Definition of ship outfitting scheduling as a resource availability cost problem and development of a heuristic solution technique

Rose, CD; Coenen, JMG; Hopman, JJ

DOI
10.5957/JSPD.32.3.150003

Publication date
2016

Document Version
Final published version

Published in
Journal of Ship Production and Design

Citation (APA)
Rose, CD., Coenen, JMG., & Hopman, JJ. (2016). Definition of ship outfitting scheduling as a resource availability cost problem and development of a heuristic solution technique. *Journal of Ship Production and Design*, *32*(3), 154 - 165. https://doi.org/10.5957/JSPD.32.3.150003

Journal of Ship Production and Design, Vol. 32, No. 3, August 2016, pp. 154–165
http://dx.doi.org/10.5957/JSPD.32.3.150003

Definition of Ship Outfitting Scheduling as a Resource Availability Cost Problem and Development of a Heuristic Solution Technique

Christopher Rose, Jenny Coenen, and **Hans Hopman**

Delft University of Technology, Maritime and Transport Technology, Delft, The Netherlands

The outfitting process of shipyards building complex ship types, such as offshore, passenger, and military vessels, is becoming more critical to efficient ship production as the number of components installed on these vessels continues to increase. Outfitting of such vessels is generally characterized by disorganization and rework due to a lack of coordination between the shipyard and subcontractors as well as insufficiently detailed planning. This paper presents a mathematical model for the outfitting planning process of a shipyard building complex vessels. A qualitative description is included for the constraints and objectives underlying the developed mathematical model for the Ship Outfitting Scheduling Problem (SOSP). A heuristic solution technique is also developed for solving the SOSP, and a test case of six midship sections from a recently constructed pipelaying vessel is presented to show the feasibility of both the mathematical model and heuristic. This test case shows that it is possible to find a high-quality planning for the SOSP with a reasonable computational effort. Furthermore, it was found that the greatest priority should be given to components that have the earliest deadlines or dependents with such deadlines. Components should also be scheduled in such a way to minimize the required movements of outfitting personnel between the different work sites of a shipyard.

Keywords: shipbuilding, scheduling, outfitting, optimization

1. Introduction

OUTFITTING IS the installing of components in a vessel, such as piping; heating, ventilating, and air conditioning (HVAC) ducts; cable trays; and equipment. In recent years, European shipyards have mainly focused on the steel building portion of the shipbuilding process and have outsourced the outfitting work to a set of specialized subcontractors. Usually, at least one different subcontractor is hired by the shipyard per outfitting discipline. The main role of the yard has been to provide the subcontractors with a general time frame during which the outfitting work must be completed and to act as the coordinator interfacing between the various subcontractors. Because each subcontractor performs their own work independently with only limited communication, the outfitting process is often characterized by delays, rework and suboptimization (Wei 2012). These problems are especially pronounced in the construction of complex vessels, such as offshore vessels, dredgers, and cruise vessels. Complex vessel types differ from simple, steel intensive cargo ships since they are usually engineered to order and produced as one-off designs or in very short series. Moreover, these problems will only become worse in the coming years as increasing mission and safety requirements dictate more complex and outfitting intense vessels (CESA 2011). For example, future vessels will need to comply with more stringent redundancy requirements that can potentially double the required number of components in certain critical systems. A more detailed explanation of the problems that result insufficiently detailed outfitting planning as well as several site examples of outfitting-related rework from a European shipyard can be found in Wei (2012).

One possible method for improving the outfitting process of shipyards building complex vessels is to generate a detailed outfitting planning that takes into account the needs of all involved parties. Such a planning could be implemented by the shipyard to provide more structured guidance to the subcontractors as to when and where they should perform their work. This type of approach

---

Manuscript received by JSPD Committee June 15, 2015; accepted September 26, 2015.

could potentially reduce the amount of rework that results from miscommunication as well as the risk of failing to complete the required outfitting work on time. To generate a detailed outfitting planning, scheduling techniques developed by the operations research community can be applied to a mathematical model of the outfitting process. Unfortunately, a sufficiently detailed mathematical model of the outfitting process has not yet been published. The first purpose of this paper is to develop and present such a model by examining the constraints and objectives that govern this process.

Fundamentally, the ship outfitting scheduling problem (SOSP) is a variation of the resource availability cost problem (RACP), also referred to as the resource investment problem (RIP) in some literature. The RACP seeks to schedule a set of tasks within a strict project deadline while minimizing the total number of resources required. A shipyard usually sets the main planning milestones of the construction of a vessel (keel laying, launching, and delivery) at contract signing. Soon after, the planning department of a shipyard creates the section building planning (Meijer et al. 2009). The section building planning indicates the assembly and erection order of the vessel's steel sections as well as the allowable time windows and deadlines for outfitting. Furthermore, the monetary penalties incurred from late vessel delivery are generally very high since owners plan charters for a vessel based on the delivery date (Schank et al. 2005). Also, no hard limitations exist on the number of outfitting teams any subcontractor can assign to a given project at a given time.

To adequately model the SOSP, the traditional RACP formulation must be expanded. First, time windows are added to specify intervals during which outfitting can be performed. The allowable time windows for outfitting each section (preoutfitting, slipway outfitting, and quay outfitting) come from the section building schedule of a ship. Because the required time to outfit a component is a function of the outfitting stage during which the component is installed, the RACP must be expanded to cover phase-dependent task execution times. Sequence dependent setup times are also included to account for the fact that additional time is required when mounting teams move between work sites. Finally, the outfitting process of shipbuilding is partly governed by complex precedence relations that cannot be modeled by the simple start–finish precedence relations traditionally included in RACP. All of these elements are incorporated in the mathematical model developed for the SOSP.

A simple heuristic method is also developed to show that it is feasible to solve the SOSP formulation with a satisfactory solution quality in a reasonable computational time. A test case of six midship sections from a pipelaying vessel is used to assess the heuristic.

2. Literature review

An extensive literature review of the ship outfitting planning process was performed by Wei (2012) who concluded that shipbuilding literature only covered the topic in a cursory manner. Wei qualitatively described some of the constraints that govern the outfitting process and developed a method for automatically generating an assembly sequence of outfitting components within a single section. A summary of her work can be found in Wei and Nienhuis (2012). Rose and Coenen (2015) expanded the work of Wei by developing and comparing several metaheurstics to automatically generate an outfitting schedule for a single section.

