Journal of
*Marine Science and Engineering*

*Article*

# Static and Discrete Berth Allocation for Large-Scale Marine-Loading Problem by Using Iterative Variable Grouping Genetic Algorithm

**Dong Yin [1], Yifeng Niu [1], Jian Yang [2,*] and Shaobo Yu [3]**

1 College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China
2 School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China
3 Army Infantry College of PLA, Nanchang 330100, China
* Correspondence: yangjianxin@scut.edu.cn; Tel.: +86-020-8711-1289

**Abstract:** In this paper, we study the static discrete berth allocation problems (BAPs) for large-scale time-critical marine-loading scenarios. The objective is to allocate the vessels to different types of berths so that all the vessels can be loaded within the minimum time under the tidal condition. The BAP is formalized as a min–max problem. This problem is rather complex as the vessels and berths are quite numerous in the large-scale marine-loading problem. We analyze this problem from a novel perspective, and find out that this problem has the characteristic of partially separable. Therefore, the iterative variable grouping genetic algorithm (IVGGA) is designed to search the near-optimal berth allocation plans. The vessels and berths are divided into subgroups, and the genetic algorithm (GA) is applied to generate the near-optimal berth allocation plans in each subgroup. To achieve the balance of loading tasks among subgroups, we propose reallocating some vessels among subgroups according to the berth allocation plans in subgroups. To guarantee the convergency of the algorithm, an iterative vessel reallocation policy is devised considering the loading tasks of different types of berths. We demonstrate the proposed algorithm in dealing with large-scale BAPs through numerical experiments. According to the results, we find that the proposed algorithm would have good performance when the number of vessels in each subgroup are kept in medium scale. Compared with the original GA, our algorithm shows the effectiveness of the iterative variable grouping strategy. The performance of our algorithm is almost not changed as the number of vessels and berths increases. The proposed algorithm could obtain efficient berth allocation plans for the large-scale marine-loading problem.

**Keywords:** discrete berth allocation problem; separable optimization problem; iterative variable grouping genetic algorithm; vessel reallocation

check for updates

**Citation:** Yin, D.; Niu, Y.; Yang, J.; Yu, S. Static and Discrete Berth Allocation for Large-Scale Marine-Loading Problem by Using Iterative Variable Grouping Genetic Algorithm. *J. Mar. Sci. Eng.* **2022**, *10*, 1294. https://doi.org/10.3390/jmse10091294

Academic Editor: Václav Píštěk

Received: 12 August 2022
Accepted: 7 September 2022
Published: 13 September 2022



## 1. Introduction

Maritime transport is one of the main transportation methods in military and civil fields. In the civil field, it is essential to a country as it plays a major role in international trading that can sustain economic development [1]. In the military field, maritime transport is also the first choice to transport people and supplies cross the sea [2]. Meanwhile, maritime transportation is one of the efficient methods in large-scale time-critical missions, such as humanitarian aid and disaster relief (HADR) actions [3]. Maritime transportation could be used to evacuate refugees from disaster areas to safe places or deploy medical equipment, food, and other relief materials to the disaster areas [4,5]. The earthquakes in Haiti in 2010 and 2021 killed thousands of people [6]. One of the main reasons for this tragic result is there is a lack of quick and orderly disaster relief actions after earthquakes. When a disaster occurs, the first 72 h are very critical for lifesaving [7]. As large-scale disasters occur from time to time, HADR actions are continuous requirements for every country. In order to alleviate the disaster impact in the affected areas, the essential objective of maritime transportation under these circumstances is to fulfill the mission within the minimum time, such that the supplies and relief teams can be distributed to the disaster regions timely. Therefore, when there is a large quantity of vessels to be used to transport the supplies and relief teams, the available berths are required to be fully used so that the vessels can be loaded as quickly as possible.

In large-scale time-critical maritime-loading missions, hundreds of different types of vessels are required to be loaded. To improve the loading efficiency, the berth-planning problem should be considered comprehensively, as it determines the following rescue resource allocation and emergency distribution. The berth allocation problem (BAP) is one of the key decisions of the berth-planning problem. According to the BAP, the actual berthing positions, as well as the start and end berthing times of various types of vessels, should be determined [8]. In the large-scale time-critical maritime transport problem, an effective berth allocation plan is crucial for (i) minimizing the overall loading time, and (ii) guaranteeing the robustness of the berth allocation plan.

The BAP could be classified as static BAP and dynamic BAP [9]. A static berth allocation model mainly considers allocating the candidate vessels to appropriate berths so that vessels are able to arrive at the assigned berths in the required time, while a dynamic berth allocation model considers allocating the candidate vessels with known future arrival information. In the HADR-related time-critical maritime transportation problem, as there are many unexpected factors, it may be inefficient to generate the dynamic berth allocation plans based on the specific arrival/departure information of vessels [10]. In order to improve the marine transportation efficiency, the main objective of the berth allocation problem is set to generate the berth allocation plan that could maximize the utilization of the berths considering the vessels to be loaded. We assume that the arrival time of vessels and the loading equipment could meet the requirement of the berth allocation plan. Therefore, the static BAP is important for the large-scale time-critical marine-loading missions. The planning horizon is based on the scale of marine-loading problem. It may range from one to several days [11].

The berths could be classified into several types according to their loading capabilities and available time windows [12]. Meanwhile, there are many different types of vessels to be loaded in large-scale maritime transportation missions. These vessels are classified into several levels based on their sizes and loading weights, namely small-tonnage vessels, medium-tonnage vessels, large-tonnage vessels and heavy-tonnage vessels [13]. In the real environment, different types of berths are capable of loading vessels with different sizes and tonnages [14]. In addition to that, there are regular strong tide waves that would have obvious influences on the water depths of many ports [8]. Some large-/heavy-tonnage vessels may be restricted by the tidal condition as they could port in the berths only when the water depth reaches certain conditions [15]. The objective of the BAP is to improve the utilization rates of the available berths by considering the constraints of vessels and berths. As there may be hundreds of vessels, and there are dozens of berths, the static BAP of large-scale time-critical maritime transport scenarios is, therefore, a high-dimensional complex optimization problem.

The genetic algorithm (GA) is widely used to resolve different types of BAPs [16]. It is a highly parallel, stochastic, and adaptive optimization algorithm that is based on the "survival of the fittest" principle, which evolves in the direction of optimal solution [17]. As there are hundreds of vessels to be loaded, the original GA is difficult to converge to the near-optimal solution as there would be too many genes in one chromosome. The cooperative coevolutionary (CC) strategy is introduced in [18] to deal with large-scale optimization problems. The main idea of CC is to decompose the original high-dimensional problem into a set of lower-dimensional subproblems, which are easier to solve. Typically, each subproblem is assigned to a subpopulation of candidate solutions according to the characteristics of the initial problem, which would then evolve according to the adopted

Genetic Algorithm. During the iterative optimization process, the cooperative information interaction among subcomponents happens in the evaluation of the fitness value.

This paper studied the static BAP of vessels of various types, considering the differences in the loading capabilities of berths and the tidal condition. The objective was set to obtain a berth allocation plan that loads all the vessels within the minimum time. The computation complexity of the original large-scale static BAP is rather high. Therefore, we further discuss this problem. As the berths that belong to one specific type are of the same capability, the overall loading time of each berth is determined by the number of assigned vessels of different types. The allocation policies of vessels are not strongly coupled. According to this feature of the BAPs and the CC strategy, we further analyze the BAPs based on the "separable" theory. We find that the large-scale BAPs are partially separable. To our knowledge, there is has not been a similar discussion on the BAPs before. Based on this characteristic, we devise the iterative variable grouping genetic algorithm (IVGGA). This algorithm searches for the approximate optimal berth allocation plans by using the iterative optimization method. According to this novel algorithm, the available vessels and the berths are grouped into subgroups as evenly as possible. The genetic algorithm is used to search the near-optimal berth allocation plan in each subgroup [19]. The maximum loading time of all these subgroups would be minimized when the loading tasks of subgroups are balanced. Therefore, we propose making a minor adjustment to the allocations of vessels among subgroups iteratively based on the near-optimal berth allocation plans generated by the GA. As there are various types of vessels and berths in each subgroup, we propose determining the vessels to be reallocated between subgroups with consideration of the overall loading time of different types of berths. According to the numerical experiment, we find that when the number of vessels in each subgroup is set to medium size, the algorithm can generate good results. Our method is demonstrated in large-scale marine-loading scenarios and compared with the original genetic algorithm. The results show that the proposed algorithm could generate approximate global optimal berth allocation plans for large-scale BAPs.

The rest of this paper is organized as follows. The related works are presented in Section 2, and we model the marine-loading problem in Section 3. In Section 4, the iterative variable grouping genetic algorithm is presented. A report of the experimental results is presented in Section 5 and discussion and conclusions in Section 6.

