Optimizing inventory strategy for modular shipbuilding

van der Beek, Tom; van Essen, Theresia; Pruijn, Jeroen; Aardal, Karen; Hopman, Hans

Publication date
2019

Document Version
Final published version

Published in
19th International Conference on Computer Applications in Shipbuilding (ICCAS 2019)

Citation (APA)
van der Beek, T., van Essen, T., Pruijn, J., Aardal, K., & Hopman, H. (2019). Optimizing inventory strategy for modular shipbuilding. In *19th International Conference on Computer Applications in Shipbuilding (ICCAS 2019)* (pp. 165-172). Royal Institution of Naval Architects.

OPTIMIZING INVENTORY STRATEGY FOR MODULAR SHIPBUILDING

T van der Beek, J T van Essen, J Pruyn, K Aardal, H Hopman, Delft University of Technology, NL

SUMMARY

The primary drivers for buying a ship from a certain yard are price, delivery time and quality. In order to decrease construction time and costs, shipbuilding companies are exploring the development of product-families to include family wide modularity and cross family standardization. Standardization is the use of identical components across multiple products, while modularity combines parts to create 'building-blocks'. This creates an opportunity for less inventory, a more efficient supply chain and shorter delivery times. Considering a network of suppliers and shipyards, the shipbuilder has to answer the following question: *Which components and pre-assembled modules should be available in which inventory?* Since the exact ship orders are not known, this can be seen as an optimization problem with uncertainty. To solve it, it is formulated as an integer linear program (ILP), and to handle the uncertainty, the Sampling Average Approximation (SAA) method is used. Several smaller instances are solved to optimality by Gurobi optimization software and the performance of this approach is evaluated along with the convergence of the SAA method. The results show convergence of the SAA method although only relatively small instances can be solved to optimality by the ILP.

NOMENCLATURE

SETS

A_s	Relative times when $q_{sia} > 0$ for any i
D	Depots
D'	Depots in the original problem without scenarios
D'_d	Depots in the multi-scenario problem, which link to depot $d \in D'$ in the original problem
D_j	Depots for yard $j \in J$
I	Component and module types, i.e., $I_c \cup I_m$
I_c	Component types
I_m	Module types
J	Yards
M	Module options
M_s	Module options which can be used by ship $s \in S$
M_{sia}	Module options of module type $i \in I_m$ which can be used by ship $s \in S$ at time $a \in A_s$
S	Ships
S_j	Ships produced in jard $j \in J$
T	Time periods
T_s	Time periods when ship $s \in S$ can be constructed
Λ	Scenarios
Λ_∞	All possible scenarios
Φ	Base inventories

PARAMETERS

a_{s0}	Earliest relative time in A_s
b_i	Cost per one base stock for component or module type $i \in I$
c_{st}	Cost for starting the construction of ship $s \in S$ in time period $t \in T_s$
f_s	Construction time of ship $s \in S$
j_s	Yard of ship $s \in S$
l_{dj}	Lead time from depot $d \in D$ to yard $j \in J$
OBJ_u	Objective value for iteration u in robustness evaluation algorithm
$OBJ_u^{\theta(v)}$	Objective value for iteration u in robustness evaluation algorithm with base inventory $\theta(v)$.
q_{sia}	Number of components of type $i \in I$ required for ship $s \in S$ at relative time $a \in A_s$
r_{mia}	Reduction at relative time $a \in A_s$ of component type $i \in I_c$ by using module $m \in M$
t_s	Order time for ship $s \in S$
Δ_u	Variation parameter for iteration u in robustness evaluation algorithm
λ_d	Lead time for depot $d \in D$ to re-order a component
φ^*_Λ	Optimal base inventory over scenario set Λ
$\theta(u)$	Base inventory of iteration u in robustness evaluation algorithm

DECISION VARIABLES

W_m	1 if module option $m \in M$ is used, and 0 otherwise
X_{st}	1 if ship $s \in S$ is constructed at time $t \in T_s$, 0 otherwise
Y_{sdit}	Number of components or modules of type $i \in I$ ordered for ship $s \in S$ at time $t \in T_s$ from depot $d \in D$
Z_{dit}	Inventory level of component type $i \in I$ at time $t \in T$ at depot $d \in D$

1. INTRODUCTION

In the current market of shipbuilding, lead times are an important aspect to gain a competitive advantage for both standardized and custom design [1]. The research presented in this paper was done within the NAVAIS project, an EU sponsored cooperation between multiple European maritime companies and Delft University of Technology to develop a modular product family for work boats and ferries. Work boats can be used for various activities on water, like construction, maintenance and inspection. Ferries are used for transporting passengers and vehicles.

