Computers & Operations Research 144 (2022) 105796

Contents lists available at ScienceDirect

Computers and Operations Research

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

Multi-objective optimization of the 3D container stowage planning problem in a barge convoy system

Amina El Yaagoubi [a,*], Mohamed Charhbili [a], Jaouad Boukachour [a], Ahmed El Hilali Alaoui [b]

[a] Normandie Univ, UNIHAVRE, Le Havre 76600, France
[b] Euro-Mediterranean of Fez, Morocco

ARTICLE INFO

Keywords:
Shifting movements
Barge convoy system
Containers loading/unloading
Stability
Bin-packing
Multi-objective optimization
NSGA-II

ABSTRACT

Mainly motivated by the massification of flows in inland shipping, where several barges can be assembled in convoys to transport a large number of containers to and from the hinterland, we introduce an original extension of the Container Stowage Planning Problem (CSPP) (which we denote the 3D – barge Convoy Container Stowage Planning Problem (3D –CCSPP)). The requirements for stowing all containers in a barge convoy entail additional constraints related to both the containers and the convoy. In this paper, the 3D –CCSPP is dealt with, on the basis of its relation to the 3D Bin-Packing Problem (3D-BPP), in which the filled objects represent the containers, and the bins indicate the convoy barges. We consider the multi-objective aspect of the problem with three realistic functions to be optimized: *shifting* movements, the convoy stability and the number of real-used barges. This is a new and effective aspect considering the state of the art of CSPP in inland shipping. We develop a multi-objective mathematical formulation of the 3D-CCSPP and propose a novel adaptation of the multi-objective evolutionary algorithm NSGA-II (Non-Dominated Sorting Genetic Algorithm-II) based on a set of heuristics introduced by the BPP resolution methods. Specific encoding method, generation of the initial population and genetic operators are designed based on the mathematical description of the problem. Extensive experiments are then carried out on a varied set of instances. The computational results demonstrate the effectiveness of the proposed approaches, they are evaluated using CPLEX solver and their performance is verified by a set of measurements which are adapted to the problem in real-world applications in relation to each objective function. Additional studies are carried out in sensitivity analysis to provide decision makers with a better knowledge of the problem. They can then choose, from the set of compromise solutions of the Pareto front, the most promising one for practical implementation based on their experience and preferences.

1. Introduction

Barges are flat-bottomed river transport units, with which different types of goods can be transported. However, their most important aspect is that they are not all, generally, independent boats. They must be towed by a tugboat (or pusher) to ensure their movement. The latter is a very powerful type of propulsion vessel that maneuvers other vessels by pushing or pulling them, either by direct contact or by means of a solid towing cable. Its cockpit can be mounted to offer the pilot a good navigation vision and also lowered to pass under bridges (Mihyeon Jeon and Amekudzi, 2005). As a result, several barges can be coupled and fixed end-to-end, forming a system called a convoy or pushed convoy, whose movement is ensured by a pusher on rivers and large-scale canals. It is not uncommon to associate several barges, in width as in length, to form a “separable” convoy (Fig. 1). It should be noted that river convoys consist of a pusher boat and a set of barges whose dimensions may vary. However, the current trend is to design units with standardized dimensions to optimize the heavy transport capabilities offered by the waterway. As a result, in the present paper, we are interested in the case where the convoy consists of a set of homogeneous barges.

In container ports, barge delays can occur at different areas and can be caused by several factors (Douma et al., 2009; Tanaka and Tierney, 2018). However, the main activities affecting the time the barge spends in the dock of the river terminal are the container unloading and loading processes. Although there is a relationship between the two processes, they are essentially carried out in two independent tasks, with loading

* Corresponding author at: Le Havre Normandie University, Le Havre, 76600, France.
E-mail addresses: amina.el-yaagoubi@univ-lehavre.fr (A. El Yaagoubi), mohamed.charhbili@univ-lehavre.fr (M. Charhbili), jaouad.boukachour@univ-lehavre.fr (J. Boukachour), a.elhilali-alaoui@ueuromed.org (A. El Hilali Alaoui).

https://doi.org/10.1016/j.cor.2022.105796
Received 7 June 2021; Received in revised form 17 March 2022; Accepted 18 March 2022
Available online 25 March 2022

being the most difficult and the most sensitive to the efficiency of operations in subsequent ports (Imai et al., 2002). Within this context, a good loading plan in the initial port allows the reduction of the time of unloading containers in their port of destination, and consequently, the reduction of the overall time of the transport. Indeed, the containers are stacked according to the "last in, first out" policy, so that to unload a certain container, it is necessary to unload all the containers stacked above, then reload them again, generating thus unproductive movements called "*shifting*", which entail cost escalation as well as delays at ports. Therefore, the operational efficiency of both the port and the barge is highly dependent on a proper container stowage planning.

In general, the Container Stowage Planning Problem (CSPP) consists in determining the most suitable arrangement of the containers in a ship, respecting a set of structural, operational and physical constraints, related to both the containers and the ship, while ensuring the ship stability and reducing the number of *shiftings*.

The present work is inspired by real-application problems, where the container stowage planning is generated, not only for a single barge, but, for a convoy system consisting of a boat pushing several homogeneous barges moored between them. We assume that each container movement consumes a uniform cost in all ports. Therefore, minimizing costs at ports requires minimizing unproductive container movements in each port (Expósito-izquierdo et al., 2017). We denote this new variant the 3D – barge Convoy Container Stowage Planning Problem (3D–CCSPP). The 3D –CCSPP is treated on the basis of its relation to the three-dimensional Bin-Packing Problem (3D-BPP) (Paquay et al., 2018), in which the filled objects represent the containers, and the bins indicate the barges of the convoy. In our study, we consider the multi-objective aspect of the problem with three realistic functions to be optimized by developing a new multi-objective mathematical formulation. Although it achieves optimal results for almost all small instances, other computational results show that it is not suitable for solving large real instances in a very short time such as required by the industry. The 3D-CCSPP is more complex, but also more realistic, and could be processed using a *meta*-heuristic. As a result, we adapted the multi-objective *meta*-heuristics NSGA-II (Non-Dominated Sorting Genetic Algorithm II), combined with heuristics of the BPP to solve large-scale instances.

To our knowledge, no study in the literature considers the case of container stowage planning in barge convoy systems and simultaneously aims at optimizing the number of *shiftings*, the total stability of the convoy and the number of real-used barges in a multi-objective formulation.

The rest of this paper is organized as follows. The second section gives a detailed overview of the related literature. The third section formally describes the container stowage planning problem in a barge convoy system (3D-CCSPP) and identifies its different characteristics, relying on the multi-objective aspect of the problem and its relation to the 3D-BPP. Then, an appropriate mathematical model is presented in the fourth section describing the different structural and operational constraints. In addition, to solve the 3D-CCSPP on large instances, the NSGA-II multi-objective approach is developed in the fifth section, combined with BPP heuristics. We then present, in the sixth section, a set of numerical results that we evaluate by proposing three performance measures adapted to the problem, as well as additional studies on sensitivity analysis to provide the stowage planner with a better knowledge of the problem. Finally, we conclude with a conclusion in the last section.

2. Related work

A very large number of the CSPP studies have been published in recent years and have mainly focused on maritime transportation between container terminals using container ships. As noted in Pacino et al. (2011), CSPP contributions can broadly be categorized into two types of approaches, single-phase approaches and multi-phase approaches.

On the one hand, in multi-phase approaches, the CSPP is treated as the Master Bay Plan Problem (MBPP), which is hierarchically divided into sub-problems, often into a Master Planning Problem (MPP) and a Slot Planning Problem (SPP). First, a general loading plan is created by distributing the containers in the bays of the ships. Then, to determine the position of each container, a phase of planning the exact locations follows and focuses on each bay at a time (Wilson and Roach, 1999; Wilson and Roach, 2000; Sciomachen & Tanfani, 2007; Delgado et al., 2012). Ambrosino et al. (2004) proposed a three-phase algorithm to solve the MBPP. The first phase involves dividing the ship into different parts and associating grouped containers with different bay subsets without specifying their actual position. Then, they assign the exact position to each container by solving a linear programming model. During the last phase, some local research exchanges are carried out to verify and eliminate the unrealizable solutions caused by the stability constraints. It is also interesting to note the work of Parreño et al. (2016), who studied the SPP considering container movements and separation rules that arise when handling containers of dangerous goods. A very recent work is that of Bilican et al. (2020), they considered the CSPP with stability restrictions and proposed a MILP formulation that aims at avoiding costs resulting from *shiftings* movements and ship instability. They used an IP formulation to get a lower bound on the total *shiftings* cost which is used in the proposed two-stage heuristic algorithm and then uses the SH algorithm to minimize the additional fuel cost due to the ship instability.

In addition, several researchers have addressed the CSPP based on its relationship to the BPP. Sciomachen & Tanfani (2003) presented a heuristic method to solve the MBPP based on its connection to the 3D-BPP. The heuristic procedure presented here is intended to exploit the potential of the 3D-BPP algorithm proposed by Martello et al. (2000). The main objective is to minimize the total loading time. Stability and stacking constraints are considered in the order of container loading and not by ship stability measurements. In their later work, Sciomachen & Tanfani (2007) went further and included dock side equipment, namely, dock cranes. The goals were to minimize the total loading time and to use the dock equipment efficiently. The approach has been validated using real test cases from the port of Genoa (Italy). However, they did not take into account the influence of the stability of the vessel as a function of the weight distribution of the containers. Likewise, Wei-Ying et al. (2005) decomposed the CSPP for a container ship serving many ports into two sub-problems in order to reduce computational complexity. The study done mainly focuses on the first sub-problem

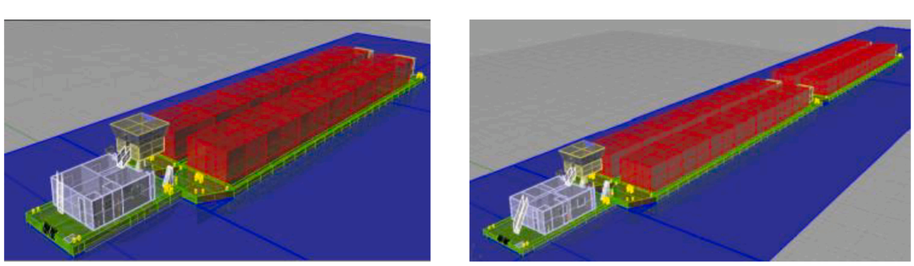

Fig. 1. Barge convoy system (van Hassel, 2011).

which is considered as a BPP where the ship bays are considered as bins, the number of positions in each bay is considered as the capacity of the corresponding bin and containers with their different characteristics are treated as objects to be loaded. The goal is to minimize *shifting* movements and the number of bays occupied by containers in each port.

On the other hand, single-phase approaches represent the cargo space of the ship as a cell-based data structure allowing the entire stowage plan to be described in one step. Avriel et al. (1998) present a model that minimizes the number of *shifting* movements in a stowage planning. However, the model has limited applicability due to its complexity. The authors therefore developed the suspensive heuristic which focuses only on the constraints of stacking containers and does not include any consideration of vessel stability. Imai et al. (2002) studied the container stowage problem by considering a single ship hold and a single stack in the yard of the container port. Using an estimated number of *shifting* motions, they found solutions with acceptable vessel stability represented by a factor called “metacentric distance” (which will be explained below). Subsequently, Imai et al. (2006) presented a multi-criteria optimization method of the CSPP by considering the stability of the ship and the number of container movements. For vessel stability measurements, the metacentric distance, the list and the trim were used, and for *shifting*, an estimated number was used. Another research was carried out by Delgado et al. (2009). In their work, they applied Constraint Programming (CP) to the CSPP. They reported that the CP approach outperformed an integer programming approach and a column generation approach in a preliminary study. However, they only tested their approach in the case of a single bay in the ship’s hold. Tavares de Azevedo et al. (2014) extended the 2D Container Ship Loading Planning Problem, (2D-CLPP), proposed by Avriel and Penn (1993), to a three-dimensional problem. They included two optimization criteria: the number of container movements and the stability of the container ship. However, they assumed that each container has the same size and unit weight. Ding and Chou (2015) examine the CSPP when a container ship successively visits a series of ports without considering stability measures. They developed a heuristic algorithm to generate stowage plans with a reasonable number of *shifting* movements and show that the algorithm works better than the suspension heuristic proposed in Avriel et al. (1998). In addition, a new study addressing a related problem, namely, the zero-*shifting* stack minimization problem was investigated by Wang et al. (2014). This problem aims to find the minimum number of stacks needed to accommodate all the containers in a multiport trip without any *shifting* movements. More recently, Parreño-Torres et al. (2021) introduce three generalizations of the CSPP, which come with several complexities: container sizes, container weights, and stability constraints. They provide integer programming formulations for the general problem as well as some special cases with identical container size and/or identical weights then they develop a matheuristic approach to solve the more general problem using the decomposable structure of the three mathematical models.

The main disadvantage of applying the one-phase approach to solve the CSPP is that this problem is NP-hard and there is no guarantee of getting an optimal solution for large vessels within a reasonable calculation time (Wilson and Roach, 1999). However, in the two-phase approach, since the main stowage plan, which is generated at the first phase, does not take into account the stacking constraints, it is possible that the plan of the locations that result, in the second phase, is not feasible, as stated in (Parreño et al., 2016). Pacino et al. (2011) address this problem with a post-optimization procedure in which containers are removed from solutions until a workable location plan is reached.

All the work reported above has been primarily focused on solving the container stowage problem in maritime container ships. To our knowledge, only few studies focus on inland vessels. Li et al. (2017) and Hu & Cai (2017) studied the CSPP in a river container ship. They applied it to the Yangzi Jiang River in China. Li et al. (2017) approached the problem through a two-phase approach. They respectively designed a Greedy Randomized Adaptive Search Procedure (GRASP) and a Heuristic Evolutionary Strategy Algorithm (HES) to solve the multi-objective optimization problem in two levels. At the first level, they focused on minimizing the difference in longitudinal weight and the number of bays used during the journey. At the second level, the containers are divided into groups based on their characteristics, such as destination, size and weight, and then the SPP is addressed with a focus on minimizing the number of *shiftings* and the heeling moment of each bay. Hu and Cai (2017) proposed a multi-objective mathematical model that only addresses stability constraints without taking into account stacking constraints related to containers. They used a heuristic algorithm to generate the initial solution while minimizing the number of *shiftings*, then this solution is optimized by a genetic algorithm considering longitudinal stability as an optimization objective. In addition, Li et al. (2018) formulated the CSPP for inland navigation on the Yangtze River by considering a multi-port stowage plan that aims to maximize capacity utilization by ensuring the use of fewer stacks. The issue considers the container weight to be uncertain due to logistical limitations at the Yangtze River terminals. The model ensures that there is no *shiftings* by allowing stacks to be made up only of containers of the same origin and destination. The stability constraints relate to the maximum weight of each stack and the tolerance of the trim and list weight, as proposed in Pacino et al. (2011). A hybrid neighborhood search algorithm is developed and tested against exact methods. More recently, Fazi (2019) formalized a typical stowage problem for a barge transporting maritime containers inland. The problem considers a dry-port transport system. They take into account stability measures during all transport phases and avoid unproductive *shiftings* movements while maximizing stowed containers. They developed a hybrid metaheuristic approach, based on local search and an industrial solver.

In this work, we address a new variant of the container stowage planning problem using single phase approach, thus, a detailed loading plan is generated according to the cellular structure of river vessels. This choice is mainly justified by the fact that, in the inland navigation case, the river units used are much smaller than mega-ships in maritime transport. To the best of our knowledge, no study in the literature considers the case of container stowage planning in barge convoy systems and simultaneously aims at optimizing the number of *shiftings*, the total stability of the convoy and the number of real-used barges in a multi-objective formulation.

3. Problem description

The 3D-CCSPP can be formally defined as follows: given a set of homogenous barges with a well-defined path, described by a sequence of ports to be visited with a uniform *shifting* cost, the problem is to find the stowage plan of a set of containers in the convoy while respecting a certain number of structural, operational and physical constraints related to both the containers and the convoy; all containers must be loaded on board and each container is identified by its size, weight, fragility and destination. The stowage plan is generated in order to optimize three objectives. The first one corresponds to the container unproductive movements during unloading operations at each port, the second represents convoy stability notion defined by the metacentric distance of each barge, while the third one is linked to the 3D bin-packing problem in order to reduce the number of barges used in the convoy (see Fig. 2).

3.1. The 3D-CCSPP’s relation with the 3D-BPP

The special structure of the barge convoy makes the 3D-CCSPP a special case of the 3D-BPP in the fixed-orientation orthogonal case (Dyckhoff, 1990). The latter is a combinatorial problem which consists of placing a set of n objects of dimensions $(w_i, h_i, d_i), i = 1, ..., n$, in the oriented case, in a minimum number of identical bins $R_j, j = 1, ..., m$, of dimensions (W, H, D) where W (resp.w_i) is the width, H (resp. h_i) is the height and D (respectively d_i) is the depth of the bins (resp. objects) so

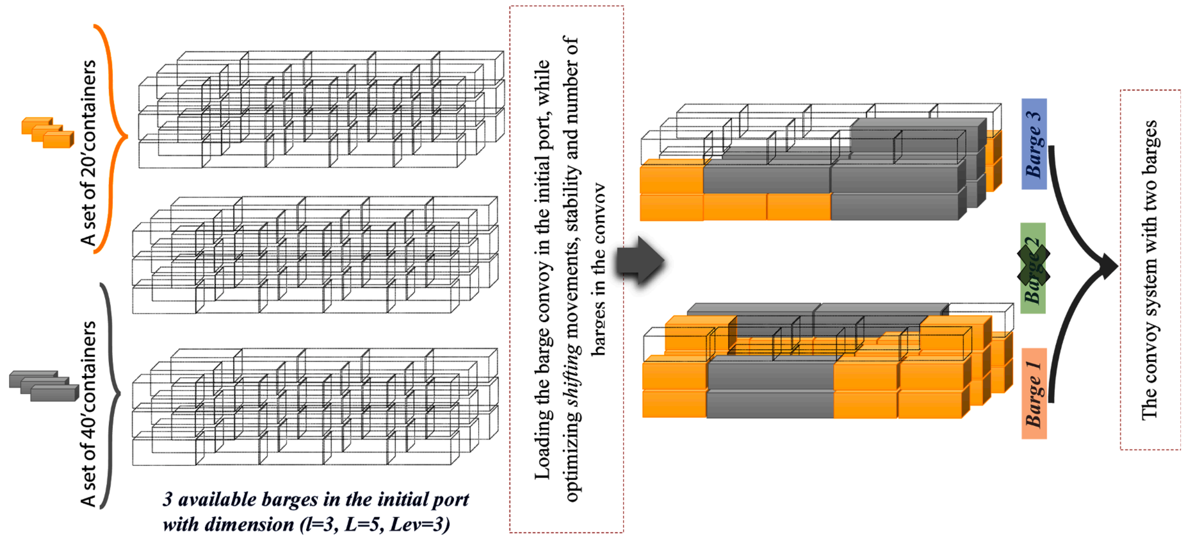

Fig. 2. Stowage planning in a barge convoy system.

that the edges of the objects are parallel to those of the bins that contain them without overflowing the bins and without any overlap. To our knowledge, there is no study in the literature dealing with the CSPP in a barge convoy as a bin-packing problem.