## 2. Related Works

### *2.1. Berth Allocation Problem*

The BAP has attracted significant attention among researchers. Imai et al. formulated a static berth allocation problem as a nonlinear integer program to minimize both the total time that the vessels spend at the berth and the degree of dissatisfaction incurred by the berthing order [20]. Ren and Tian proposed a tree search method based on a greedy heuristic algorithm considering the priority of goods [21]. Imai, Nishimura, and Padimitriou proposed solving this problem by using the grouped genetic algorithm [22]. According to their method, they added the priority factor to the operation time of each vessel to the objective function. Xu Qinghua proposed searching the optimal berth allocation plan by virtue of the multiobjective-programming theory and genetic algorithm [23]. Çağatay Iris et al. proposed dealing with the complex berth allocation and quay crane assignment problems that consider time-variant/-invariant quay crane allocation policy by using the set-partitioning models [24]. According to their method, the near-optimal solutions were obtained by using the simplified avenues. Eduardo Tadeu Bacalhau et al. proposed dealing with the dynamic berth allocation problem by using the hybrid genetic algorithm [16]. Their optimization objective was to minimize the overall loading time of all the lined vessels. The dynamic programming with state space reductions (DPSSR) method was used to reduce the local solution space in each iteration. This method could improve the efficiency of the genetic algorithm. However, this method struggles to deal with a large amount of candidate vessels and berths. Dongsheng Xu et al. studied the berth allocation problem in container terminals in which the assignment of vessels to berths was limited by water depth and tidal condition [25]. In their research, the time period was divided into two periods, namely, the low-water period (LW), defined in [0, T], and the high-water period (HW), defined in [T, ∞]. They formulated the problem as a mixed-integer linear programming (MILP) problem, which was proved to be able to solve the dynamic and static berth allocation problem. Eduardo Lalla-Ruiz et al. studied the BAP under time-dependent limitations [26]. They propose that the time periods defined by Dongsheng Xu in [25] would lead to infeasible solutions, and they proposed an alternative mathematical formulation based on the generalized set-partitioning problem, which considers a multiperiod-planning horizon and includes constraints related to berth and vessel time windows. Xavier Schepler et al. studied the discrete berth allocation problem considering the stochastic arrival times of vessels. They proposed the proactive/reactive combined approach, and the approach was proved to be efficient in dealing with the BAP with uncertain arrival times of vessels [27]. These research works have not considered the different types of vessels and berths with different load capability, which is too oversimplified to deal with the real problem. Ming-Wei Li et al. proposed a PSO-based method to deal with the multiobjective optimization problem considering the minimum additional trucking distance and the port time of vessels [14]. Their method was proved to be efficient in dealing with the multicategory vessels and multitype berths allocation problem. Çağatay Iris et al. proposed dealing with the berth allocation problem by using the adaptive large neighborhood search method [28]. According to their method, parts of the current solution were destroyed by a destroy operator; then, the remaining partial solution was reconstructed by a repair operator at each iteration. Their method greatly inspired us. They considered the influences of the uncertainty of vessel arrivals and the fluctuation in the container-handling rate of quay cranes [29]. The balance between efficiency, robustness, and recoverability was discussed. According to their study, the vessel-specific buffer times help to guarantee the plan to be robust to the possible fluctuations in the arrival times of vessels.

In most of the existing research, the authors considered dealing with small-/medium-scale problems, and the number of vessels to be loaded were no more than one hundred [27,30]. The existing methods could deal with these problems efficiently. However, there is little research on the large-scale BAP.

### 2.2. Cooperative Coevolutionary Method

Potter and Jong proposed the cooperative coevolutionary approach to deal with the optimization problems with a complex structure [18]. The global problem was decomposed into several subproblems that were defined in subranges, and the complete solutions were obtained by assembling representative members of each of the species. According to their specific problem, the fitness value was determined by the highly coupled elements. In each iteration, the fitness values of each species were relative to the updated information of other species. Yuping Wang et al. studied the formula-based variable grouping method [31]. They proposed determining the separable and nonseparable variables according to the form of the optimization objective function. They did not discuss the influences of the constraint conditions on the separability of the problem. Giuseppe A. Trunfio stated that most real-world optimization problems are partially separable. Therefore, they generated objective functions that were between separable and fully nonseparable [32]. The level of separability is considered as a measure of the difficulty of an optimization problem. They presented a new adaptive algorithm to enhance the efficiency of the CC algorithm. Elahe Sadat Hosseini et al. proposed a hierarchical subchromosome genetic algorithm (HSC-GA) to optimize the design of wireless sensor networks [33]. According to this method, the genetic algorithm is applied on each subchromosome separately. In the research [34], the global optimization variables were coded into a hierarchical chromosome; the chromosome was separated into several subchromosomes. The defined subchromosome grouped the closely related blocks together. In a future step, these subchromosomes could form the upper-level building blocks.

According to the existing research, we find that the original CC methods mainly consider the optimization problem that the fitness value is impacted by the variables in different subcomponents, and the global optimal solution would be obtained as the coevolution of variables in multiple subcomponents. This paper proposes applying this method to deal with the large-scale BAP according to the characteristics of the problem. We propose defining the available subdomains of variables by using the iterative variable grouping genetic algorithm in the BAP. The vessels would be classified into subgroups. To achieve the balance of loading tasks of berths among subgroups, some vessels would be reallocated between subgroups according to the optimal berth allocation plans generated by the genetic algorithm in each subgroup. The near-optimal berth allocation plan would be obtained by applying the "local optimization, vessels reallocation" iteration.

## 3. Modeling Large-Scale Berth Allocation Problem

### *3.1. Problem Analysis*

There are n vessels (i.e., vessels 1, 2, . . . , n) and m berths (i.e., berths 1, 2, . . . , m). For $i = 1, 2, \ldots, n$, vessel $i$ has the features of loading time $t_i$ and tonnage weight $w_i$. For $j = 1, 2, \ldots, m$, berth $j$ has the feature of loading capability. According to the characteristics of the BAP, the following constraints should be considered.

#### 3.1.1. Loading Capability

According to the tonnage weight, the vessels are classified as heavy-tonnage vessels, large-tonnage vessels, medium-tonnage vessels, and small-tonnage vessels [30]. The vessels are linearly indexed from heavy-tonnage vessels to small-tonnage vessels. Therefore, $w_{i-1} \geq w_i$, for $i = 2, 3, \ldots, n$. We assume that the number of heavy-tonnage vessels, large-tonnage vessels, medium-tonnage vessels, and small-tonnage vessels to be loaded are $n_h$, $n_l$, $n_{mid}$, and $n_s$ respectively.

We classify the berths into four types according to their loading capabilities. The type *i* berths could berth and load all kinds of vessels, and the type II berths could berth and load the large-tonnage vessels, the medium-tonnage vessels, and small-tonnage vessels. The type III berths could berth and load the medium-tonnage vessels and small-tonnage vessels. The type IV berths could berth and load small-tonnage vessels. The feasible vessel berth allocation policies are depicted in Figure 1.

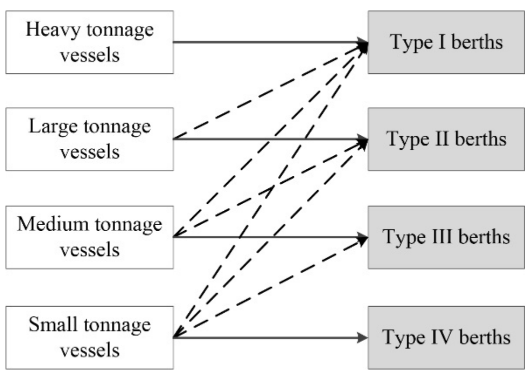



**Figure 1.** The feasible vessel berth allocation policies; the solid lines indicate the preference allocation policy of each type of vessel, and the dashed lines indicate the alternative allocation policy of each type of vessel.

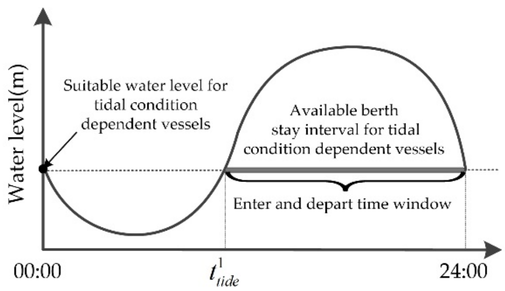
The berths are linearly indexed in such a way that at any point in time, berth $j$ is able to serve all types of vessels that berth $j + 1$ could serve ($j = 1, 2, \ldots, m - 1$). According to the loading requirements of vessels, the feasible solutions of vessel berth allocation policies should satisfy Equation (1).

$$\begin{aligned} &M_s^p \in [1, m], M_{mid}^p \in [1, m - m_1], \\ &M_l^p \in [1, m - m_1 - m_2], M_h^p \in [1, m - m_1 - m_2 - m_3] \end{aligned} \tag{1}$$

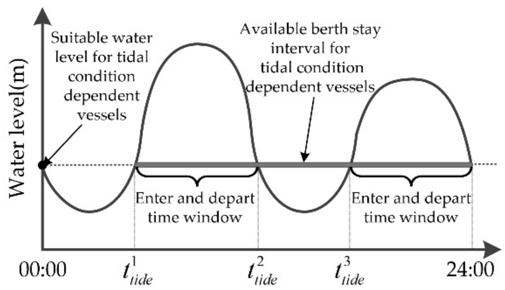
where $M_s^p$ is the vessel berth assignment policy for small vessels. $M_{mid}^p$ is the vessel berth assignment policy for medium vessels. $M_l^p$ is the vessel berth assignment policy for large vessels. $M_h^p$ is the vessel berth assignment policy for heavy vessels. $m_1$ is the number of type IV berths. $m_2$ is the number of type III berths. $m_3$ is the number of type II berths.

#### 3.1.2. The Tidal Condition

Many megaports (e.g., Port of Shanghai, Port of Tianjin, and Port of Hamburg) are tidal ports [8]. To define our problem mathematically, we first discuss the natural condition of the megaport. There are semidiurnal tidal harbors and diurnal tidal harbors [35]. These harbors are characterized as unfixed water depths. In the diurnal tide harbors, the timeline of a day can be divided into two time periods, $[0, t_{tide}^1]$ and $[t_{tide}^1, 24]$, according to the water depth of the chosen port, where $[0, t_{tide}^1]$ is the low-water period and $[t_{tide}^1, 24]$ is the high-water period. In the semidiurnal tidal harbors, the timeline of a day can be divided into four time periods, $[0, t_{tide}^1]$, $[t_{tide}^1, t_{tide}^2]$, $[t_{tide}^2, t_{tide}^3]$, and $[t_{tide}^3, 24]$, according to the water depth of the chosen port. $[0, t_{tide}^1]$ and $[t_{tide}^2, t_{tide}^3]$ are the low-water periods and $[t_{tide}^1, t_{tide}^2]$ and $[t_{tide}^3, 24]$ are the high-water periods. Here, $t_{tide}^1$ and $t_{tide}^3$ represent the time points at which the water level has reached a certain threshold where the berthing of large/heavy vessels becomes less restrictive, and time 0 and $t_{tide}^2$ are the time points at which the water level has fallen to a certain threshold. Therefore, some large-/heavy-tonnage vessels could only enter the tidal berth in the high-water time period. Meanwhile, other types of vessels could be loaded both at the low-water and high-water times. The change in water level in tidal ports is shown in Figure 2.