When buying a product, the shortest lead time would be attained if a product could be bought directly from stock. This is, for example, often the case when buying a new car. Unfortunately, for ships, this is usually not possible because of several reasons: First of all, ships are very expensive and keeping finished ships in stock carries high costs. These costs are due to deprecation of the ships, interest costs on investments, maintenance costs and storage costs. Secondly, ships are customized products. This means that the total variety of ships is very large and by keeping every possible variation in stock, it is bound to result in a large part of the stock not being sold.

Therefore, instead of producing to stock, shipyards traditionally start producing a ship after a customer orders it. The influence a customer order has on the production process can be characterized by the customer order decoupling point (CODP) [2]. This point defines how far the client order penetrates the production or distribution process. Traditional ship production is done by engineering-to-order (ETO). In ETO, both the engineering process and the production process are done after the customer order is placed. This is the earliest CODP possible (Figure 1), and minimizes the costs of keeping products in stock and the risk of overproduction to basically zero.

From earlier to later CODP, the other production types are make-to-order (MTO), assemble-to-order (ATO) and make-to-stock (MTS) (see also Figure 1). In MTO, the engineering design is available before the customer orders a product, and the production is done afterwards. In an ATO environment, there consists one or more defined product families. These exists of multiple modules. After a product is ordered, the modules are assembled to create the desired product. Lastly, MTS produces complete products and stores them in stock.

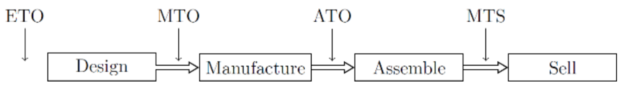

Figure 1: Types of production environments

An ATO approach has the capability of reducing lead times, while maintaining a varied product portfolio. An ATO system consists of multiple subassemblies or modules. By having these modules available in the inventory, the lead times are reduced due to the reduction in manufacturing and transportation time. The portfolio variety is kept since relatively few modules can create many combinations of the final product. In addition, this offers opportunities to reduce inventory costs [3]. When producing according to an MTO or ATO policy, a base stock policy can reduce lead times. A base stock policy is a policy where (some) components are kept in stock regardless of ship orders. When components leave the inventory, they will be re-ordered to resupply the base-stock levels.

In this paper, a modular shipbuilding approach is assumed in an ATO environment of multiple depots, component manufacturers and shipyards. Given this environment and a product family, two questions arise. The first one is the placement of components. Obviously, having components available near the shipyard can reduce lead times, but also increase inventory costs. Secondly, the question arises which modules to already pre-assemble. By pre-assembling modules, flexibility is lost, but it creates an opportunity for decreased lead times.

The contribution of this paper is providing a first optimization model for this problem. This model, combined with the SAA method, minimizes the expected costs due to storage and delayed construction by determining the base-stock levels, component sources and scheduling per yard. This provides valuable information, both for direct operations, i.e., deciding the stock levels, as for evaluating the potential benefit of modular production of product families.

In Section 2, the literature on base-stock optimization is reviewed. Subsequently, the problem is described in Section 3 and the model is given in Section 4. Section 5 expands the problem to include stochastic ship orders, Section 6 shows computational results for the implemented model and Section 7 concludes the paper and presents an outlook on future research on this topic.

2. LITERATURE STUDY

Since shipbuilding is traditionally ETO based [4], optimization approaches for the base-stock in this industry are, to the best of the authors' knowledge, non-existent. There is some research on inventory minimization for interim products during the production process [5], but to obtain more information, the scope of the literature study was expanded to other industries. However, similar industries as shipbuilding face the same problem. Since most current approaches are ETO or MTO, there is little research on base-inventory optimization for industries with large, highly customizable, and low sale volumes like industrial machinery, industrial transport systems and construction [6].

Agrawal and Cohen [7] studied the problem of determining inventory levels for a stochastic assembly system at multiple time periods to minimize the holding costs, while guaranteeing a probabilistic service threshold, i.e., a lower bound on the probability that there are no delays. This is done for a single location. A stochastic programming model is given and the optimal solution is given for small instances.

A similar problem is studied by Akçay and Xu [8]. For a single horizon period, an optimization problem is modeled to determine base stock levels and replenishment decisions to maximize profit of the completed orders. Proof is given for the NP-hardness of this problem, and the stock levels are determined with the sample average approximation method (SAA). This method creates multiple scenarios after which it optimizes the average cost function of one policy applied to all scenarios.

