1. Introduction

In recent years, small lot transportation and just-in-time delivery have come about due to diversification of user needs. Therefore, it becomes necessary to improve the efficiency of the shipment procedure in the iron and steel industry. On the other hand, the inheritance and the standardization of operator’s know-how has become a problem due to the decrease in the number of skilled operators. Especially, in the allocation planning of steel products in the hold of a ship, because the restriction which should be considered is complex and the problem scale is very large, planning systematization was difficult in the past. In this paper we study a scheduling algorithm of the allocation problem of steel products such as strip coils and sheets in the hold of a ship. Since steel products are usually stowed hierarchically in a hold, this planning becomes a three-dimensional allocation problem. First, we express the stowing-position of the products as three-dimensional integer coordinates, and model them as a combinatorial optimization problem. Next, the weight balance of the ship, loading ratio, and work efficiency of stowing are evaluated quantitatively, and they are optimized by applying the SA (Simulated Annealing) method. To obtain a preferable solution in practical time, we take the stowing know-how which the operators have in the evaluation function, and can make the search process efficient. In addition, to confirm the effectiveness of the proposed algorithm, several case studies have been carried out with actual data. Through the case studies, it has been found that the evaluation function involving the operation know-how term could reduce more unloaded coils assuring qualities of other evaluation items than the best solution obtained by the evaluation function without know-how term.

KEY WORDS: strip coils; shipment; allocation planning; combinatorial optimization; simulated annealing; automation; simulation.
search process efficient. The effectiveness of the algorithm is shown by numerical experiment using actual data.

2. Outline of Hold Allocation Planning

In the iron and steel works to which we apply the allocation planning method, about 30 ships transport steel products such as steel coils and sheets for the users and coil centers. Every ship carries more than 200 steel coils of various sizes. For the shipment, it is necessary to appropriately allocate these coils in the hold. In the allocation, the first condition that should be considered is to load all coils specified for the shipment. There exist other kinds of conditions, to be considered as follows:

(1) Dimensions of Hold

Figure 1 shows an example of the hold of a ship. A hold is composed of two or more blocks, so piles of stowed coils can not exceed the dimensions of a block, i.e. the width, length and height. Some trestles lie on the floor of each hold with different width according to the block. Therefore, unloadable blocks may exist for some coils, according to their diameter.

(2) Stability of Coils

To prevent collision of coils during transportation, there exists a lower restriction for the distance between adjacent coils as shown in Fig. 2(a). In addition, to guarantee the stability of piled coils under navigation, the size of the coils in the upper row have upper restrictions to those of the coils allocated under them as shown in Fig. 2(b). That is, the width of the upper coil can not exceed the maximum width of the lower two coils, and the bottom of the upper coil must sink more than a certain ratio from the top of the lower coil with a smaller diameter.

(3) Work Efficiency of Stowing

To reduce the moving distance in the horizontal direction of a crane, it is necessary to allocate the coils loaded on the same palette to as near a place as possible. Figure 3 shows movement of the crane from palettes to hold.

(4) Safety of Ship during Navigation

For the ship stability, it is necessary to maintain load balance of back-to-forth and right-to-left. That is, the decline in the right-to-left direction (rolling) must be 0 and that in the back-to-forth direction (pitching) must be a target angle set for stability in running as shown in Fig. 4. Usually the target inclination in the back-to-forth direction is given as the variance between the draft at the bow and the stern of a sailing situation, which is called trim.

Usually the allocation planning is determined manually by human experts based on their experiences. As mentioned above, since the conditions of this planning are complex and the number of coils is large, the problems left unsolved
are as follows:
1) Some coils remain unloaded.
2) A long time is required to adjust the balance of a ship.
3) The loading time becomes longer because a crane may move extra distance.

3. Modeling of Coil Allocation in a Hold

3.1. Decision Variables

Usually, coils are allocated in double layers in a hold and placed in rows in the right-to-left direction. Therefore, to describe the allocation of a coil in the hold, it is necessary to determine a vector \( V_i \) in Eq. (1) with four integer-variables:

\[
V_i = (a_i, x_i, y_i, z_i)
\]

where \( a_i \) is a block index of product \( i \), \( x_i \) is a column index, \( y_i \) is a row index, and \( z_i \) is an index that represents upper \((z_i = 1)\) or lower \((z_i = 2)\). \( N_a \) and \( n_i \) are the number of areas and the number of columns in the area respectively. \( N_y(a_i) \) is the number of trestles in the area \( a_i \), \( N_x(a_i) \) and \( N_z(a_i) \) are constants concerning each ship, but \( n_i \) changes depending on the width of each row which is defined as the maximum width of coils in a row. We also use virtual coordinates to express the unloaded coils. That is, a coil is considered to be unloaded, if \( V_i \neq V_j \) if \( i \neq j \).