3.2. The barge stability according to the metacentric distance GM

We recall that the vessel stability generally depends on three factors, namely, the list, the trim and the metacentric distance (GM). More discussion of these three factors can be seen in (Barrass, 2000), (Imai et al., 2006) and (Azevedo et al., 2018). In this work, we consider the metacentric distance as a measure of convoy stability and assume that, the stability problems, raised by the other two factors and those raised in the visiting ports after *shiftings*, can be solved using ballast tanks.

The GM factor indicates the distance between the metacenter (M) and the center of gravity (G) of the barge, which shows, among other things, how easy or difficult it is for the barge to capsize (Fig. 3). The greater this distance is, the more difficult it becomes for the barge to overturn (Imai et al., 2006; Zhang & Lee, 2016). In general, excessive GM may be caused by: poor ship design, improper cargo distribution, or improper distribution of the bunker and ballast. In our study, we focus on the second point.

GM mainly deals with transverse stability, which concerns vertical and cross equilibrium. It can be calculated using the following formula: $GM = G_0M + (\sum_n w_n * m_n)/\Delta_T$ where G_0M is the initial metacentric distance, it is fixed for all barges. w_n is the weight of the container n. m_n is the vertical distance between the metacentre (M) and the container n. Δ_T is the weight of the barge with cargo, defined by $\Delta_T = \Delta_S + \sum_n w_n$, where Δ_S is the weight of the barge without cargo.

3.3. Container fragility and stack dynamic weight limit

Inspired by the BPP with fragile objects (Bansal et al., 2009), the concept of container fragility (which denotes how much weight it can support on its corner castings) can define both the weight limit of each stack of the barge and ensure the order of proper loading of containers according to their weight (lightweight containers in the upper levels). In light of this, we define the stack fragility by the allowable weight that can be added to the stack, taking into account the already stacked containers (see El Yaagoubi (2019) for more details).

More formally, let N be a set of containers to be loaded into a single barge of P stacks and let $N(p_t)$ be the set of containers stacked in the stackp_t, $\forall t \in \{1, \cdots, P\}$ and $Top(n)$ the set of containers stacked on top of container n. Each container is characterized by its weight w_n and its

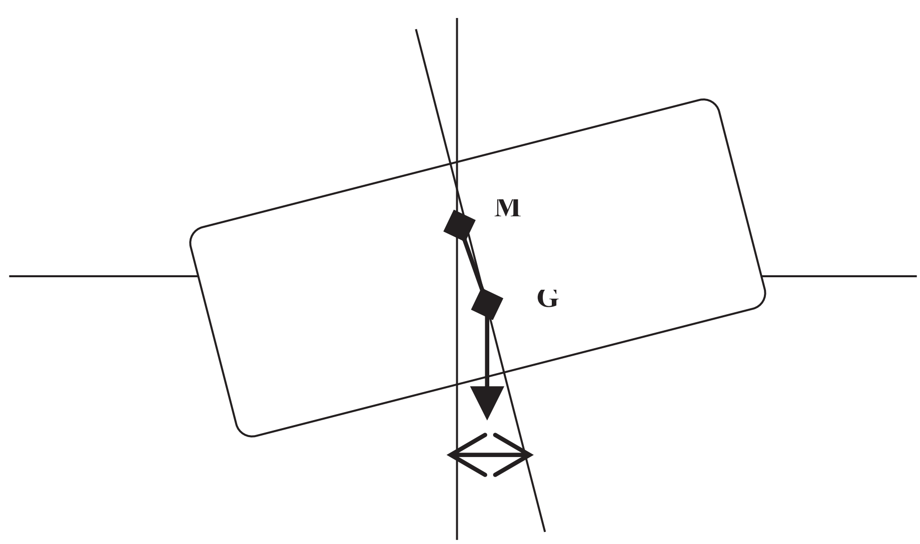

Fig. 3. The metacentric distance GM.

fragility f_n. The fragility of the stack p_t is calculated according to the following formula:

$$Fst_{p_t} = \min_{n \in N(p_t)} \left(f_n - \sum_{n' \in Top(n)} w_{n'} \right) \tag{1}$$

4. Mathematical formulation of the 3D-CCSPP

In this section, we propose an integer multi-objective mathematical model to solve the 3D-CCSPP. This modeling incorporates new and realistic constraints which reflect the actual operations during stowing containers in a convoy of homogeneous barges.

4.1. Problem assumptions

The basic assumptions used in this paper are listed as follows:

- Each barge of the convoy system has a rectangular shape,
- The number of barges available in the port is limited but sufficient to secure the transportation of all containers,
- All available barges are homogeneous and empty,
- The number of barges in the convoy does not exceed the number of barges available in the port,
- The barge convoy begins its tour from the initial port, for which we generate the stowage plan, then successively visits a finite number of ports known in advance where only unloading operations are allowed,
- Each barge is divided into a finite set of stacks. Each stack is defined by the combination of each row with each bay, thus constituting a set of slots. It can hold up to a maximum number of vertically stacked containers,
- Each slot can accommodate a 20 ft container. A TEU is considered as the discretization unit of slots inside the barge,
- The containers to be stowed are of two different sizes (40 ft and 20 ft).

4.2. Mathematical model

$$f_1 = Min \sum_{t=1}^{B} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev-1} \sum_{n \in N} \left(1 - \frac{1}{2} y_n\right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Lev} des_{n,n'} \left(y_{n'} \; U^t_{ijh'n'} + (1 - y_{n'}) \; V^t_{ijh'n'} \right) \right) \left(y_n \; U^t_{ijhn} + (1 - y_n) \; V^t_{ijhn} \right) \tag{4}$$

In order to model the 3D-CCSPP, we propose the following notations:

N_1 : the set of containers of size $40'$
N_2 : the set of containers of size $20'$

N : the set of all containers $N = N_1 \cup N_2$
L : length of the barge discretized in twenty feet
l : width of the barge discretized in eight feet
Lev : the maximum number of levels in the barge discretized in eight feet
m_h : the vertical distance between the metacenter of the barge and the level h
w_n : the weight of container n
B : the total number of barges in the initial port

f_n : the fragility of container n given in weight unit
$f_{max} = \max_{n \in N} \{f_n\}$: the maximum fragility of all containers
d_n : destination of container n

$$des_{n,n'} = \begin{cases} 1 \text{ if } d_n \text{ is before } d_{n'} \\ 0 \text{ otherwise} \end{cases}$$

$Des = (des_{n,n'})_{1 \leqslant n,n' \leqslant |N|}$ defines the succession relation between n and n'

$$y_n = \begin{cases} 1 \text{ if container } n \text{ is of type } N_1 \\ 0 \text{ if container } n \text{ is of type } N_2 \end{cases}$$

With these notations and parameters in hand, an integer nonlinear multi-objective mathematical formulation of the 3D-CCSPP is constructed by introducing the three variables below:

$$V^t_{ijhn} = \begin{cases} 1 \text{ if container } n \text{ is in slot } (i,j,h) \text{ of the barge } t, \; n \in N_2 \\ 0 \text{ otherwise} \end{cases}$$

$$U^t_{ijhn} = \begin{cases} 1 \text{ if container } n \text{ is in slot } (i,j,h) \text{ of the barge } t, \; n \in N_1 \\ 0 \text{ otherwise} \end{cases}$$

$$b_t = \begin{cases} 1 \text{ if the barge } t \text{ is used} \\ 0 \text{ otherwise} \end{cases}$$

We note that:

$$V : l \times L \times Lev \times N \times B \rightarrow \{0,1\}$$
$$(i,j,h,n,t) \rightarrow V^t_{ijhn}$$

$$U : l \times L \times Lev \times N \times B \rightarrow \{0,1\}$$
$$(i,j,h,n,t) \rightarrow U^t_{ijhn}$$

where:

$$\sum_{t=1}^{B} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev} V^t_{ijhn} = 0, \forall n \in N_1 \tag{2}$$

$$\sum_{t=1}^{B} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev} U^t_{ijhn} = 0, \forall n \in N_2 \tag{3}$$

The first objective function (f_1) makes it possible to minimize the total number of *shiftings* made during the convoy journey, it can be formulated according to equation (4):

The second objective function (f_2) is to maximize the total stability of the convoy by maximizing the sum of the metacentric distances of all the real-used barges in the convoy. To do this, we adopt the formulation of the GM stability factor proposed by Imai et al. (2002), in defining GM's contribution ratio (given in feet) as follows (5):

$$G_{nh} = \frac{w_n m_h}{\sum_{n \in N} w_n} \forall n \in N, \forall h \in \{1, \cdots, Lev\} \tag{5}$$

Then, the objective function f_2 can be formulated according to equation (6):

$$f_2 = Max \sum_{t=1}^{B} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev} \sum_{n \in N} G_{nh} \left(\left(1 - y_n\right) V^t_{ijhn} + \frac{1}{2} y_n \; U^t_{ijhn} \right) \tag{6}$$

Moreover, since in the 3D-CCSPP all the containers have to be loaded, the sum of containers weights is therefore constant. So, f_2 can be

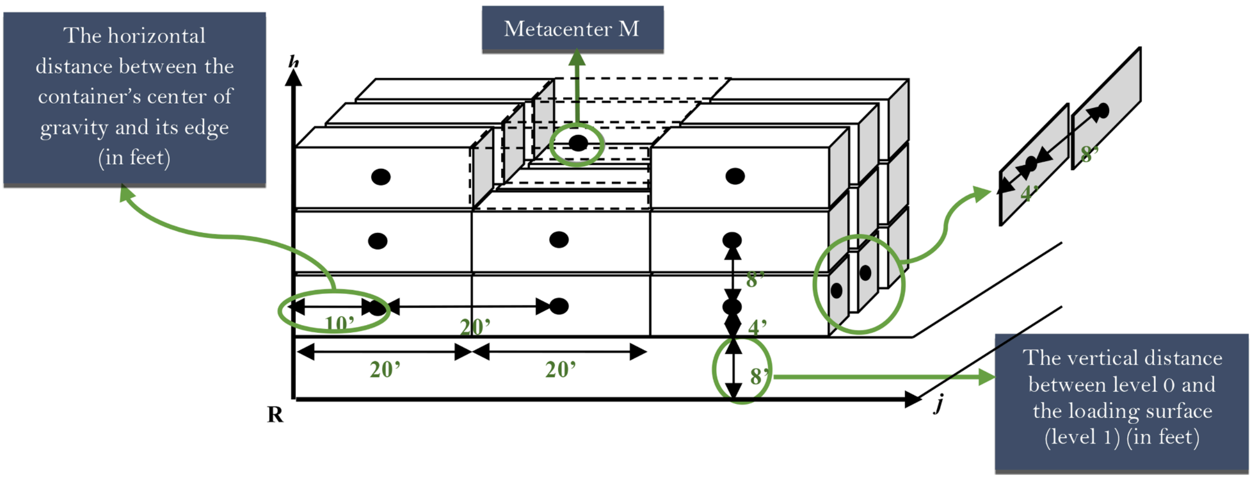

Fig. 4. The metacenter position in the barge.

reformulated by (7):

$$f_2 = Max\left(\frac{1}{\sum_{n\in N} w_n}\right)\sum_{t=1}^{B}\sum_{i=1}^{l}\sum_{j=1}^{L}\sum_{h=1}^{Lev}\sum_{n\in N}\left(w_n\, m_h\left(\left(1-y_n\right)V^t_{ijhn}+\frac{1}{2}y_n\, U^t_{ijhn}\right)\right) \quad (7)$$

The parameter m_h can be calculated as follows: firstly, we recall that, in the steady state of the barge, the metacenter M is above the center of gravity G, the two points are located on the same vertical axis of the barge. Based on the work of Imai et al. (2002), Imai et al. (2006), we assume that M is at the geometric center of the highest level whose position is $M(i,j,h) = \left(\frac{l}{2},\frac{L}{2},Lev\right)$ as shown in Fig. 4. Furthermore, the co-ordinates of M are calculated according to the following formula:

$$M\left(a_i,b_j,c_h\right) = (8i-4, 20j-10, 8h+4)$$

$$M\left(a_i,b_j,c_h\right) = \left(\frac{8l}{2}-4,\frac{20L}{2}-10, 8Lev+4\right) = (4l-4, 10L-10, 8Lev+4)$$

It should be noted that since GM is a vertical distance, all containers of a particular level have the same distance from the metacenter. That is why m_h will only depend on the height axis, hence the notation m_h. Thus, we calculate the vertical distance between the metacenter M of the barge and the level $h, \forall h \in \{1,\cdots,Lev\}$ according to equation (8):

$$m_h = \sqrt{(8Lev+4-8h-4)^2} = 8|Lev-h| \ \forall h \in \{1,\ldots,Lev\} \quad (8)$$

Therefore, the objective function (7) can be replaced by (9) as follows:

$$f_2 = Max\left(\frac{8}{\sum_{n\in N} w_n}\right)\sum_{t=1}^{B}\sum_{i=1}^{l}\sum_{j=1}^{L}\sum_{h=1}^{Lev}\sum_{n\in N} w_n|Lev-h|\left(\left(1-y_n\right)V^t_{ijhn}+\frac{1}{2}y_n\, U^t_{ijhn}\right) \quad (9)$$

Whereas the third objective function (f_3) is related to the bin-packing aspect which consists on minimizing the number of real-used barges to transport the set of containers (10).

$$f_3 = Min\sum_{t=1}^{B} b_t \quad (10)$$

The three objective functions of the mathematical model are optimized under the constraints described below:

$$\sum_{t=1}^{B}\sum_{i=1}^{l}\sum_{j=1}^{L}\sum_{h=1}^{Lev} V^t_{ijhn} = 1 \ \forall n \in N_2 \quad (11)$$

Constraints (11) ensure that all 20′ containers are loaded into the convoy and each container is loaded into one and only one slot of a single barge.

$$\sum_{t=1}^{B}\sum_{i=1}^{l}\sum_{j=1}^{L}\sum_{h=1}^{Lev} U^t_{ijhn} = 2 \ \forall n \in N_1 \quad (12)$$

$$U^t_{ij+1hn}\, U^t_{ij-1hn} \leqslant U^t_{ijhn}\left(1-U^t_{ij-1hn}\right) \quad \forall n\in N_1, \forall i\in\{1,\ldots,l\}, \forall j\in\{2,\ldots,L-1\}, \forall h\in\{1,\ldots,Lev\},\ \forall t\in\{1,\ldots,B\} \quad (13)$$

$$U^t_{ij+1hn} \geqslant U^t_{ijhn}\left(1-U^t_{ij-1hn}\right) \quad \forall n\in N_1, \forall i\in\{1,\ldots,l\}, \forall j\in\{2,\ldots,L-1\}, \forall h \in\{1,\ldots,Lev\}, \forall t\in\{1,\ldots,B\} \quad (14)$$

$$U^t_{i1hn} \leqslant U^t_{i2hn} \ \forall n\in N_1, \forall i\in\{1,\ldots,l\}, \forall h\in\{1,\ldots,Lev\}, \forall t\in\{1,\ldots,B\} \quad (15)$$

$$U^t_{iLhn} \leqslant U^t_{iL-1hn} \ \forall n\in N_1, \forall i\in\{1,\ldots,l\}, \forall h\in\{1,\ldots,Lev\}, \forall t\in\{1,\ldots,B\} \quad (16)$$

Constraints (12), (13), (14), (15) and (16) dictate that all 40′ containers are loaded and that each container is loaded into one and only one barge thus occupying two consecutive locations along the length of the container bays axis of the corresponding barge.

$$b_t \geqslant y_n U^t_{ijhn} + (1-y_n) V^t_{ijhn} \ \forall n \in N, \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Lev\}, \forall t \in \{1,...,B\} \tag{17}$$

Constraints (17) link the decision variables, they ensure that the barge t is used if at least one of its slots is occupied.

$$(h-1)\left(\sum_{n\in N}\left(y_n U^t_{ijhn} + (1-y_n)V^t_{ijhn}\right)\right) \leqslant \sum_{h'=1}^{h-1}\sum_{n'\in N}\left(y_{n'} U^t_{ijh'n'} + (1-y_{n'})V^t_{ijh'n'}\right) \quad \forall i \in \left\{1,...,l\right\}, \forall j \in \left\{1,...,L\right\}, \forall h \in \left\{2,...,Lev\right\}, \forall t \in \left\{1,...,B\right\} \tag{18}$$

Constraints (18) ensure that if a slot *(i, j, h)* of the barge t is occupied, then all locations below must be occupied as well.

$$1 - \left(y_n\ U^t_{ijhn} + (1-y_n)\ V^t_{ijhn}\right) \geqslant \sum_{\substack{n'\in N\\ n'\neq n}}\left(y_{n'}\ U^t_{ijhn'} + (1-y_{n'})\ V^t_{ijhn'}\right) \quad \forall n \in N,\ \forall i \in \left\{1,...,l\right\}, \forall j \in \left\{1,...,L\right\}, \forall h \in \left\{1,...,Lev\right\}, \forall t \in \left\{1,...,B\right\} \tag{19}$$

Constraints (19) indicate that if slot *(i, j, h)* of the barge t is occupied, then it must be occupied by one and only one container.

$$\sum_{h=1}^{Lev}\sum_{n\in N}\left(y_n\ U^t_{ijhn} + (1-y_n)\ V^t_{ijhn}\right) \leqslant Lev \ \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall t \in \{1,...,B\} \tag{20}$$

Constraints (20) ensure that the number of containers loaded in each stack of each convoy barge does not exceed the maximum number of levels in the barge. We recall that the barges constituting the convoy are homogeneous.

$$\sum_{j=1}^{L}\sum_{n\in N}\left(y_n\ U^t_{ijhn} + (1-y_n)\ V^t_{ijhn}\right) \leqslant L \ \forall i \in \{1,...,l\}, \forall h \in \{1,...,Lev\}, \forall t \in \{1,...,B\} \tag{21}$$

Constraints (21) indicate that the number of containers stowed in each row of each barge does not exceed the length L.

$$\sum_{i=1}^{l}\sum_{n\in N}\left(y_n\ U^t_{ijhn} + (1-y_n)\ V^t_{ijhn}\right) \leqslant l \ \forall j \in \{1,...,L\}, \forall h \in \{1,...,Lev\}, \forall t \in \{1,...,B\} \tag{22}$$

Constraints (22) indicate that the number of containers stowed in each bay of each barge does not exceed the width l.

$$\sum_{n\in N_2} V^t_{ijhn} \leqslant \sum_{n\in N_2} V^t_{ijh-1n} \ \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{2,...,Lev\}, \forall t \in \{1,...,B\} \tag{23}$$

All containers are fitted with corner casting imposing constraints (23) which ensure that 20′ containers cannot be stacked on top of 40′ containers.

$$\left(1-\sum_{n\in N_2}V^t_{ijhn}\right)\left(\sum_{n'\in N_1}U^t_{ijh+1n'} + \sum_{n\in N_2}V^t_{ijhn}\right) + \sum_{n\in N_2}V^t_{ijhn}\left(\sum_{n'\in N_1}U^t_{ijh+1n'} + \sum_{n\in N_2}V^t_{ijhn} - 1\right) \leqslant \sum_{n''\in N_2}V^t_{ij+1hn''}\left(\sum_{n\in N_2}V^t_{ijhn}\right) + \sum_{n'''\in N_1}U^t_{ijhn'''}\left(1-\sum_{n\in N_2}V^t_{ijhn}\right)$$
$$\forall i \in \{1,...,l\}, \forall j \in \{1,...,L-1\}, \forall h \in \{1,...,Lev-1\}, \forall t \in \{1,...,B\} \tag{24}$$

Constraints (24) show that if the slot *(i, j, h)* of a given barge t is occupied by a 20′ container and the slot *(i, j, h + 1)* of the same barge is occupied by a 40′ container, then the slot *(i, j + 1, h)* of t must be occupied by a 20′ container.

$$\sum_{h'=h+1}^{Lev}\sum_{\substack{n'\in N\\ n'\neq n}} w_{n'}\left(\frac{1}{2}y_{n'}\ U^t_{ijh'n'} + (1-y_{n'})\ V^t_{ijh'n'}\right) \leqslant f_{max} + \frac{1}{2}y_n\ U^t_{ijhn}(f_n - 2f_{max}) + (1-y_n)\ V^t_{ijhn}(f_n - f_{max}) \quad \forall n \in N, \forall i \in \left\{1,...,l\right\}, \forall j \in \left\{1,...,L\right\}, \forall h \in \left\{1,...,Lev-1\right\}, \forall t \in \left\{1,...,B\right\} \tag{25}$$

Constraints (25) ensure the dynamic weight limit of a stack, that is, the fragility of each container placed in a stack of a given barge must be greater than or equal to the sum of the weights of all the containers placed above (El Yaagoubi et al., 2018).

$$U^t_{ijhn} \in \{0,1\} \ \forall n \in N_1,\ \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Lev\}, \forall t \in \{1,...,B\} \tag{26}$$

$$V^t_{ijhn} \in \{0,1\}\ \forall n \in N_2,\ \forall i \in \{1,\dots,l\}, \forall j \in \{1,\dots,L\}, \forall h \in \{1,\dots,Lev\}, \forall t \in \{1,\dots,B\} \tag{27}$$

$$b_t \in \{0,1\}\ \forall t \in \{1,\dots,B\} \tag{28}$$

Constraints (26), (27) and (28) are the integrity constraints.