Lu and Song [9] also study an ATO system to define the base stock levels. They show that, given continuous stock level variables, the optimal solution can be found with a local search steepest descent algorithm. They compare the optimal solution against a simpler heuristic used commonly in practice, and evaluate the base stock levels for varying replenishment lead times and lead time variability.

Other variations include pre-assembly and de-assembly actions [10] and dependence between demands of multiple items [11]. Furthermore, research is done in heuristic approaches for base-stock levels [12]. Van Jaarsveld and Scheller-Wolf [13] study industrial-size ATO systems, consisting of hundreds of components, and use SAA to determine the base stock levels.

A related field of research is spare parts inventory optimization. Deciding the optimal stocking policy for spare parts has the following characteristics [14]: The demand patterns are intermittent and characterized by long sequences of zero demand, the variety of parts is very large and it is desired to minimize stocks to reduce the risk of spare parts obsolescence. These are similar to the characteristics of the shipbuilding industry.

Nozick and Turnquist [15] minimize the stock-out and holding costs for an inventory allocation problem with multiple distribution centers. Each retail outlet can order parts at exactly one distribution center, and an order is backlogged if it is unavailable at the distribution center. For this problem, they provide a non-convex formulation.

For a single depot, Van Jaarsveld et al. [16] define the problem of determining the restocking policy per component. For each component, the restocking level is optimized to minimize the holding and ordering costs. This is solved to optimality with a column generation approach, and for faster computations, a rounding heuristic is given.

Basten and Van Houtum [17] study a multi-warehouse spare parts inventory problem with a continuous review base stock policy. In this problem, the base stock levels are determined, subject to various service constraints. Service constraints are, for example, bounds on the expected waiting time per component, the rate of orders which can be fulfilled or the average availability of products. They give a greedy algorithm to find a Pareto front of minimal costs and expected back-orders.

A problem with discrete inventory levels was studied by Yang and Du [18]. They use a genetic algorithm to define the base stock levels to minimize the expected back-orders and storage costs.

A range of research is available for optimizing the base stock levels. However, certain properties of our problem are still missing in the literature. One of the main properties is the out-of-stock consequence. In most research, the consequence of an out-of-stock is either delay until all components arrive, or penalty costs. For the ship construction problem, delay of a single ship will also affect the schedule of other ships at the same yard.

Furthermore, almost all referred papers use continuous stock levels to simplify the calculations. In ship constructions, systems like cranes or firefighting systems are large, expensive and needed in low quantities. Therefore, they cannot be simplified to continuous stock.

Finally, there is the aspect of multiple depots and yards. Although multiple-location models exist in the literature, most of them follow quite simple policies. In most cases in literature, each production facility is linked to a fixed depot, and ordering will always happen from that depot with possibility of a backorder. It is also often considered that a first-come first-serve policy is used, to further simplify decisions. In shipbuilding, due to the high costs of delay, it is common to order components from another depot if needed. Therefore, the decision of which component comes from which depot should also be taken into account.

3. PROBLEM DESCRIPTION

The studied problem consists of a network of yards, depots and manufacturers, where ships are produced at yards. The required components can either be taken from depots or ordered from manufacturers. It is assumed that each ship is pre-assigned to a specific yard. At the beginning of a fixed time period, for example a year, it has to be decided what the base-inventory levels for both components and assembled modules are.

However, the exact customer orders are unknown. If they were known, the base-inventory would simply be zero, since with perfect foresight all components can be ordered in time. It is assumed that the ship orders are generated by a known random process. A realization of this random process is a set of ship orders, which is called a scenario. In this research, it is assumed that a simple Bernoulli process generates the ship orders, although in practice, a more realistic generator can be created based on expert opinions from the industry. Any generator function can be used with the same model.

Each ship model consists of a construction duration and a list of components, of which each is required at a defined time during the construction process. However, for certain sets of components, it is also possible to use a pre-assembled module instead.

The total time between ordering a product and receiving it is called the lead time. To reduce the lead time, components can be already stored in depots. This reduces the lead time based on the distance between the depot and the yard, possibly approaching zero.

In Section 4, the problem is modeled as an optimization problem where the base-stock levels per depot and the flow of components is determined to minimize the costs of delayed production and holding stock. This is done for a single scenario. Subsequently, in Section 5, the model is expanded for multiple scenarios using the SAA method.

4. MODEL DEFINITION

The problem consists of a set of ships S that have to be built. A ship is constructed from a set of components. Ships are produced in yards and ship components are stored in depots.