(a). The change of water level in diurnal tide harbors



(b). The change of water level in semidiurnal tidal harbors

**Figure 2.** The change in water level in the tidal ports.

As shown in Figure 2, as the water level of the tidal harbors keeps on changing, the tide-condition-dependent vessels can enter and depart the harbors only when the water level reaches a certain height. On the other hand, when one vessel is in the tidal harbor, it can stay at the harbor until the high-water time period in the next tide circle. When berths are assigned with an inappropriate loading plan in the tidal-condition berth, it would lead to unnecessary idle time for some berths. As shown in Figure 3, when one tidal berth is assigned with too many tide-condition-dependent vessels and other berths are assigned with few, the assigned tide-condition-dependent vessels would wait for the high-water-level period, and the low-water-level period would be left idle, while the other berths would load for the full time, which is obviously inefficient.

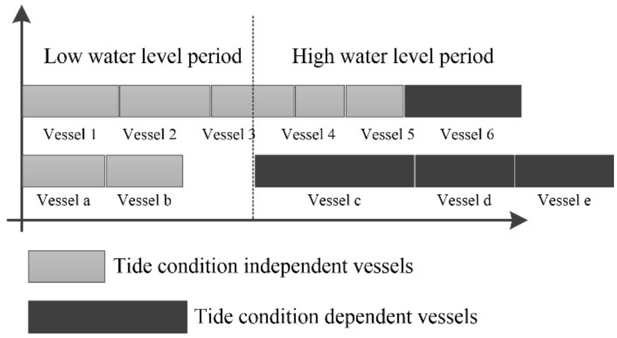



**Figure 3.** The tidal-condition-affected inappropriate berth allocation plan.

Therefore, an efficient berth allocation plan should guarantee that the tidal-condition-dependent vessels are evenly assigned to available berths, so that the maximum loading time of all the berths can be minimized.

#### 3.1.3. The Allocation Efficiency Requirement

To guarantee the robustness of the berth allocation plan, the uncertain dynamic factors that would affect the loading plan should be considered. In the practical loading process, the loading-time consumption of some vessels may be prolonged because of unexpected events, and some vessels may arrive at their designated berth place later than preplanned [29]. Therefore, the static berth allocation plan should be adjusted to deal with unexpected events, such as to reallocate some vessels to nearby available berths. As we discussed above, the large/heavy vessels can only be loaded by large-/heavy-tonnage berths in the specific time period. If the suitable berths for large vessels are occupied by small vessels, they would have to wait until these berths are idle. It would prolong the overall loading time. Therefore, appropriate avenues should be used to guarantee the efficiency of the berth allocation plan.

### 3.2. Problem Analysis

The required loading time of each vessel is defined as $t_i$. We assert that the berth stay interval of each vessel should be in one single tide circle. If one large vessel could not be fully loaded in one circle, then it would leave the harbor before the water depth becomes too low or stay for the next high-water period. Therefore, it may lead to additional work loads, which would be obviously inefficient.

If the tide-dependent vessels could not be loaded in the first tide cycle, they would wait until the next high-water period. Therefore, it may happen that the berth $j$ is left idle in the low-water period. $T^j_{idle}$ is the idle time that berth $j$ waits for the next high-water period as there are no small or medium vessels to be loaded. We assume that the vessels assigned to the berth $j$ would be loaded in $n^d_j$ days. $n^d_j$ may be determined by the assigned tide-dependent vessels.

$$n^d_j = \max(f_c(\sum_{i=1}^{n} x_{ij}t_i/24), f_c(\sum_{k=1}^{n_h} x'_{kj}t_k/T_h)) \ \forall j \in m \tag{2}$$

where $f_c$ is rounded up to the value to the next integer, $T_h$ is the time interval that the tidal-condition-dependent vessels can stay in the harbors in a day, and $n_h$ is the number of high-water-requiring vessels.

The required overall loading time of the berth $j$ is formulated as (3):

$$f_j = \sum_{i=1}^{n} x_{ij} t_i + T_{idle}^{j} \ \ \forall j \in m \tag{3}$$

where $x_{ij}$ is a binary variable. $x_{ij}$ = 1 if vessel $i$ is assigned to berth $j$. Otherwise, $x_{ij}$= 0. The value of $T_{idle}^{j}$ can be determined as below:

$$T_{idle}^{j} = \max(\sum_{i=1}^{n} x_{ij} t_i, f_c(\sum_{k=1}^{n_h} x'_{kj} t_k / \mathrm{T}_h) * 24 + f_{\mathrm{mod}}(\sum_{k=1}^{n_h} x'_{kj} t_k / \mathrm{T}_h)) + t_{tide}^{1} - \sum_{i=1}^{n} x_{ij} t_i \tag{4}$$

where $x'_{kj}$ is a binary variable. $x'_{kj}$ = 1 if vessel $k$ is assigned to berth $j$. Otherwise, $x'_{kj}$= 0.

According to Equation (4), we find that the value of $T_{idle}^{j}$ is zero if the $j$th berth is allocated with appropriate types of vessels. When there are too many tidal-condition-dependent vessels, the $j$th berth would take some more idle time before it finishes the overall loading tasks.

The objective of the BAP is to find the berth allocation plan that minimizes the maximum berthing time of all the available berths and improves the berth allocation efficiency. The objective function is

$$f = \max_{j \in [1,\ldots,m]} (\sum_{i=1}^{n} x_{ij} t_i + T_{idle}^{j} + \sigma n_j^{lth}) \tag{5}$$

s.t.

$$\sum_{j=1}^{m} x_{ij} = 1, i \in [1, n] \tag{6}$$

$$x_{ij} = 0, i \in [1, n - n_s], j \in [m_h + m_{mid} + m_l + 1, m] \tag{7}$$

$$x_{ij} = 0, i \in [1, n - n_s - n_{mid}], j \in [m_h + m_l + 1, m] \tag{8}$$

$$x_{ij} = 0, i \in [1, n - n_s - n_{mid} - n_l], j \in [m_l + 1, m] \tag{9}$$

$$T_{idle}^{j} = \max(\sum_{i=1}^{n} x_{ij} t_i, f_c(\sum_{k=1}^{n_h} x'_{kj} t_k / \mathrm{T}_h) * 24 + f_{\mathrm{mod}}(\sum_{k=1}^{n_h} x'_{kj} t_k / \mathrm{T}_h)) + t_{tide}^{1} - \sum_{i=1}^{n} x_{ij} t_i \tag{10}$$

where $m$ is the number of available berths; $n$ is the number of vessels to be loaded. $n_h$ is the number of high-water-requiring vessels. $\sigma n_j^{lth}$ is devised to evaluate the efficiency of the berth allocation plan, $\sigma$ is a coefficient, and $n_j^{lth}$ is the number of low-/medium-tonnage vessels that are allocated to large-/heavy-tonnage berths. The Equation (4) requires each vessel to be loaded to one berth. Equation (5) defines the feasible berths of medium-/large-/heavy-tonnage vessels. Equation (6) defines the feasible berths of large-/heavy-tonnage vessels. Equation (7) defines the feasible berths of heavy-tonnage vessels. Equation (8) defines the value of $T_{idle}^{j}$.

In addition to that, the tidal condition should be considered. Therefore, when the vessel $i$ is tide-dependent and $x_{ij}$ = 1, then the following constraint should be satisfied for the diurnal tide harbors:

$$s_i, s_i + t_i \in [(k-1) * 24 + T_j^1, k * 24], k \in [1, n_j^d] \tag{11}$$

The following constraint should be satisfied for the semidiurnal tidal harbors:

$$s_i, s_i + t_i \in [(k-1)*24 + T_j^1, (k-1)*24 + T_j^2] \cup [(k-1)*24 + T_j^3, k*24], k \in [1, n_j^d] \quad (12)$$

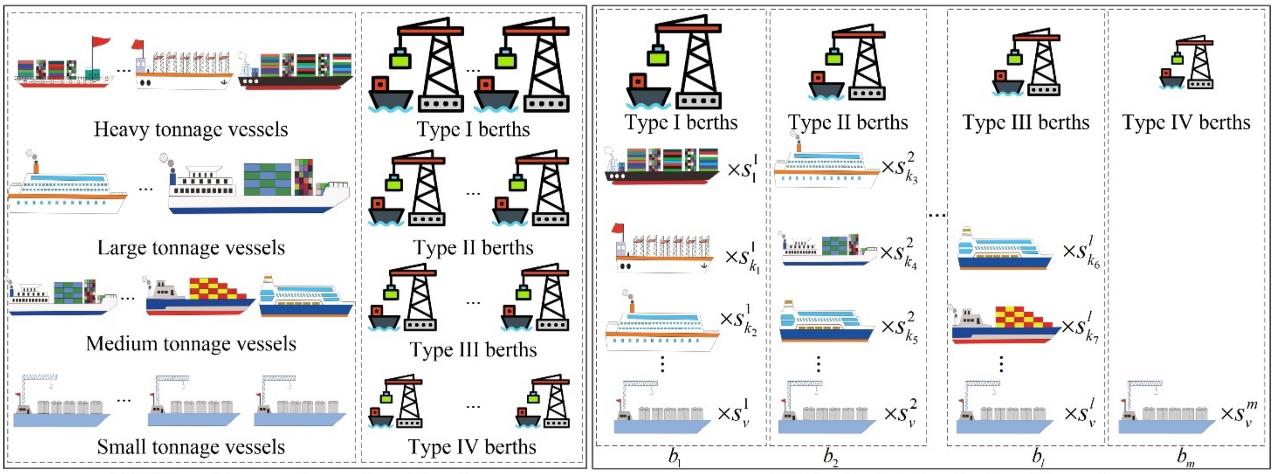
Equations (11) and (12) require the tide-dependent vessels to execute the loading tasks within the high-water period.