To linearize the model, we note $Z^t_{ijhn} = y_n\ U^t_{ijhn} + (1 - y_n)\ V^t_{ijhn}$, thus, we replace the objective function (4) by (29) by introducing a third decision variable $W^t_{ijhnh'n'}$ such as $W^t_{ijhn} = Z^t_{ijhn} * Z^t_{ijh'n'}$ and adding constraints (30) and (31) as follows:

$$f_1 = Min \sum_{t=1}^{B} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev-1} \sum_{n \in N} \left(1 - \frac{1}{2} y_n\right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Lev} des_{n,n'} W^t_{ijhnh'n'} \right) \tag{29}$$

$$W^t_{ijhnh'n'} \geqslant Z^t_{ijhn} + Z^t_{ijh'n'} - 1\quad \forall t \in \{1,..,B\}, \forall i \in \{1,..,l\},\ \forall j \in \{1,..,L\},\ \forall h \in \{1,..,Lev\}, \forall n \in N, \forall h' \in \{1,..,Lev\}, \forall n' \in N \tag{30}$$

$$W^t_{ijhnh'n'} \leqslant \frac{1}{2}\left(Z^t_{ijhn} + Z^t_{ijh'n'}\right)\quad \forall t \in \{1,..,B\}, \forall i \in \{1,..,l\},\ \forall j \in \{1,..,L\},\ \forall h \in \{1,..,Lev\}, \forall n \in N, \forall h' \in \{1,..,Lev\}, \forall n' \in N \tag{31}$$

5. Problem resolution

It is well known that the container stowage planning is an NP-hard optimization problem (Avriel et al., 2000). A good stowage plan should place each container in the most convenient location possible. However, the optimization of such a plan is difficult and complex because the possibilities are very numerous. For example, for a small container ship with a capacity of only 20 TEUs and 20 containers waiting to be loaded, there are 2.4×10^{18} possibilities. In addition, a convoy of, for example, 5 barges and 100 containers to be loaded offers even more possibilities. It may be difficult to use accurate methods to solve this problem in a timely manner. Therefore, we use the metaheuristic approach to solve it. In general, *meta*-heuristics provide a satisfactory but not necessarily optimal solution. The algorithms developed in this paper are based on the multi-objective concept by integrating a set of 1D bin-packing heuristics. A detailed description of these algorithms is given in the following sections.

5.1. Multi-objective aspect of the resolution approach

As well explained, the 3D-CCSPP is a multi-objective optimization problem aimed at optimizing the following three objective functions:

- Minimize the total number of *shiftings* throughout the tour,
- Maximize the total stability of the convoy according to the meta-centric distance factor,
- Minimize the number of barges actually used in the convoy.

In order to get the best solution, we need to be able to evaluate each of these functions and determine which of the different alternatives found best meets all the objectives (Hu & Cai, 2017). To solve this problem, we will use evolutionary algorithms that study different alternatives and make them "evolve" until a stop criterion is reached. As a result, this paper presents a new alternative for solving the 3D-CCSPP by searching for the optimal Pareto Front. This optimization framework is based on scalable calculation techniques, in particular, for the NSGA-II evolutionary multi-objective algorithm, which is at the heart of the

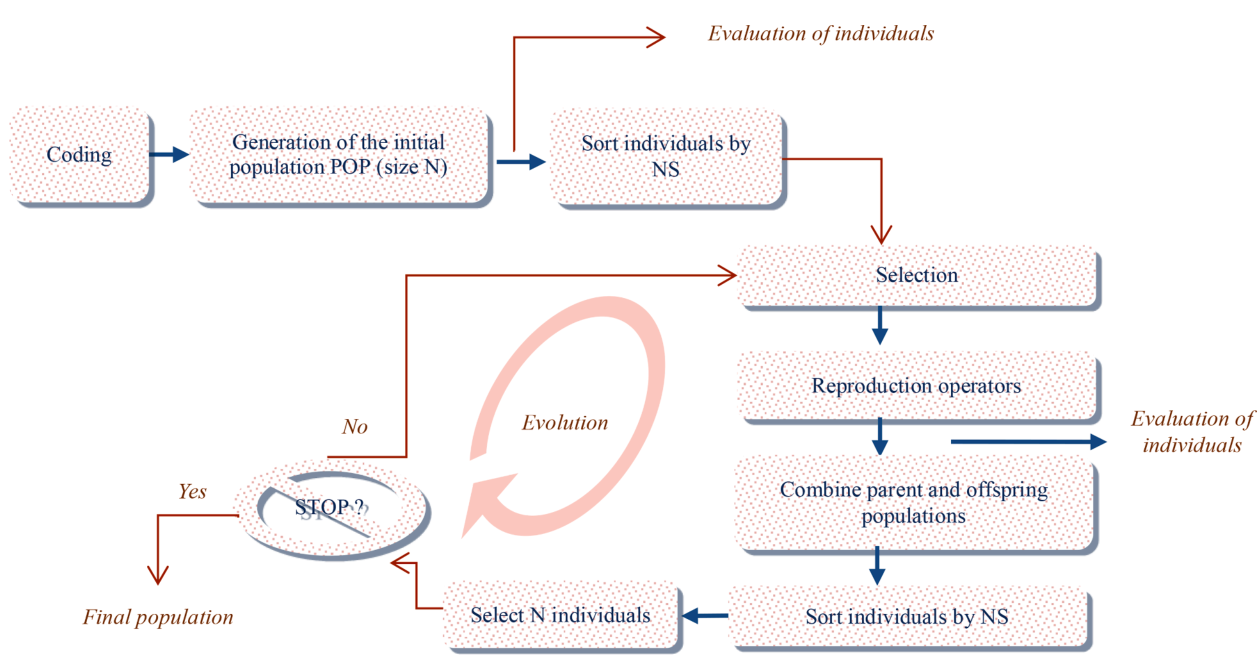

Fig. 5. Diagram of the NSGA-II.

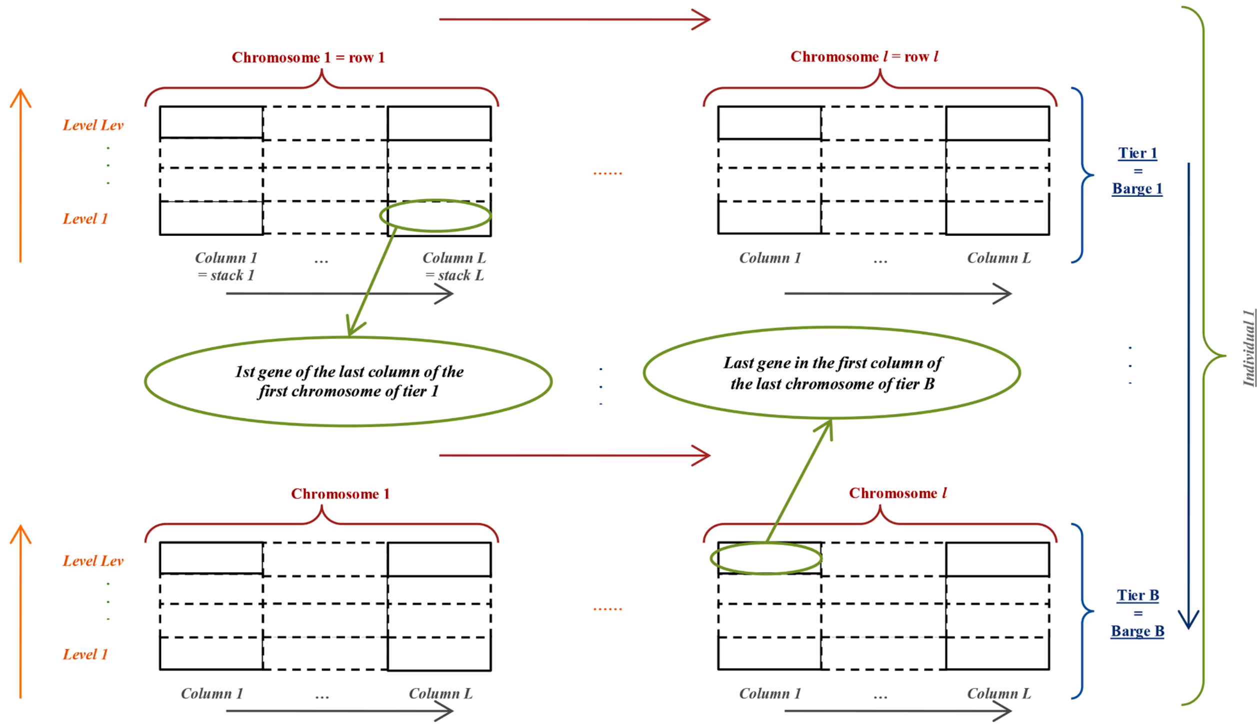

Fig. 6. The proposed encoding of the 3D-CCSPP.

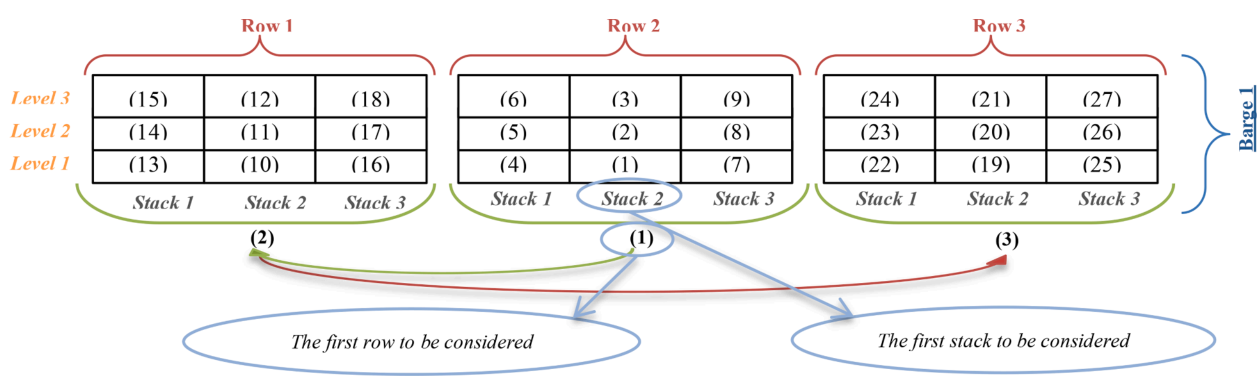

Fig. 7. Loading rule RP1.

proposed methods. In addition to its simplicity, the use of the NSGA-II in this paper is explained by the fact that it has not yet been applied to the container stowage planning problem while taking into account the five following factors: container size, dynamic weight limit of the stack, number of *shiftings*, number of real-used barges in the convoy and the stability factor. We design three new NSGA-II adaptations based on two bin-packing heuristics to solve the 3D-CCSPP. To do this, we first recall the basic concept of the NSGA-II algorithm.

5.2. General principle of NSGA-II

Over the years, the evolutionary approach NSGA-II has shown very good performance in terms of efficiency and effectiveness in addressing problems with multiple objectives. It is a robust research algorithm based on the principles of biological evolution. It provides, through the notion of population, relevant mechanisms to approach the Pareto optimal solution, namely, Pareto selection, elitism and diversification, more details are given in (Deb et al., 2002) and (Tan et al., 2002).

5.3. Adaptation of NSGA-II for the 3D-CCSPP

In this section, we adapt the different steps of the NSGA-II, illustrated in Fig. 5, to the 3D-CCSPP.

5.3.1. Encoding phase

The first step in NSGA-II is to properly define each individual in the population. This step associates with each point of the search space a specific data structure, called a chromosome. In the 3D-CCSPP, each

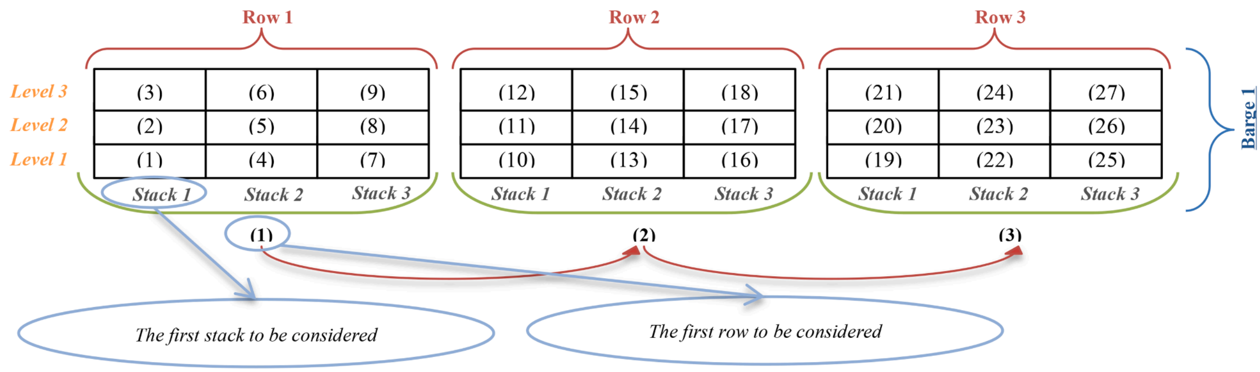

Fig. 8. Loading rule RP2.

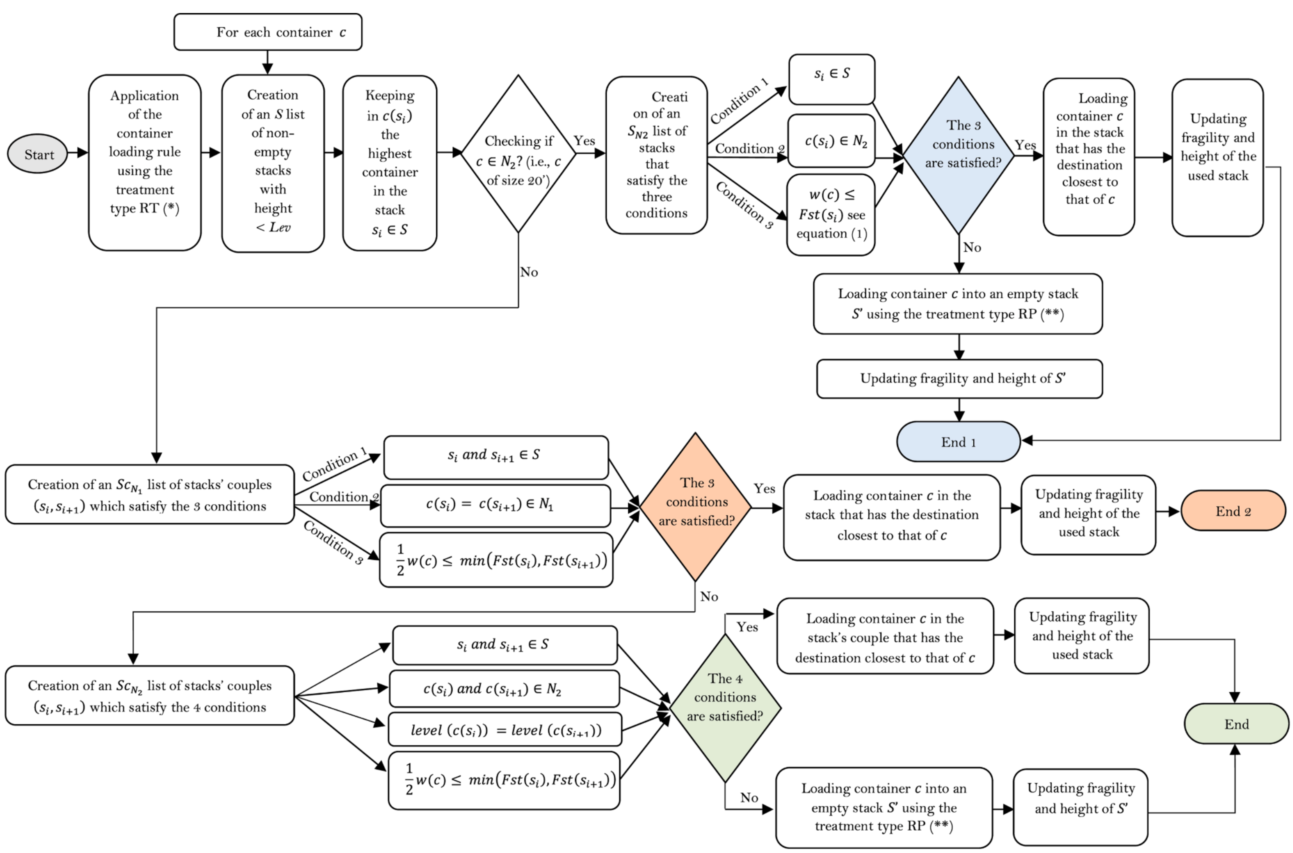

Fig. 9. General structure of the H_UniBarge heuristic.

solution is represented by a set of artificial chromosomes indicating the rows of each barge of the convoy. They are illustrated in several tier sets, where each tier set represents a barge. We note that the number of tier sets is equal to the number of barges available in the port, the number of chromosomes constituting a tier set equal to the number of rows of the corresponding barge and the total number of chromosomes constituting the individual is equal to the number of rows of the whole convoy. Each chromosome (row) consists of a set of columns representing the stacks, where the number of columns in each row is defined by the number of bays in the corresponding barge. Each stack consists of a set of genes indicating the slots in each barge, where the number of genes in each stack is equal to the number of levels of the corresponding barge (Fig. 6). Such a representation offers certain facilities. Indeed, the genotype becomes structured and can be broken down into different identifiable parts thus allowing its use by the crossing and mutation operators. We thus have two cases:

- In the case where the location is occupied, the gene is illustrated as a square containing a triplet (n, d_n, N_l) with "n" the identifier of the container, "d_n" the port of destination of the container and "$N_{l,l\in\{1,2\}}$" the container size where index 1 indicates the size 40′ and index 2 represents 20′,
- In the case where the location is empty, the gene is illustrated as a square containing the character "-".

