A thesis for the Degree of Master of Engineering

A Study on the Determination of the Objective functions considering the Placement Sequence in the Shipbuilding-customized Nesting Algorithm

Hyebin Lee

Department of Ocean System Engineering

GRADUATE SCHOOL

JEJU NATIONAL UNIVERSITY

2014. 2.

A Study on the Determination of the Objective functions considering the Placement Sequence in the Shipbuilding-customized Nesting Algorithm

Hyebin Lee

(Supervised by professor Il-Hyung Cho)

A thesis submitted in partial fulfillment of the requirement for the degree of Master of Engineering

2014. 02.

This thesis has been examined and approved.

....

Thesis director, Dong-Guk Paeng, Associate Professor, Dept. of Ocean System Engineering

....

Il-Hyung Cho, Professor, Dept. of Ocean System Engineering

....

Chong Hyun Lee, Associate Professor, Dept. of Ocean System Engineering

----

Date

Department of Ocean System Engineering
GRADUATE SCHOOL
JEJU NATIONAL UNIVERSITY

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The importance of the nesting process has been steadily increasing to reduce the consumption quantity of the raw material. In the shipbuilding, the compactness of the resultant layout is also recommended, so that the remnant plates are effectively used for the following nesting process. In this paper, the objective functions of the two main algorithms, the placement algorithm and the order decision algorithm, focuses on two main purposes, to lower the scrap ratio and to obtain the efficiently well-defined remnants out of a raw plate after the nesting process by securing the longer available length of the remnants.

Firstly, a research on the composition of the placement algorithm objective function is carried out by performing various numerical experiments using weighting factor method and targeting the nesting results of ship parts to inspect how it affects scrap efficiency and allocation characteristics. Using various weighting factor sets, the nesting results are evaluated in accordance with the above purposes and compared with each set for each ship part groups. The experiment results show different efficiency depending on the part groups. Thus, it is suggested that the nesting algorithm fitness function should be constructed differently depending on the characteristics of the parts and the needs of the users.

In next research to obtain the most efficient nesting sequence, Simulated Annealing is used. Especially, the ratio of the total area of the parts to be nested and the area of the raw plate is calculated and compared with the user defined threshold so that the resultant layout is evaluated by the selected objective function between the scrap ratio and the available length of the remnant. Through this process, the best nesting sequence and placement is decided. An objective function of the placement algorithm in this research is fixed at the one which gives the reasonably efficient nesting results. From the experiments on the ship blocks, it is founded that the appropriate determination of the objective functions in each algorithm and the interaction between those two functions lead to more improved nesting efficiency.

Chapter 1

INTRODUCTION

1.1 Background

Since the price of raw materials steadily increases every year and forms a large proportion of the production cost, it is well known fact that the manufacturing industries such as aerospace, garment and appliances have made great efforts at development and customization of the nesting algorithm which determines the most efficient layout of the pieces avoiding overlap for cutting them out of a stock-sheet.

Shipbuilding industry, one of the biggest manufacture industries, is no exception. The nesting in shipbuilding industry is processed for each block of the ship in order to reduce the time during the part sorting and delivery procedure. Each block consists of numerous parts and each part has various sizes and shapes. In addition, the nesting results should include the NC codes which contain the cut-path, bevel, lead-in, and lead-out information, so forth. These facts make the nesting more complex than that in other industries.

Generally, the primary aim in the manufacturing industries is to minimize the cost and time in process. In order to satisfy this purpose, objects in the nesting process are to minimize scrap area, to improve the reusability of waste at the end of the stock sheet or to reduce the computing time. Also, cutting time may be considered by minimizing the movement of the cutter in cutting operation.

As a result, the interest in the nesting problem has proportionally increased and many researches have been carried out for decades to improve the quality of the nesting results. Those researches have approached in various ways so as to satisfy their own nesting purposes.

1.2 Literature Review

The first obstacle which researchers face in nesting problems is the geometry. Many researches have approached to the nesting problem of irregular shaped geometry. The effort to represent shapes more accurately and to improve efficiency of the time and cost caused by the complexity of irregular shaped pieces has led to develop the various representation approaches: No-Fit-Polygon (NFP) [1], Pixel Representation [2] and etc. (e.g. Direct trigonometry [3], Phi function [4]). Especially, the general introduction of NFP and Pixel Representation will be presented in section 2.1 General.