## 4. Iterative-Variable-Grouping-Genetic-Algorithm-Based Method

### *4.1. The Characteristics of the Large-Scale Berth Allocation Problem*

As we discussed above, the BAP is an integer-programming problem. The computational complexity of this problem is strongly dependent on the number and types of vessels and berths. The GA is efficient to deal with integer-programming problems [36]. In the large-scale marine transport problem, there may be hundreds of different types of vessels and berths. If we treat this problem as a whole, a complete chromosome may consist of hundreds of genes, and the feasible solution of each gene may range from tens to hundreds. The original GA would then be difficult to obtain the near-optimal berth allocation plans for large-scale BAPs [37].

To deal with this complex problem, we will further discuss the characteristics of the BAP. Our BAP aims to search for the optimal berth allocation plan that minimizes the loading time of berths and guarantees the efficiency of the plan. In the large-scale marine-loading problem, there are many vessels of the same type and berths with the same loading capability, as shown in Figure 4a. The required loading time of each berth is determined by the number and type of vessels that are allocated to it. As shown in Figure 4b, one $k_2$ type vessel could be assigned to berth 1 or berth 2. That is to say, the detailed assignment of one given vessel would not make significant differences to the berth allocation plans. Therefore, we were able to find out that most vessels and berths are not highly relevant for the large-scale marine-loading problem.




(a)Different types of vessels and berths (b)The feasible berth allocation policy

**Figure 4.** The illustration of the berth allocation problem.

To further discuss the characteristic of the BAP, we provide the definition of separability.

Separability means that the influence of a variable on the fitness value is independent of any other variables [38]. According to this definition, we can find out that a function f: $\mathbf{R}^d \to \mathbf{R}$ is separable if:

$$\arg\min_{x_1,\ldots,x_d} f(x_1,\ldots,x_d) = \left(\underset{x_1}{\operatorname{argmin}} f(x_1,\ldots),\ldots,\underset{x_d}{\operatorname{argmin}} f(\ldots,x_d)\right) \quad (13)$$

According to the equation, we can find out that in the separable problem, the optimal solutions of the original problem could be obtained by solving several univariate problems.

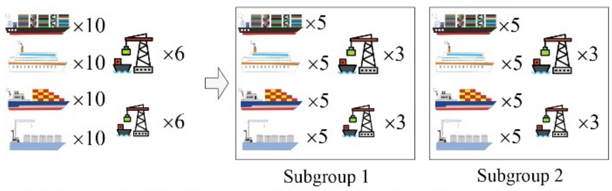
In the large-scale marine-loading problem, there are many different types of vessels and berths. The required loading time of each berth is determined by all the vessels that are allocated to it rather than one specific vessel, which is denoted as (2). Therefore, the BAP is not fully separable.

On the other hand, there are many different types of berths that could load one specific vessel. Therefore, there are no specific highly relevant variables in our BAP problem. When the number of each type of vessels and berths is divided into several subgroups $S_l = \{V_l, B_l\}, l \in [1, n_s]$, and the loading tasks of these subgroups are equal, then the original BAP can be solved as several subproblems in these subgroups [39], which is shown as Figure 5. We can obtain the equation:

$$f = \max\{f_1, \ldots f_{n_s}\} \tag{14}$$

where $f$ is the maximum loading time of the BAP defined in (3), and $f_l, l \in (1, n_s)$ is the maximum loading time of the berths in the lth subgroup, which is generated by solving the Equation (15):

$$f_l = \min(\max_{j \in \{1, m_b^l\}} \{\sum_{k=0}^{n_v^l} x_{kj} t_k + T_{idle}^j\} + \sigma n_j^{lth}) \tag{15}$$




(a) the original berth allocation problem

(b) the berth allocation problems in two subgroups

**Figure 5.** A simple example that solves the original BAP by solving two subproblems.

Therefore, the original BAP can be solved by the "divide and conquer strategy". Figure 4 depicts a simple problem where the vessels and berths can be equally divided into two groups.

However, in most cases, the vessels and berths are not in proportion to each other. Therefore, with regard to the practical problems, the number of each type of vessels and berths among subgroups need not be equal. An appropriate grouping strategy should guarantee that the berth–vessel loading tasks of subgroups are almost the same, such as when there are 12 different vessels and five berths. These vessels and berths can be divided into two groups {(seven vessels, three berths), (five vessels, two berths)} or {(nine vessels, four berths), (three vessels, one berth)} as long as the loading tasks of berths in two groups are almost the same. According to the characteristic of the BAPs, we find that most large-scale BAPs can be treated as several subproblems. Therefore, the large-scale BAPs are partially separable [40].

It is hard to determine the appropriate grouping strategies in most cases, as there are various types of vessels and berths, and these vessels and berths are not in proportion to each other. Therefore, we can only obtain the local optimal berth allocation plan for the BAPs. The local optimal solutions satisfy the Equation (16):

$$f \leq \max\{f_1, \ldots f_{n_s}\} \tag{16}$$

The Equation (16) indicates that the "divide and conquer strategy" cannot guarantee obtaining the ideal results. However, in practical problems, we can obtain the near-optimal berth allocation plans by iteratively adjusting the loading tasks among subgroups according to the optimal berth allocation plans in each subgroup.

### *4.2. Iterative Variable Grouping Genetic Algorithm*

As we discussed above, there is not a proportional relationship between the number of each type of vessels and berths in most marine-loading problems. It is difficult to determine the appropriate grouping strategy directly. We could obtain the optimal berth allocation plan in each subgroup by using the optimization algorithms. The required loading time of berths in different subgroups reflect the differences in the loading tasks among subgroups. To generate the balance of the loading tasks among subgroups, some of the vessels were reallocated among subgroups based on the optimal berth allocation plans in subgroups. We propose the iterative variable grouping genetic algorithm. The main operations of this algorithm are to divide the vessels and berths into subgroups according to the initial grouping strategy, to generate the near-optimal berth allocation plans for each subgroup by using the genetic algorithm, and to reallocate the vessels among subgroups according to the near-optimal berth allocation plans of subgroups.

#### 4.2.1. Initial Grouping Strategy

The primary concern of the grouping strategy is that there are available berths for the vessels in each subgroup.

Furthermore, we should determine the scale of vessels and berths in each subgroup considering the features of the BAP and the characteristics of the genetic algorithm.

According to the nature of the cooperative coevolutionary method, small group sizes are suitable for fully separable problems, making the optimization of each subcomponent easier, and large group sizes increase the probability of grouping together interacting variables in nonseparable problems [32]. As the allocation of one vessel is not highly related with any other specific vessels in the large-scale BAPs, the sizes of subgroups need not be large.

In addition to that, the GA can produce near-optimal solutions for small to medium-size instances efficiently, and it is inefficient in dealing with larger instances [37].

According to the discussion above, we propose the initial grouping strategy as below:

(1) The number of vessels and berths should be kept in medium scale in each group;
(2) To improve the computational efficiency, the number of each type of vessels and berths should be divided as evenly as possible;
(3) Each type of vessel should be distributed to the subgroups according to the number of correspondence berths.

#### 4.2.2. Searching the Near-Optimal Plans by Using the Genetic Algorithm

We searched the optimal berth allocation plan in each subgroup by using the genetic algorithm. We regarded the berths in each group as service counters, and the vessels were treated as guests. The objective was to serve all the guests within the minimum time. The evolution process of the GA was realized by three evolution operators: selection operator (SO), crossover operator (CO), and mutation operator (MO) [41]. The main parts of the genetic algorithm are described as below:

- Chromosome

The chromosome expresses the allocations of all the vessels in one subgroup. It should be guaranteed that the chosen berths are able to load the assigned vessels.

- Population

The population in the evolution process is created by a random process that ensures each chromosome is feasible.

- Crossover

The crossover process is applied between two individuals (parents) that are chosen randomly. This process creates two new individuals (children).

- Mutation

Every chromosome in the population is likely to mutate. We set the value of the randomly chosen gene to mutate in its feasible range. The mutation probability would decline as the population came close to the optimal solution.

Some of the best chromosomes (an elite set of the selected chromosomes) may be moved without any change to the next generation (elitism process).

- Fitness function

As the primary objective of the BAP is to load all vessels within the minimum time, the fitness function needs to evaluate the overall loading time of berths in each subgroup. In addition to that, to guarantee the efficiency and robustness of the BAP strategy, the high-tonnage berths should be assigned with as few small/medium vessels as possible. Therefore, these berths could be used to deal with unexpected events. For example, when some vessels are temporarily added to the loading list, the idle high tonnage berths could serve all types of vessels without modifying the original plan. When the loading tasks of some vessels are prolonged, the unoccupied high-tonnage berths could be used to load more types of subsequent vessels than low-tonnage berths.

