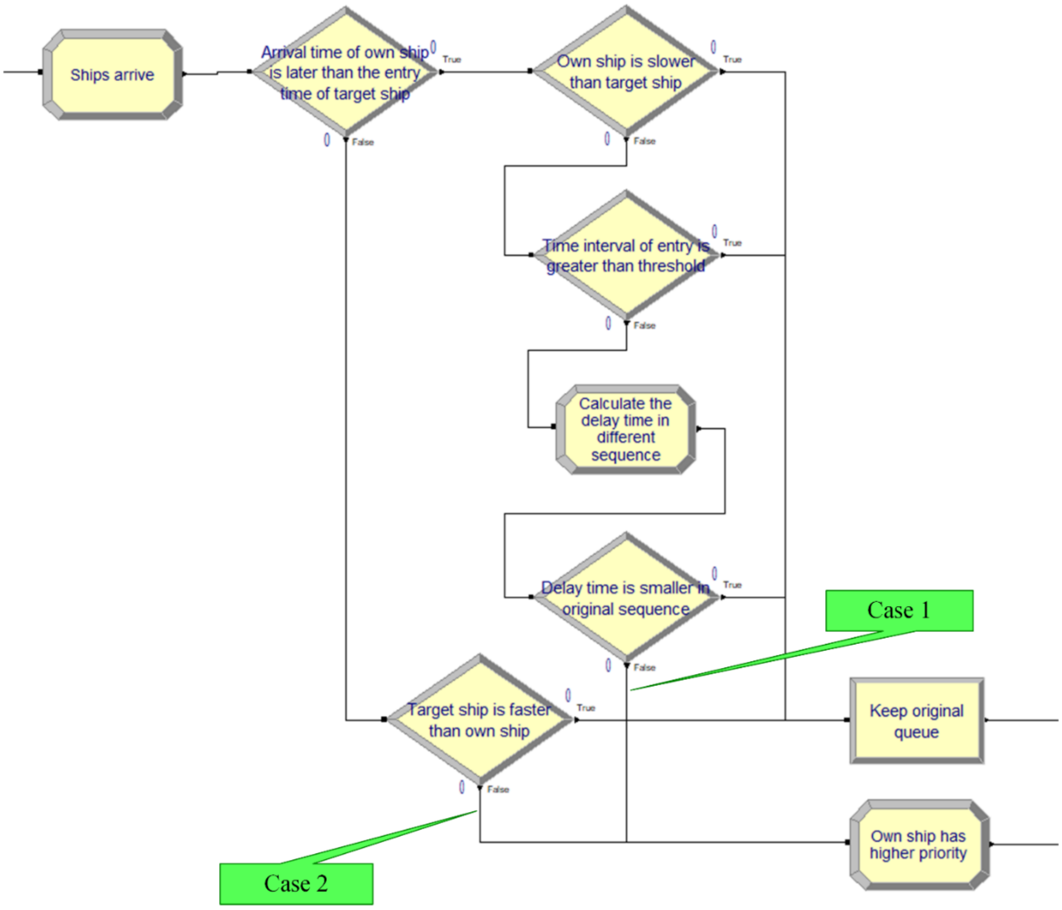
5. Experiments and analysis

5.1 Simulation model

The experiments performed for a restricted one-way waterway are presented in this section. Monte Carlo simulations, which are widely accepted in ship traffic studies, are used to make an approximate evaluation by the law of large numbers, [32]. Arena software (version 14.0) [33] was used to construct a ship traffic simulation model that includes four sub-models: arriving, self-organizing cooperation, navigation and departure. A top-level view of the model is shown in Fig. 4(a). Taking the process as an example, the arriving ships follow some kind of defined probability distribution. A ship's attributes (including arrival time and speed) are assigned randomly when it is created by the ship arrival module. In the ship self-organizing cooperation module in Fig. 4(b), each ship then adjusts the entry sequence using SSOCA. After determining the entry sequence, the ship enters the waterway and navigates under the constraints of safe distance. Finally, the ship leaves the waterway through the ship departure module.