Three-dimensional coordinates are calculated based on \( V_i \) and the constraints between adjacent coils. Then, the balance of a ship is calculated using the coordinates and the weight of each coil and the balance data sheet of a ship.

3.2. Evaluation Function

To obtain the optimum solution of the allocation problem, we defined several kinds of criterion functions as follows:

(1) Balance of a Ship
Roll and Pitch of a ship in both right-to-left direction and back-to-forth direction is to be minimized. To represent the goodness of balance of a ship, we define \( E_1 \) as shown in Eq. (2).

\[
E_1 = d|t - \bar{t}|, \quad \alpha > 0 \quad \text{...............(2)}
\]

where \( d \) is inclination degree to starboard, \( t \) is difference of hull depth under the surface of the water between stern and prow (trim), \( \bar{t} \) is a target of \( t \), and \( \alpha \) is a scaling parameter.

(2) Total Number of Unloaded Coils
The number of coils left unloaded is to be minimized. To represent the smallness of the number of unloaded coils, we define \( E_2 \) as shown in Eq. (3).

\[
E_2 = M - \sum_i \min\{1, a_i\} \quad \text{...............(3)}
\]

where \( M \) is the total number of coils.

(3) Total Distances of Crane Movement
The sum of the distances of the crane movement is to be minimized. To represent extra movement of the crane simply, we use frequency of the movement between blocks. If a pallet \( p \) will be placed in front of the block \( B_p \) that is used most frequently in the pallet, the frequency of movement between different blocks is assumed as the number of coils that cannot be allocated in block \( B_p \). So we define \( E_3 \) as shown in Eq. (4).

\[
E_3 = \sum_p \sum_{a \in B_p} \min\{1, a\} \quad \text{...............(4)}
\]

These criterion functions are combined in the form of a weighted linear summation. The resulting value is estimated in the optimization algorithm. Each variable to be determined is a three-dimensional integer. Therefore the problem is a type of integer programming and the combinatorial optimization algorithm is employed to obtain the solution.

4. Optimization Algorithm for Coil Allocation Problem

4.1. Application of SA Method

We use the Simulated Annealing (SA) method\(^8\) to search the optimal allocation of coils in a hold. The SA method is well-known as one of the search algorithms that can give a quasi-optimal solution in practical time for a large-scale combinatorial optimization problem. In addition the SA algorithm is very simple and its programming and operation are easy even for complex problems. These are important factors for continual operation of the scheduling system in the field.

The process flow of SA algorithm is shown as follows.

1) Generation of a initial solution
A feasible solution is generated randomly and registered as the first provisional solution with its evaluation value.

2) Setting initial temperature
The temperature of the initial state that becomes the starting point of annealing is set. This starting temperature is usually high enough.

3) Neighborhood creation
The provisional solution is changed randomly to another solution in its neighborhood. This operation is made by modifying a some of the decision variables of the provisional solution. The new solution given here becomes the next solution candidate.

4) Calculating evaluation value of the candidate
Evaluation value \( E \) is calculated according to a given function.

5) Judgement of updating the provisional solution
The candidate solution is accepted with probability \( P_a \) as

\[
P_a = \begin{cases} 
1 & \text{(if } \Delta \leq 0) \\
\exp\left(-\frac{\Delta}{T}\right) & \text{(if } \Delta > 0) \
\end{cases} \quad T > 0 \quad \text...............(5-1)
\]

\[
\Delta = E - \bar{E} \quad \text...............(5-2)
\]
where $E$, $E\tilde{E}$ and $T$ denote the evaluation value of the candidate solution, that of the provisional solution and the current temperature, respectively. If the candidate solution is accepted, it becomes the new provisional solution.

6) Judgement of equilibrium condition
Based on the number of times of creating candidates and the number of times of accepting them, the equilibrium condition in temperature $T$ is judged. If the judgement result is equilibrium, go to 7), otherwise go back to 3).