In relation to the placement of the parts, Bang [5] and Kang [6] worked for the optimal allocation with the Pixel Representation. Each research proposed its own objective functions which contribute to minimize the waste of the raw sheet. Weng and Kuo [7] focused on the development of the algorithm which decreases the usage length to have the larger remained sheet material with the Pixel Representation and bottom-left filling algorithm which has been usually applied to nesting problems. They put a set of many irregular shapes on a sheet material whose size is relatively larger than the total size of the shapes. Besides, many other considerations on the placement of pieces have continued to suggest the nesting algorithm with Genetic Algorithm, Simulated Annealing, etc.

With regard to the nesting sequence, Babu and Babu [2] used GA to obtain the optimized nesting sequence and rotation angle. Dowsland *et al.* [8] used nine different orderings in their experience. Ryu and Kim [9] and Kim and Lee [10] adopted GA and optimized the allocation sequence satisfying the conditions.

Traditionally and generally in the shipbuilding, however, most of the automatic nesting process arranges the pieces with the sequence depending on the area or the aspect ratio. After this process, the experienced and skilled workers additionally adjust the arrangement of pieces. From the productivity point of view in workings, it is time-consuming work and may not give the most efficient result.

1.3 Specific Aims

In this paper, the criteria for judgment of nesting efficiency with a focus on the quality of the resultant layout as well as the consumption quantity of the raw plates are set as follows. First criterion is the minimum scrap area by placing the parts as many as possible in a raw plate. Second one is the compactness of the resultant layout. Depending on the accuracy of the nesting result at the stage of the raw plate estimation, the remnant plates could be produced at the stage of the detail design. In order to improve the effectiveness of the remaining stock which contributes to reduction of the production cost by being used for next block nesting process, it would be better if the shape of the remnant is close to the useful shape such as the rectangle. Furthermore, the longer the length of the remnant has, the more effectively it can be used.

The nesting algorithm here is structured by two main algorithms: *placement algorithm* and *order decision algorithm*. The former is the algorithm which determines *the location of the pieces* onto the raw plate and the rotation angle of the pieces. The latter is the one to find the optimized *nesting sequence* which satisfies the nesting purpose. Each algorithm has its own objective function and the interaction between two algorithms determines the resultant layout.

The nesting algorithm of my thesis is based on Pixel Representation. To form the objective function of the placement algorithm, we select the terms from the previously suggested objective functions, reconstruct the objective function by adopting weighting factor method to satisfy the nesting purpose of this research and verify whether it is appropriate to the shipbuilding-customized nesting by carrying out the numerical experiments. When it comes to the order decision algorithm, Simulated Annealing (SA) is utilized to find the nesting sequence which boosts the nesting efficiency compared to the traditional sequence.

Hence, this research shows the advancement of efficiency of the nesting results by considering not only the translation and rotation of the parts but the interaction of two main algorithms.

1.4 Thesis Outline

The paper is structured as follows.

Chapter 2 provides the description of the method for representing the geometry in the case of straight lines, curvilinear features and the method for classifying the inside and outside of the geometry. Also, the rotation of the geometry is introduced.

Chapter 3 shows the first main algorithm which is the placement algorithm and its objective function. The numerical experiment and the results are also in this chapter.

Chapter 4 presents the order decision algorithm. This chapter handles Simulated Annealing and the objective function.

Chapter 5 reports the experimental results comparing on the nesting sequences with SA and evaluates those results according to the criteria on which this research focuses.

Chapter 6 concludes this study, and suggests future studies.

Chapter 2

REPRESENTATION OF GEOMETRY

2.1 General

The nesting problem has required powerful geometric tools which can handle geometries of various shapes and sizes. As a result, many researches have been carried out using many methodologies. The well-known techniques are No-Fit Polygon (NFP), which is the resulting polygon which can be found by tracing one shape around the boundary of another [11] (Figure 1), and Pixel Representation, which is an approach that divide the continuous stock sheet into discrete areas [12]. Pixel Representation is not only simple to code, but easy to represent non-convex and complex pieces. Furthermore, this method is reasonably fast when checking the geometric feasibility of the layouts. Since it cannot exactly represent pieces with non-orthogonal edges, Pixel Representation increases in the size of the matrices so as to improve the representation's accuracy by refining the size of the grid unit. As a result, this representation requires greater memory usage and running time for feasibility checks [12]. Although the NFP is an excellent tool for conducting intersection tests between pairs of polygons, it has not been widely applied for two-dimensional packing problems in both the literature and real world manufacturing industries. Undoubtedly this is due to the NFP's complex implementation and the lack of available robust algorithms [11]. In the case of ship parts, many arcs compose the geometric shape and even the circle (e.g.