(a)



(b)

Fig. 4 An Arena-based ship traffic simulation model. (a) Top-level view of the model, (b) Modules used for self-organizing cooperation.

In Fig. 4(b), the ship has priority to enter the channel in two cases. In case 1, when the arrival time of the ship is later than the entry time of the previous ship, three requirements must be satisfied: (1) its speed is larger than the previous ship, (2) the time interval between its arrival time and the entry time of previous ship is sufficiently small, and (3) the delay time can be shortened. In case 2, when the arrival time of own ship is earlier than the entry time of the previous ship, the ship has higher priority only when the first requirement in case 1 is satisfied.

5.2 Parameter settings

In accordance with Eqs. (16), (21) and (25), the delay time of a ship is related to the time interval between ships' arrival times, speeds, channel length and safe distances. These simulation parameters and the ranges of the fixed steps over which they can updated (in the last column) are summarized in Tab. 5. According to a statistical analysis of traffic flow data for the *Xiashimen Waterway* in China, the ship arrival rule was subject to a Poisson distribution with a rate of arrival of 3 ships per hour, and the ship speeds obeyed a normal distribution. The base case values of mean speed and standard deviation of speed were designed as 10 knots and 2 knots, respectively. The speeds were confined in the range between 4 and 40 knots. 99.73% of speed values fell within [4, 16] in accordance with three-sigma rule of thumb [34], so that ignoring values outside that range would pose very little influence on the normal probability distribution. Waterway lengths from 8 to 12 nautical miles and safe distances from 1000 to 1400 meters were used in the model. Meanwhile, the safe distances for all ships were the same in a simulation. The above presumptions can be adjusted to real cases via statistical analyses of historical data on such parameters. Five inputs are therefore introduced: rate of arrival, mean speed, standard variation of speed, waterway length and safe distance.

Tab. 5 Parameter settings

Factor	Base case value	Fixed step	Range
Rate of arrival (h^{-1})	3	0.5	2.0-4.0
Mean speed (kn)	10	0.5	9-11
Standard deviation of speed (kn)	2	0.1	1.8-2.2
Waterway length (nm)	10	1	8-12
Safety distance (m)	1200	100	1000-1400

Considering that waiting and ship speed reductions would influence traffic efficiency, the average of delay time is used to undertake a quantitative analysis of those influences. The average delay time is calculated as

$$\frac{1}{m}\sum_{i=1}^{m} T_i^{delay} = \frac{1}{m}\sum_{i=1}^{m}(T_i^{wait} + T_i^{dec}) = \frac{1}{m}\sum_{i=1}^{m}\left[(t_i^{out} - t_i^{arr}) - \frac{L_{wat}}{v_i}\right] \tag{27}$$

where t_i^{out} is the departure time of *ship i*, which is obtained through simulation. On the one hand, (t_i^{out}- t_i^{arr}) displays the actual time consumed for *ship i* owing to waiting or deceleration. On the other hand, L_{wat} / v_i is the shortest time consumed for *ship i* without waiting and deceleration. The difference between the two comprises the delay for *ship i*.

Furthermore, the number of ships in all simulations was set to 2000, and each simulation was repeated 200 times. To ascertain the warm-up period, a scatter plot was used to observe when the system reached a stable state. The horizontal axis of Fig. 5 represents simulation running time, and the vertical axis represents average delay time. As can be seen, 72 hours (approximately 260×10^3 seconds) of warm-up was sufficient.