According to the feature of the problem, we define the fitness function of the lth subgroup as (17).

$$f_l = 1/(\max_{j\in\{1,m_b^l\}}\{\sum_{k=0}^{n_v^l} x_{kj}t_k + T_{idle}^j\} + \sigma n_j^{lth}) \tag{17}$$

where $\max_{j\in\{1,m_b^l\}}\{\sum_{k=0}^{n_v^l} x_{kj}t_k + T_{idle}^j\}$ depicts the maximum loading time of berths in one subgroup, and $\sigma n_j^{lth}$ denotes the inefficient berth–vessel allocations. As we stated above, $n_j^{lth}$ is the number of low-/medium-tonnage vessels that are allocated to large-/heavy-tonnage berths, and $\sigma$ is a small coefficient that guarantees $\sigma n_j^{lth}$ is significantly smaller than the former part of (17). The better chromosomes would have large values of $f_l$.

- Selection method

There are several types of selection techniques, such as roulette wheel, rank, tournament, Boltzmann, and stochastic universal sampling [42]. As there would be a large number of allocation plans with almost the same fitness values, the fitness function would be almost flat in most areas. To improve the computational efficiency of the GA, the chromosomes that are have more potential should be guaranteed to have higher selection probabilities. As the roulette wheel technique is more agile, we define the selection method based on the roulette wheel technique. The selection probability of an individual is set as Equation (18):

$$ps(\mathrm{a}_i) = \hat{f}_i/\sum_{j=1}^{n_p^l}\hat{f}_j, \forall i \in n_p^l \tag{18}$$

where $\hat{f}_j$ is defined as: $\hat{f}_j = \left[\hat{f}_j/\max_{i\in 1,n_p^l}(\hat{f}_i)\right]^{n_e}$, and $n_e$ is a positive value that is larger than 1; $n_p^l$ is the size of the population in the lth subgroup. According to this equation, the individuals that are have more potential would be selected with higher probability.

#### 4.2.3. Iterative Vessel Reallocation Policy

The differences in the maximum loading time between subgroups indicate the imbalances in the loading tasks of related subgroups. The loading tasks and service capability of subgroups should be modified to balance, such that the differences in loading time between subgroups reach a small value. We propose adjusting the grouping strategy based on the berth allocation plans generated by the GA. According to the berth allocation plans, we could determine the required loading time of each berth. The maximum loading time of berths in each subgroup could be used as the measure of the loading tasks of the subgroup.

The differences in the maximum loading time among subgroups could be reduced by reallocating some vessels between the subgroups with larger maximum loading time and minor maximum loading time. As a result, the loading tasks of subgroups would be close to being balanced. We propose that the appropriate grouping strategy should satisfy the Equation (19):

$$\max_{l\in[1,n_s]}\{T_l^{\max}\} - \min_{l\in[1,n_s]}\{T_l^{\max}\} \leq t'_{\min} \tag{19}$$

where $T_l^{\max} = \max_{k\in[1,l]}\{T_k\}$ is the maximum loading time of berths in the $l$th group by using GA, $n_s$ is the number of subgroups, and $t'_{\min}$ is the minimum loading time difference. The value of $t'_{\min}$ depends on the required loading time of different types of vessels. Therefore, the vessel reallocation policy could be used iteratively to achieve balance in the loading tasks among subgroups.

As there are many different types of berths and vessels in each subgroup, to improve the computational efficiency, it is crucial to determine the appropriate type of vessels to be reallocated between subgroups based on the detailed loading-time differences. As vessels could be classified by the loading tonnage type, we propose the vectorized vessel reallocation policy. We assumed that the GA could generate the optimal berth allocation plans for each subgroup. According to the fitness function we defined above, small-tonnage vessels are preferred to be allocated over small-tonnage berths; medium-tonnage vessels are preferred to be allocated over medium-tonnage berths. As there are four types of berths, we could obtain the vector $\mathbf{T}_l^v =< T_{l,1}^{\max}, \ldots, T_{l,4}^{\max} >$, where $T_{l,j}^{\max}$ is the maximum overall loading time of all the $j$th type of berths. We could obtain more information about the differences between related subgroups by virtue of $\mathbf{T}_l^v$. Accordingly, we propose the vessel reallocation policy:

Firstly, choose the type of vessels to reallocate between subgroups by virtue of $\mathbf{T}_l^v$. When the loading tasks of $j$th type vessels are unbalanced between the two subgroups $l$ and $h$, then there would be an obvious difference between $T_{l.j}^{\max}$ and $T_{h.j}^{\max}$. Suppose that $T_{l.j}^{\max} > T_{h.j}^{\max}$, then some $j$th type vessels in the subgroup l would be chosen to reallocate to the subgroup $h$.

Secondly, determine the vessel exchange policy. The variation in the overall loading time of subgroup l and subgroup $h$ would be $-t_j$ and $t_j$, respectively, if we reallocate one $j$th type vessel between them. As the value of $t_j$ may be large, the large variation would make the iterative reallocation method difficult to converge to the balance of loading tasks among subgroups. Therefore, it would be inefficient if we only reallocated the chosen vessel from one subgroup to another subgroup in each step. To improve the computational efficiency, we applied the vessel exchange policy. According to this policy, when one high-tonnage vessel is chosen to reallocate from the subgroup l to the subgroup $h$, one small-tonnage vessel in the subgroup h would be reallocated to the subgroup $l$. Therefore, the variation in the overall loading time of the subgroup $h$ would be:

$$\Delta T_h = t_s - t_j \tag{20}$$

where $t_j$ is the loading time of the chosen vessel and $t_s$ is the loading time of one small-tonnage vessel. We obtain $t_s < t_j$.

By using the vessel reallocation policy, the loading tasks of related subgroups are guaranteed to be changed slightly; this would lead to the rapid convergence of the balance of the loading tasks among subgroups. The iterative reallocation process would stop as the value of $\max_{l\in[1,n_s]}\{T_l^{\max}\} - \min_{l\in[1,n_s]}\{T_l^{\max}\}$ reaches zero or does not change in a certain time step. The flow chart of the iterative variable grouping genetic algorithm is shown as Figure 6.

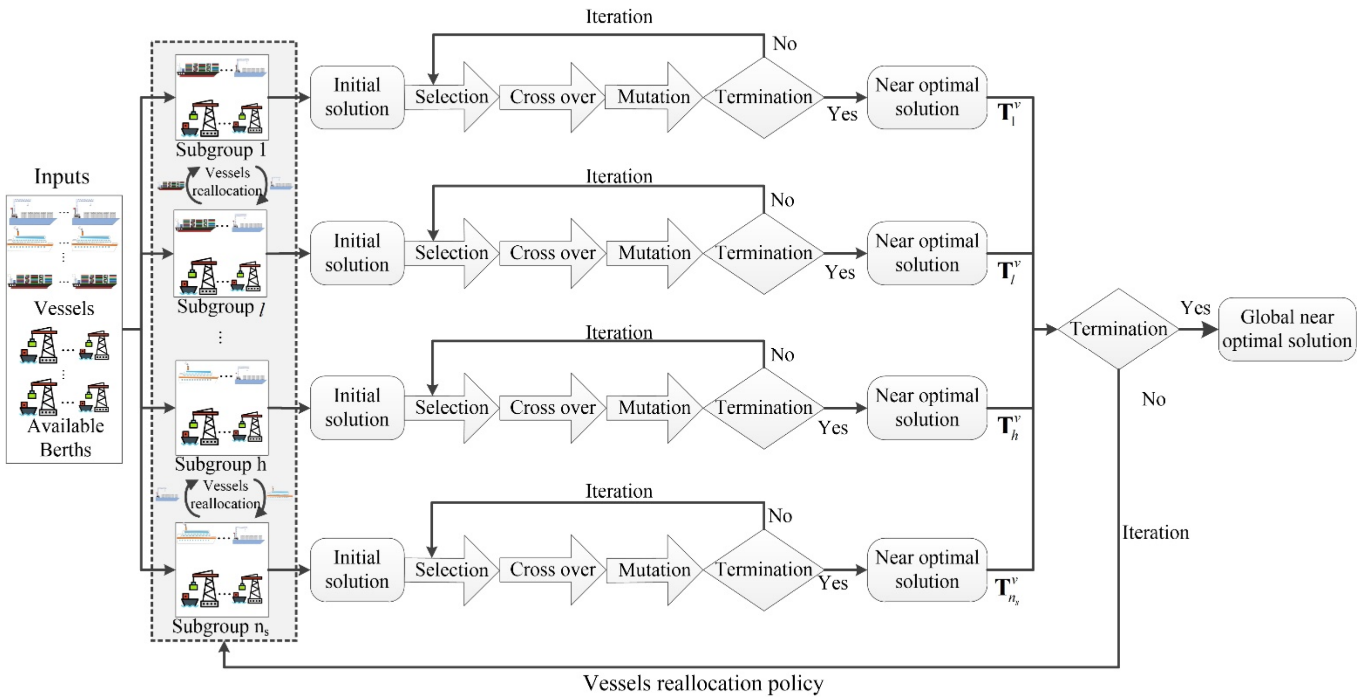



**Figure 6.** A flowchart for the detailed steps for the proposed iterative variable grouping genetic algorithm.

## 5. Computational Experiments

In this section, we demonstrate the proposed algorithm in large-scale marine-loading scenarios. Our approach was implemented in MATLAB 2018 on a 2.5 GHz Intel i5 quad-core processor with 12 GB memory running on the Windows 10 operating system.

### *5.1. Large-Scale Berth Allocation Scenario*

Many researchers demonstrate their algorithms in small-/middle-term vessel allocation scenarios [43]. In this paper, we demonstrate the proposed algorithm through dealing with the large-scale berth allocation problem.

#### 5.1.1. Scenario Introduction

In the large-scale time-critical berth allocation scenarios, the number of vessels to be allocated and the number of available berths may be in the hundreds. We anticipate there are 138 available berths. The information of available berths is shown in Table 1.

**Table 1.** Information of available berths.

| Types of Berths | Number | ID Range |
|---|---|---|
| 1 | 29 | $B_1$~$B_{29}$ |
| 2 | 40 | $B_{30}$~$B_{69}$ |
| 3 | 41 | $B_{70}$~$B_{110}$ |
| 4 | 28 | $B_{111}$~$B_{138}$ |

There are different types of vessels to be loaded. Each type of vessel has different features in loading time and tonnage type, which are shown as Table 2.