doubling part) exists. Adopting NFP for these types of geometry requires tremendous time and cost. Also, NFP cannot accurately represent curvilinear features because it approximates curves as straight lines. For these reasons, we adopted the Pixel Representation for representing the geometry of each part which composes of straight lines and curvilinear features.

The geometrical information on the candidate parts is from the generic file of TRIBON M3, the commercial CAD system has been usually used in the shipbuilding [13]. The information includes the x-y coordinates of the vertices for all features and additionally the amplitude in the case of curvilinear features.

2.2 Resolution

The resolution is one of the most important variables of the Pixel Representation method since it determines the computing time and the accuracy of the resultant layout. The higher resolution for improving the quality of the layout requires more computing time and vice versa. Therefore, users have to give higher priority between the time and quality.

The error could be occurred due to rounding off. All the data of geometry from TRIBON M3 is real number. However, the Pixel Representation method rounds off these real numbers to the nearest integers. As the repeated computation by discretizing the value, the error is accumulated and it might result the overlap when the final integer matrices are drafted to TRIBON M3 to obtain the resultant image. For this reason, it is required to compensate the data. Through the several numerical experiments and comparison of the results between this nesting algorithm and TRIBON M3, the following relationship about the error is found (Figure 2.1). Before drafting the nesting results, the compensation value depending on the rotation angle of each piece is added to the result data.

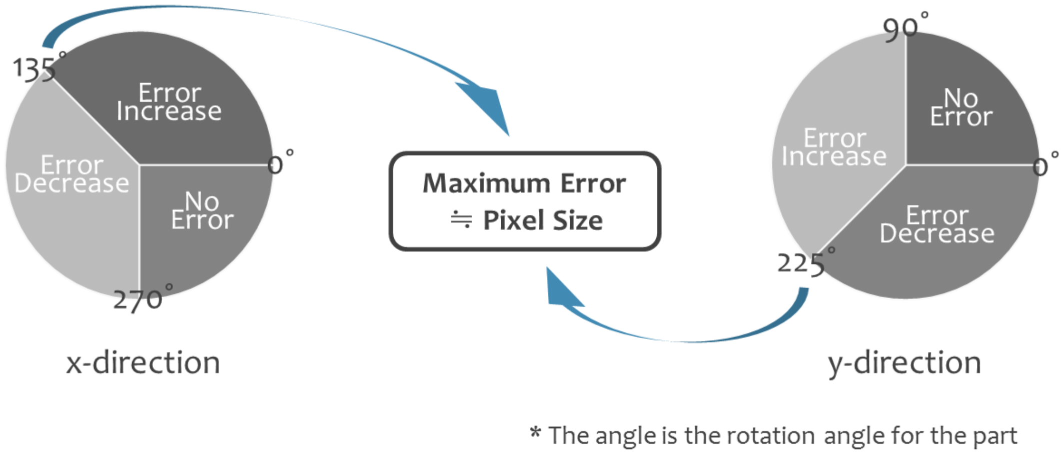

Figure 2.1 Error occurred when drafting the data to TRIBON M3

2.3 Determination of Cell Data

A grid is generated as a matrix having the same width and length as those of the geometry and all pixels are coded as ‘2’. After then, the pixels which represent the contours of the piece and internal features like simple holes or complex geometry are found and coded as ‘1’. Based on the contour pixels, the other pixels are separated into inside or outside of the parts. While the inner pixels are coded as ‘1’, ‘0’ denotes the outside of the piece. Each pixel has the data about the previous and next contour pixel to use for classifying the other cells into the inside or outside of the piece. A matrix for a raw plate is also generated with data ‘2’.