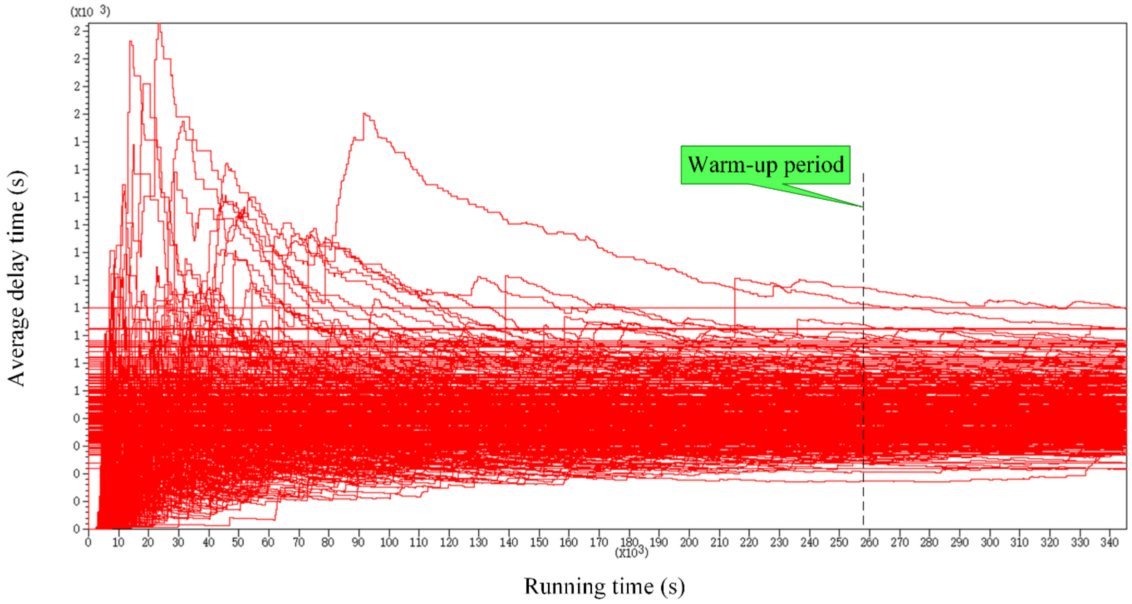


Fig. 5 Scatterplot from the ship traffic simulation model for the base case

5.3 Results and analysis

To undertake a comprehensive analysis, the following five simulation scenarios were used.

1. Rate of arrival is a variable varied by fixed steps over its range, while the other parameters remain unchanged.

2. Mean speed is a variable varied by fixed steps over its range, while the other parameters remain unchanged.

3. Standard deviation of speed is a variable varied by fixed steps over its range, while the other parameters remain unchanged.

4. Waterway length is a variable varied by fixed steps over its range, while the other parameters remain unchanged.

5. Safety distance is a variable varied by fixed steps over its range, while the other parameters remain unchanged.

By removing the data from the warm-up period, Arena generates mean values of the outputs at a 95% confidence level. The average delay times with rate of arrival, mean speed, standard deviation of speed, waterway length and safe distance under the different scenarios are presented in Fig. 6. The performance is analyzed by comparing the ship traffic simulation results determined using the SSOCA model with those of the other models in Fig. 6. In the first place, it can be seen that the factors had significant positive correlations with average delay time, with the exception of mean speed. The average delay time tended to increase with rate of arrival, standard deviation of speed, waterway length and safe distance but experienced a decreasing trend with mean speed. The tendency proves that traffic efficiency would decrease for more frequently arriving ships, smaller mean speeds, greater speed differences, greater waterway lengths and larger safe distances.