The set J contains all yards and the set D contains all depots. For each ship $s \in S$, the production yard j_s is given. The set $S_j \subseteq S$ contains all ships to be produced in yard $j \in J$. A yard $j \in J$ can only order components from depots $D_j \subseteq D$. Depot 0 represents ordering from manufacturers.
The problem is modeled in discrete time with T being the set of time periods. A ship $s \in S$ can be constructed in all time periods after the order time t_s. These time periods for ship $s \in S$ are contained in set $T_s \subseteq T$. This also serves as the time periods when components can be ordered for ship $s \in S$. The reason for this is that it is assumed that a ship order is not known until it has arrived. Therefore, ordering components before the ship is ordered is forbidden.
The duration of the construction of ship $s \in S$ is f_s time periods. The lead time when ordering from depot $d \in D_j$ to yard $j \in J$ is l_{dj}. After ordering, the depot restores its base inventory by ordering the same amount. This takes λ_d time periods.
The binary decision variable X_{st} defines the starting time, being equal to one if the construction of ship $s \in S$ starts in time period $t \in T_S$, and zero otherwise. The variable Y_{sdit} defines the number of components or modules of type $i \in I$ leaving depot $d \in D$ at time $t \in T$, to be used for the construction of ship $s \in S$. The inventory levels are defined by the decision variables Z_{dit}. Z_{dit} defines the level of component or module type $i \in I$ in depot $d \in D$ at the start of time period $t \in T$.
The cost function, Expression 1, is based on the start time of construction for each ship and the holding costs for each component as base-stock. The cost of starting the construction of ship $s \in S$ in time period $t \in T_s$ is given by c_{st}. Since the construction time f_s is constant, this is equal to setting the cost for finishing the construction of ship $s \in S$. The cost per component or module type $i \in I$ in base stock is equal to b_i, and the total base stock is the number of components $i \in I$ in a depot at time 0, plus the number of components which are directly shipped at time 0.

$$\min \sum_{s \in S} \sum_{t \in T_s} c_{st} X_{st} + \sum_{d \in D \setminus \{0\}} \sum_{i \in I} \left(Z_{di0} + \sum_{s \in S} Y_{sdi0} \right) b_i \tag{1}$$

Constraint 2 requires all ships to be constructed. Constraint 3 imposes that for all ships produced at a certain yard, only one can be produced simultaneously. This is a simplification, as in reality the production process of multiple ships overlap. However, for a broad initial planning, it can be assumed sequentially.

$$\sum_{t \in T_s} X_{st} = 1, \qquad \forall s \in S \tag{2}$$

$$\sum_{s \in S_j} \sum_{t'=t-f_s+1}^{t} X_{st'} \leq 1, \qquad \forall t \in T, j \in J \tag{3}$$

Ships are built from components. The set I_c contains all component types. The number of components required for each ship is defined by q_{sia}. This defines the number of components of type $i \in I_c$ needed for ship s, at a time periods after starting construction. The time passed after starting construction is called the *relative time*. It is also possible to require components before the start of construction by setting $q_{sia} > 0$ for $a < 0$. This is illustrated in Figure 2.

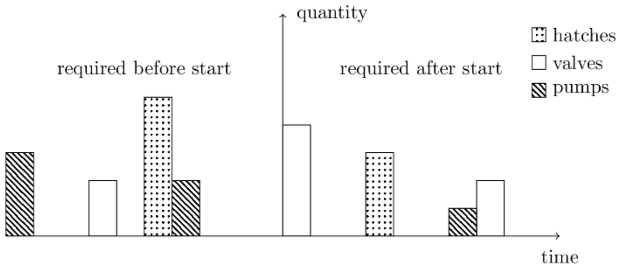

Figure 2: Component requirements for a single ship

Instead of constructing a ship fully from components, it is also possible to use modules. Modules are pre-constructed sets which can be installed instead of using a set of separate components. The physical *modules* are contained in the set I_m with each $i \in I_m$ representing a module kept in stock. However, the effect of using a module can differ. The same module $i \in I_m$ can, depending on in which ship it is installed and at which point in the construction process, replace different components at different times.
For example, consider a ship where a module $i \in I_m$ is placed in a spatially restricted location. Later in the construction process, the rest of the ship might block the path to the installation location, so the module has to be installed relatively early. Loose components are smaller, and therefore, might be installed later. Another ship that uses the same module might not have these restrictions, and therefore, the same module $i \in I_m$ might be needed much later, thus, providing more benefit based on component requirements. It is also possible that certain modules include functions not required on a ship, but which are included to maintain the generality of the modules. This means that the same module $i \in I_m$ might replace a slightly different set of components on one ship compared to another.
For this reason, *module options* are introduced. A module option $m \in M$ represents the possibility to use a certain module $i \in I_m$ at a certain ship and at a certain time in the construction process. The binary variable W_m is equal to one if module option $m \in M$ is used, and zero otherwise. The components inside module $m \in M$ are denoted by the parameter r_{mia}. If module $m \in M$ is used, then at each relative time $a \in A_s$ of the construction process, the number of components of type $i \in I_c$ needed is reduced by r_{mia}. The set M_s contains all module options for ship $s \in S$, and the set M_{sia} represents all module options of module type $i \in I_m$, for ship $s \in S$, which can be used at time $a \in A_s$.
Components and modules are to be ordered at depot $d \in D$ for yard $j \in J$ exactly l_{dj} time periods before they are