5.3.2. Generation of the initial population

We note that the NSGA-II is highly dependent on the initial population which is often generated in a random manner, however, when the problem to be solved is subject to several constraints, it becomes difficult to find a randomly feasible solution. This is the case of the 3D-CCSPP. We have, thus, designed two loading heuristics based on the 1D-BPP to generate the initial population of the 3D-CCSPP.

5.3.2.1. Loading rules. The proposed heuristics consist of stowing the containers iteratively in the barges. To do this, we first introduce two types of treatment:

- « RT »: the order in which the containers are examined,
- « RP »: the barge and the position in which we seek to place the container.

For each type, we propose, respectively, two loading rules explained as follows:

- Loading rule "RT1": The set of N containers to be loaded is sorted, first by size, giving priority to 20′ containers, then in the descending order of their weight.
- Loading rule "RT2": The set of N containers to be loaded is sorted, first in the descending order of their weight, then by their size, i.e., for the set of containers with the same weight, we load 20′ containers first.

The two loading rules ("RT1" and "RT2") can help avoid bad situations where 20′ containers are stowed on top of 40′, or, when heavier containers are stowed on top of lighter ones. Moreover, in order to introduce diversity into the individuals constituting the initial population, we randomize the first container to be loaded, thus giving two new rules "RTN1" and "RTN2".

- Loading rule "RTN1": The first n containers to be loaded from the set N ($n > 2$) are chosen at random, and then the rest are loaded according to the RT1 rule.
- Loading rule "RTN2": The first n containers to be loaded from the set N ($n > 2$) are chosen at random, and then the rest are loaded according to the RT2 rule.
- Loading rule "RP1": Following the order shown in Fig. 7, the containers are loaded row by row starting from the middle of the barge, along the width axis, and continuing in an alternative way to the two sides, port and starboard, stack by stack, filling the stack which is in the middle of the row, along the length axis, and continuing towards the two ends bow and stern of the barge in an alternative way, and this starting from the bottom level of each stack towards the highest level. Each gene (i.e., slot) contains a number between two parentheses indicating its priority for load operations. That is, when loading containers, the chosen slot is determined according to its priority, starting with the one with the lowest number (El Yaagoubi et al., 2020). In addition, this loading rule is based on the bin-packing heuristic First Fit, which will be explained later (Martello et al., 2000).
- Loading rule "RP2": Following the sequence shown in Fig. 8, the containers are loaded row by row starting with the first row on the port side of the barge and continuing sequentially to the starboard side, stack by stack, filling the one at the lower left corner of the bow and continuing to the stern of the barge in a sequential manner, starting from the bottom level of each stack towards the highest level. In the same way, this rule follows the First Fit heuristic principle of the bin-packing problem.

5.3.2.2. Container loading heuristic in a single barge H_UniBarge. To load a set of containers in a given barge, we propose the H_UniBarge heuristic which is inspired by the one proposed by El Yaagoubi et al. (2020), and based on the loading rules RTN1, RTN2, RP1 and RP2 (see the flow chart illustrated in Fig. 9).

To introduce even more diversity to the solutions generated by H_UniBarge heuristic, we use two additional randomization procedures. On the one hand, since we have developed two constructive loading rules, RTN1 and RTN2, we choose one at random to elaborate the solution at each execution of the heuristic (indicated by (*) in the flow chart (Fig. 9)). On the other hand, instead of always starting the loading from the lower-left corner of the bow, we randomly choose, at each iteration, a loading rule according to the priority order of the positions in the barge, RP1 or RP2 (indicated by (**) in flow chart (Fig. 9)).

5.3.2.3. Loading heuristic in a barge convoy system H_MultiBarge. To stow a set of containers in a set of barges constituting a convoy, we propose the H_MultiBarge heuristic that hybridizes the H_UniBarge heuristic with the 1D bin-packing problem heuristics.

- Next Fit (NF) heuristic: In this method we only consider one open barge at a time. The containers are processed according to a given order based on the rules RTN1 and RTN2. The containers are stowed successively in the open barge as long as there is room for the current container, otherwise the current barge is closed, and a new barge is opened.

- First Fit (FF) heuristic: Initially only one barge is considered, and the containers are treated according to a given order (RTN1 and RTN2). When there is no more room in the first barge to stow the current container, a second barge is then opened but without closing the first. In an intermediate stage where we have a set B of open barges numbered from 1 to | B | in the order of their first use, a container *c* in progress is stored in the barge of the smallest number that can contain it. In the case where no barge can contain *c*, a new barge (| B | + 1) is then used without closing the others. The order in which the containers are handled is crucial for the quality of the solution.

By combining the H_UniBarge heuristic with the bin-packing heuristics just presented, we obtain two heuristics of container loading in a barge convoy HNF_MultiBarge (Algorithm 1) and HFF_MultiBarge (Algorithm 2). In order to introduce even more diversity into the individuals of the initial population, we use another additional randomization procedure. Since we have developed two container loading heuristics in a barge convoy, HNF_MultiBarge and HFF_MultiBarge, we choose one at random for each generation of an individual in the population. Due to all the randomization rules used, we can generate a very diverse initial population of size *n*.

Algorithm 1
General structure of the constructive heuristics HNF_MultiBarge.

Inputs
Open the first barge of the convoy
Apply the container loading rule according to the type of treatment RT (*)
for each container c **do**
  if there is room in the current open barge **then**
    S = non-empty stack of height $< Lev$
    $c(s_i)$ = the container that is located in the heighest level of the stack $s_i \in S$
    if $c \in N_2$ **then**
      $S_{N2} = \{s_i / s_i \in S$ and $c(s_i) \in N_2$ and $w(c) \leq Fst(s_i)\}$
      if $S_{N2} \neq \varnothing$ **then**
        $S_{N2}^{min} = \underset{s \in S_{N2}}{\text{argmin}}(des(c(s))$ (the stack that has the destination closest to that of c)
        Load container c into the stack S_{N2}^{min}
       Update the fragility and height of the stack S_{N2}^{min}
      else
       Load container c into an empty stack depending on the type of treatment RP (**)
       Update the fragility and height of the stack
      end if
    else
      $Sc_{N1} = \left\{ (s_i, s_{i+1}) / s_i,\ s_{i+1} \in S\ \&\ c(s_i) = c(s_{i+1}) \in N_1\ \&\ \frac{1}{2} w(c) \leqslant min(Fst(s_i), Fst(s_{i+1})) \right\}$
      if $Sc_{N1} \neq \varnothing$ **then**
        $Sc_{N1}^{min} = \underset{s \in Sc_{N1}}{\text{argmin}}(des(c(s))$
        Load container c into the stack Sc_{N1}^{min}
       Update the fragility and height of the stack Sc_{N1}^{min}
      else
       $Sc_{N2} = \left\{ (s_i, s_{i+1}) / s_i\ ,\ s_{i+1} \in S\ \&\ c(s_i), c(s_{i+1}) \in N_2\ while\ c(s_i) \neq c(s_{i+1})\ and\ level\ (s_i) = level\ (s_{i+1})\ \&\ \frac{1}{2} w(c) \leqslant min(Fst(s_i), Fst(s_{i+1})) \right\}$
       if $Sc_{N2} \neq \varnothing$ **then**
        $Sc_{N2}^{min} = \underset{s \in Sc_{N2}}{argmin}(des(c(s)))$
        Load container c in the stack Sc_{N2}^{min}
       Update the fragility and height of the stack Sc_{N2}^{min}
      else
       Load container c into an empty stack depending on the type of treatment RP (**)
       Update the fragility and height of the stack
      end if
     end if
    end if
  else
    Open a new barge and close the old one
    $c--$
  end if
end for

Algorithm 2
General structure of the constructive heuristics HFF_MultiBarge.

```
Inputs
Open the first barge of the convoy
Apply the container loading rule according to the type of treatment RT (*)
B = {openbarges}
for each container c do
  for each barge b from 1 to |B| do
   if there is room in the barge b then
     S = non-empty stack of height < Lev
     c(s_i) = the container that is located in the heighest level of the stack s_i ∈ S
     if c ∈ N_2 then
       S_N2 = {s_i/s_i ∈ S& c(s_i) ∈N_2 &w(c) ≤ Fst(s_i)}
       if S_N2 ≠ ∅ then
        S_N2^min = argmin_{s∈S_N2}(des(c(s)) (the stack that has the destination closest to that of c)
         Load container c into the stack S_N2^min
         Update the fragility and height of the stack S_N2^min
        break
       else
        Load container c into an empty stack depending on the type of treatment RP (**)
        Update the fragility and height of the stack
       break
      end if
     else
      Sc_N1 = {(s_i, s_{i+1})/s_i, s_{i+1} ∈ S&c(s_i) = c(s_{i+1}) ∈ N_1&(1/2)w(c) ≤ min(Fst(s_i), Fst(s_{i+1}))}
      if Sc_N1 ≠ ∅ then
        Sc_N1^min = argmin_{s∈Sc_N1}(des(c(s))
         Load container c into the stack Sc_N1^min
        Update the fragility and height of the stack Sc_N1^min
        break
      else
        Sc_N2 = {(s_i, s_{i+1})/s_i, s_{i+1} ∈ S&c(s_i), c(s_{i+1}) ∈N_2withc(s_i) ≠c(s_{i+1})&level(s_i) = level(s_{i+1})&(1/2)w(c) ≤ min(Fst(s_i), Fst(s_{i+1}))}
        if Sc_N2 ≠ ∅ then
          Sc_N2^min = argmin_{s∈Sc_N2}(des(c(s))
           Load container c into the stack Sc_N2^min
           Update the fragility and height of the stack Sc_N2^min
          break
         else
           Load container c into an empty stack depending on the type of treatment RP (**)
           Update the fragility and height of the stack
           break
         end if
        end if
      end if
     end if
    end if
  end for
  if c is not loaded yet
    Open a new barge b + 1 without closing the old one
  B = B ∪ {b + 1}
    Load c into the barge b + 1 depending on the type of treatment RP (**)
  end if
end for
```

5.3.3. Selection operator

The selection operator has for mission to choose, in the present population, the future parents necessary for the reproduction stage. It is also used for rebuilding the population in the replacement stage. In the 3D-CCSPP, the binary tournament selection due to its simplicity, efficiency and ability to produce an acceptable solution and handle either minimization or maximization issues without any structural changes (Deb, 2000; Blickle, 2000). It consists of randomly selecting two individuals from the population and selecting the one with the highest quality according to the crowded comparison operator.

The crowded comparison operator ($<_n$) is used to guide the process of selection with the uniform distribution of Pareto solutions. Each individual i of the population is characterized by: the rank of non-domination i_{rang} and the crowding distance i_{dist}. We define the partial order $<_n$ of two individuals i and j as follows:

$$i <_n j if \left(i_{rang} < j_{rang}\right) or i_{rang} = j_{rang} and i_{dist} > j_{dist}$$

In other words, the solution belonging to the smallest-order Pareto front is preferred if i and j belong to two Pareto fronts. Otherwise, we choose the solution that has the largest crowding distance.

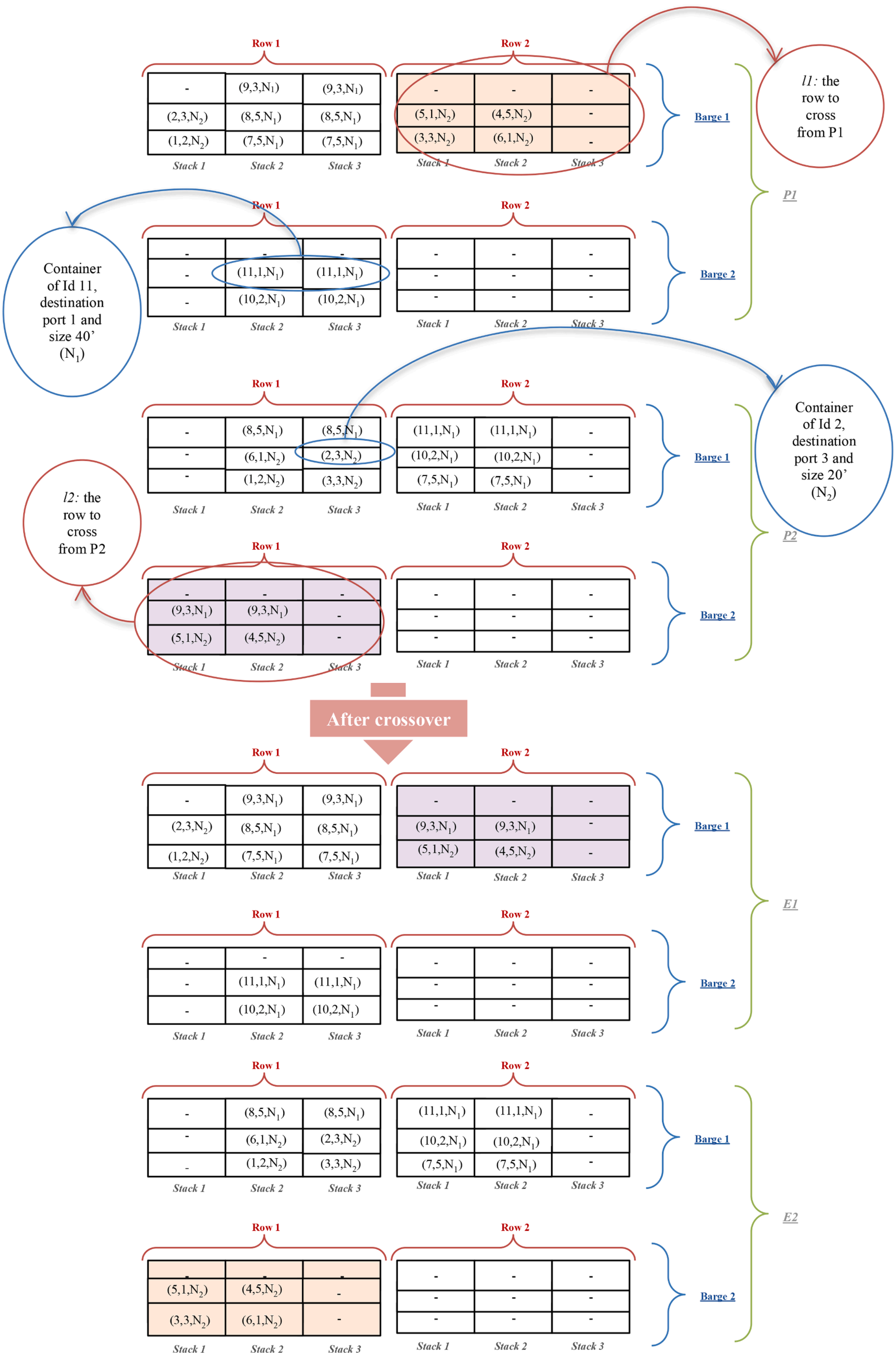

Fig. 10. The proposed crossover operator of the 3D-CCSPP.

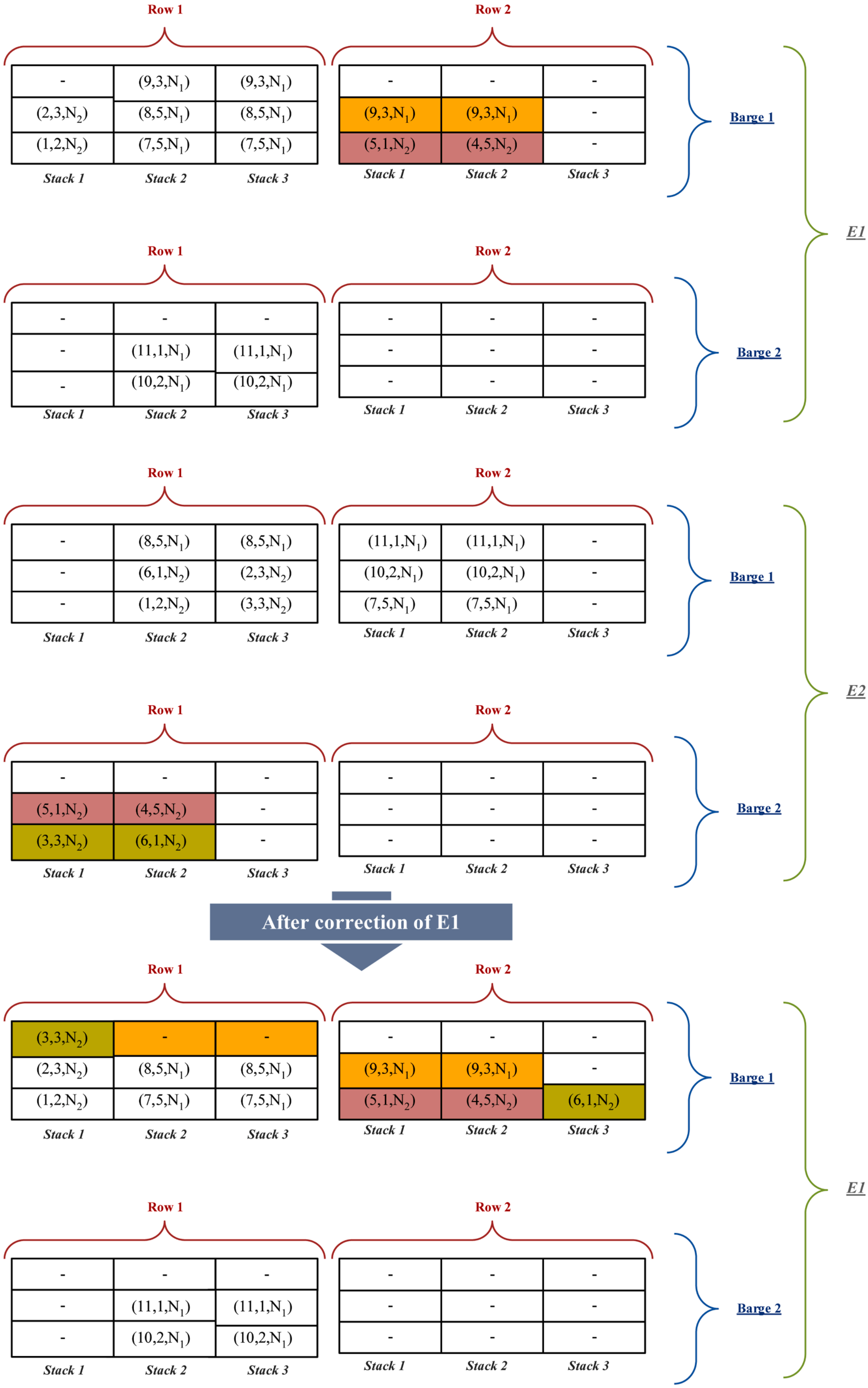

Fig. 11. Correction strategy of offspring E1.