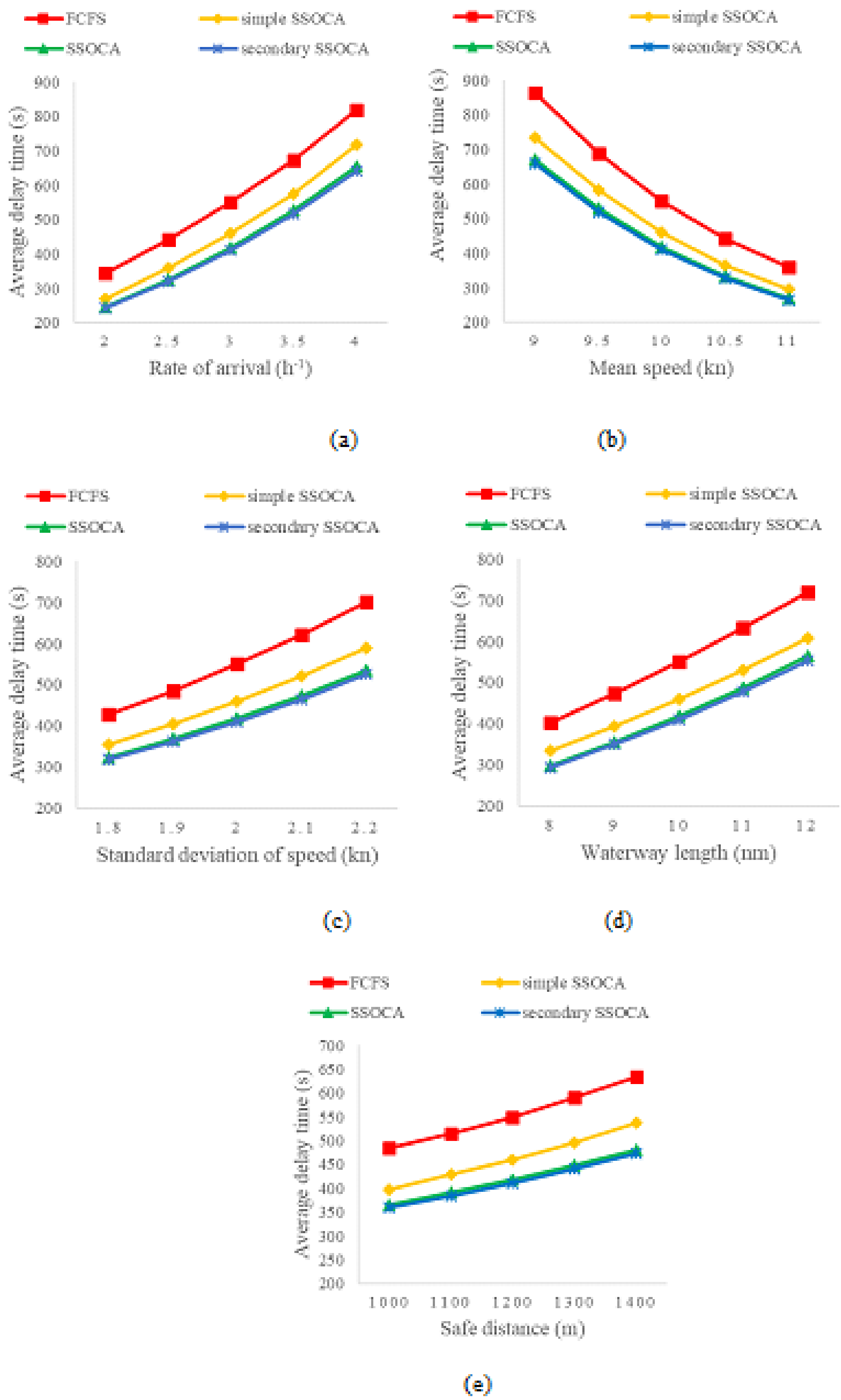


Fig. 6 Average delay times for different (a) rates of arrival, (b) mean speed, (c) standard deviations of speed, (d) waterway lengths and (e) safe distances.

It can be seen that the four lines are almost parallel and that the delay time for the FCFS model was always the highest, which shows that delay time was reduced with the variation of each factor in some of the methods. For the base case, the average delay time was reduced from 550 seconds for FCFS to 418 seconds for SSOCA, i.e., 24.0% of the average delay time was eliminated. This indicates that the adjustments in the sequences of traffic flow in the waterways were greatly improved and the phenomenon of speed deceleration was alleviated when using the SSOCA model. Moreover, in comparison with FCFS, SSOCA's advantage always held under different combinations of traffic flow parameters. It should be noted that four ships or more may change their sequences at higher arrival rates. Under such circumstances, the algorithm could also yield better performance, even if the possible additional interactions exist. The average delay time was reduced to 460 seconds for the simple SSOCA model applied to the base case. Only 16.3% of the reductions in delays showed that simple SSOCA was inferior to SSOCA. The average delay time was further reduced to 410 seconds for the base case using the secondary SSOCA model, which shows that the secondary SSOCA is superior to SSOCA. However, the SSOCA and secondary SSOCA results nearly overlapped, which means that only a small improvement was obtained with the SSOCA model. One reason is that a ship could seldom change its sequence beyond two times, considering the arrival time interval and speed variance between the ships set up in the experiments.