The RACP was first introduced by Möhring (1984) who also proved that it was nondeterministic polynomial time (NP) complete. This means that it is not possible to develop an algorithm to solve the general case of this problem in polynomial time. The RACP is the dual problem of the resource constrained project scheduling problem (RCPSP), which seeks to minimize a project deadline under resources constraints as opposed to minimizing the required number of resources under time constraints (Hartmann & Briskorn 2010) Overall, the available scheduling literature for the

Nomenclature

A = set of active precedence relations
a_k = total number of resources available of type $k \in R$
A_{tk} = set of activities being executed at time t that require a resource of type $k \in R$
B_k = set of precedence relations which guarantee the first item to be mounted in each component group $k \in G$ belongs to F_k
b_i = base duration of activity $i \in N$
C_k = the cost function of a resource $k \in R$
ca_i = construction area of activity $i \in N$
CT_{it} = time window of component $i \in N$ at time t
d_i = duration of activity $i \in N$
dl_i = deadline of activity $i \in N$
EDD_i = priority for earliest deadline of activity $i \in N$
F_i = set of all starting activities in a component group $i \in G$
f = function for calculating setup time
fb_i = base time multiplication factor of activity $i \in N$
FCFS_i = priority for first come first served of activity $i \in N$
ftw_j = base duration multiplication factor of time window $j \in \mathrm{TW}$
G = set of component groups
lb_k = lower bound of resources required of type $k \in R$
M = set of mountable components
M_i = set of all activities in component group $i \in G$
MOT_{it} = priority for minimizing outfitting time of activity $i \in N$
MST_{it} = priority for minimizing setup time of activity $i \in N$
MTD_i = priority for mounting time dependents of activity $i \in N$
N = set of activities
P = set of finish-start precedence relations
PC_i = set of all penetrating components of component group $i \in G$
pc_{ik} = activity executed immediately prior to activity $i \in N$ using the same resource $k \in R$
Q = set of exclusion type precedence relations
R = set of renewable resource types
r_i = resource required of activity $i \in N$
S = set of components with special priority
s_i = start time of activity $i \in N$
st_i = setup time of activity $i \in N$
t = time
TW = set of all time windows
U_{ij} = set of finish-start precedence relations for component group $i \in G$ and starting activity $j \in F_i$
W_{ij} = set of all possible start times for activity $i \in N$ in time window $j \in \mathrm{TW}$

RACP is extremely scarce, contrasting with the RCPSP, which has been extensively studied. Möhring (1984) and Demeulemeester (1995) show that the RACP can be solved using a set of RCPSPs where the number of available resources are varied until a schedule is found that just meets required project deadline. Drexl and Kimms (2001) used Lagrangian relaxation and a column generation technique to determine lower and upper bounds of the classic RACP formulation.

Several variations of the RACP have also been studied. Neumann and Zimmermann (1999) examined the RACP with time windows and three different objective functions, Hsu and Kim (2005) addressed the multimode variation of the problem and Yamashita et al. (2007) analyzed the RACP under uncertain task execution times. Other formulation variations required for the SOSP, such as sequence-dependent setup times, have been addressed for the RCPSP (see Hartmann & Briskorn (2010) for a comprehensive literature review). Due to their relatively similar problems structures, extensions to the RCPSP formation can often be applied to the RACP. Although most of the extensions which are required to model the SOSP have been addressed individually in literature, no formulation has been found which is sufficient to fully model the SOSP.

Complex precedence constraints have only been covered in a cursory manner by all scheduling literature, which works almost exclusively with simple precedence relations known a priori. Möhring et al. (2004) defined and/or precedence constraints for parallel machine scheduling, which state that at least one of a set of activities must be completed to start another. Kuster and Jannach (2006) examined exclusion-type precedence constraints for a RCPSP formulation of the airport turnaround handling process. These constraints specify when two tasks cannot be executed simultaneously. Overall, the available scheduling literature on complex precedence constraints is inadequate to fully model the SOSP.

3. Process description

The outfitting tasks included in this analysis are those completed by small mounting teams, often employed by specialized subcontractors. Typical outfitting tasks include the installation of pipes, cable trays, HVAC ducts, foundations, and equipment. The majority of component installation tasks are of this type, except for the installation of large equipment. Large equipment, such as the main engines and generator sets, is often installed by larger teams that are employed by the equipment suppliers. Each outfitting team can only work on the installation of components belonging to their discipline. The objective of the outfitting process is to minimize the number of outfitting teams required for a shipbuilding project. The assumption is made that once an outfitting team is hired it will continue to work for the duration of the shipbuilding project to stress the importance of having a level workforce (Meijer et al. 2009).

A complex vessel can be composed of up to 50,000 components (Wei et al. 2010) that are each individually considered. Every component has a deadline, which is either dictated by the sea trial date, the testing date of the system to which the component belongs or the latest point during at which the component can be installed due to size restrictions.

Components can only be installed during certain outfitting stages (or time windows) that correspond to the shipbuilding stages of the sections and compartments to which the components belong. The definition of these time windows comes from the section building schedule of a vessel. Figure 1 shows the shipbuilding stages typically included in a section building schedule of a complex vessel. The first two stages, assembly and preoutfitting, are performed in the section assembly area. During the assembly stage, the panels of the section are welded together. The preoutfitting stage, which typically overlaps slightly with the assembly stage, offers for the installation of some components while the section is still very easy to access. Next, the section is transported to the paint hall. Depending on the shipyard planning, the section may be temporarily placed in a storage location before or after this stage. The section is then erected on the slipway/drydock and is available for slipway outfitting while the rest of the sections are being erected. Once all of the sections are erected, the ship is launched and then moored at the quay until it is ready for sea trials.

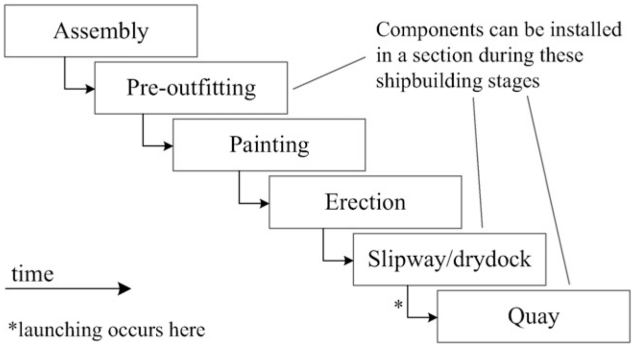


Fig. 1 Shipbuilding stages of complex vessel construction

However, not every component can be installed during every outfitting stage. If a component is located on a section boundary, the component cannot be installed during preoutfitting. This is to prevent the component from interfering with the alignment and welding portions of the erection process. The size of a component can also dictate during which stages it can be installed. At some point in time, large components can no longer be placed into a compartment due to the access restrictions. The deadlines of such components are dictated by the erection of the surrounding section that restricts access to the section. Finally, components of soft and easily damageable material should be installed at the slipway or quay to prevent damage (Wei 2012).

The mounting time of a component is dependent on the outfitting stage during which the component is installed. Schank et al. (2005) describes the reason for the additional installation time associated with the later mounting stages and proposes a series of factors to estimate this time increase. The contributing factors are the additional travel time of mounting teams, components and tools associated with slipway and quay outfitting as well as the increasingly cramped working conditions. The setup time required before installing a component is a function of the previous work location of the mounting team. When a mounting team changes work sites, additional time is not only required for the movement of personnel and tools, but also for the mounting team to familiarize itself with the new environments (reading drawings, finding power supplies and exits, etc.) (Wei 2012).