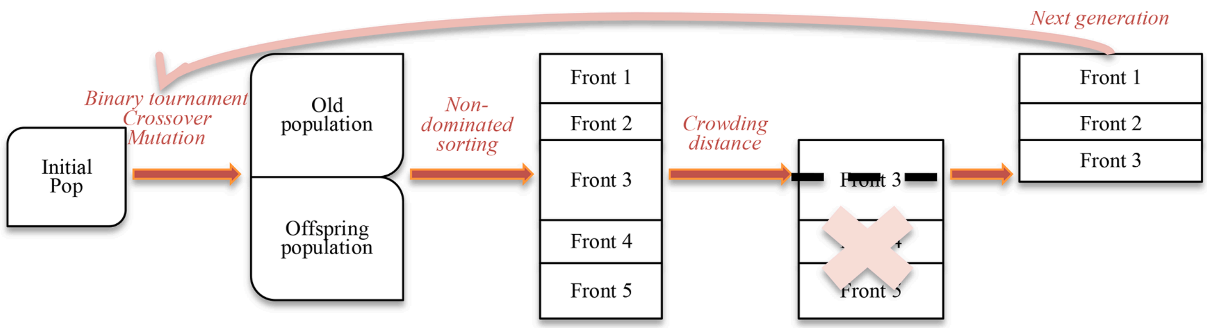

Fig. 12. A schematic iteration of the NSGA-II approach.

Table 1
Parameter settings of NSGA-II.

Parameter	*Value*
Population size	$\in \{50, 100, 150\}$
Crossover probability (Pc)	0.9
Mutation probability (Pm)	0.05
Maximal number of iterations (tmax)	500

5.3.4. Reproduction phase

5.3.4.1. Crossover operator. In the 3D-CCSPP, the resulting container loading plans in the convoy are associated with several difficulties and may be impractical if the crossover operator is not properly designed. It is therefore impossible to directly apply conventional crossing operators. In what follows, we design a new crossover operator specific to the 3D-CCSPP to deal with these difficulties, namely, the loading constraints imposed by the size of the containers (20′ and 40′) and the dynamic weight limit of each stack. The proposed crossover proceeds in two phases:

- The first phase consists in choosing at random two non-empty rows to cross (l_i and l_j) which can have different indices belonging to two different barges, each belonging to a parent, then exchanging them creating two children.
- The feasibility of these new individuals must be verified during the second phase using the proposed Feasibility Checking Heuristic (FCH).

In Fig. 9, we illustrate the operation of the proposed crossover. In this example, the rows to be exchanged between the two parents P1 and P2 are *l1* (row 2 of barge 1 of P1) and *l2* (row 1 of barge 2 of P2) represented by two different colors. We note that the feasibility of two offsprings E1 and E2 must be verified.

Table 2
Convoy barge system characteristics.

Id_Convoy	*Nmax*	*Dim_Barge*			*B_CapM*	*C_CapM*
		l	*L*	*Lev*		
Cv1	2	2	4	2	16	32
Cv2	2	2	3	4	24	48
Cv3	2	2	5	3	30	60
Cv4	2	2	4	4	32	64
Cv5	2	3	4	3	36	72
Cv6	2	3	6	3	54	108
Cv7	6	5	6	3	90	540
Cv8	5	4	7	4	112	560
Cv9	5	5	8	4	160	800
Cv10	5	5	10	4	200	1000
Cv11	3	6	12	5	360	1080
Cv12	3	7	11	6	462	1386

Table 3
Weight range of containers.

Size	*L (Leight)*	*M (Medium)*	*H (Heavy)*
20′	[3–15]	[16–24]	[25–29]
40′	[4–15]	[16–24]	[25–31]

5.3.4.2. Feasibility Checking heuristics (FCH). The crossover operation may result in infeasible individuals. It is therefore necessary to verify and correct them. The basic verification rules of the 3D-CCSPP are: the absence or repetition of containers, container size constraints and stack fragility constraints. The crossover operator proposed in this work makes it possible to reduce the three feasibility verification rules into one, since, in our case, the notion of non-feasibility can exist only in terms of the absence or repetition of the containers in the convoy. For example, in Fig. 10, the non-feasibility of the two offsprings E1 and E2 is observed only in terms of the repetition of some containers and the absence of others, and not in terms of container size constraints and weight limit of the stack. As a result, we propose the Feasibility Checking Heuristic (FCH) which allows reloading the chosen rows l_i and l_j, respectively, in a random way, while considering the containers that are not yet loaded and those already loaded. More precisely, if there is a container in row l_j (the chosen row of the second parent) which does not exist in row l_i (the chosen row of the first parent), we delete it from the other uncrossed rows of the offspring E_i to correct (the one from the first parent and row l_j) while repairing the generated empty slots and updating the fragility and height of the corresponding stacks. Otherwise, if there is a container in row l_i which does not exist in row l_j, we reload it in the first suitable position of the offspring E_i according to rules RP1 or RP2.

We note that the removal of containers, when correcting the offspring individual, is done at the level of the other rows except the

Table 4
Containers characteristics.

InstC	*|N|*	*N1*		*N2*		*N_TEU*
		|N1|	*W TM*	*|N2|*	*W TM*	
InstC1	7	4	L/M	3	H	11
InstC2	10	2	L/H	8	M	12
InstC3	14	10	M/H	4	L	24
InstC4	15	3	H	12	L/M	18
InstC5	16	8	M	8	L/H	24
InstC6	20	15	L	5	M/H	35
InstC7	25	11	L/M	14	H	36
InstC8	27	7	L/H	20	M	34
InstC9	32	20	L	12	M/H	52
InstC10	70	34	L/M/H	36	L/M/H	104
InstC11	100	75	L/M/H	25	L/M/H	175
InstC12	200	150	L/M/H	50	L/M/H	350
InstC13	300	160	L/M/H	140	L/M/H	460
InstC14	400	211	L/M/H	189	L/M/H	611
InstC15	500	261	L/M/H	239	L/M/H	761

crossed one, otherwise the offspring will be identical to its parent. Fig. 11 illustrates the operation of the correction strategy proposed, using the offspring E1 obtained from the crossover operation shown in Fig. 10. In this example and according to FCH, E1 is not feasible in terms of the repetition of container 9 and the absence of containers 3 and 6. The red color represents the containers to be kept, the orange color indicates the containers to be removed from the other rows of E1 except the crossed row (l_1), and the green color represents the containers to be reloaded in the convoy. When removing repeated 20′ containers, we must always check the size of the containers above them to avoid the generation of empty slots between stack levels. To do so, we propose the Heuristic of Deleting a Twenty-foot container (HDT). Indeed, to delete the container c from it slot (i,j,h), we search, in the above levels $(>h)$, for a 40′ container that occupies the slot $(i,j+1,k>h)$, if there is any, we remove the 20′ container which is stacked in the slot $(i,j+1,h)$ and add it to the list of containers to be reloaded in the correction step according to FCH, otherwise, we remove the 20′ container which is in the slot $(i,j-1,h)$ and add it to the list of containers to be reloaded in the correction phase according to FCH.

5.3.4.3. Mutation operator. The role of the mutation operator is to introduce the diversity of solutions to overcome local optima. It consists of modifying one or more genes of an individual selected by the selection operator. In the 3D-CCSPP, we propose two mutation strategies:

- Mutation M1: Choose two barges of the convoy randomly, then, swap between a random row of the first barge with another random row of the second barge.
- Mutation M2: Choose a set B of real-used barges in the convoy in a random manner, then, for each barge $t \in B$, swap two randomly chosen rows from the same barge.

5.3.5. Replacement phase

The basic iterations of NSGA-II are described in Fig. 12. In each generation of NSGA-II, genetic operators are applied to the parent population to obtain a child population of equal size. Parents and children are then combined to form a temporary population on which the Fast Non-Dominated Sorting (NS) approach and crowding distance operator (Deb et al., 2002) are applied to construct the next-generation parent population of the same size as the initial one.

6. Computational results

To test the quality and performance of the NSGA-II and the proposed model, we consider instances based on a combination of real instances of river flows from the HAROPA project proposed by the Greater Port of Le Havre (GPMH) with a set of randomly generated small, medium and large instances, thus presenting different situations encountered by inland navigation services. All our data sets and results are available on https://github.com/AminaElya/Optimization-of-3D-CCSPP.git. All approaches used for the 3D-CCSPP resolution have been implemented with Java using NetBeans 8.0. By default, all experiments were conducted on an Intel® Core ™ i5-4570 CPU @ 3.20 GHz, 4 GB RAM. The final setting of the parameters of the proposed algorithm is shown in Table 1. The values presented in this table are the result of several intensive studies that were conducted to refine the NSGA-II. We note that the size of the population (n) is set alternately at 50, 100 and 150. Indeed, we didn't choose a fixed value because we have compared the solutions found in

Table 5
Instance generation of the 3D-CCSPP.

InstCCSP	*Id_Convoy*	*InstC*	*P*	*InstCCSP*	*Id_Convoy*	*InstC*	*P*
CCSP1	Cv1	InstC1	5	CCSP37	Cv9	InstC10	15
CCSP2		InstC2	5	CCSP38			25
CCSP3		InstC3	5	CCSP39			35
CCSP4	Cv2	InstC1	5	CCSP40		InstC11	15
CCSP5		InstC2	5	CCSP41			25
CCSP6	Cv3	InstC3	5	CCSP42			35
CCSP7			10	CCSP43		InstC12	15
CCSP8	Cv4	InstC4	5	CCSP44			25
CCSP9			10	CCSP45			35
CCSP10		InstC5	5	CCSP46	Cv10	InstC13	15
CCSP11			10	CCSP47			25
CCSP12	Cv5	InstC6	5	CCSP48			35
CCSP13			10	CCSP49		InstC14	15
CCSP14		InstC7	5	CCSP50			25
CCSP15			10	CCSP51			35
CCSP16		InstC8	5	CCSP52		InstC15	15
CCSP17			10	CCSP53			25
CCSP18	Cv6	InstC9	15	CCSP54			35
CCSP19	Cv7	InstC10	15	CCSP55	Cv11	InstC13	15
CCSP20			25	CCSP56			25
CCSP21			35	CCSP57			35
CCSP22		InstC11	15	CCSP58		InstC14	15
CCSP23			25	CCSP59			25
CCSP24			35	CCSP60			35
CCSP25		InstC12	15	CCSP61		InstC15	15
CCSP26			25	CCSP62			25
CCSP27			35	CCSP63			35
CCSP28	Cv8	InstC10	15	CCSP64	Cv12	InstC13	15
CCSP29			25	CCSP65			25
CCSP30			35	CCSP66			35
CCSP31		InstC11	15	CCSP67		InstC14	15
CCSP32			25	CCSP68			25
CCSP33			35	CCSP69			35
CCSP34		InstC12	15	CCSP70		InstC15	15
CCSP35			25	CCSP71			25
CCSP36			35	CCSP72			35

Table 6
Numerical results, using CPLEX, of the small instances of Table 5 and their corresponding status, by optimizing the number of *shiftings* “Shift”, the stability GM “Stab” and the number of barges of the convoy “BNb”, where «$\alpha = 1$, $\beta = 0$ & $\gamma = 0$», «$\alpha = 0$, $\beta = 1$ & $\gamma = 0$» and «$\alpha = 0$, $\beta = 0$ & $\gamma = 1$», respectively.

Instance		Number of variables	Number of constraints	Shift				Stab				BNb			
Id	*(|N|-P-B-l-L-Lev)*			Obj Val	CPLEX Status	Time Status	Iterations	Obj Val	CPLEX Status	Time Status	Iterations	Obj Val	CPLEX Status	Time Status	Iterations
CCSP1	(7–5-2–2-4–2)	3811	7190	0*	P.O.	0.22	**8**	6*	P.O.	0.27	51	1*	P.O.	0.25	101
CCSP2	(10–5-2–2-4–2)	7363	14,140	0*	P.O.	1.02	48	10.78*	P.O.	3.16	1588	1*	P.O.	3.64	1555
CCSP3	(14–5-2–2-4–2)	13,891	26,820	0*	P.O.	2.17	68	7.68*	P.O.	2.28	158	2*	P.O.	4.72	189
CCSP4	(7–5-2–2-3–2)	10,419	20,310	0*	P.O.	3.61	61	13.89*	P.O.	14.23	33	1*	P.O.	13.3	467
CCSP5	(10–5-2–2-3–4)	20,643	40,600	0*	P.O.	15.11	505	15*	P.O.	24.02	688	1*	P.O.	25.73	318
CCSP6	(14–5-2–2-5–3)	37,803	73,906	0*	P.O.	23.31	107	7.71*	P.O.	2.45	138	2	P.O.	2.42	394
CCSP7	(14–10-2–2-5–3)	37,803	73,906	0*	P.O.	16.39	127	12.84*	P.O.	16.94	138	1*	P.O.	28.03	394
CCSP8	(15–5-2–2-4–4)	60,483	119,374	0*	P.O.	47.44	1019	13.62*	P.O.	38.55	532	1*	P.O.	65.74	3576
CCSP9	(15–10-2–2-4–4)	60,483	119,374	0*	P.O.	210.66	43,269	13.36*	P.O.	43.52	394	1*	P.O.	69.94	3576
CCSP10	(16–5-2–2-4–4)	68,611	135,360	0*	P.O.	93.83	179	8.72*	Current	3600	370,985	1*	P.O.	108.77	2541
CCSP11	(16–10-2–2-4–4)	68,611	135,360	0*	P.O.	108.89	24,054	9*	Current	3600	1126	1*	P.O.	28.33	2541
CCSP12	(20–5-2–3-4–3)	90,723	178,822	0*	P.O.	67.45	586	10.01*	Current	3600	67,955	2	Current	3600	22,722
CCSP13	(20–10-2–3-4–3)	90,723	178,822	0*	P.O.	372.77	44,461	7.21*	Current	3600	81,678	2	Current	3600	57,820
CCSP14	(25–5-2–3-4–3)	140,403	277,208	0*	P.O.	166.97	1582	14.76*	P.O.	145.83	3253	2	Current	3600	198,948
CCSP15	(25–10-2–3-4–3)	140,403	277,208	0*	P.O.	175.38	959	14.76*	P.O.	147.09	3253	2	Current	3600	136,899
CCSP16	(27–5-2–3-4–3)	163,299	322,524	0*	P.O.	175.61	1309	14.97*	P.O.	161.41	3550	2	Current	3600	11,486
CCSP17	(27–10-2–3-4–3)	163,299	322,524	0*	P.O.	169.34	948	14.8*	P.O.	166.02	3654	1*	P.O.	371.69	30,999
CCSP18	(32–5-2–3-6–3)	342,147	677,242	0*	P.O.	1342.98	5827	–	Out of Mem.	–	–	–	Out of Mem.	–	–

P.O.: the objective value of the solution is optimal (Proven Optimal).
Current: the best objective value reached at the end of an hour of calculation without the solver being able to recognize it as being optimal.
Out of Mem.: insufficient memory resources.
*: reach the lower bound limit for the « Shift » and « BNb » columns and higher than GM desired for the « Stab » column.

the Pareto front by each population's size for a set of instances, and noticed that, for each instance, the best front was found using a different size of population ($\in \{50, 100, 150\}$). We note that, to decide which configuration is better considering a multi-objective problem, we compared the fronts obtained two at a time, and selected the one that has the most solutions which dominate those of the other, if there are any. Otherwise, we have considered the number of solutions found in the Pareto front as a decision measure. Moreover, in our preliminary experiments, we tried to give different combinations of P_c (probability of crossing) and P_m (probability of mutation) on a set of instances, while preserving the other parameters $\{\{Pc = 0.5; Pm = 0.5; n = 50\}, \{Pc = 0.3; Pm = 0.6; n = 50\}, \{Pc = 0.9; Pm = 0.05; n = 50\}, \{Pc = 0.5; Pm = 0.5; n = 100\}, \{Pc = 0.3; Pm = 0.6; n = 100\}, \{Pc = 0.9; Pm = 0.05; n = 100\}, \{Pc = 0.5; Pm = 0.5; n = 150\}, \{Pc = 0.3; Pm = 0.6; n = 150\}, \{Pc = 0.9; Pm = 0.05; n = 150\}\}$. For each instance and each combination, 20 independent analyzes were performed. We can confirm that the effects of the mutation probability show that a small P_m is likely to improve the values of the solutions obtained when P_c is larger. Therefore, we define$P_c = 0.9$ and$P_m = 0.05$ as final parameters.

6.1. Instance generation of the 3D-CCSPP

We generate a set of small, medium, and large instances. First, we describe the characteristics of all barge convoys used in this paper to generate the final instances of the 3D-CCSPP. A series of 12 instances, corresponding to convoys of real sizes carrying capacities ranging from 32 to 1386 TEU, was created. Table 2 describes the characteristics of each convoy: the identifier "Id_Convoy", the maximum number of barges in the convoy "Nmax", the number of rows, the number of bays, the number of levels and the maximum capacity of each barge is indicated in columns "l", "L", "Lev" and "B_CapM", respectively. Finally, the maximum capacity of each convoy is given in the last column "C_CapM".

In a second phase, we report in Table 4 the parameters relating to the set of containers considered in this paper. For each instance "InstC" indicated in the first column, a number |N| of containers of mixed size must be transported (|N| = |N1|+|N2|, where |N1| indicates the number of 40′ containers and |N2| the number of 20′). |N|, |N1| and |N2| are displayed in the second, third and fifth columns respectively. While, columns 4 and 6 indicate, respectively, the weight "W" of containers, it is given in metric ton and is generated randomly within the weight ranges "L", "M" and "H" (as stated in Table 3). The total number of TEUs in each instance is reported in the last column "N_TEU". We assume that the fragility of each container is equal to 3 times its weight.

Finally, the 3D-CCSPP instances are created by combining Table 2 and Table 4 in Table 5. For each instance "InstCCSP", a barge convoy "Id_Convoy" must be docked in the initial port to stow the entire "InstC" containers and visit "P" ports during the tour.