To identify the traffic parameters that contribute significantly to the uncertainties of the models' average delay times , sensitivity analyses [12] of the four models were performed. The results are presented in Tab. 6. Mean value of ship speed was most sensitive to average delay time. Recalling the decrease tendency shown in Fig. 6(b), large ship speeds are beneficial for improving the efficiencies of restricted waterways. The sensitivity of the standard deviation of speed was approximately half the mean speed and was followed by waterway length and rate of arrival. The safety distance was the least sensitive to average delay time. Noting that a constant safety distance was assumed in the simulations, it can be inferred that the assumption did not have an apparent influence on the results. Nevertheless, more investigations examining different safety distance values are necessary. Meanwhile, comparing the four approaches, the sensitivities for SSOCA were basically smaller than those for FCFS and the simple SSOCA model, which reflects the advantage of the SSOCA model vis-à-vis stability.

Tab. 6 Parameter sensitivity of the four models

Method	Rate of arrival	Mean speed	Standard deviation of speed	Waterway length	Safety distance
FCFS	1.71	6.03	3.25	1.88	1.08
SSOCA	1.46	4.81	2.54	1.60	0.83
simple SSOCA	1.60	5.26	2.80	1.64	0.50
secondary SSOCA	1.43	4.71	2.45	1.54	0.81

According to Fig. 5, there was a large uncertainty in the results of the simulation model. The 95% confidence intervals are listed in the 3rd column of **Pogreška! Izvor reference nije pronađen.**. The half width of confidence interval for delay time was approximately 1.5% of the mean value, which shows that the mean values were adequately precise when the simulations were repeated 200 times. Furthermore, the smaller confidence interval for the SSOCA model was found to be comparable with those for FCFS and the simple SSOCA, indicating that satisfactory convergences can be achieved.

Tab. 7 Comparisons among FCFS, SSOCA, simple SSOCA, secondary SSOCA and all permutation for the base case

Model	Average delay time (s)	Confidence intervals (s)	Computation cost	Optimization
FCFS	550	[542.28 557.72]	0	None
SSOCA	418	[412.21 423.79]	m	Local
Simple SSOCA	460	[453.04 466.96]	m	Local
Secondary SSOCA	410	[404.26 415.74]	$2m$	Local
All permutation	Unknown	Unknown	$m!$	Global

In addition, as an important part of computational complexity theory, the analysis of algorithms [35] provides theoretical estimates for the resources needed by any algorithm that solves a given computational problem. These estimates provide insights into reasonable directions in the search for efficient algorithms. In consideration of low computational complexity, a computation cost model substituted for usual run-time analysis [36] was applied to compare the algorithms. The computation cost model assigns a constant cost to every computation for each ship. The amounts of computations for the different models are also listed in **Pogreška! Izvor reference nije pronađen.**. In the FCFS model, ships are only required to report their own details to the VTS, and no computations are needed. A ship needs only make a single set of computations in the SSOCA and simple SSOCA models, but twice the computations are made in the secondary SSOCA model, which is unsatisfactory. Furthermore, as far as the central control mode is concerned, the number of computations even reaches m factorial when using the all permutation algorithm with m ships. Although the all permutation model is more likely to realize global optimizations, computational cost increases greatly with the number of ships. In a word, the SSOCA model is the best option for balancing performance and computation.