The remaining constraints governing the outfitting process relate to the assembly sequence of components. The first of these ensures a collision-free installation path. Figure 2 shows the motivation

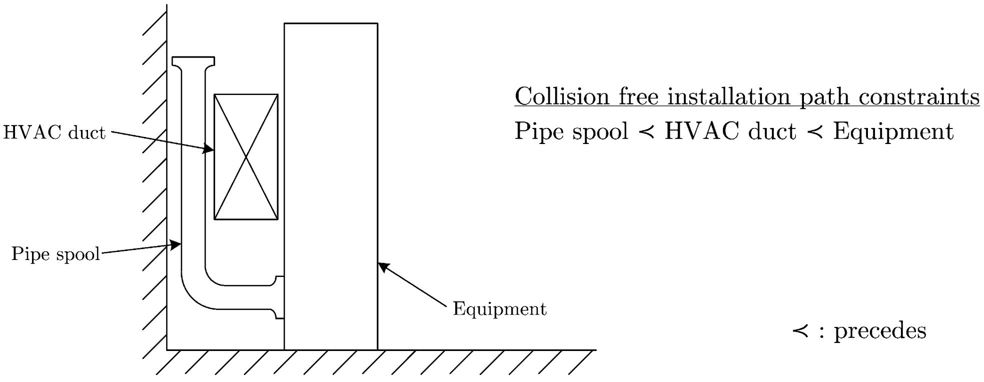


Fig. 2 Motivation for collision-free installation path constraints

behind these constraints. The mounting sequence of the components in this figure should clearly be as stated in the figure to ensure that workers are able to place components without needing to remove previously mounted ones. Furthermore, such a sequence ensures sufficient access and an open working space for the mounting teams.

Wei (2012) developed a method for defining the collision-free installation path constraints between components in section by looking for one-dimensional interferences between the components in the vertical direction. However, as noted by Wei, this method fails to take into account the steel structure of a section and the three-dimensional (3D) nature of the outfitting process. Wei recommended an improvement to her own method which is implemented here. First, each component is assigned to whichever boundary is closest to that component. Figure 3 shows an example of how a group of components in a section are assigned to the boundary which is closest to them.

Next, the one-dimensional interferences between each of the components associated to a boundary are calculated in the normal direction of that boundary. When an interference exists between two components, the component closer to the boundary must be mounted before the one which is further away. Figure 4 contains an

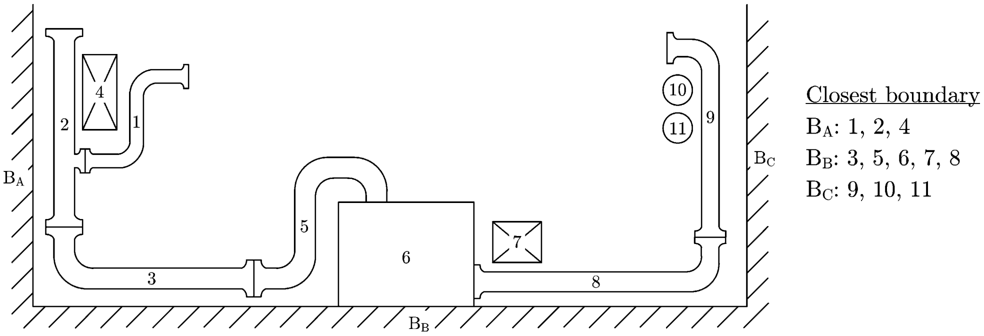


Fig. 3 Assignment of components to closest boundary

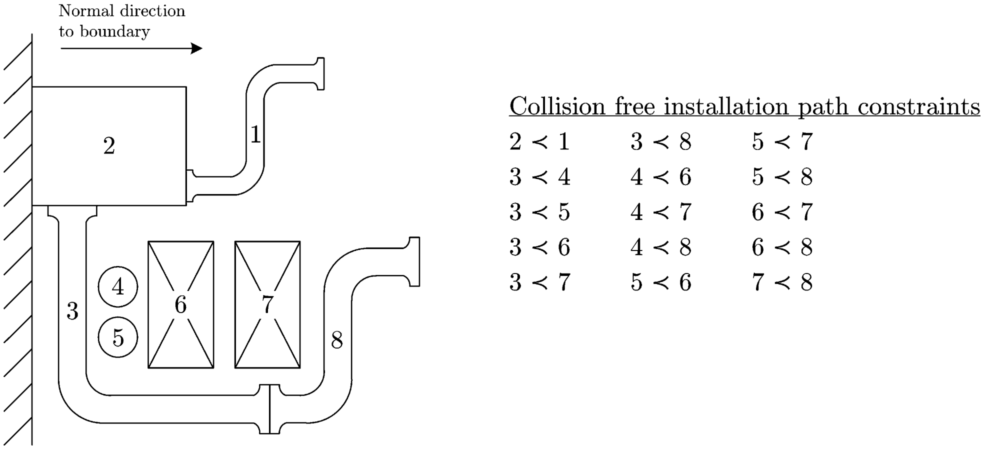


Fig. 4 Collision free installation path constraints between components

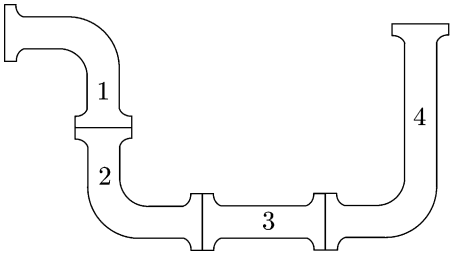
example of the collisions-free installation path constraints that would exist for a group of components associated with the same boundary.

The next set of assembly sequence constraints are those related to installation continuity. Wei (2012) shows that is bad practice to mount a pipe spool or HVAC duct between two already mounted ones due to the additional alignment work required when trying to match both ends of pipe simultaneously. Furthermore, assembly teams usually prefer to start mounting a pipe from a penetration piece, also for alignment reasons. Therefore, if a pipe has at least one penetrating pipe spool, one of those components should be mounted first. If a pipe has no penetrations, any spool of that pipe can be mounted first. Figures 5 through 7 show the installation continuity constraints for pipes with zero, one and two penetration pieces, respectively. Note that these constraints become even more complex for pipes with branches, but the underlying logic is identical. Due to the complex nature of the pipe routing in ship sections, a set of infeasible constraints could occur when combining the collision-free installation path and installation continuity constraints. In this case, the installation continuity constraints should be relaxed as the additional alignment work associated with violating these constraints is less than the work required for removing and remounting an already mounted component (especially if that component requires an assembly team of a different discipline).

The last set of assembly sequence constraints are based on the concept of maintaining a minimum safe working distance between mounting teams. A detailed rational behind these constraints is found in Wei (2012). In summary, to ensure that mounting teams do not encroach in each other's working areas, components that are separated by less than some minimum safe working distance should not be installed simultaneously. This results in a complex set of precedence relations between those components, where it is irrelevant which component is mounted first, yet the installation of one of the components must precede the other.