7) Judgement of convergence condition
Based on the temperature $T$ or the total iteration number, convergence condition is judged. If the convergence condition is satisfied, SA process is completed, otherwise go to 8).

8) Updating temperature
Temperature parameter is changed to a lower value. That is, $T$ is updated to $\rho T$ ($0<\rho<1$). Then go back to 3).

4.2. Generation of an Initial Solution
Since SA is one of the single-point search algorithms, the initial solution influences calculation time for optimal solution search. Total number of unloaded coils is the most important factor in this problem. Moreover, as mentioned below, it is the most difficult to decrease unloaded coils in SA algorithm. Therefore it is necessary to generate an initial solution which reduce unloaded coils as much as possible in a short time. To obtain such an initial solution, we used a heuristic method. In this method, allocation position of each coil is determined sequentially based on the priority of blocks and that of coils. The priority setting method is shown as follows:

a) Priority of blocks
For each hold block $k$ ($k=1,2,\ldots,N_o$), we defined the index $F_b(k)$ to express easiness of stowing as shown in Eq. (6),

$$F_b(k) = \frac{C_k}{N_y(k)L_k} \sum_{i\in\Omega} \frac{N_y(i)L_i}{L_k}$$  \hspace{1cm} (6)

where $C_k$, $N_y(k)$ and $L_k$ are the number of coils that can be allocated in block $k$, the number of trestles in block $k$ and the back-to-forth length of block $k$. That is, the block that has smaller $F_b(k)$ has more difficulty in stowing, so it is selected with higher priority.

b) Priority of coils
For each coil $i$ ($i=1,2,\ldots,M$), we defined the index to express easiness of stowing as shown in Eq. (7),

$$F_c(i) = \sum_{k\in\Omega} \frac{N_y(k)L_k}{L_i}$$  \hspace{1cm} (7)

where $\Omega$ is the set of blocks in which coil $i$ can be allocated. That is, the coil that has smaller $F_c(i)$ has more difficulty in stowing, so it is allocated with higher priority. If two or more coils have same value of $F_c(i)$, the heavier coil is allocated with higher priority, because the heavier coil is more difficult to be stowed on upper layer.

The lower rows are created according to the priority defined in a). Then, the coils are retrieved according to the priority defined in b), and selected if their diameters are allowed in the block. Next, the coils that can be stowed on the upper rows are retrieved in the order of weight, and selected if their width and diameters are allowed to those of the lower two coils. Thus the blocks and the coils which are difficult to be stowed are selected with higher priority, so more unloaded coils can be decreased in the initial solution compared with selecting allocation position randomly.

4.3. Neighborhood Creation
In the SA algorithm, the neighborhood solution is created by changing the stowage positions of the partial coils. However, exchanging the elements of a decision vector $V_i$ randomly causes occurrence of duplicate coil allocation. Therefore, we use three ways to change the stowage positions as shown in Fig. 5 without operating $V_i$ directly. These are summarized as follows:

a) Replacement between two coils
The allocation vectors of two coils selected randomly are replaced with each other if the stowage state after replacement can satisfy the dimensional restrictions of coils. Replacement between a loaded coil and an unloaded coil is acceptable.

b) Move to empty position of coil
The stowage position of a coil selected randomly is changed to an empty position that is also selected randomly if the dimensional restrictions are satisfied after replacing allocation. The stowage position or an empty position can be that of unloaded coils, so the number

![Fig. 5. Neighborhood creation in SA.](image-url)
of unloaded coils may be changed in this operation.
c) Replacement between two columns
Though the two operations mentioned above are related to coil movement, two columns in the same block are exchanged in this operation. The goodness of balance regarding trim can be changed in this operation, because decline in right-to-left direction, the number of unloaded coils and crane movement between different blocks are not influenced by this operation.

Basically, a way is selected randomly among a), b) and c) according to the probability set beforehand. However, the number of unloaded coils can be decreased only in the case which unloaded coils are moved to empty positions in b). Therefore selecting the way of neighborhood creation and the stowage position randomly increase the calculation time required for convergence. To avoid this problem, we use the following way of neighborhood creation:

1) If there are unloaded coils, the way b) is applied to all of them. If there is an empty position to which they can be moved, neighborhood creation process is completed, otherwise go to 2).

2) A way is selected randomly among a), b) and c) according to the probability set beforehand. Then, the probability of the way c) should be set smaller than the other two ways, because the way c) influences only the evaluation regarding trim.

3) The stowage positions are changed according to the way selected in 2).