required. If ordering in advance is allowed, the optimal solution might 'cheat' the requirement of not allowing component orders before a ship order arrives. Consider two ships 1 and 2 that both need components of type $i \in I_c$, and that are produced in the same yard with $t_1 < t_2$. If ship 1 has some time between the order time t_1 and the start of construction, it might order component type $i \in I_c$ earlier than required such that the depot is restocked in time for ship 2.

The component requirements are set by Constraint 4. First, A_s is introduced, representing all relative times when $q_{sia} > 0$, for any $i \in I$. The variable a_{s0} defines the earliest relative time in A_s. The first term on the left hand side represents all orders for ship $s \in S$ of component $i \in I_c$, that arrive in time for relative time $a \in A_s$. The second term is a cancellation term, which makes sure that the constraint is always satisfied when $X_{st} = 0$. When $X_{st} = 1$, the second term reduces to zero. Constraint 4 is thus only relevant when $X_{st} = 1$. The first term on the right hand side equals the required components at relative time a. The last term represents the reduced requirements due to modules being used. Similarly, Constraint 5 requires enough modules to arrive. The first term of this constraint does not include depot 0, since modules are to be taken only from depots and not from manufacturers.

$$\sum_{d \in D_{j_s}} Y_{sdi(t-l_{dj_s}+a)} + (1 - X_{st}) q_{sia} \geq q_{sia} - \sum_{m \in M_s} r_{mia} W_m, \quad \forall s \in S, t \in T_s, a \in A_s, i \in I_c \tag{4}$$

$$\sum_{d \in D_{j_s} \setminus \{0\}} Y_{sdi(t-l_{dj_s}+a)} + (1 - X_{st}) |M_{sia}| \geq \sum_{m \in M_{sia}} W_m, \quad \forall s \in S, t \in T_s, a \in A_s, i \in I_m \tag{5}$$

Constraint 6 and 7 update the inventory at each time period by removing the ordered and adding the re-ordered components. The moment of ordering new components is done at the time of the arriving ship order. Although this is not always the same time as the component is ordered by the yard, it represents the earliest possible time that a depot can know that a certain component will be shipped, and therefore, anticipate for potential shortages. Furthermore, Constraints 8 to 11 represent all decision variables.

$$Z_{dit} = Z_{di(t-1)} - \sum_{s \in S} Y_{sdit}, \quad \forall d \in D \setminus \{0\}, i \in I, t \in T \setminus \{0\}, t < \lambda_d \tag{6}$$

$$Z_{dit} = Z_{di(t-1)} - \sum_{s \in S} \left(Y_{sdit} + Y_{sdi(t-\lambda_d)} \right), \quad \forall d \in D \setminus \{0\}, i \in I, t \in T \setminus \{0\}, t \geq \lambda_d \tag{7}$$

$$X_{st} \in \{0,1\}, \quad \forall s \in S, t \in T_s \tag{8}$$

$$Y_{sdit} \geq 0 \text{ and integer}, \quad \forall s \in S, d \in D_{j_s}, i \in I, t \in T_s \tag{9}$$

$$Z_{dit} \geq 0 \text{ and integer}, \quad \forall d \in D, i \in I, t \in T \tag{10}$$

$$W_m \in \{0,1\}, \quad \forall m \in M \tag{11}$$

This model can be seen as the optimization of one known scenario of ship orders. In Section 5, the model is expanded to incorporate stochastic ship orders by using the SAA-method.

5. SAMPLE AVERAGE APPROXIMATION

The model introduced in Section 4 optimizes one scenario. However, since the base stock values have to be determined before the scenario is known, it is desired to have a base-stock policy that has the lowest expected cost for all possible scenarios. For this, a generator is used. A generator is a stochastic function, that produces a single scenario of ship orders.