4. Mathematical model

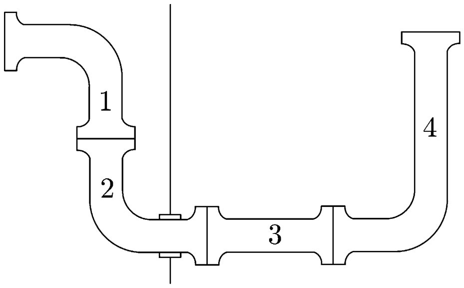
To define the ship outfitting scheduling problem, the classic RACP formulation was adapted to include phase-dependent task execution times, sequence-dependent setup times, time windows, complex precedence constraints, and individual task deadlines. The number of resources required per task was also limited to one



Installation continuity constraints

$[1 \prec 2 \prec 3 \prec 4]$ - or -
$[2 \prec 3 \prec 4,\ 2 \prec 1]$ - or -
$[3 \prec 4,\ 3 \prec 2 \prec 1]$ - or -
$[4 \prec 3 \prec 2 \prec 1]$

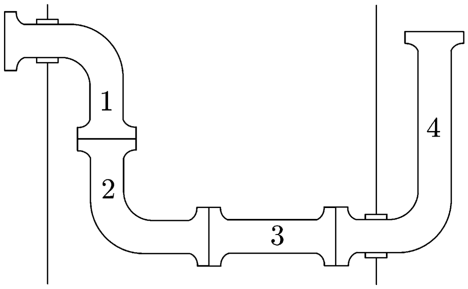
Fig. 5 Installation continuity constraints for pipe with no penetrations



Installation continuity constraints

$2 \prec 3 \prec 4$
$2 \prec 1$

Fig. 6 Installation continuity constraints for pipe with one penetration



Installation continuity constraints

$[1 \prec 2 \prec 3 \prec 4]$ - or -
$[4 \prec 3 \prec 2 \prec 1]$

Fig. 7 Installation continuity constraints for pipe with two penetrations to reflect the nature of the ship outfitting process (multiple outfitting teams do not work on the same activity).

R is a set of renewable resource types (in this case outfitting disciplines), and a_k represents the total number of resources (in this case mounting teams) of type $k \in R$ available during the project duration. C_k is the cost function of a resource over the project duration.

N is a set of activities (in this case mounting tasks), where each activity $i \in N$ has a start time s_i, duration d_i, deadline dl_i, base duration b_i, base time multiplication factor fb_i, construction area ca_i and requires one unit of resource $r_i \in R$. $\mathrm{pc}_{ik} \in N$ is the activity executed immediately prior to activity i using the same resource $k \in R$. If the resource is idle prior to starting activity i then $pc_{ik} = \emptyset$. The setup time st_i of an activity is calculated by a predefined function, f, so that $\mathrm{st}_i = f(\mathrm{ca}_i, \mathrm{ca}_{\mathrm{pc}_{ik}})$. $A_{ik} \subseteq N$ represents the set of activities being executed at time t that require a resource of type k.

TW is a set of all time windows (in this case outfitting stages). The start and finish time of each of these time windows is determined by the section building schedule. W_{ij} represents a set of all possible start times for activity $i \in N$ where that activity is executed in time window $j \in \mathrm{TW}$. If an activity cannot be executed in during a time window (due to that components weight, material, or distance to section boundary) then $W_{ij} = \emptyset$. ftw_j is the base duration multiplication factor associated with executing a task in time window j.

P is a set of finish–start precedence relations between two activities representing the collision-free installation path constraints. Q is set of exclusion type precedence relations between two tasks representing the minimum safe working distance constraints. M_i is a set of all activities in component group $i \in G$, where G is a set of all component groups. F_i is a set of all activities in a component group that could be the first activity of M_i to be executed where $F_i \subseteq M_i$. PC_i is a set of all activities in component group i that involve mounting a penetrating component, where $PC_i \subseteq M_i$. $F_i = PC_i$ if $PC_i \neq \emptyset$ and otherwise $F_i = M_i$. U_{ij} is the set of finish–start precedence relations required to satisfy the component group installation continuity constraints for component group $i \in G$ where $j \in F_i$ is the first component of i to be mounted.

$$\min \sum_{k \in R} C_k(a_k) \tag{1}$$

$$d_i = \mathrm{fb}_i \times b_i + \mathrm{st}_i \qquad \forall i \in N \tag{2}$$

$$\mathrm{fb}_i = \mathrm{ftw}_j \quad \text{if } s_i \in W_{ij} \qquad \forall i \in N, \forall j \in \mathrm{TW} \tag{3}$$

$$s_i + d_i \leq \mathrm{dl}_i \qquad \forall i \in N \tag{4}$$

$$s_i \in W_{ij} \qquad \forall i \in N, j \in \mathrm{TW} \tag{5}$$

$$s_i + d_i \leq s_j \qquad \forall (i, j) \in P \tag{6}$$

$$s_i + d_i \leq s_j \quad \text{or} \quad s_j + d_j \leq s_i \qquad \forall (i, j) \in Q \tag{7}$$

$$s_i + d_i \leq s_j \qquad \forall (i, j) \in U_{kl}, \forall k \in G, l \in F_k \tag{8}$$

$$\sum_{i \in A_{tk}} i \leq a_k \qquad \forall k \in R, \forall t \tag{9}$$

Equation (1) is the objective function, which minimizes the total cost. Equations (2) and (3) calculate the execution time of each activity. Equations (4) and (5) ensure that each activity is completed before its deadline and in the allowable time window. Equations (6–8) guarantee that all collision-free installation path, minimum safe working distance, and component group installation continuity constraints are met. Equation (9) sets the minimum number of required resources.

5. List scheduling heuristic

The following section describes the list scheduling heuristic developed to solve the SOSP formulated in the previous section. List scheduling is a greedy, priority-based heuristic scheduling algorithm that constructs a planning by individually scheduling each component in an order determined by the component priorities and the problem constraints. This type of solution technique was selected due to its low computational requirements. Because the RACP is NP-complete, any exact solution technique would take far too great of a computational effort to solve the SOSP for a complete complex vessel in a reasonable amount of time. Furthermore, Rose and Coenen (2015) showed that metaheuristics (including genetic algorithms, simulated annealing, and particle swarm optimization) required extensive computational times to schedule the components of a single vessel section.

List scheduling heuristics, however, have been shown to provide solutions to the RCPSP in very fast computational times (Brucker et al. 1999). In this paper, the approach of solving the RACP by solving the feasibility problem of a set of RCPSP with different resource limitations, presented by Möhring (1984) and Demeulemeester (1995), was implemented. As a result, it was critical to have a solution technique that could solve the RCPSP quickly. This is especially important when solving the case of an entire complex vessel. Another advantage of using a list scheduling heuristic was that it was possible to implement the complex precedence constraints of the SOSP without drastically altering the algorithm.

One main drawback of using list scheduling heuristics for scheduling is that they have been shown to have generally poor solution qualities and inconsistent performance compared to complex heuristics methods and metaheurstics (Brucker et al. 1999). The severity of this drawback depends on the problem structure. A test case was used to assess the severity of this drawback for the SOSP.

Figure 8 shows the algorithm of the list scheduling heuristic developed to solve the SOSP. The same notation from the mathematical model presented in the previous section is used in this figure. A parallel schedule generation scheme was selected instead of a serial schedule generation scheme since it was found to require less computational effort by the authors.

In order to determine the lower bound (Step 1 in Fig. 8) all precedence relationships between the components were relaxed and the setup time was set to zero for each component. Steps 5 to 14 of the algorithm shown in Fig. 8 were then used to schedule all of the components. Initially only one resource of each type was used, and the number of resources of a certain type was only increased each iteration if unscheduled components requiring that resource type existed at the conclusion of the previous iteration.