6. Conclusions

The SSOCA model, a self-organizing cooperation strategy, has been proposed in this paper. Delay time is presented as an evaluation indicator for traffic efficiency and includes wait time and deceleration time. A mathematic model of delay time with different sequences was deduced based on following theory. A self-organizing cooperation model was offered that enables slower ships to assign higher priorities to faster ships. By obtaining information on nearby target ships, each ship can interactively choose the optimum sequence in accordance with a local benefit delay time model. An Arena-based ship traffic simulation model was constructed to compare the SSOCA model with the FCFS model and two other associated models. The results show that SSOCA model can effectively reduce the average delay times suffered by ships to acceptable levels. Furthermore, the model can also obtain satisfying results with different combinations of factors, including high arrival rates. Moreover, a trade-off between efficiency and computation is achieved by allocating computation burden to each ship.

In future work, an anchorage and berth cooperation model can be considered by taking their capacities as constraints in order to make the self-organizing model more reasonable and practical. In addition, ultra-large-scale ships can only enter waterways during periods of high tides due to their deep draughts. In addition, pilotage is compulsory in particular areas, but there are often too few pilots for ships. These restrictions can also be considered in the models.

Acknowledgement

This research was supported by the National Natural Science Foundation of China (51479157), and National Key Technologies Research & Development Program (2017YFC0804900, 2017YFC0804904).

REFERENCES

[1] U. S. Kim, S. W. Lee, and C. H. Yang, "Risk Analysis of VLCS based on Trend of Cargo Growth," (in English), *Asian Journal of Shipping & Logistics*, vol. 25, no. 1, pp. 19-39, 2009. https://doi.org/10.1016/s2092-5212(09)80010-2

[2] O. Zwikael and J. Smyrk, *Project Management for the Creation of Organisational Value*. (in English), Springer London, 2011, p. 196. https://doi.org/10.1007/978-1-84996-516-3

[3] K. Hsu, T. Kou, and Z. Wu, "Optimum scheduling model for ship in/outbound harbor in one-way traffic fairway," (in Chinese), *J. Dalian Maritime Univ*, vol. 4, no. 24, p. 038, 2008.

[4] D. Mavrakis and N. Kontinakis, "A queueing model of maritime traffic in Bosporus Straits," (in English), *Simulation Modelling Practice and Theory*, vol. 16, no. 3, pp. 315-328, Mar 2008. https://doi.org/10.1016/j.simpat.2007.11.013

[5] X. Zhang, J. Lin, Z. Guo, and T. Liu, "Vessel transportation scheduling optimization based on channel–berth coordination," (in English), *Ocean Engineering*, vol. 112, pp. 145-152, 2016. https://doi.org/10.1016/j.oceaneng.2015.12.011

[6] X. Zhang, X. Chen, M. Ji, and S. Yao, "Vessel Scheduling Model of a One-Way Port Channel," (in English), *Journal of Waterway Port Coastal & Ocean Engineering*, vol. 143, no. 5, p. 04017009, 2017. https://doi.org/10.1061/(asce)ww.1943-5460.0000388

[7] J. F. Zhang, T. A. Santos, C. G. Soares, and X. P. Yan, "Sequential ship traffic scheduling model for restricted two-way waterway transportation," (in English), *Proceedings of the Institution of Mechanical Engineers Part M-Journal of Engineering for the Maritime Environment*, vol. 231, no. 1, pp. 86-97, Feb 2017. https://doi.org/10.1177/1475090215621580

[8] G. Praetorius, E. Hollnagel, and J. Dahlman, "Modelling Vessel Traffic Service to understand resilience in everyday operations," (in English), *Reliability Engineering & System Safety*, vol. 141, pp. 10-21, Sep 2015. https://doi.org/10.1016/j.ress.2015.03.020

[9] F. G. Coomber, M. D'Inca, M. Rosso, P. Tepsich, G. N. di Sciara, and A. Moulins, "Description of the vessel traffic within the north Pelagos Sanctuary: Inputs for Marine Spatial Planning and management implications within an existing international Marine Protected Area," (in English), *Marine Policy*, vol. 69, pp. 102-113, Jul 2016. https://doi.org/10.1016/j.marpol.2016.04.013

[10] F. Xiao, L. Han, C. V. Gulijk, and B. Ale, "Comparison study on AIS data of ship traffic behavior," (in English), *Ocean Engineering*, vol. 95, no. 3, pp. 84-93, 2015. https://doi.org/10.1016/j.oceaneng.2014.11.020