6.2. Numerical results and discussion

6.2.1. Model resolution by CPLEX

To validate the proposed model, we solve a set of small instances using CPLEX solver by optimizing a single objective at a time using three parameters associated with the three objective functions (α for the first, β for the second and γ for the third) (see Table 6). A one-hour time limit has been imposed on the solver. The first two columns under the "Instance" heading report the different characteristics of each instance where « Id », *«$|N|$ », « P », « B », « l », « L », and « Lev »* represent respectively, the identifier of the instance, the number of containers to be stowed, the number of ports to visit, the maximum number of barges in the convoy, the width, length and height of barges. The third and fourth columns indicate, respectively, the number of variables and constraints in the model for each instance. The values under the heading "Obj Val" represent the best corresponding objective value obtained by CPLEX, the values under the heading "CPLEX Status" indicate the status of the solver at the end of the calculation, the values under the "Time

Table 7
Numerical results obtained by NSGA-II and CPLEX for small instances.

InstCCSP	Shift							Stab							BNb						
	Best sol	CPLEX			NSGA-II			Best sol	CPLEX			NSGA-II			Best sol	CPLEX			NSGA-II		
		Obj Val	CPU (s)	Dev	Obj Val	CPU (s)	Dev		Obj Val	CPU (s)	Dev	Obj Val	CPU (s)	Dev		Obj Val	CPU (s)	Dev	Obj Val	CPU (s)	Dev
CCSP1	0	**0**	0.22	0	**0**	7	0	6	**6**	0.27	0	**6**	7	0	1	**1**	0.25	0	**1**	7	0
CCSP2	0	**0**	1.02	0	**0**	7	0	10.78	**10.78**	3.16	0	10.43	7	0.35	1	**1**	3.64	0	**1**	7	0
CCSP3	0	**0**	2.17	0	**0**	5	0	7.68	**7.68**	2.28	0	**7.68**	5	0	2	**2**	4.72	0	2	5	0
CCSP4	0	**0**	3.61	0	**0**	7	0	13.89	**13.89**	14.23	0	12.99	7	0.9	1	**1**	13.3	0	2	7	1
CCSP5	0	**0**	15.11	0	**0**	11	0	15	**15**	24.02	0	12.14	11	2.86	1	**1**	25.73	0	2	11	1
CCSP6	0	**0**	23.31	0	**0**	12	0	7.71	**7.71**	2.45	0	7.65	12	0.06	2	**2**	2.42	0	2	12	0
CCSP7	0	**0**	16.39	0	**0**	12	0	12.84	**12.84**	16.94	0	10.12	12	2.72	1	**1**	28.03	0	1	12	0
CCSP8	0	**0**	47.44	0	**0**	12	0	13.62	**13.62**	38.55	0	11.54	12	2.08	1	**1**	65.74	0	2	12	1
CCSP9	0	**0**	210.66	0	**0**	12	0	13.36	**13.36**	43.52	0	12.03	12	1.33	1	**1**	69.94	0	2	12	1
CCSP10	0	**0**	93.83	0	**1**	12	1	11.34	8.72*	3600	2.62	**11.34**	12	0	1	**1**	108.77	0	2	12	1
CCSP11	0	**0**	108.89	0	**0**	13	0	10.64	9*	3600	1.64	**10.64**	13	0	1	**1**	28.33	0	2	13	1
CCSP12	0	**0**	67.45	0	**0**	13	0	11.15	10.01*	3600	1.14	**11.15**	13	0	1	**2***	3600	1	1	13	0
CCSP13	0	**0**	372.77	0	**0**	13	0	9.74	7.21*	3600	2.53	**9.74**	13	0	2	**2***	3600	0	2	13	0
CCSP14	0	**0**	166.97	0	**0**	13	0	14.76	14.76	145.83	0	10.86	13	3.9	2	**2***	3600	0	2	13	0
CCSP15	0	**0**	175.38	0	**0**	13	0	14.76	**14.76**	147.09	0	10.79	13	3.97	2	**2***	3600	0	2	13	0
CCSP16	0	**0**	175.61	0	**0**	14	0	14.97	**14.97**	161.41	0	10.91	14	4.06	2	**2***	3600	0	2	14	0
CCSP17	0	**0**	169.34	0	**0**	14	0	14.8	**14.8**	166.02	0	11.13	14	3.67	1	**1**	371.69	0	2	14	1
CCSP18	0	**0**	1342.98	0	**0**	14	0	13.18	-	-	-	**13.18**	14	0	2	-	-	-	2	14	0

* interrupted by the user.

Table 8
Information from individuals of the Pareto optimal front found by the NSGA-II for large instances.

Inst CCSP	*Pareto Front*														
	S1			*S2*			*S3*			*S4*			*S5*		
	Shift	*Stab*	*BNb*	*Shift*	*Stab*	*BNb*	*Shift*	*Stab*	*BNb*	*Shift*	*Stab*	*BNb*	*Shift*	*Stab*	*BNb*
CCSP19	15	8.87	2	12	8.44	2	13	8.54	2	10	7.78	2	5	7.36	2
CCSP20	9	8.4	2	10	8.74	2	11	8.85	2	5	7.71	2	7	8.05	2
CCSP21	12	8.26	2	14	8.7	2	7	7.06	2	5	7.04	2	9	7.66	2
CCSP22	14	10.05	3	15	10.24	3	13	9.95	3	10	9.56	3	8	9.33	3
CCSP23	19	8.52	2	21	8.64	2	17	8.22	2	15	8.06	2	11	8	2
CCSP24	16	8.51	2	21	8.61	2	13	8.27	2	15	8.37	2	24	8.78	3
CCSP25	31	14.76	4	39	14.94	4	45	15.11	4	38	14.92	4	30	14.74	4
CCSP26	47	13.74	4	56	13.78	4	36	13.57	4	55	13.75	4	37	13.72	4
CCSP27	40	12.17	4	52	12.26	4	39	12.16	4	36	12.13	4	44	12.23	4
CCSP28	7	8.23	2	14	9.08	2	12	8.93	2	11	8.81	2	8	8.61	2
CCSP29	16	10.9	2	22	11.38	2	25	11.49	2	11	9.61	2	15	10.5	2
CCSP30	18	10.05	2	15	10.02	2	22	10.16	2	13	9.51	2	8	9.38	2
CCSP31	32	10.25	2	23	9.73	2	26	9.93	2	20	9.51	2	19	9.46	2
CCSP32	39	12.04	2	34	11.68	2	31	11.5	2	28	11.19	2	22	10.99	2
CCSP33	39	9.76	2	30	9.59	2	28	9.5	2	22	8.82	2	24	9.29	2
CCSP34	72	15.51	4	74	15.55	4	65	15.38	4	57	15.28	4	53	15.07	4
CCSP35	51	13.88	4	52	13.91	4	48	13.82	4	66	13.95	4	57	13.94	4
CCSP36	45	15.91	4	42	15.9	4	53	16.25	4	40	15.7	4	46	16.14	4
CCSP37	14	10.55	2	21	11.14	2	19	10.79	2	25	11.29	2	18	10.6	2
CCSP38	16	5.48	1	18	5.62	1	15	5.44	1	22	5.72	1	23	6.29	1
CCSP39	24	6.24	1	19	5.78	1	21	6.19	1	20	5.85	1	17	5.31	1
CCSP40	34	11.29	2	31	11	2	22	10.89	2	28	10.97	2	18	10.16	2
CCSP41	44	12.3	2	40	12.14	2	30	11.72	2	36	11.95	2	21	11.16	2
CCSP42	22	9.51	2	18	9.07	2	16	8.98	2	19	9.43	2	17	3.99	2
CCSP43	48	11.16	3	45	11.08	3	51	11.26	3	54	11.32	3	40	10.99	3
CCSP44	83	11.27	3	61	10.88	3	74	11.2	3	72	10.9	3	51	10.55	3
CCSP45	54	10.93	3	59	11.33	3	55	11.31	3	65	11.35	3	48	10.6	3
CCSP46	139	11.34	3	116	11.19	3	131	10.95	3	107	10.16	3	98	10.12	3
CCSP47	131	11.13	3	130	11.08	3	122	10.77	3	126	10.96	3	101	9.11	3
CCSP48	95	10.55	3	102	10.61	3	109	10.97	3	105	10.7	3	107	10.85	3
CCSP49	191	13.92	4	163	13.4	4	170	13.62	4	165	13.52	4	193	13.95	4
CCSP50	157	12.34	4	160	12.53	4	187	12.64	4	177	12.6	4	168	12.58	4
CCSP51	153	18.91	4	176	19.01	4	158	18.95	4	147	18.72	4	149	18.83	4
CCSP52	216	18.85	5	180	18.59	5	178	18.56	5	189	18.75	5	202	18.79	5
CCSP53	177	19.81	5	181	19.94	5	168	19.74	5	190	20.08	5	164	19.72	5
CCSP54	213	16.8	5	236	17.05	5	221	16.96	5	216	16.89	5	212	16.74	5
CCSP55	119	9.85	2	120	9.97	2	91	9.49	2	105	9.84	2	92	9.83	2
CCSP56	122	10.09	2	111	9.98	2	112	10	2	106	9.62	2	108	9.88	2
CCSP57	155	11.14	2	130	10.6	2	145	10.92	2	139	10.87	2	137	10.79	2
CCSP58	156	14.56	3	142	14.22	3	148	14.52	3	166	14.6	3	168	14.81	3
CCSP59	225	14.5	3	205	13.97	3	218	14.01	3	223	14.06	3	222	14.05	3
CCSP60	182	14.91	3	177	14.73	3	185	14.94	3	169	14.72	3	187	14.98	3
CCSP61	266	14.72	3	260	14.51	3	265	14.56	3	252	14.4	3	259	14.41	3
CCSP62	274	14.59	3	236	14.35	3	231	14.26	3	244	14.39	3	257	14.48	3
CCSP63	226	14.49	3	215	14.4	3	217	14.44	3	224	14.47	3	221	14.46	3
CCSP64	126	11.25	2	108	11	2	116	11.22	2	127	11.27	2	111	11.13	2
CCSP65	122	12.18	2	119	12	2	117	11.99	2	100	10.22	2	92	9.16	2
CCSP66	139	11.29	2	125	11	2	126	11.23	2	116	10.6	2	121	10.7	2
CCSP67	154	16.56	3	148	16.5	3	141	16.28	3	144	16.3	3	120	12.66	3
CCSP68	163	15.16	3	160	15.07	3	161	15.13	3	168	15.2	3	159	15.02	3
CCSP69	182	13.9	3	191	14.26	3	187	14.22	3	183	14	3	204	14.28	3
CCSP70	221	16.68	3	225	16.72	3	222	16.7	3	230	16.81	3	218	16.54	3
CCSP71	285	9.01	3	260	8.9	3	293	9.03	3	257	8.85	3	246	8.84	3
CCSP72	244	14.5	3	239	14.44	3	242	14.45	3	230	14.36	3	228	14.21	3

Status" heading represent the total execution time required to reach the corresponding statutes (in seconds), and the "Iterations" heading represents the number of iterations needed to find the solution for the corresponding objective.

On the one hand, Table 6 confirms the validity of our proposed mathematical model for all the small instances studied. Indeed, the values obtained by minimizing each of the two objective functions "shift" and "BNb" have reached the theoretical lower bound which is defined, respectively, by zero *shifting* and the number of containers N_TEU divided by the B_CapM, and in the case of maximizing the objective function "Stab", all the obtained values are greater than the smallest metacentric distance desired in practice by experts (GM = 3.28084 feet) (Imai et al. (2002), Imai et al. (2006)). On the other hand, Table 6 also shows the complexity of the problem, with a number of variables ranging from 3811 for the smallest instance (CCSP1) to 342,147 for the largest (CCSP18), and a number of constraints ranging from 7190 to 677242, hence the use of an approximate method (NSGA-II *meta*-heuristic). We can also notice that for each instance, the number of iterations and the execution time differ from one objective to another. This can be justified by the fact that we have set the CPLEX settings to default. Indeed, the solver typically preprocesses problems by simplifying the constraints, reducing their size and eliminating redundancy. Its presolver tries to reduce the size of a problem by decreasing the number of rows and columns, i.e., making inferences about the nature of any optimal solution to the problem. Its aggregator, which can be applied once or more, tries to remove variables and rows by substitution. Therefore, according to each objective, the preprocessing done influences the total solution speed.

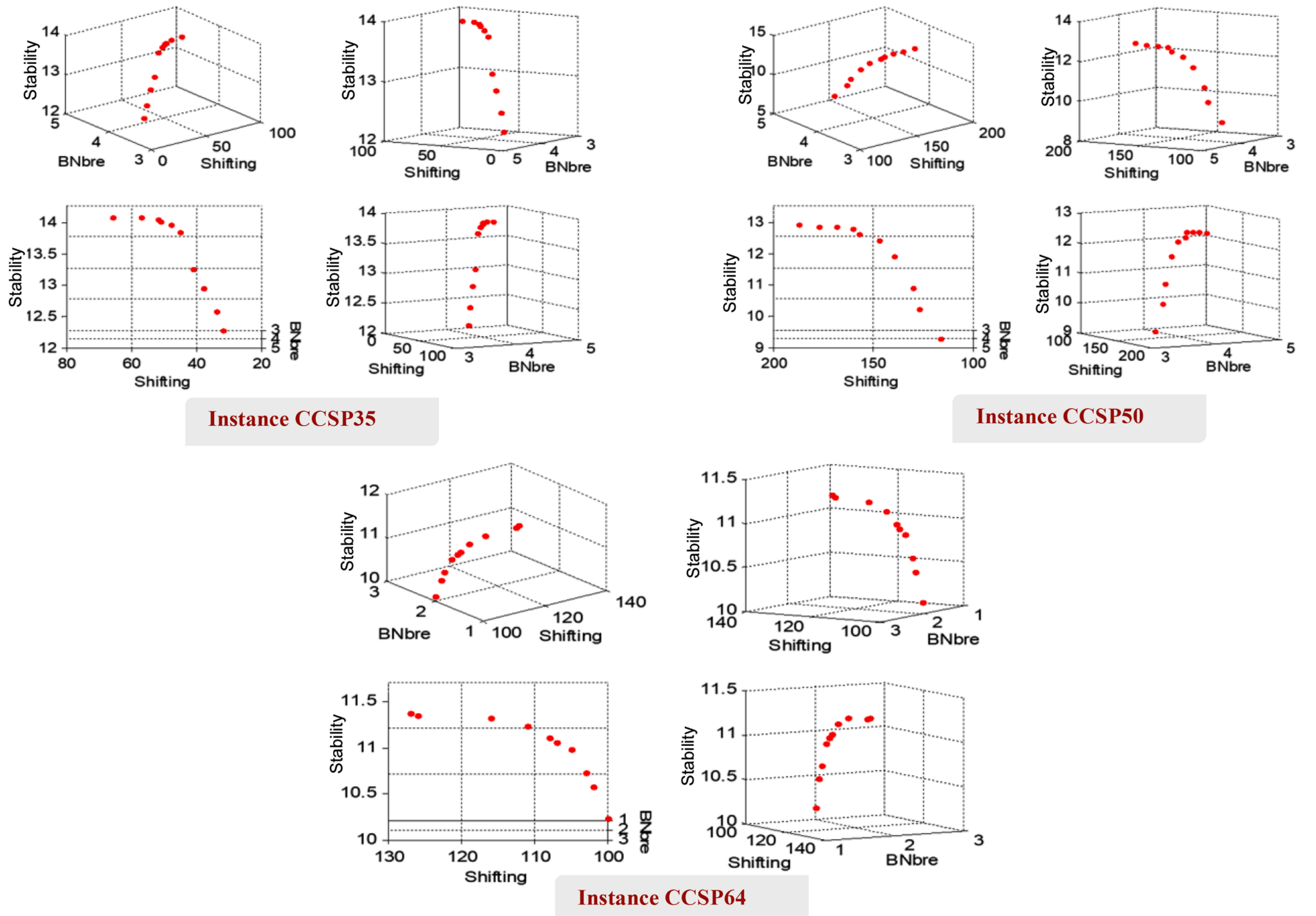

Fig. 13. 3D optimal Pareto front obtained by NSGA-II for some instances.

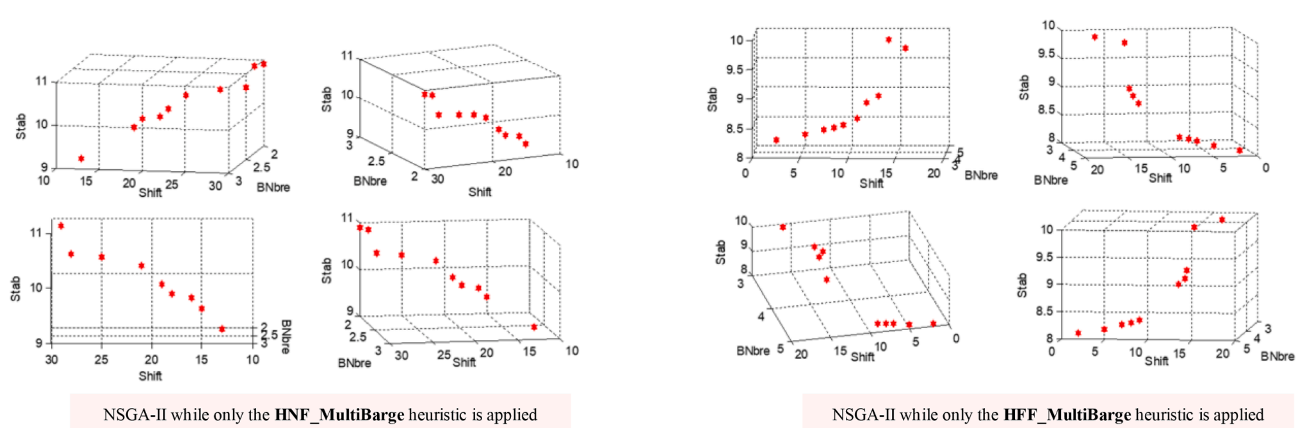

Fig. 14. 3D optimal Pareto Fronts of instance CCSP19 obtained by NSGA-II, using only one heuristic (HNF_MultiBarge or HFF_MultiBarge) for the generation of the initial population.