The performance of a list scheduling heuristics is very dependent on the priority rule used to select which component should be scheduled (Steps 8 to 9 in Fig. 8). Furthermore, the correlation

```
Step 1.  Determine a lower bound, $lb_k$, of $a_k \forall k \in R$
Step 2.  Set $a_k = lb_k \ \forall k \in R$
Step 3.  Set $S = \emptyset$ where $S$ is the set of components with special priority
Step 4.  Create active set of precedence relations $A$ where $A$ contains $P$
         and a set of precedence relations $B_k$ which guarantee that the first
         item to be mounted in each component group $k \in G$ belongs to $F_k$
Step 5.  Set time $t = 0$
Step 6.  Determine set of mountable components $M$ at $t$ where $M \subset N$.
         To be mountable, the following must be satisfied:
             (a) Component $m \in M$ must also satisfy $m \in A_{tk}$ were $k \in R$
                 is the resource required for $m$
             (b) $(j, m) \notin A$, $\forall j \in N$ where $s_j + d_j < t$
             (c) $t + d_m \leq dl_m$
             (d) $s_m \in W_{m,j}$ where $j \in TW$
Step 7.  If $M = \emptyset$ go to step 14
Step 8.  Select component $c \in M \cap S$ with highest priority
Step 9.  If $M \cap S = \emptyset$, select component $c \in M$ with highest priority
Step 10. Determine $d_c$ and schedule $c$ at $t$ and add $c$ to set of scheduled
         components $C$
Step 11. $\forall (c, j) \ or \ (j, c) \in Q$ add $c \prec j$ to $A$
Step 12. If $c$ is the first component of a component group $k \in G$, remove
         $B_k$ from $A$ and add $U_{kc}$ to $A$
Step 13. Go to step 6
Step 14. Increase $t$, if $t < max(dl_i \forall i \in N)$ go to step 6
Step 15. If $C \neq N$ add any component $i \in N \setminus C$ to $S$ which meets one of
         the following conditions
             (a) $dl_i < max(dl_k \forall k \in N)$
             (b) $(i, j) \in A \cup B_k$ where $j \in S$
Step 16. Repeat steps 4-15 until no addtitional components are added to $S$
Step 17. If $C \neq N$, increase $a_k \ \forall k \in R$ which are required by a task in $N \setminus C$
         and go to step 2
```

Fig. 8 Algorithm of list scheduling heuristic for SOSP

between solution quality and priority rule is a function of the problem structure (Boctor 1990; Franck & Neumann 1996). Therefore, several different priority rules were tested and their performance compared. The following priorities rules were considered:

1) **Minimizing outfitting time (MOT)**: Components are given priority based on the outfitting stage during which they would be scheduled at the current time. Equation (10) shows how this priority is assigned, where CT_{it} is the time window of component i at time t.

$$\mathrm{MOT}_{i,t} = \begin{cases} 1 & \text{if } \mathrm{CT}_{it} = \text{preoutfitting} \\ 2 & \text{if } \mathrm{CT}_{it} = \text{slipway outfitting} \\ 3 & \text{if } \mathrm{CT}_{it} = \text{quay outfitting} \end{cases} \tag{10}$$

2) **Minimizing setup time (MST)**: Components are given priority in such a way to minimize the setup time required of the task being scheduled. Equation (11) was used to assign this priority.

$$\mathrm{MST}_{it} = \begin{cases} 1 & \text{if } \mathrm{ca}_i = \mathrm{ca}_{\mathrm{pc}_{ik}} \\ 2 & \text{if } \mathrm{CT}_{it} = \mathrm{CT}_{\mathrm{pc}_{ik}t} \\ 3 & \text{otherwise} \end{cases} \tag{11}$$

3) **Earliest deadline of dependents (EDD)**: Priority was given to components based on the deadline of its dependents. Equation (12) shows how this priority was assigned, where D_i is the set of all possible dependents of a component.

$$\mathrm{EDD}_i = \min\left(\mathrm{dl}_i, \mathrm{dl}_j\right) \quad \forall j \in D_i \tag{12}$$

4) **Mounting time of dependents (MTD)**: Component priority was based on the total mounting time of the component and its dependents. This priority was assigned using

Equation (13). For this rule, a higher value corresponds with a higher priority.

$$\mathrm{MTD}_i = b_i + \sum_{j \in D_i} b_j \qquad (13)$$

5) **First come first serve (FCFS)**: Priority was given to components based on when they first became available to be mounted. Equation (14) was used to assign this priority.

$$\mathrm{FCFS}_i = \min(u) \quad u \in W_{ij} \quad \forall j \in \mathrm{TW} \qquad (14)$$

These rules were selected because of their low computational requirements. Each rule could either be determined a priori or dynamically with a simple calculation. Furthermore, these rules also cover a wide variety of different aspects that could drive the ship outfitting process, such as the required setup time, the precedence network between components, the component deadlines and the outfitting time windows.

Moreover, it was not possible to directly implement any rules that were derived from the critical path method (CPM), such as latest start time, earliest start time and minimum slack time. Such rules are often present in other literature using priority-based list scheduling, such as Boctor (1990) and Franck and Neumann (1996). Dubois et al. (2003) show that it is not possible to directly implement CPM-based rules without using stochastic calculations where task durations and deadlines are not known a priori. In the case of the SOSP, calculating CPM values is even more complex because the task durations are a function of the time window during which the component is scheduled and a varying setup time. Furthermore, the precedence network itself is not fixed.

Because the priority rules shown above often resulted in ties between components, a second priority was used to break the ties. Random selection was used in the case that two components shared the highest priority for both rules.

6. Test case

In order to test the mathematical model and list scheduling heuristic formulated for the SOSP, a test case was performed on six sections of a pipelaying vessel. The examined vessel was built by Royal IHC in their Krimpen location and delivered in 2014. These sections represent a 17 m stretch of the vessels midship that contains a high voltage switchboard room, moon pool, pump room, and winch room. This set of sections was selected since they contain a variety of different types of spaces, ranging from an outfitting intensive high voltage space to the relatively open area surrounding the moon pool. Furthermore, no major compartments spanned between the selected sections and adjacent ones, which meant that is was possible to consider the sections in isolation. Table 1 shows some general characteristics of the selected sections.

The following outfitting disciplines and tasks were considered, which represent over 95% of the outfitting components in the examined sections:

1) **Piping**: pipe spools, valves, strainers, pumps, minor equipment
2) **HVAC**: ducts, fans, minor equipment
3) **Electrical**: cable trays, light fixtures, minor equipment
4) **Ironworks**: foundations, stairs, ladders, platforms, railings

Painting and insulation were not included because these two tasks are generally performed independently from the included tasks (Wei 2012). Furthermore, large components (weighing more than 500 kg) were excluded. This was done because the installation of such components is typically not done by the two man mountings team resources considered by the mathematical model. Furthermore, the planning of these components, such as the main engines and generators, is often dictated by the delivery schedule of the suppliers.

Table 1 Test section characteristics

ID	Components	Location	Type
A	100	Centerline	Double bottom
B	56	Starboard	Double bottom
C	90	Port	Double bottom
D	535	Starboard	Aux machinery
E	739	Port	Aux machinery
F	305	Centerline	Moonpool

Mounting times of pipe spools, cable trays, and HVAC ducts were estimated using the equations developed by Wei (2012). In her research, Wei also determined the mounting time of roughly 100 pieces of equipment. A polynomial was fit through this data to estimate the mounting time of a piece of equipment from its mass. To determine the mounting times of ironworks, the total number of hours associated by the shipyard to these outfitting tasks was divided by the total mass of these components. An experienced outfitting supervisor was asked to estimate the mounting time of light fixtures.