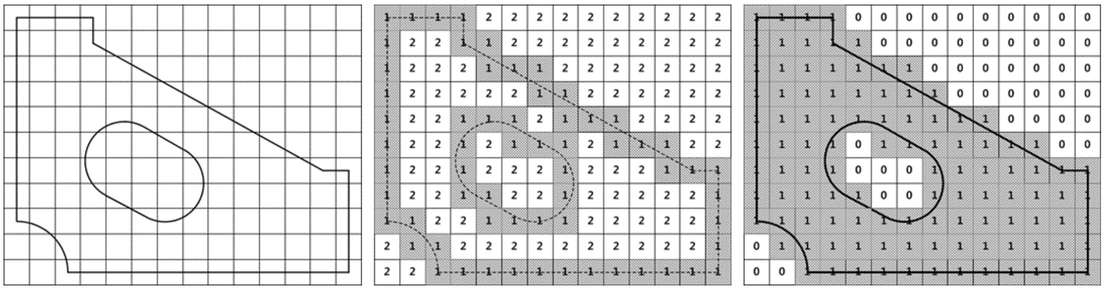

Figure 2.2 Process of Grid representation for a bracket-type part

Polygon consists of lines and curvilinear features. Those types of the features are represented as a grid by the appropriate algorithms. The details of this procedure will be followed. The drawing direction follows counterclockwise.

2.3.1. Straight Lines

Straight lines are codified based on Bresenham’s algorithm [14] excepting the trivial cases which don’t need to utilize Bresenham’s algorithm such as horizontal and vertical lines and the line whose slope is equal to 1. Already, in many computer graphics, it is used as it assures the speed although it produces relatively low quality lines by comparison with most other algorithms [15]. The algorithm performs line drawing computations using only integer additions and comparisons. However, the x-y coordinate sets of two end points are given as the floating number. To compute those sets for determination of the path between the end points, the sets should be rounded off. Given that x-y coordinate of start point is (x_1, y_1), the next point can be either $(x_1 + 1, y_1)$ or $(x_1 + 1, y_1 + 1)$. In short, it is determined based on the midpoints. If the midpoint is below the line, the next pixel becomes the former, and vice versa. To code the relationship between the midpoints and the real line, the process starts from the line equation [16].

$$\mathrm{y} = \frac{dy}{dx}x + c \tag{1}$$

$$\mathrm{F(x,y) = 2xdy - 2ydx + 2cdx} \tag{2}$$

by multiplying the integer '2'. If an arbitrary point (x, y) is on the line, $\mathrm{F(x,y)}$ is zero. In the case that the point is above the line, $\mathrm{F(x,y)}$ is negative. Otherwise, $\mathrm{F(x,y)}$ becomes positive. Substituting the first midpoint $\left(x_1 + 1,\ y_1 + \frac{1}{2}\right)$ into $\mathrm{F(x,y)}$ and using $\mathrm{F}(x_1, y_1) = 0$, we get

$$\mathrm{F}\left(x_1 + 1, y_1 + \frac{1}{2}\right) = \mathrm{2dy - dx} \tag{3}$$

and it is defined this value as 'Decision variable'. If this value is smaller than zero, the northeast pixel is chosen as the next one. If the variable is larger than zero, the next pixel is the east pixel. Depending on which pixel is selected as the next point, the next midpoint can be $\left(x_1+2, y_1+\frac{1}{2}\right)$ or $\left(x_1+2, y_1+\frac{3}{2}\right)$. In order to evaluate the sign of $\mathrm{F}(\mathrm{x}, \mathrm{y})$ for the following midpoints, the decision variable can be obtained by the relationship below.

Assuming that an arbitrary midpoint is (x_m, y_m), the relationship between the decision variable for the midpoint and for the successive midpoint is

$$\mathrm{F}(x_m+1, y_m) - F(x_m, y_m) = 2dy \tag{4}$$

when the current pixel is the east one of the previous pixel. In the case that the current pixel is the northeast one of the previous one, the relationship is

$$\mathrm{F}(x_m+1, y_m+1) - F(x_m, y_m) = 2dy - 2dx \tag{5}$$

By supplementing the value of Equation 4 or 5 to the decision variable until the end of the line, the decision variable can be updated and the pixel representing the line is determined.

After the calculation, the data in the pixels representing the end points and the path is converted ‘1’.