The tidal condition is considered in this complex scenario. We set the $v_1$, $v_{15}$, $v_{16}$, and $v_{17}$ type vessels as tidal-condition-dependent vessels. In this scenario, we set the harbor as a diurnal tide harbor. We set the low-water time period to be [0:00–5:00] and high-water time period to be [5:00–24:00].

We set the value of $\sigma$ to be 0.01. It guarantees that the value of $\sigma n_l^{lth}$ does not affect the evaluation of the loading time of each berth.

**Table 2.** The information of vessels to be loaded.

| Vessel Type | $v_1$ | $v_2$ | $v_3$ | $v_4$ | $v_5$ | $v_6$ | $v_7$ | $v_8$ | $v_9$ | $v_{10}$ | $v_{11}$ | $v_{12}$ | $v_{13}$ | $v_{14}$ | $v_{15}$ | $v_{16}$ | $v_{17}$ | $v_{18}$ | $v_{19}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Number | 76 | 23 | 17 | 63 | 13 | 10 | 1 | 1 | 31 | 12 | 10 | 200 | 36 | 8 | 8 | 13 | 2 | 17 | 213 |
| Loading time | 6 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 1 | 1 | 10 | 15 | 18 | 5 | 4 |
| Tonnage type | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 2 | 1 | 1 | 3 | 3 |

### 5.1.2. Parameters' Validation

To guarantee the computational efficiency, the vessels and berths should be allocated to each subgroup as evenly as possible. Therefore, the number of berths in one subgroup is correlated to the number of vessels belonging to this subgroup. Two main parameters should be determined, namely the number of vessels in each subgroup and the value of $n_e$.

To determine the optimal parameters, we validated the performance of the genetic algorithm in different combinations of the number of vessels, and we set the value range of the number of vessels in each subgroup to be (30, 40, 50, 60, 70, 80, 90, 100) and the value range of $n_e$ to be (1, 2, 3, 4, 5, 6, 7, 8 ). The minimum values that were generated in four samples are shown in Figure 7.

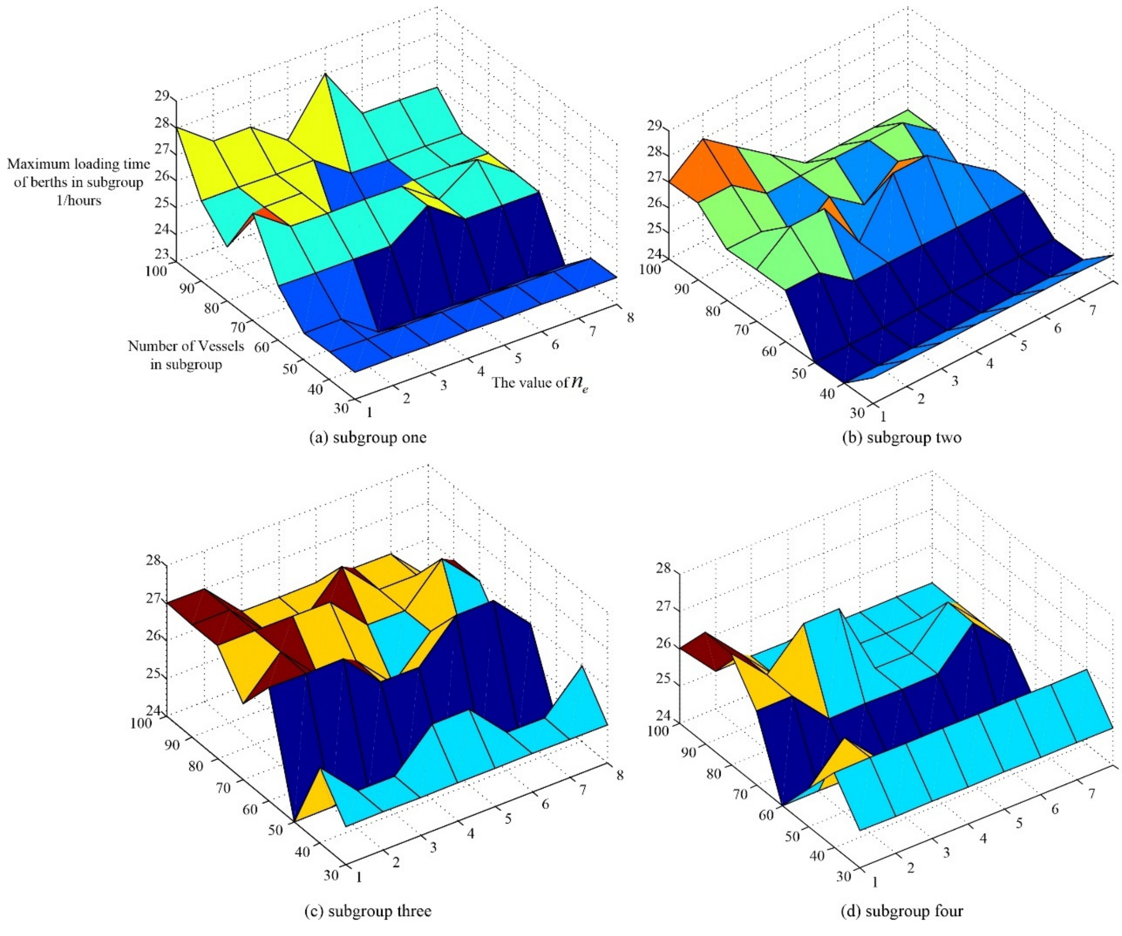



**Figure 7.** The maximum loading time of subgroups when applying different values on two parameters. (**a**) subgroup one; (**b**) subgroup two; (**c**) subgroup three; (**d**) subgroup four.

As the number of vessels and berths is correlated in each subgroup, when the number of vessels is set to a small value, the number of berths that belong to the same subgroup would also be limited. This would lead to obvious differences in the {vessels, berths} data in different subgroups. The IVGGA would be inefficient to generate the ideal berth allocation plan in that situation. On the other hand, when the number of vessels in each subgroup is too large, the GA would be inefficient in generating the optimal berth allocation plans. According to Figure 7, we find that when the number of vessels is set to 50, and the value of $n_e$ is set in the range (3, 4, 5, 6, 7, 8), the IVGGA has better performance. We consider that if the value of $n_e$ is set too high, the algorithm may be prematurely converging the solution to a local minimum. Therefore, we define the vessels in each subgroup to be about 50, and the value of $n_e$ to be 4. The berths and vessels are, therefore, grouped into 15 subgroups. According to the initial grouping strategy, the number of vessels in each subgroup would be in the range (49, 52), and the number of berths would be in the range (9, 10). The numbers of each type of vessel and berth are evenly distributed into each subgroup as possible.

5.1.3. Performance Evaluation

The IVGGA was applied to search the approximate optimal berth allocation solution. Firstly, the initial local optimal solutions were obtained based on the initial grouping information, as shown in Figure 8. It shows that in some subgroups, the maximum loading time was 26 h, and the berths in some other subgroups could fulfill the loading tasks within 23 h. To reduce the maximum loading time of all the available berths, the vessels should be reallocated among the subgroups.

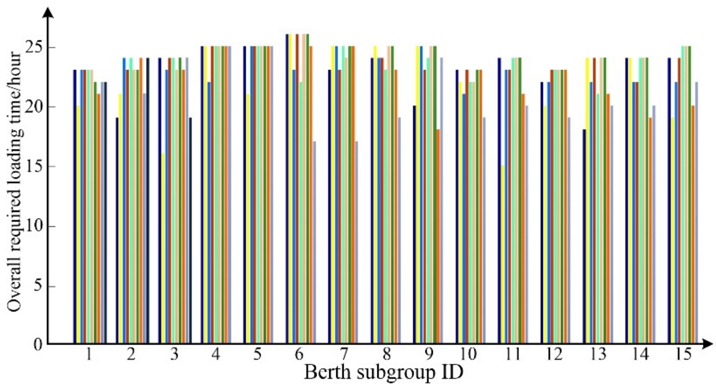



**Figure 8.** The loading time of each berth according to the initial grouping strategy.

After the iterative reallocation of vessels between subgroups, the differences of maximum loading time between each subgroup reached 0. The loading time of each berth is shown in Figure 9. It shows that the maximum loading time was 24 h.

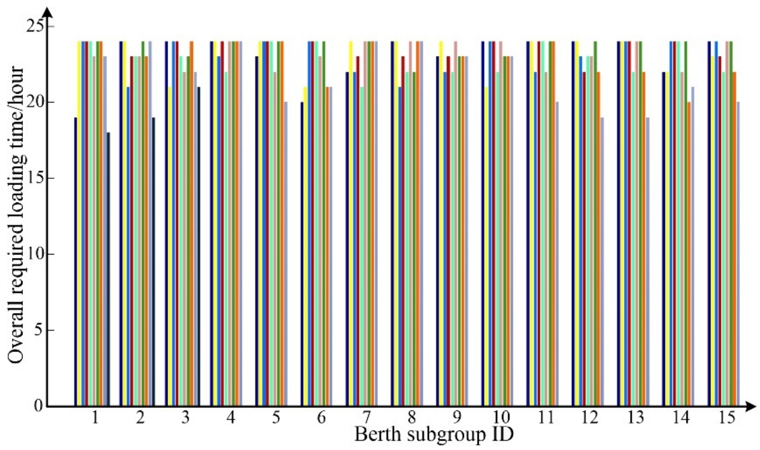



**Figure 9.** The loading time of each berth according to the berth allocation plan generated by the Iterative Variable Grouping GA.

The changes in vessels of each subgroup, from using the proposed vessel reallocation policy, are shown in Table 3.