Mounting time factors were estimated using Schank et al. (2005) as guidance. Schank et al. (2005) preformed a surveyed four European Union shipyards asking the yards to estimate these factors. The results of this survey were used as guidance for the authors to select the factors used for the test case. Factors of 1.0, 1.5, and 2.0 were chosen for preoutfitting, slipway outfitting, and quay outfitting, respectively. The required setup time taken to be 0 minute if the mounting team was previously in the same work location (section or compartment), 15 minutes if the mounting team was in the same area (section assembly area, slipway, or quay), and 30 minutes otherwise. The setup time values were based on shipyard observation of the authors.

Because Royal IHC depends on subcontractors for the installation of cable trays and HVAC ducts, these components were not broken into small chunks suitable for mounting in the 3D model of the examined vessel. Instead, these components were modeled as entire lines. The methodology developed by Wei (2012) was used to break these lines into roughly 3-m long pieces suitable for installation. Wei cites that the preferred installation size of HVAC ducts in the maritime industry corresponds with spools of roughly this size.

The latest point in time a component could be installed based on size constraints were determined assuming that the six sections were built in isolation. A material constraint was included to specify that no equipment was to be installed in the preoutfitting stage. This constraint was based on the author's observation of the shipyard examined in the test case. A distance of 2.5 m, as proposed by Wei (2012), was used to generate the exclusion-type precedence constraints to ensure a minimum safe working distance between mounting teams.

Finally, the section building schedule of the vessel was shortened to make the test case more realistic. Otherwise, a single resource of each discipline would have sufficient time to complete all of the outfitting tasks. The section building schedule was reduced to 13% of its original length, which is proportional to the weight of the six test sections relative to the total ship. Although this type of reduction results in a somewhat skewed section building scheduling, it is sufficient for determining the feasibility of the mathematical model and list scheduling heuristic developed for the SOSP. However, a test case of a complete complex vessel should be performed before using these methods to draw conclusions about the outfitting process of shipyards building complex vessels.

All geometric-dependent input information, mainly the precedence constraints, was derived from a 3D cad model of the test case vessel. This model contained detailed coordinates describing the position and shape of each of the considered outfitting components as well as the vessel's steel structure. This allowed for the development of a preprocessing script that automatically generated the collision-free installation path, minimum safe working distance, and pipe continuity constraints. Because of the large number outfitting components in a complex vessel, manually generating these constraints would not be feasible in reasonable amount of time. Furthermore, component properties (such as room and section associations, penetration, etc.) were derivable when not explicitly recorded by the shipyard.

However, using the 3D model of a vessel as input for creating an outfitting planning implies that this model must exist at the time the schedule is generated. This is not the case for most modern shipyards building complex vessels, which are generally time-sensitive projects which are built as fast as possible. These yards generate their production plans in parallel with detailed engineering drawings in an effort to minimize the time required to deliver a custom-built vessel. A full analysis of the problem of using a detailed 3D model as input for generating outfitting plans and its implications on can be found in Rose and Coenen (2014). Rose and Coenen found that the introduction of automatic design tools in the ship design process has the potential to increase the amount of detailed data early in the ship design process, which in turn allows for the better implementation of automatic scheduling approaches such as the one proposed in this paper.

7. Results

Figures 9 and 10 show the resource distribution functions found for two different combinations of priority rules. Both of these functions show that in the beginning of the outfitting process,