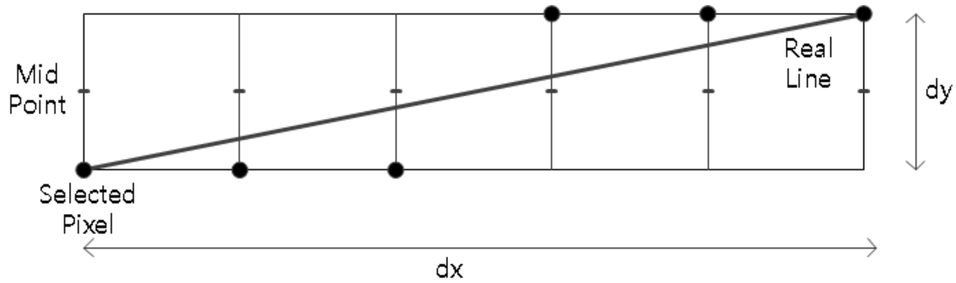

Figure 2.3 Selection of pixels based on midpoints

2.3.2. Curvilinear features

As mentioned before in section 2.1 General, geometric data for each curvilinear feature includes the x-y coordinate sets of two end points and amplitude. Amplitude is given as a vector, as shown in Figure 2.4. If the absolute value of the amplitude is smaller than the user-defined tolerance, the curvilinear feature is approximated to the straight line and represented by Bresenham's algorithm. Otherwise, the feature is generated as a matrix using CIRCLE_ARC_GRID algorithm [17].

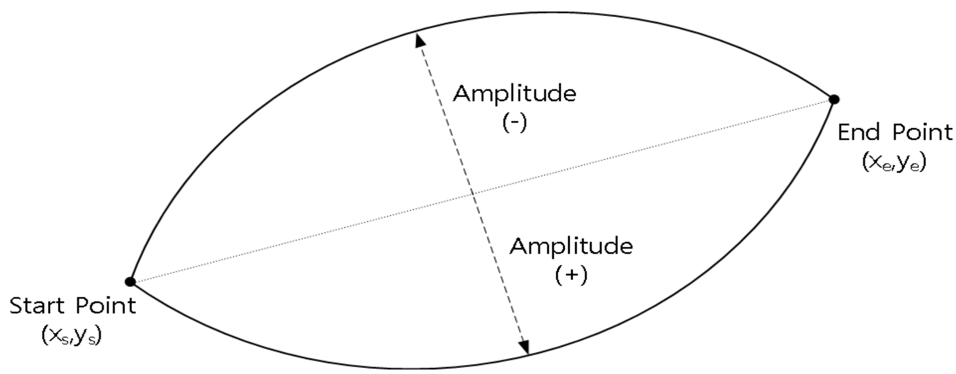

Figure 2.4 Geometric data for curvilinear feature

In detail, the process is as follow. At first, a radius and x-y coordinate set of the center point of the arc are found using the given data. In this step, two center points are found. So, the appropriate set should be selected from the relationship with an amplitude vector. With the coordinate set, each angle between the center and the start point and between the center and the end point is found (Figure 2.5). The grids which represent the curvilinear feature are determined through Equation 6 and 7 by increasing or decreasing '$angle$', which constantly varies from the angle from the start point to the end point.

$$\mathrm{x} = x_c + r \times \cos(angle \times \frac{\pi}{180}) \tag{6}$$

$$\mathrm{y} = y_c + r \times sin(angle \times \frac{\pi}{180}) \tag{7}$$

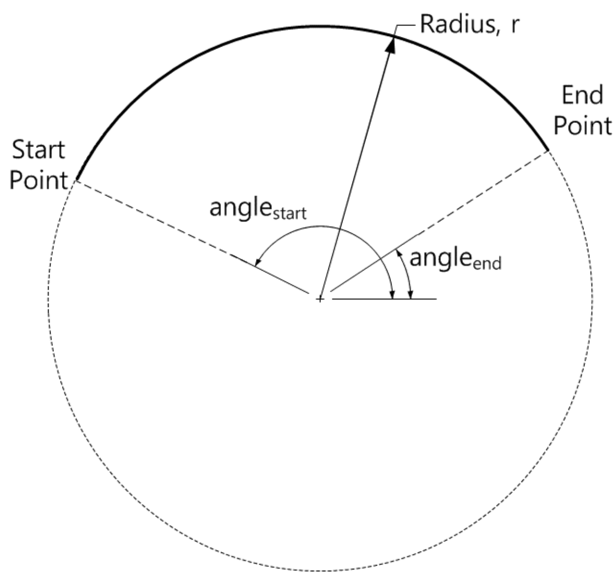

Figure 2.5 Angle of start and end point

2.3.3. Fill Algorithm

Using the contour girds of the polygon, the remained pixels in a grid are classified as the inside or outside of the polygon. The data is set as ‘1’ into the inside or ‘0’ in the case of the outside.