[11] X. Yang, X. Yang, and Z. Wang, *et al.*. "Marine Traffic Simulation of Congestion in Restricted Waterway with Mont Carlo Method," (in Chinese), *Navigation of China,* vol. 39, no. 2, pp. 40-44, 2016. http://dx.chinadoi.cn/ 10.3969/j.issn.1000-4653.2016.02.010

[12] H. Wang, J. Liu, K. Liu, J. Zhang, and Z. Wang, "Sensitivity analysis of traffic efficiency in restricted channel influenced by the variance of ship speed," (in English), *Proceedings of the Institution of Mechanical Engineers Part M Journal of Engineering for the Maritime Environment,* no. 2, p. 147509021772054, 2017. https://doi.org/10.1177/1475090217720549

[13] D. Zubillaga *et al.*, "Measuring the Complexity of Self-organizing Traffic Lights," (in English), *Entropy*, vol. 16, no. 5, pp. 2384-2407, 2014. https://doi.org/10.3390/e16052384

[14] F. V. Westrenen and G. Praetorius, "Maritime traffic management: a need for central coordination?," (in English), *Cognition Technology & Work,* vol. 16, no. 1, pp. 59-70, 2014. https://doi.org/10.1007/s10111-012-0244-5

[15] C. Gershenson, "Self-organizing urban transportation systems," (in English), *Physics*, pp. 269-279, 2009. https://doi.org/10.1007/978-3-642-24544-2_15

[16] T. C. Collier and C. Taylor, "Self-organization in sensor networks," (in English), *Journal of Parallel & Distributed Computing,* vol. 64, no. 7, pp. 866-873, 2004. https://doi.org/10.1016/j.jpdc.2003.12.004

[17] W. R. Ashby, "Principles of the self-organizing dynamic system," (in English), *Journal of General Psychology,* vol. 37, no. 2, p. 125, 1947. https://doi.org/10.1080/00221309.1947.9918144

[18] L. A. Safonov, E. Tomer, V. V. Strygin, Y. Ashkenazy, and S. Havlin, "Multifractal chaotic attractors in a system of delay-differential equations modeling road traffic," (in English), *Chaos,* vol. 12, no. 4, p. 1006, 2002. https://doi.org/10.1063/1.1507903

[19] Y. Shu, W. Daamen, L. Han, and S. P. Hoogendoorn, "Influence of external conditions and vessel encounters on vessel behavior in ports and waterways using Automatic Identification System data," (in English), *Ocean Engineering*, vol. 131, pp. 1-14, 2017. https://doi.org/10.1016/j.oceaneng.2016.12.027

[20] B. S. Kerner, "Experimental Features of Self-Organization in Traffic Flow," (in English), *Physical Review Letters*, vol. 81, no. 17, pp. 3797-3800, 1998. https://doi.org/10.1103/physrevlett.81.3797

[21] D. P. Watling and G. E. Cantarella, "Model Representation & Decision-Making in an Ever-Changing World: The Role of Stochastic Process Models of Transportation Systems," (in English), *Networks & Spatial Economics*, vol. 15, no. 3, pp. 843-882, Sep 2015. https://doi.org/10.1007/s11067-013-9198-2

[22] B. Płaczek, "A self-organizing system for urban traffic control based on predictive interval microscopic model," (in English), *Engineering Applications of Artificial Intelligence*, vol. 34, no. 3, pp. 75-84, 2014. https://doi.org/10.1016/j.engappai.2014.05.004

[23] P. G. Gipps, "A behavioural car-following model for computer simulation," (in English), *Transportation Research Part B: Methodological,* vol. 15, no. 2, pp. 105-111, 1981. https://doi.org/10.1016/0191-2615(81)90037-0

[24] L. Qi, Z. Zheng, and L. Gang, "A cellular automaton model for ship traffic flow in waterways," (in English), *Physica A: Statistical Mechanics and its Applications,* vol. 471, no. Supplement C, pp. 705-717, 2017/04/01/ 2017. https://doi.org/10.1016/j.physa.2016.12.028