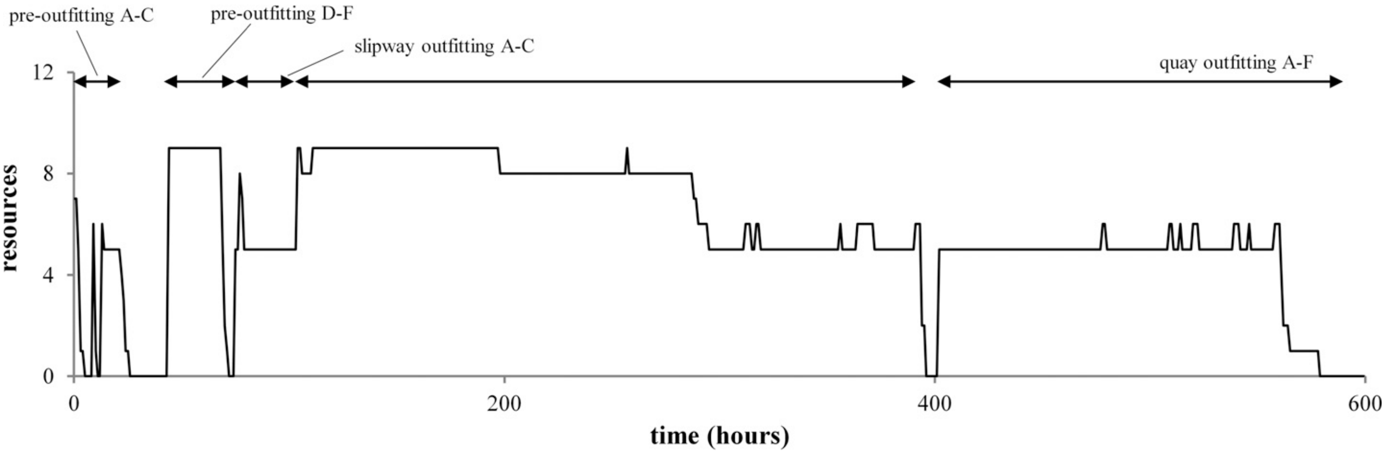


Fig. 9 Resource distribution function for EDD/MST (main/aux) priority rule

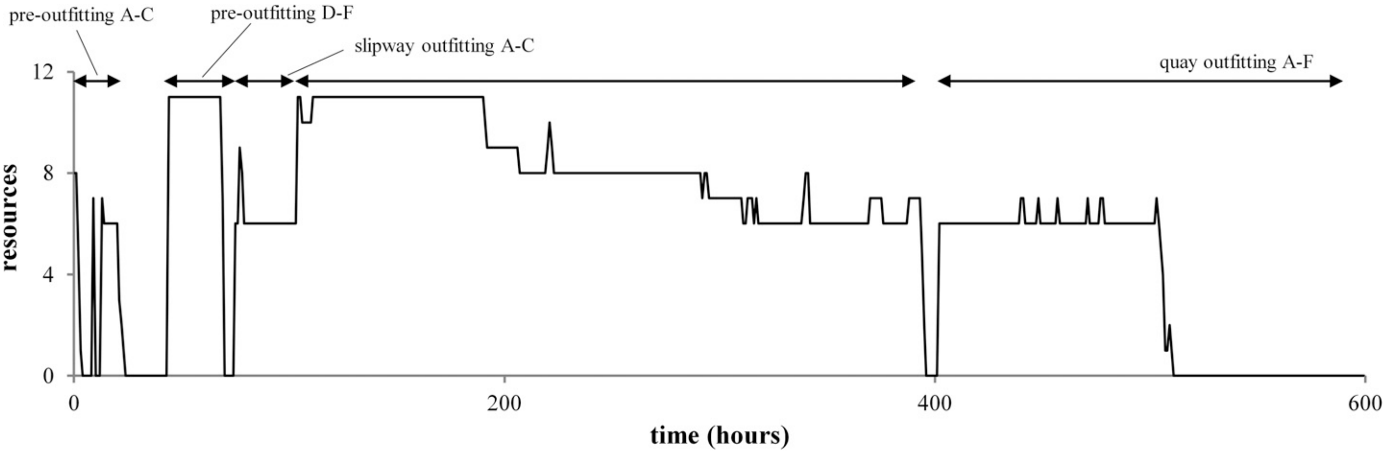


Fig. 10 Resource distribution function for EDD/MOT (main/aux) priority rule

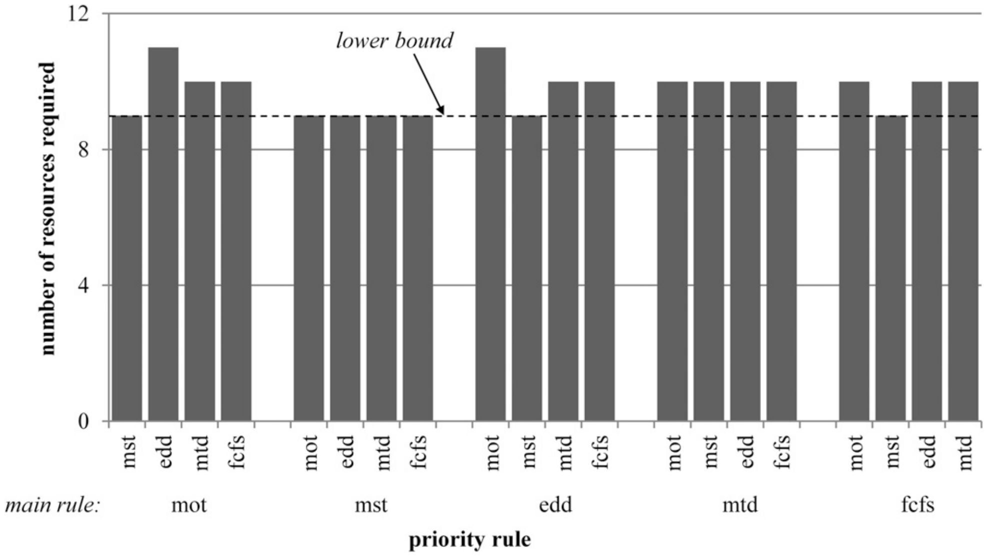


Fig. 11 Total number of resources required

during the preoutfitting of sections A–C, insufficient work existed to occupy all of the available resources. This is because these sections are double bottom sections, which are much smaller than midship sections and have fewer outfitting components. Double bottom sections also do not usually have components that belong to the HVAC or electrical disciplines. All available resources were fully occupied while the remaining sections were preoutfit. During the slipway outfitting phase, the number of required resources decreased. The occurred as outfitting disciplines completed all of their required work. Around 400 hours there was a brief pause in all outfitting work as the ship was launched. These figures show that the priority rules EDD/MST created a superior outfitting planning compared to EDD/MOT since fewer total resources were required.

Figure 11 shows the minimum number of required resources for each combination of priority rules. This figure shows that the different priority rules did not result in vastly different results. This is partially due to the small size of the test case and partially due to the relatively under-constrained nature of SOSP. In general, the best solutions were found by any combination of priority rules that included the MST criteria. All four schedules generated using this criteria as the main rule found the best solution of any of the priority rule combinations. This occurs because the MST priority rule seeks to eliminate unnecessary setup time by assigning mounting teams tasks in the same work area as their previous task whenever possible, reducing the total required outfitting man hours.

To assess the quality of the schedules generated by the list scheduling heuristic, the number of resources required were compared to the number indicated by the lower bound calculated in the first step of the algorithm. This lower bound is also shown in Fig. 11. The lower bound indicates that at least nine resources are required, the same solution found by some of the combinations of priority rules. This means that the list scheduling heuristic found the optimal solution for the six section test case presented here.

The results found from the test case indicate that the list scheduling heuristic did not result in poor solutions as is often seen when this type of algorithm is applied to other scheduling formulations. One possible reason for this is the generally under-constrained nature of ship outfitting. Table 2 shows the average number of constraints of each type per component for the examined six sections. This table shows that the most common constraint present by a large margin were those related to maintaining a minimum safe working distance. However, these constraints do very little to constrain the allowable mounting times of components since they merely state that two components cannot be installed at the same time. This slightly limits the allowable installation window of one of the two components, but does not enforce any hard precedence relations between the components. This table also shows that on average each component had less than one collision-free installation path or installation continuity constraint. As a result of the under-constrained nature of the problem, the heuristic was usually able to keep the resources fully occupied until almost all of the components of that discipline were finished. This caused the generally flat nature of the required resource function seen in Figs. 9 and 10. The small spikes seen in these figures resulted from the few hard precedence relations present in the test case.

Table 2 Average number of constraints per component

Constraint type	Constraints per component
Collision-free installation path	0.26
Installation continuity	0.71
Minimum safe working distance	28

The majority of the computational time of the heuristic algorithm was used performing steps 5 to 14, which attempts to construct a feasible schedule using the current resource limits and priorities. If a feasible schedule was not found, this process would be repeated with modified resource limits and components with special priority until such a schedule was found. Figure 12 shows the number of scheduling iterations that were required for each combination of priority rules until a feasible solution was found. This number is roughly proportional to the computational time. Figure 12 indicates that the least number of iterations was required when EDD was the main priority rule. This occurred because this

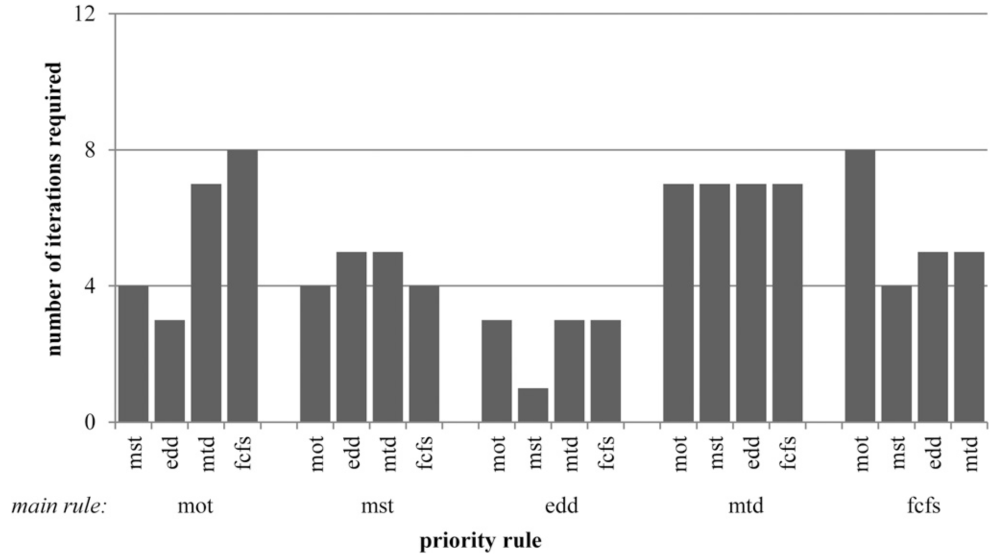


Fig. 12 Total number of scheduling iterations required

priority rule generally gave the highest priority to components, which would eventually be assigned special priority, reducing the required number of required iterations.