From the leftmost pixel until the grid data is ‘1’ that is the contour, it is converted to ‘0’ which means the pixel is the outside of the piece. The contour pixel firstly met is called ‘$\text{Point}_{\text{ref}}$ 1’ and the last one continuously connected to ‘$\text{Point}_{\text{ref}}$ 1’ in the same row, ‘$\text{Point}_{\text{ref}}$ 2’. If the pixel is not a consecutive contour one in the row, the cross product is calculated to determine whether the pixel is the outside or inside of the polygon. As mentioned before, every contour pixel has the information on the previous and next contour cell as the reference points, A and B in Figure 2.6 because the geometry is generated clockwise. Using the cross product of a vector between one of the $\text{Point}_{\text{ref}}$ and its previous or next cell and another vector between the $\text{Point}_{\text{ref}}$ and the pixel being examined, the pixel data is determined.

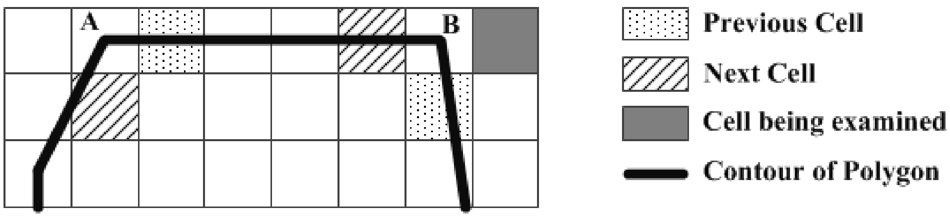

Figure 2.6 Grid Information

2.4 Rotation

As the candidate parts in the shipbuilding have the irregular shape and various size, the resultant layout could be diverse by rotating the piece. From this reason, the rotation of the pieces is required during the placement process in order to raise efficiency of the nesting process. To determine the appropriate rotation angle, the geometry has to be rotated finely from 0 to 360 degree. However, a lot of parts in every block need this process and the enormous computing time for the process is required. The interval of trial rotation angle decides not only the quality of the layout but also the computing time. Therefore, the angle interval for rotating the geometry has to be reasonable.

There are two methods for rotating geometry. First, geometry itself could be rotated around the center of the rotation [6]. The second is to rotate the grid of geometry around the center of the rotation [5]. In this paper, we adopted the first method and it updates the grid information for every rotation angle. Geometric data which includes two x-y coordinate sets for each end point is rotated clockwise on the center of the circumscribed quadrilateral of the candidate part by easily calculating Equation 8 (Figure 2.7). After the rotation process, the process of representing the geometry is repeated from the calculation of the contour.

$$\begin{pmatrix} x_r \\ y_r \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x - x_c \\ y - y_c \end{pmatrix} \tag{8}$$

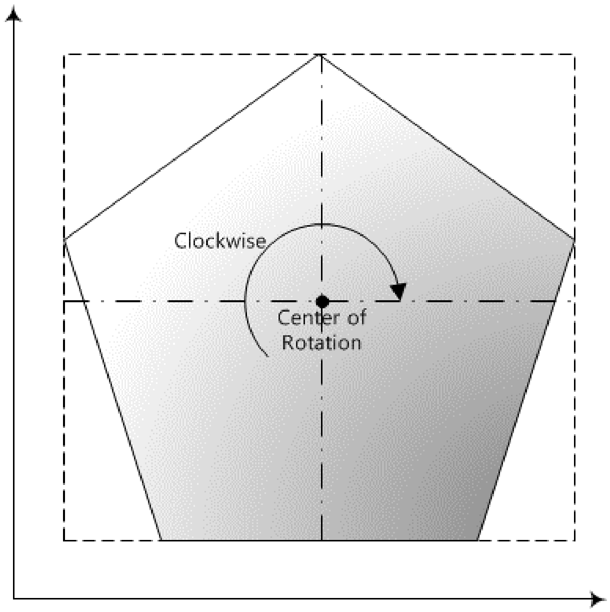

Figure 2.7 Rotation of geometry

After the rotation, the compensation process is necessary. This process translates the rotated geometry to have the minimum values of x and y coordinates to 0 each. This is to reduce memory to be used if the minimum x and y are more than 0 or to prevent the error if minimum values are negative (Figure 2.8).