6.2.2. NSGA-II results

In order to assess and measure the NSGA-II results quality, we start by evaluating them from the single-objective comparison point of view (using CPLEX results). The first column of Table 7 contains the identifier of the instances studied, the columns "Shift", "Stab" and "BNb" represent, respectively, the number of *shiftings*, the GM stability (given in feet) and the number of real-used barges in the convoy. In each of these three columns we present the results obtained by the CPLEX solver and the NSGA-II. The values under the heading "Best sol" indicate the best solution found by these two resolution methods. For the solutions obtained by CPLEX, the value of the corresponding objective function indicated in the "Obj Val" column represents the optimal solution, or the best bound found within 3600 s. For the solutions found by the NSGA-II, the value of the corresponding objective function given in the "Obj Val" column indicates the average of the values of all the individuals of the Pareto front. The CPU corresponding to the execution of each instance is

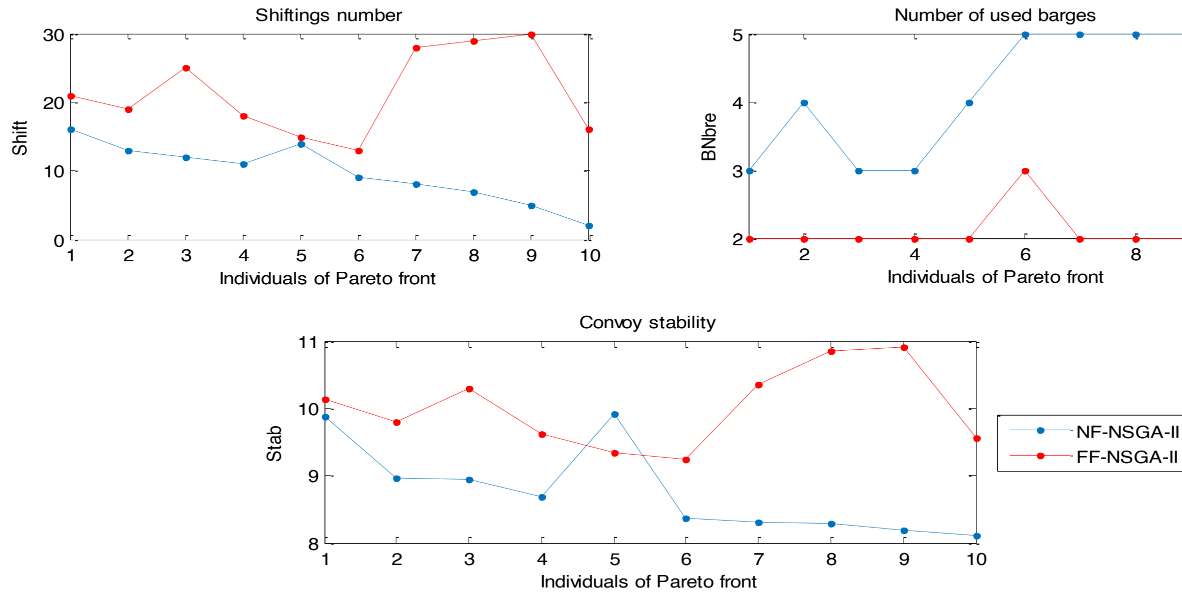

Fig. 15. Comparison between NF-NSGA-II and FF-NSGA-II.

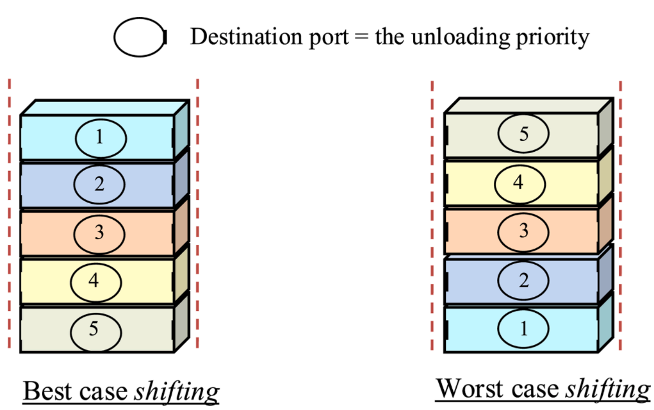

Fig. 16. The *shifting* in the worst- and best-case scenarios.

reported in the "CPU (s)" columns. The values under the "Dev" heading indicate the deviation from the best solution found by the NSGA-II or CPLEX, which is shown in the "Best sol" column (*Dev* = *Best sol* – *Obj Val*).

The results of Table 7 confirm the performance of NSGA-II adapted to 3D-CCSPP. They clearly show the effectiveness of this approach when solving small instances by offering optimal solutions for most instances.

The computation results obtained on the large instances by the NSGA-II are summarized in Table 8. In practice, it is always difficult to display all the optimal solutions of the Pareto front because of its large space of solutions. Consequently, we report only a subset of solutions of the Pareto front randomly chosen. The first column of the table gives the identifier of each instance, the columns under the heading "Pareto Front" draw the set of solutions chosen from the Pareto front obtained "S1", "S2", "S3", "S4" and "S5". Each of these five columns reports the values of the solutions obtained for the three objective functions studied "Shift", "Stab" and "BNb", indicating respectively the number of *shiftings*, the GM stability (in feet) and the number of barges used.

Among the proposed alternatives, represented by the set of compromise solutions of the Pareto front, the planner of the containers' stowage in the convoy is responsible for choosing the one that best meets his needs, based on his personal experience and preferences. Of course, other studies can be carried out to help make better decisions which depend on the *shiftings'* fees in the visiting ports, the availability of using ballasts to help ensure the convoy stability and the cost of using more barges in the convoy to transport all the containers.

6.2.2.1. Pareto front representation. As an example, Fig. 13 illustrates the set of solutions obtained for each of the instances CCSP35, CCSP50 and CCSP64 in a 3D Pareto front representation. Each sub-figure gives a different view of the front. Each red dot represents a solution, that is, a stowage plan meeting all the constraints established by the model. The x-axis represents the number of *shiftings* throughout the convoy journey, the y-axis represents the total convoy stability according to the GM metacentric distance, and the depth axis indicates the number of barges used in the convoy to transport the set of containers. We notice that the individuals of the Pareto front are well distributed all along the front; this is due to the use of the crowding distance in the NSGA- II. We can also find that the number of barges used in each convoy remains the same from one individual to another for the current instance, which generates a clear 2D view of the front.

6.2.3. Analysis and discussion

6.2.3.1. The influence of initial population nature. To show how the final NSGA-II population might be affected by the nature of the initial population that is generated, originally, using a random choice between the two proposed heuristics (HNF_MultiBarge and HFF_MultiBarge) during each execution, we give a more general comparison between the two heuristics, according to the three objective functions. We choose the CCSP19 instance as an example; we illustrate, in Fig. 14, the Pareto front obtained by the NSGA-II using only the HNF_MultiBarge heuristic when building the initial population (denoted by NF-NSGA-II), and the one obtained by NSGA-II when only the HFF_MultiBarge heuristic is used (denoted by FF-NSGA-II). This confirms that the type of heuristic used during the generation of the initial population strongly influences the values of the solutions found in the last front.

More precisely, we illustrate, in Fig. 15, the values of the solutions of each individual from the Pareto front according to the number of *shiftings*, the number of barges actually used and the total stability of the convoy. We find that the NF-NSGA-II gives the best results in terms of minimizing the number of *shiftings*, while the FF-NSGA-II gives the best results in terms of minimizing the number of barges used and maximizing the metacentric distance of the convoy. This is due to the nature

Table 9
Performance analysis of the results obtained by NSGA-II in terms of the number of *shiftings* compared to the one in worst case-scenario.

InstCCSP	*BNbAv*	*Shift moyen*		*Worst_sh*	*R*
		Av	*Av int>*		
CCSP19	2	11	11	180	93,89
CCSP20	2	8,4	9	180	95
CCSP26	4	46,2	47	360	86,94
CCSP42	2	18,4	19	480	96,04
CCSP43	3	47,6	48	720	93,33
CCSP44	3	68,2	69	720	90,42
CCSP45	3	56,2	57	720	92,08
CCSP53	5	176	176	1500	88,27
CCSP54	5	219,6	220	1500	85,33
CCSP55	2	105,4	106	1440	92,64
CCSP56	2	111,8	112	1440	92,22
CCSP57	2	141,2	142	1440	90,14
CCSP58	3	156	156	2160	92,78
CCSP59	3	218,6	219	2160	89,86
CCSP60	3	180	180	2160	91,67
CCSP61	3	260,4	261	2160	87,92
CCSP62	3	248,4	249	2160	88,47
CCSP63	3	220,6	221	2160	89,77
CCSP64	2	117,6	118	2310	94,89
CCSP65	2	110	110	2310	95,24
CCSP66	2	125,4	126	2310	94,55
CCSP67	3	141,4	142	3465	95,9
CCSP68	3	162,2	163	3465	95,3
CCSP69	3	189,4	190	3465	94,52
CCSP70	3	223,2	224	3465	93,54
CCSP71	3	268,2	269	3465	92,24
CCSP72	3	236,6	237	3465	93,16

Table 10
Performance analysis of the GM stability factor obtained by NSGA-II compared to 1 m and 1.5 m per barge.

InstCCSP	*GM_min*	*GM_max*	*Stab Av*	*Dev*	*Dev* min	*Dev* max
CCSP19	6.56168	9.84252	8.195	**0**	–	–
CCSP20	6.56168	9.84252	8.346	**0**	–	–
CCSP21	6.56168	9.84252	7.743	**0**	–	–
CCSP22	9.84252	14.76378	9.822	0.21	x	–
CCSP23	6.56168	9.84252	8.286	**0**	–	–
CCSP24	6.56168	9.84252	8.504	**0**	–	–
CCSP25	13.12336	19.68504	14.89	**0**	–	–
CCSP26	13.12336	19.68504	13.712	**0**	–	–
CCSP27	13.12336	19.68504	12.189	7.12	x	–
CCSP28	6.56168	9.84252	8.729	**0**	–	–
CCSP29	6.56168	9.84252	10.772	9.44	–	x
CCSP30	6.56168	9.84252	9.821	**0**	–	–
CCSP31	6.56168	9.84252	9.773	**0**	–	–
CCSP32	6.56168	9.84252	11.479	16.63	–	x
CCSP33	6.56168	9.84252	9.392	**0**	–	–
CCSP34	13.12336	19.68504	15.355	**0**	–	–
CCSP35	13.12336	19.68504	13.898	**0**	–	–
CCSP36	13.12336	19.68504	15.977	**0**	–	–
CCSP45	9.84252	14.76378	11.101	**0**	–	–
CCSP46	9.84252	14.76378	10.75	**0**	–	–
CCSP47	9.84252	14.76378	10.609	**0**	–	–
CCSP48	9.84252	14.76378	10.733	**0**	–	–
CCSP49	13.12336	19.68504	13.68	**0**	–	–
CCSP50	13.12336	19.68504	12.535	4.48	x	–
CCSP51	13.12336	19.68504	18.882	**0**	–	–
CCSP52	16.4042	24.6063	18.704	**0**	–	–
CCSP53	16.4042	24.6063	19.858	**0**	–	–
CCSP70	9.84252	14.76378	16.689	13.04	–	x
CCSP71	9.84252	14.76378	8.923	9.34	x	–
CCSP72	9.84252	14.76378	14.389	**0**	–	–

« **x** » in the **Dev min** column indicates that the corresponding deviation is from **GM_min.**
« **x** » in the **Dev max** column indicates that the corresponding deviation is from **GM_max.**
« – » no deviation from the desirable GM.

of the NF and FF heuristics of the bin-packing problem. Particularly, considering only one barge at a time, in the HNF_MultiBarge heuristic, can result in a significant loss of usable space, which is not the case for the HFF_MultiBarge heuristic. This explains the use of more barges in the NF-NSGA-II compared to the FF-NSGA-II. On the other hand, the use of more barges could reduce the number of *shiftings*. Thus, each method has its own advantage. For this reason and for diversity purpose, we have generated the initial population of NSGA-II based on a random choice, at each run, between the two proposed loading heuristics HNF_MultiBarge and HFF_MultiBarge.

6.2.3.2. Performance analysis of stowage plans obtained by NSGA-II. We believe that our research envisions a novel and real application of the CSPP while considering barge convoy systems. To the best of our knowledge, this is the first study to consider this new variant which has not yet been addressed by other researchers; and which made it difficult to compare the results obtained with those in the literature. Therefore, we propose performance measures to evaluate and validate the quality of the NSGA-II stowage plans against the three objective functions studied, we propose three performance measures. The first relates to the number of *shiftings* in the worst case-scenario, the second relates to the desirable metacentric distance in real-world application and the third relates to the number of barges used in the best case-scenario.

6.2.3.2.1. Performance analysis according to the number of shiftings. We define the "worst case-scenario for a stack" as the worst number of *shiftings* needed to empty a single stack of a single barge as shown in Fig. 16. Therefore, we can calculate the total number of *shiftings* of the whole convoy of barges in the worst-case scenario by the following formula:

$$Worst_{Sh} = \sum_{t=1}^{B'} \sum_{i=1}^{l} \sum_{j=1}^{L} \sum_{h=1}^{Lev-1} (Lev - h) = B' \times l \times L \times \sum_{h=1}^{Lev-1} (Lev - h)$$

In addition, for an instance of the 3D-CCSPP with a convoy of B' real-used barges, each with dimension $l \times L \times Lev$, we introduce the *shifting* rate (Ding and Chou, 2015), which is a relative performance measure for stowage plans, according to the following equation, where *Sh* indicates the total number of *shiftings* obtained during the entire convoy tour:

$$R = \frac{Worst_{Sh} - Sh}{Worst_{Sh}} \times 100$$

We report, **in** Table 9, the *shiftings* rate R compared to the worst case-scenario for the set of instances resolved by the NSGA-II in Table 8. The identifier of each instance is given in column 1. The average value of the number of barges used and the number of *shiftings* performed for the five individuals of the Pareto front are indicated, respectively, in columns 2 and 3. We recall that a barge is considered 'totally' used even if at least one container is loaded on board. Therefore, the column "BNbAv" represents the smallest integer greater than the average number of barges used. The values under the heading "Av" indicate, therefore, the initial average value and the values under the heading "Av int>" represent the integer part of this average + 1. The number of *shiftings* in the worst case-scenario, calculated for each instance, is given in the column "Worst_sh". The rate of *shifting* is reported in the last column "R".

Based on the analysis of the results presented in Table 9, we can confirm the high quality of the NSGA-II stowage plans in relation to the values of the objective function "*shifting*". Indeed, for most of the studied instances, the difference between the number of *shiftings* found and that in the worst cases-scenario is greater than 90%. In other words, the solutions obtained are within 10% of the best-case scenario. Actually, not being within 1% or less of the best scenario could be justified by the implication of the multi-objective aspect which seeks the compromise between the three objective functions and not only optimizes the *shiftings* movements.

6.2.3.2.2. Performance analysis according to the stability factor GM. In this section, we evaluate the quality of the solutions obtained by the

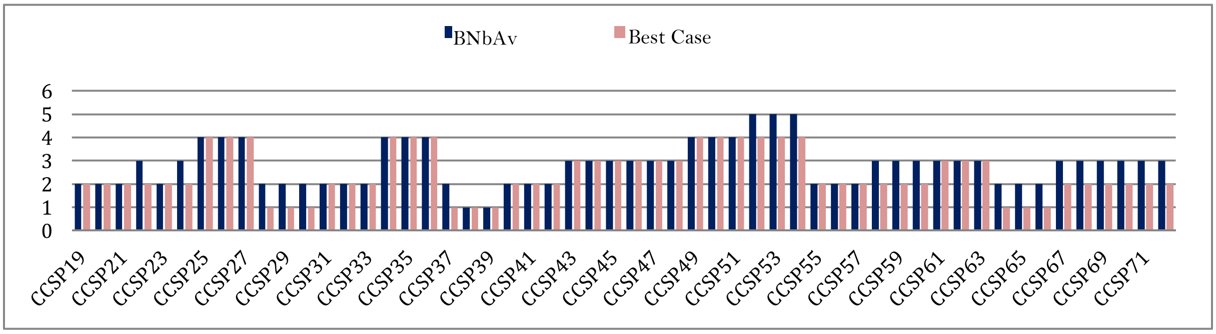

Fig. 17. Comparison between the number of barges real-used in the convoy obtained by NSGA-II and that in the best case-scenario.

NSGA-II in relation to the desirable metacentric distance in practice. In a general case, the rules of the maritime transport inspection provide that the value of GM must be greater than 1 m if the containers are stowed, and preferably, that it must not exceed 1.5 m (Imai et al., 2002; Imai et al., 2006). Therefore, to evaluate our results, we compare the mean NSGA-II GM values for the five Pareto front individuals reported in Table 8 with the desirable range $[GM\_min, GM\_max] = [1, 1.5]$ meters of a given barge. In addition, the nature of the 3D-CCSP problem, which is based on the use of a set of barges and not one, imposes the redefinition of the interval $[GM\_min, GM\_max]$ for each instance according to the number of barges used in the convoy. Table 10 summarizes the results obtained after the comparison of desirable GM with that found by our algorithm. The first column indicates the identifier of the instance. The range $[GM\_min, GM\_max]$ corresponding to each instance is given in columns 2 and 3, $[GM\_min, GM\_max] = [3.28084*BNb,\ 4.92126*BNb]$. The fourth column gives the average stability obtained for the five individuals of the Pareto front. GM deviation from the desirable range is given in the « Dev » column according to the following cases:

- If $Stab\_Av < GM\_min$, then $Dev = \frac{GM\_min - Stab\_Av}{GM\_min} \times 100$
- If $Stab\_Av > GM\_max$, then $Dev = \frac{Stab\_Av - GM\_max}{GM\_max} \times 100$
- If $Stab\_Av \in [GM\_min, GM\_max]$, then $Dev = 0$

Although the resulting GM value is small for some instances (indicated by the “x” character in the “Dev min” column), most solutions seem reasonable, with GM distance limited to the corresponding range.

The present work aims to generate a stowage plan for a set of containers only in the first port of the journey while focusing on the new “multi-barge” aspect. Hence, the metacentric distance is calculated only in the first port. Indeed, *shiftings* movements may affect the barge stability because of the unloading and reloading operations done. In this case, we consider using ballast tanks to avoid any generated instability of the convoy.

6.2.3.2.3. Performance analysis according to the number of real-used barges. To evaluate the quality of the solutions obtained by the NSGA-II compared to the number of barges actually used in the convoy, we introduce the number of real-used barges in the convoy in the best case-scenario “Best Case”, as follows:$BestCase = N\_TEU/B\_CapM$.

In Fig. 17, we illustrate the comparison results of the solutions found in terms of number of real-used barges by the NSGA-II with the best case-scenario, in which, the average value of the number of barges used of the five individuals of the Pareto front of Table 8 is indicated by “BNbAv”. We can confirm that the results obtained by the NSGA-II are of very good quality. Indeed, we note that the NSGA-II returns solutions with, at most, a single barge in the case where the deviation from the “Best Case” is not zero. Moreover, the observed deviation may be justified by the fact that the best case-scenario does not consider the size constraints of the containers and the dynamic weight limit of the stacks, which are, on the other hand, considered in the NSGA-II. This analysis opens up new avenues for further researches, such as, studying the influence of the barge filling rate on the total cost of containers transport.

6.2.3.3. Sensitivity analysis. The purpose of sensitivity analysis is to identify important parameters that dominate model behaviors (Gan et al., 2014) and quantify the effect of uncertainties in input parameters on the variability of a model’s output factors (Pianosi et al., 2016). An input parameter is any element that can be changed before model execution and an output factor or response in our case is given by the objective functions, which are optimized.

In order to quantify this sensitivity, new techniques based on statistical theories have emerged such as regression, tests, statistical learning, Monte Carlo, Sobol’s index, etc. An extensive review of various sensitivity analysis methods is given by Iooss & Lemaître (2015). In our study, we opt for the linear regression method, which consists of using several variables called explanatory variables to explain the dependent variable. The linear regression method uses standardized regression coefficients as direct measures of sensitivity. This method is appropriate when the hypothesis of linearity is confirmed. For this, the calculation of the coefficient of determination R^2 is carried out (Glantz & Slinker, 2001); hence, the linearity can be confirmed if R^2 is large. In addition, the choice of this method is based on its advantages that are concretized

Table 11
Summary statistics of inputs and outputs data.