The best combination of priority rules both for solution effort and computational effort was using EDD as the main rule and MST as the auxiliary rule. However, when using these two rules, random selection was regularly used to break ties as there were many components with identical priorities. This occurred because a majority of the components had the same deadline (the end of the quay outfitting stage) and the various sections and compartments often had multiple components, which were available to be mounted at any point in time. Therefore, the authors suggest an investigation into finding a suitable third priority rule. In this case, however, the implementation of such a rule would not improve the solution quality since the generated schedule already matched the lower bound. This may not be the case, however, when examining a larger group of sections or an entire vessel.

The test case was written in the native PostgreSQL scripting language (pl/pgSQL) and run on a dedicated server with 2.5 GHz quadcore Intel Xeon CPU, 16 GB RAM and 2 TB raid hardrive system. Unfortunately, it is not possible to determine the required computational effort of this calculation because the server is shared by multiple users. However, a scheduling iteration was generally created in roughly 15 minutes. Furthermore, little effort was done to optimize the code for performance or to assign appropriate indexes.

8. Conclusions

The heuristic list scheduling technique developed was able to find a high-quality solution to the proposed formulation for the SOSP using a reasonable computational effort for a test case of six sections of a pipelaying vessel. Figures 11 and 12 show that the best performance of the developed list scheduling heuristic, both in terms of solution quality and computational time, was seen when using EDD and MST as the main and auxiliary priority rules, respectively. This indicates that when planning outfitting tasks it is important to first consider those components that have the earliest deadlines, as well as the other components on which those components depend. A shipyard should also try to minimize the number of unnecessary movements performed by the mounting teams between different work sites during the outfitting process.

The test case also indicates that, in general, the ship outfitting process is under constrained. This reinforces the decision to use a priority-based list scheduling heuristic for outfitting planning, as such algorithms tend to exhibit favorable performance in problems with few restrictions. Furthermore, the large problem size of the SOSP matches well with the low computational requirements of these type of algorithms.

To further test both the mathematical model and heuristic presented in this paper, a test case of an entire vessel should be conducted. First, this would test the validity of the proposed SOSP formulation on a more realistic test case. Moreover, a complete ship would contain section types, such as accommodation sections, which were not included in the test case presented in this paper. Examining a whole vessel would also allow a shipyard to evaluate a section building planning in terms of outfitting. For example, feedback would be created to assess the appropriateness of the time windows given to each section to perform the required outfitting work. Furthermore, a quantitative analysis could be performed to assess the potential benefits of training multiskilled outfitting workers who are able to perform mounting tasks of several different disciplines.

Ultimately, this type of analysis could be used by a shipyard to guide the planning of its outfitting process with the goal of reducing the number of resources required for these tasks. Such a planning could also be used by the shipyard as a coordination tool between the specialized outfitting subcontractors or as a reference for tracking the outfitting process of a ship.

Acknowledgments

The authors would like to thank Royal IHC for providing access to the data used to perform the test case presented in this paper.

References

Boctor, F. F. 1990 Some efficient multi-heuristic procedures for resource-constrained project scheduling, *European Journal of Operational Research*, **49**, 1, 3–13.

Brucker, P., Drexl, A., Mhring, R., Neumann, K., and Pesch, E. 1999 Resource-constrained project scheduling: Notation, classification, models, and methods, *European Journal of Operational Research*, **112**, 1, 3–41.

CESA. 2011 Annual report 2010-2011. Technical Report, Community of European Shipyards Associations. It can be found at http://www.cesa.eu/links_downloads#.

Demeulemeester, E. 1995 Minimizing resource availability costs in time-limited project networks, *Management Science*, **41**, 10, 1590–1598.

Drexl, A. and Kimms, A. 2001 Optimization guided lower and upper bounds for the resource investment problem, *The Journal of the Operational Research Society*, **52**, 3, 3, 340–351.

Dubois, D., Fargier, H., and Fortemps, P. 2003 Fuzzy scheduling: Modelling flexible constraints vs. coping with incomplete knowledge, *European Journal of Operational Research*, **147**, 2, 231–252.

Franck, B. and Neumann, K. 1996 Priority-rule methods for the resource-constrained project scheduling problem with minimal and maximal time lags-an empirical analysis, *Proceedings*, The 5th International Workshop on Project Management and Scheduling, Poznan, Poland, pp. 8891.

Hartmann, S. and Briskorn, D. 2010 A survey of variants and extensions of the resource-constrained project scheduling problem, *European Journal of Operational Research*, **207**, 1, 1–14.

Hsu, C.-C., and Kim, D. S. 2005 A new heuristic for the multi-mode resource investment problem, *The Journal of the Operational Research Society*, **56**, 4, 406–413.

Kuster, J. and Jannach, D. 2006 Handling airport ground processes based on resource-constrained project scheduling. In Advances in Applied Artificial Intelligence, *Proceedings*, 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2006, June 27–30, Annecy, France, pp. 166–176.

Meijer, K., Pruyn, J., and Klooster, J. 2009 Erection scheduling support tool, *Proceedings*, The 8th International Conference on Computer and IT Applications in the Maritime Industries, TuTech GmbH, May 10–12, Budapest, Hungary. pp. 326–332.

Möhring, R. H. 1984 Minimizing costs of resource requirements in project networks subject to a fixed completion time, *Operations Research*, **32**, 1, 89–120.

Möhring, R. H., Skutella, M., and Stork, F. 2004 Scheduling with and/or precedence constraints, *SIAM Journal on Computing*, **33**, 2, 393–415.

Neumann, K. and Zimmermann, J. 1999 Resource levelling for projects with schedule-dependent time windows, *European Journal of Operational Research*, **117**, 3, 591–605.

Rose, C. and Coenen, J. 2014 Current state of automated design tools for increasing availability of data early in the ship design process. *Proceedings*, The International Conference of Maritime Technology, ICMT2014, July 7–9, Glasgow, Scotland.

Rose, C. and Coenen, J. 2015 Comparing four metaheuristics for solving a constraint satisfaction problem for ship outfitting scheduling, *International Journal of Production Research*, **53**, 19.

Schank, J., Pung, H., Lee, G., Arena, M., and Birkler, J. 2005 Outsourcing and outfitting practices: Implications for the ministry of defence shipbuilding programmes. Technical Report, RAND Corporation, Santa Monica, CA.

Wei, Y. 2012 Automatic generation of assembly sequence for the planning of outfitting process in shipbuilding, Dissertation, VSSD, Delft University of Technology.

Wei, Y. and Nienhuis, U. 2012 Automatic generation of assembly sequence for the planning of outfitting processes in shipbuilding, *Journal of Ship Production and Design*, **28**, 2, 49–59.

Wei, Y., Nienhuis, U., and Moredo, E. 2010 Two approaches to scheduling outfitting processes in shipbuilding, *Journal of Ship Production*, **26**, 1, 20–28.

Yamashita, D. S., Armentano, V. A., and Laguna, M. 2007 Robust optimization models for project scheduling with resource availability cost, *Journal of Scheduling*, **10**, 1, 67–76.