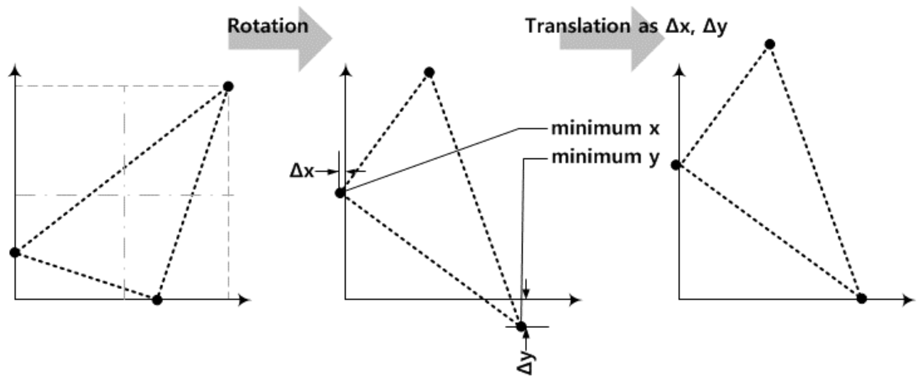

Figure 2.8 Compensation process

Chapter 3

PLACEMENT ALGORITHM

3.1 Location

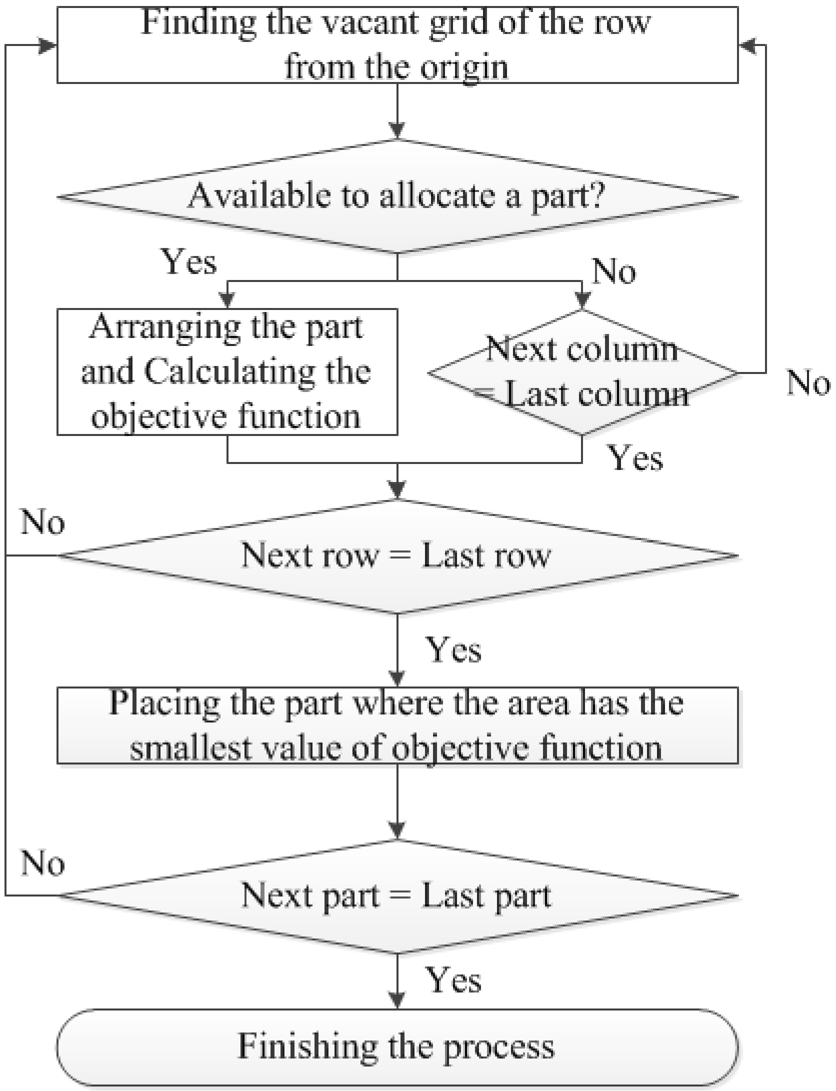

Figure 3.1 Flow chart

The placement algorithm follows the one proposed by Lee and Ruy [18]**.** Placing one piece on the sheet is simply done by adding each swamped cells of two matrices. The whole procedure is shown at Figure 3.1 as a flow chart. To find the feasible area where a part will be nested on the sheet, we consider the area inside the circumscribed quadrilateral including all the parts being already nested (Figure 3.2). The part is temporally placed on closest grid of the first row on the sheet from the origin. The data of each cell in a raw plate is ‘2’ if there is no nested part or ‘3’ if it is occupied by the part. In the case of parts, the data of all the grids is ‘1’ or ‘0’, as mentioned before. If each sum of two corresponding cells in the area is smaller than or equal to ‘3’, which means there is no overlap, the objective function is calculated and the same process is repeated on the following row until the last row of the search area to find other feasible places. If the overlap occurs, the part is moved to the next column of the same row to check the feasibility. The procedure like this also allows the hole-filling which is one of the crucial merits to improve the quality of the nesting layout.

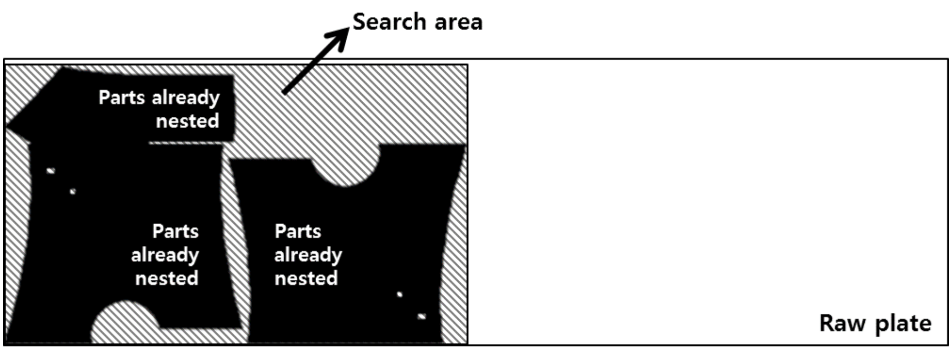

Figure 3.2 Search area for determining location of piece

Every time when the objective function is calculated, the result is compared with the current smallest value. If the result is smaller than the smallest, the value is updated to this result and the location is also updated. Otherwise, the value and location keep the current value and location. Once the process finishes, the part is allocated in the location which has the smallest value of the objective function. The objective function determines the whole layout. For this reason, the objective function must consist of terms which show the purpose of the user correctly.