**Table 3.** The vessel reallocation results.

| Subgroup ID | Number of Vessels after Adjustment | Added Vessels | Removed Vessels |
|---|---|---|---|
| 1 | 52 | $v_{13}$:1,$v_{15}$:1 | $v_{12}$:2 |
| 2 | 52 | $v_2$:1,$v_{14}$:1,$v_{19}$:1 | $v_1$:1,$v_{13}$:2, |
| 3 | 52 | $v_1$:1,$v_2$:1,$v_{15}$:1 | $v_{13}$:2,$v_{16}$:1 |
| 4 | 52 | $v_1$:1 | $v_{16}$:1 |
| 5 | 46 | $v_3$:1,$v_{10}$:1,$v_{13}$:1,$v_{16}$:1,$v_{18}$:1 | $v_4$:1,$v_5$:1,$v_{12}$:7,$v_{15}$:1 |
| 6 | 46 | $v_1$:1,$v_{12}$:1 | $v_4$:1,$v_9$:1,$v_{13}$:2,$v_{18}$:1,$v_{19}$:2 |
| 7 | 53 | $v_{12}$:2,$v_{13}$:1,$v_{19}$:1 | $v_{16}$:1 |
| 8 | 52 | $v_4$:1,$v_5$:1,$v_9$:1,$v_{12}$:2,$v_{19}$:1 | $v_1$:3,$v_{15}$:1 |
| 9 | 52 | $v_1$:1,$v_9$:1,$v_{12}$:2,$v_{13}$:1 | $v_3$:1,$v_{15}$:1,$v_{19}$:1 |
| 10 | 51 | $v_4$:1,$v_{12}$:1,$v_{13}$:1,$v_{15}$:1 | $v_2$:1,$v_9$:1 |
| 11 | 49 | $v_{12}$:1,$v_{13}$:1,$v_{16}$:1 | $v_2$:1,$v_{10}$:1,$v_{14}$:1 |
| 12 | 50 | $v_1$:1 | |
| 13 | 49 | $v_{16}$:1 | $v_1$:1 |
| 14 | 49 | | |
| 15 | 49 | | |

In addition, the IVGGA shows the capability of guaranteeing the robustness of the berth allocation plan. Figure 10 shows that the medium-tonnage vessels are mainly served by the Type 2 and Type 3 berths, and the small-tonnage vessels are mainly served by the Type 4 berths. As shown in Figure 9, the blue bars depict the overall loading times of type 1 and type 2 berths. The yellow bars and red bars depict the overall loading times of type 3 and type 4 berths, respectively. It shows that there are idle times for most high-tonnage berths. Therefore, the type 1 and type 2 berths could be used to deal with unexpected events as these berths could serve most vessels. In instances when some vessels arrive later than the predefined start time, or the loading tasks of some vessels are not finished in the scheduled time, the idle berths could be used to load the follow-up vessels (the detailed berth allocation plan is recorded in Supplementary Materials).

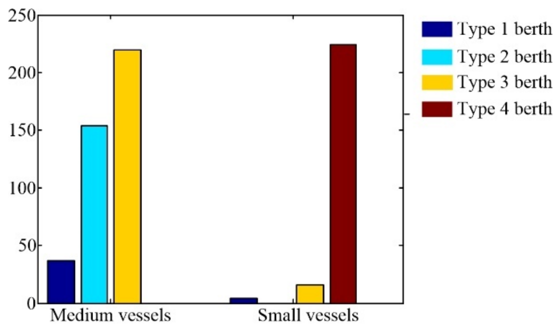



**Figure 10.** The allocations of small and medium vessels to different types of berths.

### *5.2. Comparison with Original Genetic Algorithm*

In this computational experiment, we compared the proposed algorithm with the original GA. We considered that there were 19 types of vessels to be loaded, and the loading time and tonnage type of these vessels are listed in Table 2. These vessels are classified as four types according to their tonnage data. The available berths could also be classified into four types according to their loading capabilities, as recorded in Table 1. We devised various scenarios for different numbers of vessels and available berths, as shown in Table 4. We created many instances for each scenario, and the number of available berths varied in a certain range. We assert that the number of vessels in each subgroup should be no more than 50. In addition to that, to guarantee the consistency of the original GA and the IVGGA, we assert that the length of the chromosome is linearly correlated with the number of vessels in each subgroup. Therefore, the number of genes increase with the number of vessels. Figure 11 shows the maximum loading time of available berths according to the optimization berth allocation results generated by the original GA and the IVGGA algorithm.

**Table 4.** The information of vessels and berths in each scenario.

| Scenario Type | Number of Vessels | The Range of Number of Available Berths |
|---|---|---|
| 1 | 77 | (15, 16) |
| 2 | 151 | (29, 31) |
| 3 | 228 | (43, 45) |
| 4 | 302 | (56, 58) |
| 5 | 379 | (70, 72) |
| 6 | 453 | (84, 86) |
| 7 | 530 | (97, 99) |
| 8 | 604 | (111, 113) |
| 9 | 681 | (125, 127) |
| 10 | 755 | (140, 141) |

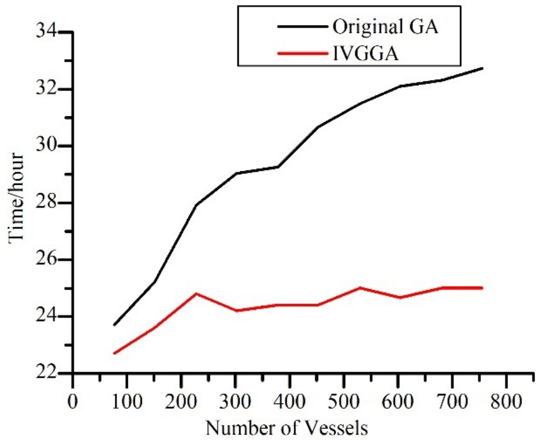



**Figure 11.** The results obtained by using the original genetic algorithm and the iterative variable grouping genetic algorithm.

According to the results shown in Figure 11, we find that the IVGGA outperforms the original GA. When the number of vessels increases to 100, the original GA becomes inefficient in generating near-optimal solutions. On the other hand, the IVGGA can generate good results even when the number of vessels reaches 700+. Therefore, we can reach the conclusion that the proposed algorithm can deal with large-scale BAPs.

## 6. Discussion and Conclusions

Large-scale maritime transport is the main transportation method in humanitarian aid and disaster relief (HADR) actions. As transporting the supplies and relief teams timely is very important for lifesaving and alleviating the disaster impact in the affected areas, to make full use of the available vessels and berths is critical in HADR actions. This paper demonstrated an iterative variable grouping genetic algorithm (IVGGA) for dealing with the static large-scale berth allocation problem (BAP). The optimal berth allocation plan should minimize the time to load all the vessels, and guarantee that the plan is robust to unexpected events. The effect of the tidal condition was considered.

When there are a great number of vessels to be loaded, the original genetic algorithm (GA) would be inefficient in generating near-optimal berth allocation plans. According to the characteristics of the static large-scale BAPs, we found that it was partially separable. Therefore, we propose using the "divide and conquer strategy". We divided the vessels and berths into subgroups, and the GA was applied to search the near-optimal berth allocation plans in subgroups. With consideration of the unbalance of loading tasks among subgroups, we propose iteratively reallocating the vessels between the subgroups with a larger fitness value and a minor fitness value. To improve the computational efficiency, we propose the vessel reallocation policy. Our approach proved to be of high performance in generating a near-optimal berth allocation plan in the complex scenario involving more than 700 vessels to be loaded. According to the results of the computational experiments, we found that our method is highly scalable.

As we mainly focused on the marine-loading problem in large-scale time-critical missions, what we considered the most is to maximize the utilization of available berths, and to fulfill the overall loading tasks in the minimum time. Therefore, we focused on the static large-scale BAP in this study. However, as this problem is partially separable, the near-optimal solution generated by using the proposed method would not fix the assignment of each specific vessel to each berth. On the contrary, we could determine the detailed allocation plan for vessels considering the position of vessels and berths. Furthermore, when the detailed allocation plan was determined, the order of loading could be determined according to the arrival times of related vessels. In our future research, we will study the online vessel–berth allocation problem based on the static berth allocation plans. In addition to that, as we have considered the robustness in the proposed algorithm, small-tonnage vessels and medium-tonnage vessels are preferred to be allocated to the shallower and smaller berths. The berths that could serve the high-/heavy-tonnage vessels would be reserved for unexpected events rather than loading the small-/medium-tonnage vessels. Therefore, these berths could be used to load the additional vessels or the vessels that arrive late.

**Supplementary Materials:** The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/jmse10091294/s1. Table S1: The detail berth allocation plan.