Consider a set of scenarios Λ, a set of possible base inventories Φ, and an optimal cost $c_\lambda(\varphi)$, which minimizes the costs in Expression 1 for scenario $\lambda \in \Lambda$ while applying base inventory $\varphi \in \Phi$. Now, the optimal base inventory φ_Λ^* for scenarios Λ is defined as the base inventory that minimizes the average value over all scenarios:

$$\frac{1}{|\Lambda|} \sum_{\lambda \in \Lambda} c_\lambda(\varphi_\Lambda^*) \leq \frac{1}{|\Lambda|} \sum_{\lambda \in \Lambda} c_\lambda(\varphi), \quad \forall \varphi \in \Phi. \tag{12}$$

Thus, a single inventory policy is applied to multiple scenarios as illustrated in Figure 3. For infinitely many scenarios produced by the generator, φ^* would be optimal in expectation. The idea behind Sample Average Approximation (SAA) [19] is to randomly generate scenarios, and with an increasing number of scenarios, φ_Λ^* will approach $\varphi_{\Lambda_\infty}^*$, where Λ_∞ represents all possible scenarios.

To apply the SAA method on the inventory problem, a combined problem is created. Each depot and each yard in the original problem is added once per scenario to the combined problem. Each yard from a scenario λ can only receive components from depots corresponding to the same scenario. The ship orders of scenario λ are then set to the corresponding yards of scenario λ.

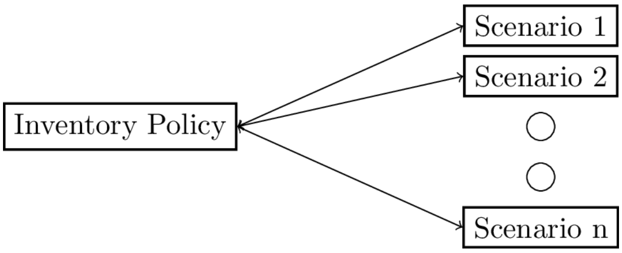

Figure 3: Multiple scenarios

This creates |Λ| separate problems. To link them, the base inventory should be equal. Since each depot in the original problem is added once per scenario, denote D' as the original set of depots, and D'_d as the set of depots in the combined problem which originate from the original depot $d \in D'$. Then, adding Constraint 13 creates the SAA combined problem by setting the base inventories equal across all scenarios.

$$Z_{di0} = Z_{d'i0}, \qquad \forall d \in D', d' \in D'_d \tag{13}$$

6. COMPUTATIONAL RESULTS

The model introduced in Sections 4 and 5 was solved by the ILP solver Gurobi Optimizer and tests were performed on generated data. This was done to evaluate the size of solvable instances and the convergence properties of the SAA method. The instances were run on the hpc cluster of Delft University of Technology. A total memory limit of 32gb was imposed. Each optimization job used 8 cores, which each had a clock frequency of 2.40 GHz.
The required ship types were generated by using the component ordering list of existing shipbuilding projects. Components were sampled from these lists, combined with component commonality and required ordering time. The modules were defined by sampling multiple component types and a time, and a module would then exists for all selected components required before the selected time. The costs were then hand fitted in order to guarantee a computational challenge, i.e., not have the optimum at either ordering each part or having everything in stock. The considered time period is 3 years, divided into monthly intervals.

The randomly generated tests were varied on the number of components, modules, yards and scenarios. For each combination of these amounts, five random instances were generated and optimized with a time limit of 3 hours. An instance *x* is considered larger than instance *y* if there is no property (modules, yards, etc.) in instance *y* with a higher value than the property in instance *x*, and at least one property of instance *x* is larger than the property in instance y. In Table 1, all solved instances which did not have a larger solved instance are shown with the average solving time. The same concept, but reversed, applies to the property that an instance is smaller than another. In Table 2, all unsolved instances are shown which do not have a smaller unsolved instance.

Components	Module options	Scenarios	Yards	Avg. Time [s]
10	0	5	8	2412
10	0	10	3	216
10	506	8	3	790
14	0	5	5	2249

Table 1: Largest solvable instances

Components	Module options	Scenarios	Yards
5	0	5	5
10	0	3	5
10	0	5	3

Table 2: Smallest unsolvable instances

In total, 93 instances were solved, and 189 were not. It can be seen that size is not the only factor determining the difficulty, as there are solved instances larger than some unsolved, as can be seen in Table 1 and 2. However, these results give some measure for the size of solvable instances.