Variable	*N*	*Mean*	*Std. Dev.*	*Min*	*Pctl. 25*	*Pctl. 75*	*Max*
Shift	72	75.742	82.831	0	6.55	141.25	268.2
Stab	72	11.696	2.988	5.71	9.825	13.955	19.858
BNb	72	2.558	0.961	1	2	3	5
Nmax	72	3.875	1.463	2	2.75	5	6
l	72	4.597	1.544	2	3.75	5.25	7
L	72	7.778	2.864	3	6	10.25	12
Lev	72	4.042	1.027	2	3	4.25	6
|N1|	72	113.458	90.58	2	30.5	172.75	261
W ™ N1	72	1982.75	1588.163	37.5	505	3023.125	4567.5
|N2|	72	87.153	84.493	3	23.75	152.25	239
W ™ N2	72	1403.181	1344.559	36	400	2436	3824
P	72	20.556	10.633	5	15	27.5	35

Table 12
The Pearson correlation matrix.

	Shift	*Stab*	*BNb*	*Nmax*	*l*	*L*	*Lev*	*|N1|*	*W ™ N1*	*|N2|*	*W ™ N2*	*P*
Shift	1											
Stab	0.599***	1										
BNb	0.584***	0.774***	1									
Nmax	−0.012	−0.010	0.405***	1								
l	0.737***	0.289*	0.372**	0.339**	1							
L	0.871***	0.405***	0.417***	0.161	0.881***	1						
Lev	0.722***	0.382***	0.216	−0.137	0.748***	0.789***	1					
|N1|	0.940***	0.671***	0.745***	0.206	0.789***	0.869***	0.671***	1				
W ™ N1	0.939***	0.668***	0.746***	0.207	0.788***	0.869***	0.672***	1.000***	1			
|N2|	0.981***	0.623***	0.600***	0.010	0.745***	0.879***	0.713***	0.932***	0.932***	1		
W ™ N2	0.980***	0.625***	0.598***	0.001	0.742***	0.875***	0.710***	0.930***	0.930***	1.000***	1	
P	0.401***	0.128	0.325**	0.543***	0.614***	0.552***	0.365**	0.494***	0.494***	0.393***	0.387***	1

Computed correlation used pearson-method with listwise-deletio.

in its simplicity and its low computational cost.

Table 11 gives a statistical summary of the inputs and outputs variables given by NSGA-II. The first three lines represent the outputs variables "Shift", "Stab" and "BNb", which indicate the mean of the reported solutions in the Pareto Front according to each objective function (respectively, the number of *shifting* movements, the stability of the convoy and the number of barges used in the convoy). The rest of the lines represent the inputs variables. We note that "W ™ N1" and "W ™ N2" are given by the mean weight for all 40' and 20' containers, respectively. Moreover, to avoid collinearity between the variables, some inputs which are the subject of a crossing of other inputs are not considered in the regression model, i.e., "B_CapM" and "C_CapM".

To further verify the linearity between the variables, a correlation analysis was performed, and the results found are presented in Table 12. It is quite evident that all the outputs variables, namely, the number of *shifting* movements, the stability of the convoy and the number of barges used in the convoy, are positively and significantly correlated with the other inputs variables. For example, the most correlated ones with "Shift" are respectively, "|N2|" (0.981***), "W ™ N2" (0.980***), "|N1|" (0.940***), "W ™ N1" (0.939***) and "L" (0.871***), all significant at 0.01 level.

In Table 13, the sensitivity analysis is performed at three stages. The first stage gives the regression model that explains the first output "Shift". In this model, the regression coefficients that measure sensitivity significantly at 99% confidence level is "Nmax" (−5.834***). This coefficient is negatively significant, which means that a decrease in the number of barges available leads to an increase in the number of shifts. The second stage gives the regression model that explains the second output "Stab" which represents the stability of the convoy according to the metacentric distance. In this model, the regression coefficients that measure sensitivity significantly at 99% confidence level are, "l" (−1.390***), and "Lev" (1.339***). Another significant coefficient, at 95% confidence level, is "L" (−0.516**). Whereas the coefficients "Nmax" (0.471*), "|N1|" (0.272*) and "W ™ N1" (−0.013*) measure sensitivity significantly at 90% confidence level which is less significant than the previous coefficients. This is justified by the influence of 40' containers constraints and the barges dimensions on the calculation of the metacentric distance explained in sections 4. The third stage gives the regression model that explains the last output "BNb". In this model, the regression coefficients measure sensitivity significantly within 99% confidence level, namely, "Nmax" (0.313***), "l" (−0.349***) and "L" (−0.181***), on the other hand, the coefficients "|N2|" (−0.058*), and "W ™ N2" (0.004*) measure sensitivity significantly at 90% confidence level. This is justified by the influence of the number of available barges to form the convoy as well as their dimensions and the 20' containers constraints related to their stowage restrictions explained in sections 4 and 5. In addition, the coefficient of determination R^2 varies from 0.973, 0.699 and 0.878 according to each model with values considered large.

As shown in Table 13, the p-value associated with the F-statistic is < 0.01, which means that independent variables are related to the output

Table 13
The regression model analysis results.

	Dependent variable		
	Shift	*Stab*	*BNb*
Nmax	−5.834***	0.471*	0.313***
	(2.151)	(0.259)	(0.053)
l	−0.559	−1.390***	−0.349***
	(3.133)	(0.378)	(0.077)
L	0.242	−0.516**	−0.181***
	(1.959)	(0.236)	(0.048)
Lev	0.400	1.339***	0.141
	(3.616)	(0.436)	(0.089)
|N1|	0.272	0.272*	0.012
	(1.137)	(0.137)	(0.028)
W ™ N1	0.0003	−0.013*	0.0002
	(0.065)	(0.008)	(0.002)
|N2|	1.451	−0.166	−0.058*
	(1.209)	(0.146)	(0.030)
W ™ N2	−0.049	0.011	0.004*
	(0.075)	(0.009)	(0.002)
P	0.267	−0.023	−0.005
	(0.238)	(0.029)	(0.006)
Constant	2.969	10.055***	2.031***
	(13.291)	(1.603)	(0.328)
Observations	72	72	72
R^2	0.973	0.699	0.878
Adjusted R^2	0.969	0.655	0.860
Residual Std. Error (df = 63)	14.552	1.756	0.360
F Statistic (df = 8; 63)	248.720***	15.972***	49.480***

*p < 0.1; **p < 0.05; ***p < 0.01.

variables. In other words, the model has a predictive capability. Therefore, the results generated by the proposed model and solution approach and the additional studies carried out in sensitivity analysis provide the containers stowage planner with a better knowledge of the problem to be solved, and thus make it easy to choose a better final decision.

7. Conclusion

This paper introduces the 3D – barge Convoy Container Stowage Planning Problem (3D-CCSPP), a new and effective variant of the Container Stowage Planning Problems (CSPP) which is widely encountered in real-life situations of freight transportation due to massification purpose. Indeed, in addition to considering container sizes, dynamic limit of the stacks' weights, number of *shifting* movements and the total stability, the planner is not limited to using a single barge to transport containers, but, he envisages the possibility of using a set of barges constituting a convoy. Thus, several real requirements related to both the containers and the convoy must be considered.

This new variant is treated based on its relation to the Bin-Packing problem. Firstly, a multi-objective integer mathematical model for the 3D-CCSPP is proposed considering three realistic objectives: the number of *shiftings* throughout the journey, the stability of the convoy according to the metacentric distance and the number of barges used in the convoy. To show the efficiency of the proposed model, series of experimental tests are performed using the CPLEX solver. Indeed, the 3D-CCSPP is more complex, but also more realistic. As a result, a new adaptation of the NSGA-II is proposed while introducing a set of loading rules based on the well-known bin-packing problem heuristics. Furthermore, the paper presents a set of computational experiments to evaluate the solution method and analyze its properties. That is, a comparison with the optimal or best-known solutions given by CPLEX solver for small instances is done. For medium and large instances, for which CPLEX cannot find an optimal solution, a set of performance measures are generated to analyze the quality of the solutions obtained by the proposed method. These latter show the effectiveness of the used methods. Indeed, the optimized stowage plans generated meet all the defined requirements in terms of reducing the number of used barges, the unnecessary containers' movements throughout the route, and ensuring the stability of the barge convoy. Additional studies are carried out in sensitivity analysis to provide the planner with a better knowledge of the problem. The results are finally displayed to him as a Pareto front, it is then his turn to select the solution that best fits his needs based on his experience and preferences.

To the best of our knowledge there is no published paper considering the presented problem. It is a new and efficient aspect given the state of the art of CSPP. In fact, researches have mainly focused on the resolution of CSPP in maritime container ships and only a few relevant studies have focused on inland navigation vessels, without, however, considering convoy barge systems (Li et al., 2017; Li et al., 2018; Hu & Cai, 2017; Fazi, 2019). With the proposed model and method, it is possible to help inland shipping planners to generate optimized stowage plans for a set of containers in a barge convoy system in accordance with the *shiftings* number, the stability of the convoy and the number of used barges.

The present work opens new avenues for further research nails. Indeed, more aspects of the CSPP problem could be treated. It could be interesting to investigate the influence of other factors (such as the trim and the heel) on the convoy stability. Moreover, since new containers could be waiting to be transported in each port throughout the route, it would be interesting to consider all movements from loading operations in the initial port to the arrival of containers at their final destinations including new containers loaded from intermediate ports. And this also implies taking into account the three objectives studied, including stability, in each port.

CRediT authorship contribution statement

Amina El Yaagoubi: Conceptualization, Methodology, Software, Writing – original draft, Visualization, Investigation. **Mohamed Charhbili:** Methodology, Validation, Formal analysis, Writing – review & editing. **Jaouad Boukachour:** Writing – review & editing, Supervision, Validation, Investigation. **Ahmed El Hilali Alaoui:** Writing – review & editing, Supervision, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the anonymous reviewers for their insightful comments and suggestions, which enabled us to improve and clarify our manuscript.

References

Ambrosino, D., Sciomachen, A., Tanfani, E., 2004. Stowing a containership: the master bay plan problem. Transportation Research Part A: Policy and Practice 38 (2), 81–99.
Avriel, M., Penn, M., 1993. Exact and approximate solutions of the container ship stowage problem. Computers & industrial engineering 25 (1–4), 271–274.
Avriel, M., Penn, M., Shpirer, N., 2000. Container ship stowage problem: complexity and connection to the coloring of circle graphs. Discrete Applied Mathematics 103 (1–3), 271–279.
Avriel, M., Penn, M., Shpirer, N., Witteboon, S., 1998. Stowage planning for container ships to reduce the number of shifts. Annals of Operations Research 76, 55–71.
Azevedo, A.T., de Salles Neto, L.L., Chaves, A.A., Moretti, A.C., 2018. Solving the 3D stowage planning problem integrated with the quay crane scheduling problem by representation by rules and genetic algorithm. Applied Soft Computing 65, 495–516.
Bansal, N., Liu, Z., Sankar, A., 2009. Bin-packing with fragile objects and frequency allocation in cellular networks. Wireless Networks 15 (6), 821–830.
Barrass, B., 2000. Ship stability: notes and examples. Elsevier.
Blickle, T., 2000. Tournament selection. Evolutionary computation 1, 181–186.
Bilican, M.S., Evren, R., Karatas, M., 2020. A Mathematical Model and Two-Stage Heuristic for the Container Stowage Planning Problem with Stability Parameters. IEEE Access 8, 113392–113413.
Deb, K., 2000. Introduction to selection. Evolutionary computation 1, 166–171.
Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.A.M.T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation 6 (2), 182–197.
Delgado, A., Jensen, R.M., Schulte, C., 2009. In: September). Generating optimal stowage plans for container vessel bays. Springer, Berlin, Heidelberg, pp. 6–20.
Delgado, A., Jensen, R.M., Janstrup, K., Rose, T.H., Andersen, K.H., 2012. A constraint programming model for fast optimal stowage of container vessel bays. European Journal of Operational Research 220 (1), 251–261.
Ding, D., Chou, M.C., 2015. Stowage planning for container ships: a heuristic algorithm to reduce the number of shifts. European Journal of Operational Research 246 (1), 242–249.
Douma, A., Schutten, M., Schuur, P., 2009. Waiting profiles: An efficient protocol for enabling distributed planning of container barge rotations along terminals in the port of Rotterdam. Transportation Research Part C: Emerging Technologies 17 (2), 133–148.
Dyckhoff, H., 1990. A typology of cutting and packing problems. European Journal of Operational Research 44 (2), 145–159.
El Yaagoubi, A., 2019. *Contribution à l'optimisation du chargement et du déchargement des conteneurs dans le cas des transports routier et fluvial* (Doctoral dissertation. Le Havre Normandie University & Sidi Mohamed Ben Abdellah University).
El Yaagoubi, A., Alaoui, A.E.H., Boukachour, J., 2018, April. Multi-objective river-sea-going container barge stowage planning problem with container fragility and barge stability factors. In: 2018 4th International Conference on Logistics Operations Management (GOL). IEEE, pp. 1–6.
El Yaagoubi, A., Alaoui, A.E.H., Boukachour, J., 2020, October. A Heuristic Approach For Solving Container-on-Barge Stowage Planning Problem Based On Bin-Packing First-Fit Algorithm. In: 2020 5th International Conference on Logistics Operations Management (GOL). IEEE, pp. 1–6.
Expósito-izquierdo, C., Lalla-ruiz, E., De Armas, J., Melián-batista, B., Moreno-Vega, J. M., 2017. Maritime container terminal problems. In: *Handbook of Heuristics*. Springer, pp. 1–27.
Fazi, S., 2019. A decision-support framework for the stowage of maritime containers in inland shipping. Transportation Research Part E: Logistics and Transportation Review 131, 1–23.
Gan, Yanjun, Duan, Qingyun, Gong, Wei, Tong, Charles, Sun, Yunwei, Chu, Wei, Ye, Aizhong, Miao, Chiyuan, Di, Zhenhua, 2014. A comprehensive evaluation of various sensitivity analysis methods: A case study with a hydrological model. Environmental modelling & software 51, 269–285.
Glantz, S.A., Slinker, B.K., 2001. Primer of applied regression & analysis of variance, ed. McGraw-Hill Inc, New York.
Hu, M., Cai, W., 2017. In: April). Multi-objective optimization based on improved genetic algorithm for containership stowage on full route. IEEE, pp. 224–228.
Imai, A., Nishimura, E., Papadimitriou, S., Sasaki, K., 2002. The containership loading problem. International Journal of Maritime Economics 4 (2), 126–148.
Imai, A., Sasaki, K., Nishimura, E., Papadimitriou, S., 2006. Multi-objective simultaneous stowage and load planning for a container ship with container rehandle in yard stacks. European Journal of Operational Research 171 (2), 373–389.
Iooss, B., Lemaître, P., 2015. A review on global sensitivity analysis methods. In *Uncertainty management in simulation-optimization of complex systems*. Springer, Boston, MA, pp. 101–122.
Li, J., Zhang, Y., Ji, S., 2017, August. A two-phase approach for inland container ship stowage on full route on Yangtze River. In: 2017 4th International Conference on Transportation Information and Safety (ICTIS). IEEE, pp. 799–805.
Li, J., Zhang, Y., Ma, J., Ji, S., 2018. Multi-port stowage planning for inland container liner shipping considering weight uncertainties. IEEE Access 6, 66468–66480.
Martello, S., Pisinger, D., Vigo, D., 2000. The three-dimensional bin-packing problem. Operations Research 48 (2), 256–267.
Mihyeon Jeon, C., Amekudzi, A., 2005. Addressing sustainability in transportation systems: definitions, indicators, and metrics. Journal of infrastructure systems 11 (1), 31–50.
Pacino, D., Delgado, A., Jensen, R.M., Bebbington, T., 2011. In: September). Fast generation of near-optimal plans for eco-efficient stowage of large container vessels. Springer, Berlin, Heidelberg, pp. 286–301.

Paquay, C., Limbourg, S., Schyns, M., 2018. A tailored two-phase constructive heuristic for the three-dimensional Multiple Bin Size Bin Packing Problem with transportation constraints. European Journal of Operational Research 267 (1), 52–64.

Parreño, F., Pacino, D., Alvarez-Valdes, R., 2016. A GRASP algorithm for the container stowage slot planning problem. Transportation Research Part E: Logistics and Transportation Review 94, 141–157.

Parreño-Torres, Consuelo, Çalık, Hatice, Alvarez-Valdes, Ramon, Ruiz, Rubén, 2021. Solving the generalized multi-port container stowage planning problem by a matheuristic algorithm. Computers & Operations Research 133, 105383.

Pianosi, F., Beven, K., Freer, J., Hall, J.W., Rougier, J., Stephenson, D.B., Wagener, T., 2016. Sensitivity analysis of environmental models: A systematic review with practical workflow. Environmental Modelling & Software 79, 214–232.

Sciomachen, A., Tanfani, E., 2003. The master bay plan problem: a solution method based on its connection to the three-dimensional bin-packing problem. IMA Journal of Management Mathematics 14 (3), 251–269.

Sciomachen, A., Tanfani, E., 2007. A 3D-BPP approach for optimising stowage plans and terminal productivity. European Journal of Operational Research 183 (3), 1433–1446.

Tan, K.C., Lee, T.H., Khor, E.F., 2002. Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons. Artificial intelligence review 17 (4), 251–290.

Tanaka, S., Tierney, K., 2018. Solving real-world sized container pre-marshalling problems with an iterative deepening branch-and-bound algorithm. European Journal of Operational Research 264 (1), 165–180.

Tavares de Azevedo, A., Cassilda Maria, R., De Sena, G.J., Chaves, A.A., Neto, L.L.S., Moretti, A.C., 2014. Solving the 3D container ship loading planning problem by representation by rules and meta-heuristics. International Journal of Data Analysis Techniques and Strategies 6 (3), 228–260.

van Hassel, E., 2011. *Developing a small barge convoy system to reactivate the use of the small inland waterway network*. Universiteit Antwerpen, Belgium.

Wang, N., Zhang, Z., Lim, A., 2014, June. The stowage stack minimization problem with zero rehandle constraint. In: *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, Cham, pp. 456–465.

Wei-Ying, Z., Yan, L., Zhuo-Shang, J., 2005. Model and algorithm for container ship stowage planning based on bin-packing problem. Journal of Marine Science and Application 4 (3), 30–36.

Wilson, I.D., Roach, P.A., 1999. Principles of combinatorial optimization applied to container-ship stowage planning. Journal of Heuristics 5 (4), 403–418.

Wilson, I.D., Roach, P.A., 2000. Container stowage planning: a methodology for generating computerised solutions. Journal of the Operational Research Society 51 (11), 1248–1255.

Zhang, Z., Lee, C.Y., 2016. Multiobjective approaches for the ship stowage planning problem considering ship stability and container rehandles. IEEE Transactions on Systems, Man, and Cybernetics: Systems 46 (10), 1374–1389.