4.4. Applying Know-how in Allocation Planning
Since allocation for more than 200 coils must be determined in the stowage planning we are concerned with here, its combination becomes extremely large-scale and a long calculation time is required, even by SA algorithm. Therefore, a new methodology is needed to reduce calculation time. On the other hand, operational know-how exists in the coil allocation planning. For example, the following know-hows are used to increase the number of loaded coils.
a) Gathering coils with similar width in a row may increase the number of rows in a block.
b) Allocation of two coils with similar diameter adjacent-
c) Replacement between two columns

To accelerate solution search, we use these operational know-hows in the following evaluation function E4 which should be minimized;

\[ E_4 = w_1 \lambda_1 E_1 + w_2 \lambda_2 E_2 + w_3 \lambda_3 E_3 + w_4 \lambda_4 E_4 \] ........................(9)

where \( w_1, w_2, w_3 \) and \( w_4 \) are positive weight parameters, and \( \lambda_1, \lambda_2, \lambda_3 \) and \( \lambda_4 \) are parameters to standardize the sensitivity of each evaluation item.

5. Numerical Simulation
To check the effectiveness of the proposed algorithm for our allocation planning of steel products, numerical simulations are carried out using actual operational data. Case studies examined here are as follows:
- Influence of weight parameter for ship balance
- Effect of using operational know-how
- Influence of weight parameter for the number of unloaded coils
- Effect of using operational know-how

In every case study, the number of coils to be stowed is 212, and the loading capacity of the ship is about 1 560 tons. This planning data is an actual shipment result, which has no unloaded coils and almost no deviation from desired values in balance.

(a) Influence of a Weight Parameter \( w_1 \) Concerning Ship Balance
In the evaluation function in Eq. (9), \( w_1 \) is changed from 0 to 1 under the condition that \( w_2, w_3 \) and \( w_4 \) are fixed to 1, 1 and 0, respectively. Figure 6(a) shows variation of the number of unloaded coils and the frequency of movement between blocks for \( w_1 \). In the same way, Fig. 6(b) shows differences of trim and those of inclination degree in the right-to-left direction from their desired values. The following feature is understood in Fig. 6. Both of the evaluation values about balance tend to converge easily when \( w_1 \) increases a little. On the other hand, unloaded coils and movement between blocks continue to increase when \( w_1 \) increases. Therefore, \( w_1 \) should be set much smaller than \( w_2 \) or \( w_3 \). In this case, it is found that \( w_1 \) should be about 0.1 to 0.2.

(b) Influence of a Weight Parameter \( w_2 \) Concerning the Number of Unloaded Coils
In the evaluation function in Eq. (9), \( w_2 \) is changed from 0 to 6. \( w_1 \) is fixed to 0.1 based on the result obtained above. \( w_3 \) and \( w_4 \) are fixed to 1 and 0 as set in case (a), respectively. Figure 7 shows variation of the number of unloaded coils and the frequency of movement between blocks for \( w_2 \). In this case, the maximum value of differences of trim is 0.7 [cm] and that of inclination degree in the right-to-left direction is 0.0 [deg]. The following feature is understood in Fig. 7. The number of unloaded coils cannot be decreased to less than around 4–6 even if \( w_2 \) is increased. On the other hand, crane movements between blocks keep increasing gradually when \( w_2 \) increases. Therefore, there is a limitation to decrease unloaded coils by using weight \( w_2 \) alone. Moreover, setting \( w_2 \) too high causes extra crane movement. In this case, it is found that \( w_2 \) should be about 1 to 2.

(c) Effect for Using Operational Know-how
Through above two case studies, we found that the balance of a ship after shipment can be adjusted easily by the
weight parameter, but that the number of unloaded coils cannot be 0 only by tuning the weight parameter. In this case, we set $w_4$, which is the weight concerning operation know-how, larger than 0 and check the effect, using operation know-how. In the evaluation function in Eq. (9), $w_4$ is changed from 0 to 6. $w_1$, $w_2$ and $w_3$ are fixed to 0.1, 1 and 1 respectively. Figure 8 shows variation of the number of unloaded coils and the frequency of movement between blocks for $w_4$. In this case, maximum value of differences of trim is 1.1 [cm] and that of inclination degree in the right-to-left direction is 0.0 [deg.]. The following is understood in Fig. 8. Unloaded coils are decreased less than the best solution without $w_4$ in case b). That is, more coils can be loaded by adding an evaluation value which does not relate the number of unloaded coils directly to the evaluation function. In the SA process, the reason for this can be explained as follows:

1) Width of coils in each row and the diameter of adjacent coils are arranged even if the number of unloaded coils cannot change.
2) Arranging width of coils in each row creates the spaces in which additional coils can be allocated.
3) Since the width of the adjacent coils and their diameter become close, the amount of space in the upper position where additional coils can be stowed increases.