In practice, ships have hundreds of components that need to be taken into account in the inventory optimization [20]. Therefore, although a proof of concept of the optimization model is given, the solvable instances at this moment are too small for real life applications. However, it must also be noted that since this is a strategic problem, 3 hours is relative short, as even a few weeks solving time might be allowed.

Furthermore, the convergence of the SAA method is evaluated. This is done in the following way. First, set u = 1, and solve u scenarios. Denote the objective as OBJ_u and the resulting base inventory as $\theta(u)$. Now, increase u by one and add a scenario. Solve this once normally, and once with the base inventory from the previous step fixed. Call the objective of the latter $OBJ_u^{\theta(u-1)}$. The variation metric Δ_u is then introduced by Equation 14. This represents the relative cost of using the base inventory from the previous iteration, compared to the cost of using the optimal base inventory. A robust base inventory should not worsen too much if more scenarios are added, and therefore, a low Δ_u represents a robust base inventory.

$$\Delta_u = \frac{OBJ_u^{\theta(u-1)} - OBJ_u}{OBJ_u^{\theta(u-1)}}$$

Eq. 1

This was repeated for 10 random problems, which were optimized for 10 scenarios. The average Δ_u values are shown in Figure 4. Here, a clearly decreasing trend is visible.

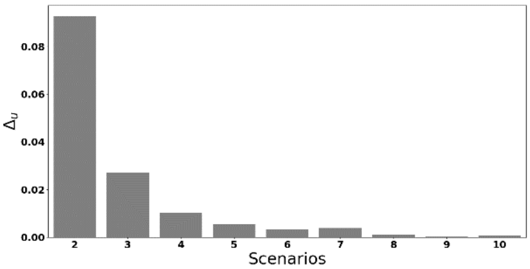

Figure 4: Average variation when adding more scenarios

7. CONCLUSIONS

This paper gives a first mathematical model for the base stock inventory problem modified for the shipbuilding industry. Along with this, initial tests were done to evaluate the performance of directly solving the ILP. These tests conclude that only relatively small instances are solved, and therefore, more work is required to optimize real world instances.

The current model captures multiple aspects: Scheduling, base inventory, component flow and modularization. Although all these aspects are certainly connected, it creates a very large model. Therefore, a possible step for future research is to split these aspects and optimize a subset of them.

However, it is also interesting to expand the model. One possibility for this is the assumption that all construction happens sequentially. Instead of this assumption, it would be interesting to take into account yard resources and the requirement of these during the construction process. On a related note, the current model for the construction process is very simplified. Expanding this to a more realistic process will result in a better representation of the benefits of modularity. Finally, in reality, there is the option of de-assembling or modifying modules, which can be a good alternative for keeping multiple modules in stock. Therefore, the model might be generalized to include this as well.

From an analytic point of view, it would be interesting to analyze the complexity of the problem and the stochastic properties. The results suggest convergence of the SAA method for increasing number of scenarios, but no guarantee can be given at this point. Finally, the solution method has to be improved to solve larger instances. The direction of this will be a result of the complexity analysis, possibly being a (meta)heuristic method, or an optimal approach.

8. ACKNOWLEDGEMENTS

The authors would like to thank all partners of the NAVAIS project for assistance during this research. The EU H2020 framework sponsors the NAVAIS project.

9. REFERENCES

[1] M. Semini, D. E. G. Haartveit, E. Alfnes, E. Arica, P. O. Brett, and J. O. Strandhagen, “Strategies for customized shipbuilding with different customer order decoupling points,” *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.*, vol. 228, no. 4, pp. 362–372, Nov. 2014.

[2] T. M. A. Ari Samadhi and K. Hoang, “Shared computer-integrated manufacturing for various types of production environment,” *Int. J. Oper. Prod. Manag.*, vol. 15, no. 5, pp. 95–108, May 1995.

[3] Z. Atan, T. Ahmadi, C. Stegehuis, T. de Kok, and I. Adan, “Assemble-to-order systems: A review,” *Eur. J. Oper. Res.*, vol. 261, no. 3, pp. 866–879, 2017.

[4] M. H. Mello and J. O. Strandhagen, “Supply chain management in the shipbuilding industry: Challenges and perspectives,” *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.*, vol. 225, no. 3, pp. 261–270, 2011.

[5] Z. Liu, D. K. H. Chua, and K. W. Yeoh, “Aggregate production planning for shipbuilding with variation-inventory trade-offs,” *Int. J. Prod. Res.*, vol. 49, no. 20, pp. 6249–6272, 2011.

[6] I. Zennaro, S. Finco, D. Battini, and A. Persona, “Big size highly customised product manufacturing systems: a literature review and future research agenda,” *Int. J. Prod. Res.*, vol. 0, no. 0, pp. 1–24, Mar. 2019.