**Author Contributions:** Conceptualization, D.Y. and J.Y.; methodology, D.Y.; software, J.Y.; validation, D.Y., Y.N., J.Y. and S.Y.; formal analysis, Y.N.; investigation, S.Y.; resources, S.Y.; data curation, Y.N.; writing—original draft preparation, Y.N.; writing—review and editing, D.Y. and J.Y.; visualization, S.Y.; supervision, J.Y.; project administration, J.Y.; funding acquisition, J.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China under Grants 61906211 and 61876187.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data available in a publicly accessible repository that does not issue DOIs. Publicly available datasets were analyzed in this study. These data can be found here: https://download.csdn.net/download/thanksgivingguy/86399373 (accessed on 11 August 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. UNCTAD. Review of Maritime Transport. 2021. Available online: https://unctad.org/system/files/official-document/rmt2021ch1_en.pdf (accessed on 30 October 2021).
2. Nica-Rus, L.; Kaiter, E.-H. Outdated Ships and Their Future During (Non) Pandemic Times. *IOP Conf. Ser. Earth Environ. Sci.* **2021**, *635*, 012006. [CrossRef]
3. Munyaka, J.-C.B.; Yadavalli, V.S.S. Decision support framework for facility location and demand planning for humanitarian logistics. *Int. J. Syst. Assur. Eng. Manag.* **2021**, *12*, 9–28. [CrossRef]
4. Sarma, D.; Singh, A.; Das, A.; Bera, U.K. A Post-Disaster Humanitarian Relief Logistic Model: Evacuation and Transportation. In Proceedings of the 2018 3rd International Conference for Convergence in Technology (I2CT), Pune, India, 6–8 April 2018.
5. Ozkapici, D.B.; Ertem, M.A.; Aygüneş, H. Intermodal humanitarian logistics model based on maritime transportation in Istanbul. *Nat. Hazards* **2016**, *83*, 345–364. [CrossRef]
6. Von Hillebrandt-Andrade, C.; Vanacore, E. Citizen science for studying earthquakes. *Science* **2022**, *376*, 246–247. [CrossRef] [PubMed]
7. Huang, K.; Jiang, Y.; Yuan, Y.; Zhao, L. Modeling multiple humanitarian objectives in emergency response to large-scale disasters. *Transp. Res. Part E* **2015**, *75*, 1–17. [CrossRef]

8. Zhen, L.; Liang, Z.; Zhuge, D.; Lee, L.H.; Chew, E.P. Daily berth planning in a tidal port with channel flow control. *Transp. Res. Part B* **2017**, *106*, 193–217. [CrossRef]
9. Buhrkal, K.; Zuglian, S.; Røpke, S.; Larsen, J.; Lusby, R.M. Models for the discrete berth allocation problem: A computational comparison. *Transp. Res. Part E Logist. Transp. Rev.* **2011**, *47*, 461–473. [CrossRef]
10. Sheu, J.-B. An emergency logistics distribution approach for quick response to urgent relief demand in disasters. *Transp. Res. Part E* **2007**, *43*, 687–709. [CrossRef]
11. Iris, Ç.; Lalla-Ruiz, E.; Lam, J.S.L.; Voß, S. Mathematical programming formulations for the strategic berth template problem. *Comput. Ind. Eng.* **2018**, *124*, 167–179. [CrossRef]
12. Cordeau, J.-F.; Laporte, G.; Legato, P.; Moccia, L. Models and tabu search heuristics for the berth-allocation problem. *Transp. Sci.* **2005**, *39*, 526–538. [CrossRef]
13. Cahyono, R.T.; Flonk, E.J.; Jayawardhana, B. Discrete-Event Systems Modeling and the Model Predictive Allocation Algorithm for Integrated Berth and Quay Crane Allocation. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1321–1332. [CrossRef]
14. Li, M.-W.; Hong, W.-C.; Geng, J.; Wang, J. Berth and quay crane coordinated scheduling using multiobjective chaos cloud particle swarm optimization algorithm. *Neural. Comput. Appl.* **2017**, *28*, 3163–3182. [CrossRef]
15. Cheimanoff, N.; Fontane, F.; Kitri, M.N.; Tchernev, N. A reduced VNS based approach for the dynamic continuous berth allocation problem in bulk terminals with tidal constraints. *Expert Syst. Appl.* **2021**, *168*, 114215. [CrossRef]
16. Bacalhau, E.T.; Casacio, L.; de Azevedo, A.T. New hybrid genetic algorithms to solve dynamic berth allocation problem. *Expert Syst. Appl.* **2021**, *167*, 114198. [CrossRef]
17. Zhu, Y. Container Ship Three-Dimensional Loading Problem Based on Hybrid Genetic Algorithm. Master's thesis, Huazhong University of Science & Technology, Wuhan, China, 2016.
18. Potter, M.A.; De Jong, K.A. A cooperative coevolutionary approach to function optimization. In *Proceedings of the Parallel Problem Solving from Nature—PPSN III International Conference on Evolutionary Computation, The Third Conference on Parallel Problem Solving from Nature, Jerusalem, Israel, 9–14 October 1994*; Davidor, Y., Schwefel, H.P., Männer, R., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 1994; Volume 866, pp. 249–257.
19. Li, M.; Yu, B.; Qi, M. PGGA: A predictable and grouped genetic algorithm for job scheduling Future Generation Computer. *Systems* **2006**, *22*, 588–599.
20. Imai, A.; Nagaiwa, K.; Chan, W.T. Efficient planning of berth allocation for container terminals in Asia. *J. Adv. Transp.* **1997**, *31*, 75–94. [CrossRef]
21. Ren, J.; Tian, Y.; Sawaragi, T. A tree search method for the container loading problem with shipment priority. *Eur. J. Oper. Res.* **2011**, *214*, 526–535. [CrossRef]
22. Imai, A.; Nishimura, E.; Papadimitriou, S. Berth allocation with service priority. *Transp. Res. Part B Methodol.* **2003**, *37*, 437–457. [CrossRef]
23. Xu, Q.; Ji, D.; Ge, Y. Research on Project Optimization of Ship Loading Berth Allocation Based on Genetic Algorithm. *Fire Control. Command. Control* **2017**, *4*, 171–176.
24. Iris, Ç.; Pacino, D.; Ropke, S.; Larsen, A. Integrated Berth Allocation and Quay Crane Assignment Problem: Set partitioning models and computational results. *Transp. Res. Part E* **2015**, *81*, 75–97. [CrossRef]
25. Xu, D.; Li, C.-L.; Leung, J.Y.-T. Berth allocation with time-dependent physical limitations on vessels. *Eur. J. Oper. Res.* **2012**, *216*, 47–56. [CrossRef]
26. Lalla-Ruiz, E.; Expósito-Izquierdo, C.; Melián-Batista, B.; Moreno-Vega, J.M. A Set-Partitioning-based model for the Berth Allocation Problem under Time-Dependent Limitations. *Eur. J. Oper. Res.* **2016**, *250*, 1001–1012. [CrossRef]
27. Schepler, X.; Absi, N.; Feillet, D.; Sanlaville, E. The stochastic discrete berth allocation problem. *EURO J. Transp. Logist.* **2019**, *8*, 363–396. [CrossRef]
28. Iris, Ç.; Pacino, D.; Ropke, S. Improved formulations and an Adaptive Large Neighborhood Search heuristic for the integrated berth allocation and quay crane assignment problem. *Transp. Res. Part E Logist. Transp. Rev.* **2017**, *105*, 123–147. [CrossRef]
29. Iris, Ç.; Lam, J.S.L. Recoverable robustness in weekly berth and quay crane planning. *Transp. Res. Part B* **2019**, *122*, 365–389. [CrossRef]
30. Kasm, O.A.; Diabat, A.; Cheng, T.C.E. The integrated berth allocation, quay crane assignment and scheduling problem: Mathematical formulations and a case study. *Ann. Oper. Res.* **2020**, *291*, 435–461. [CrossRef]
31. Wang, Y.; Liu, H.; Wei, F.; Zong, T.; Li, X. Cooperative Co-evolution with Formula-based Variable Grouping for Large-Scale Global Optimization. *Evol. Comput.* **2018**, *26*, 569–596. [CrossRef]
32. Trunfio, G.A. Adaptation in Cooperative Coevolutionary Optimization. In *Adaptation and Hybridization in Computational Intelligence*; Fister, I., Fister, I., Jr., Eds.; Adaptation, Learning, and Optimization; Springer: Cham, Switzerland, 2015; Volume 18, pp. 91–109.
33. Hosseini, E.S.; Esmaeelzadeh, V.; Eslami, M. A Hierarchical Sub-Chromosome Genetic Algorithm (HSC-GA) to Optimize Power Consumption and Data Communications Reliability in Wireless Sensor Networks. *Wirel. Pers. Commun.* **2015**, *80*, 1579–1605. [CrossRef]
34. Chen, Y. Introducing Subchromosome Representations. In: Extending the Scalability of Linkage Learning Genetic Algorithms. *Stud. Fuzziness Soft Comput.* **2006**, *190*, 91–99.
35. Zhang, B.; Chen, S.; Gu, G. Tidal types and characteristics of the harbors along the Guangxi coast. *J. Mar. Sci.* **2010**, *28*, 9–16.

36. Hua, Z.; Huang, F. A variable-grouping based genetic algorithm for large-scale integer programming. *Inf. Sci.* **2006**, *176*, 2869–2885. [CrossRef]
37. Shen, Y.; Peng, K.; Chen, K.; Li, J. Evolutionary crew scheduling with adaptive chromosomes. *Transp. Res. Part B Methodol.* **2013**, *56*, 174–185. [CrossRef]
38. Auger, A.; Hansen, N.; Mauny, N.; Ros, R.; Schoenauer, M. *Bio-Inspired Continuous Optimization: The Coming of Age*; Invited Talk at CEC: Piscataway, NJ, USA, 2007.
39. Bertsekas, D.P. *Nonlinear Programming*, 3rd ed.; Athena Scientific: Belmont, MA, USA, 2017.
40. Kallioras, N.A.; Lagaros, N.D.; Avtzis, D.N. Pity beetle algorithm-A new metaheuristic inspired by the behavior of bark beetles. *Adv. Eng. Softw.* **2018**, *121*, 147–166. [CrossRef]
41. Zhai, Q.; He, Y.; Wang, G.; Hao, X. A general approach to solving hardware and software partitioning problem based on evolutionary algorithms. *Adv. Eng. Softw.* **2021**, *159*, 102998. [CrossRef]
42. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [CrossRef]
43. Hsu, H.-P.; Chiang, T.-L.; Wang, C.-N.; Fu, H.-P.; Chou, C.-C. A Hybrid GA with Variable Quay Crane Assignment for Solving Berth Allocation Problem and Quay Crane Assignment Problem Simultaneously. *Sustainability* **2019**, *11*, 2018. [CrossRef]