On the other hand, crane movements between blocks keep increasing when $w_4$ increases. Therefore, we must avoid setting $w_4$ too high. In this case, it is found that $w_4$ should be about 1.

In these case studies, we could not obtain a result in which there is no unloaded coil. There are some cases where the standard restrictions are relaxed according to the situation. These exceptional operations have a certain risk. Moreover, it is difficult to model including them. Therefore, we think it is practical that the optimal solution is determined automatically and small tuning is done in the field.

6. Construction of Planning System and Its Application

Based on the above study, we constructed an allocation-planning system of strip coil products for field test in Kakogawa works. The optimization algorithm is coded by C programming language for an IBM RS/6000 workstation with Power PC 66MHz. Figure 9 shows the outline of our system.

The product data is selected by prior shipment planning system and transferred from the host computer. It includes
the size and weight of each coil. The pallet number and the holding place of each coil are created by a stock control system in the host computer. The ship data is stored in the workstation. It includes information to calculate ship balance, hold dimensions, width of trestles in each block and target of trim. Stowage planning part in workstation calculates optimal allocation of the coils based on various scheduling parameters, which are weight in evaluation function, thresholds of constraint conditions, annealing parameters of SA. Coil allocation designed by stowage planning part is returned to host computer, and provided to the workers, warehouses, cranes and transport vehicles on the shipment berth. Figure 10 shows an example of output display of allocation result.

The maximum calculation time of SA by this system is about 30 min. This calculation time is about 1/3 compared with the planning time by the skilled operator. So we regard it as practicable level. Moreover, as a result of testing in the actual operation, it was observed that loading time was reduced by more than 5% because of a decrease in useless movements and a reduction in the time required to decide allocation position.

7. Conclusion

In this paper, we studied an allocation problem of steel products in the hold of a ship. First, we expressed the allocation position of coils as the integer vector that consists of a block index, row address, column address and upper-lower index, and modeled as a combinatorial optimization problem. Next, SA algorithm is applied to it with an evaluation function that is formulated in a combination of weight balance after shipment, loading efficiency and work efficiency of crane. In addition, we added know-how based evaluation term to the evaluation function. This know-how is that width of the coils in the same column should be arranged and the diameter of the adjacent coils should be close. Through the case studies tuning the weight parameters by using actual data, it was found that the evaluation function involving the operation know-how term could reduce more unloaded coils assuring qualities of other evaluation items than the best solution obtained by the evaluation function without know-how term.

REFERENCES

1) T. Umeda, N. Tomochika, M. Konishi, A. Kitamura, S. Kanamura and S. Takami: CAMP-ISIJ, 10 (1997), No. 5, 1008.
2) R. Lemaster and U. Mertz: Special Interest Group on Computer & Human Interaction, ACM, New York, (1990), 369.
3) F. F.-H. Liu and C.-J. Hsiao: Operations Research Society, 48 (1997), No. 7, 726.
4) T. Terui and E. Fukuda: J.M.E.S.J., 27 (1992), No. 2, 116.
5) T. Sakamoto, S. Komatsu and K. Yasumoto: Kawasaki Steel Giho, 26 (1994), No. 4, 41.
6) E. Yamakawa and S. Fukumura: The 1991 Spring National
7) S. Kirkpatrick, C. D. Galatt and M. P. Vecchi: *Science*, 220 (1983), No. 4598, 671.

8) M. D. Huang, F. Romeo and A. Sangiovanni-Vincentelli: Proc. of IEEE Int. Conf. of Computer-Aided Design, IEEE, Washington, (1986), 381.

9) M. I. Campbell, C. H. Amon and J. Cagan: *Trans. ASME. J. Electronic Packaging*, 119 (1997), No. 2, 106.

10) U. Nagakura, H. Watatani, S. Kanamura, Y. Hirose, T. Koseki and H. Takezaki: *Kobe Steel Eng. Rep.*, 46 (1996), No. 2, 23.