[25] Z. Pietrzykowski and J. Magaj, "The Problem of Route Determination in Ship Movement in a Restricted Area," (in English), *Annual of Navigation*, vol. 19, no. 2, pp. 53-69, 2012. https://doi.org/10.2478/v10367-012-0018-4

[26] Y. Wang, J. Zhang, X. Chen, X. Chu, and X. Yan, "A spatial–temporal forensic analysis for inland–water ship collisions using AIS data," (in English), *Safety science*, vol. 57, pp. 187-202, 2013. https://doi.org/10.1016/j.ssci.2013.02.006

[27] D. G. Rand and M. A. Nowak, "Human cooperation," (in English), *Trends in Cognitive Sciences*, vol. 17, no. 8, pp. 413-425, 2013/08/01/ 2013. https://doi.org/10.1016/j.tics.2013.06.003

[28] R. Karapin and L. Feldman, "Cooperation," (in English), *POLITY*, vol. 49, no. 3, pp. 325-327, 2017. https://doi.org/10.1086/692532

[29] C. Gershenson and D. A. Rosenblueth, "Self-organizing traffic lights at multiple-street intersections," (in English), *Complexity*, vol. 17, no. 4, pp. 23–39, 2012. https://doi.org/10.1002/cplx.20392

[30] H. Greidanus, M. Alvarez, T. Eriksen, and V. Gammieri, "Completeness and Accuracy of a Wide-Area Maritime Situational Picture based on Automatic Ship Reporting Systems," (in English), *JOURNAL OF NAVIGATION*, vol. 69, no. 1, pp. 156-168, 2016. https://doi.org/10.1017/s0373463315000582

[31] J. Zhang, D. Zhang, X. Yan, S. Haugen, and C. G. Soares, "A distributed anti-collision decision support formulation in multi-ship encounter situations under COLREGs," (in English), *Ocean Engineering*, vol. 105, pp. 336-348, 2015. https://doi.org/10.1016/j.oceaneng.2015.06.054

[32] F. Goerlandt and P. Kujala, "Traffic simulation based ship collision probability modeling," (in English), *Reliability Engineering & System Safety*, vol. 96, no. 1, pp. 91-107, 2011. https://doi.org/10.1016/j.ress.2010.09.003

[33] W. D. Kelton, R. P.Sadowski, and D. T.Sturrock, *Simulation with Arena(Third Edition)*. (in Chinese), China Machine Press, 2007, pp. 206-227.

[34] T. S. Chang, K. Tone, and C. H. Wu, "DEA models incorporating uncertain future performance," (in English), *European Journal of Operational Research*, vol. 2015, no. 2, pp. 532-549, 2015. https://doi.org/10.1002/9781118946688.ch30

[35] D. H. Greene and D. E. Knuth, "The analysis of algorithms," (in English), *Proc International Congress of Mathematicians*, vol. 1, pp. 269-274, 1970.

[36] C. Asanya and R. Guha, "Direct private query in location-based services with GPU run time analysis," (in English), *The Journal of Supercomputing*, journal article vol. 71, no. 2, pp. 537-573, February 01 2015. https://doi.org/10.1007/s11227-014-1309-4

Submitted: 06.12.2017.
Accepted: 14.03.2018.

Hongbo Wang[1, 2], whonbo@hotmail.com
Jingxian Liu[1, 2, 3], ljxteacher@sohu.com
Jinfen Zhang[3,*], Corresponding Author, jinfen.zhang@whut.edu.cn
Kezhong Liu[1], kzliu@whut.edu.cn
Xugang Yang[1], yxg123@whut.edu.cn
Qing Yu[1], qing.yu@whut.edu.cn
[1]School of Navigation, Wuhan University of Technology, Wuhan, China
[2]Hubei Inland Shipping Technology Key Laboratory, Wuhan University of Technology, Wuhan, China
[3]National Engineering Research Center for Water Transport Safety, Wuhan University of Technology, Wuhan, China