[7] N. Agrawal and M. A. Cohen, “Optimal material control in an assembly system with component commonality,” *Nav. Res. Logist.*, vol. 48, no. 5, pp. 409–429, 2001.

[8] Y. Akçay and S. H. Xu, “Joint Inventory Replenishment and Component Allocation Optimization in an Assemble-to-Order System,” *Manage. Sci.*, vol. 50, no. 1, pp. 99–116, Jan. 2004.

[9] Y. Lu and J.-S. Song, “Order-Based Cost Optimization in Assemble-to-Order Systems,” *Oper. Res.*, vol. 53, no. 1, pp. 151–169, 2005.

[10] D. J. Thomas and D. P. Warsking, “A periodic inventory model for stocking modular components,” *Prod. Oper. Manag.*, vol. 16, no. 3, pp. 343–359, 2007.

[11] M. K. Dogru, M. I. Reiman, and Q. Wang, “A Stochastic Programming Based Inventory Policy for Assemble-to-Order Systems with Application to the W Model,” *SSRN Electron. J.*, vol. 58, no. 4, pp. 849–864, 2007.

[12] M. Albrecht, “Determining near optimal base-stock levels in two-stage general inventory systems,” *Eur. J. Oper. Res.*, vol. 232, no. 2, pp. 342–349, 2014.

[13] W. van Jaarsveld and A. Scheller-Wolf, “Optimization of Industrial-Scale Assemble-to-Order Systems,” *INFORMS J. Comput.*, vol. 27, no. 3, pp. 544–560, Aug. 2015.

[14] Q. Hu, J. E. Boylan, H. Chen, and A. Labib, “OR in spare parts management: A review,” *Eur. J. Oper. Res.*, vol. 266, no. 2, pp. 395–414, 2018.

[15] L. K. Nozick and M. A. Turnquist, "A two-echelon inventory allocation and distribution center location analysis," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 37, no. 6, pp. 425–441, 2001.

[16] W. van Jaarsveld, T. Dollevoet, and R. Dekker, "Improving spare parts inventory control at a repair shop," *Omega*, vol. 57, no. October 2017, pp. 217–229, Dec. 2015.

[17] R. J. I. Basten and G. J. van Houtum, "System-oriented inventory models for spare parts," *Surv. Oper. Res. Manag. Sci.*, vol. 19, no. 1, pp. 34–55, 2014.

[18] Y. Yang and Y. Du, "Multi-objective optimization model for multi-echelon spare parts supply system under uncertain circulation," in *2017 Second International Conference on Reliability Systems Engineering (ICRSE)*, 2017, no. Icrse, pp. 1–5.

[19] A. J. Kleywegt, A. Shapiro, and T. Homem-de-Mello, "The Sample Average Approximation Method for Stochastic Discrete Optimization," *SIAM J. Optim.*, vol. 12, no. 2, pp. 479–502, 2002.

[20] V. Dixit, R. K. Srivastava, and A. Chaudhuri, "Project network-oriented materials management policy for complex projects: A Fuzzy Set theoretic approach," *Int. J. Prod. Res.*, vol. 53, no. 10, pp. 2904–2920, 2015.

10. AUTHORS BIOGRAPHY

Tom van der Beek holds the current position of PhD candidate at Delft University of Technology. His research area is inventory optimization in shipbuilding.

Theresia van Essen holds the current position of Assistant Professor at Delft University of Technology. Her research focuses on developing new models and novel optimization based methods for problems from practice in the field of transportation and health care.

Jeroen Pruyn holds the current position of Assistant Professor at Delft University of Technology. He is responsible for the field of Maritime Operations and Management, both the education and research. His interests are primarily focused on the building, operating and recycling of flexible ships with low impacts. His previous experience includes Green Ship Recycling, Shipyard Simulation & Optimization, Systems Engineering for ship design, etc.

Karen Aardal holds the current position of Full Professor at Delft University of Technology. She is responsible for Optimization within Delft Institute of Applied Mathematics. Her research interests include integer optimization algorithms, approximation algorithms, applications of machine learning, and the modeling and resolution of problems within logistics and health care.

Hans Hopman holds the current position of Full Professor in Ship Design and is the head of the Department of Maritime and Transport Technology at Delft University of Technology. His research focusses on the improvement of the design process for complex ships and marine systems integration. He started his academic career in Delft in 2006 after having worked nearly 25 years for the Directorate Materiel Royal Netherlands Navy where he was involved in many naval procurement programs.