3.2 Objective functions

As aforementioned, the purposes of this nesting process are to reduce the scrap ratio and to improve the quality and efficiency of the remaining stock after the nesting. Especially, the shape of the remnant would be better if it is rectangle having a long length as much as possible. So as to satisfy these purposes, in this algorithm, five terms are considered for the objective function. These five terms are selected from the previously suggested objective functions. The objective function suggested by Kang [6] reduces the scrap ratio and the one suggested by Weng and Kuo [7] contributes to make the remnant shape efficiently. The weighting factor method is utilized so that we compare the results by changing the weighting factor sets. The weighting factor for each term indicates the importance and contribution in the objective function. All the weighting factors in the objective function add up to one.

min.

$$F = \alpha f_x + \beta f_y + \gamma f_{xy} + \delta Usage\ Length + \varepsilon Usage\ Density$$

$\alpha, \beta, \gamma, \delta, \varepsilon$: Weighting Factor

$$\alpha + \beta + \gamma + \delta + \varepsilon = 1$$

The concept of f_x, f_y and f_{xy} was proposed by Kang [6]. f_x and f_y drive the part to be located on the left-side and on the bottom of the sheet, respectively. To calculate those, in his research, f_x counts the pixels which are left-sided contour of the candidate part and f_y is the whole area under a part which is being nested. f_x and f_y here also act as Kang's, but they are a little differently calculated. In the case of f_x, the pixels only in the circumscribed quadrilateral are counted in x-direction as illustrated in the right of Figure 3.3, and normalized. To calculate f_y here, the number of the pixels is counted from just under a part being nested until the contour pixel of other nested parts or the boundary pixel of a raw plate appears, and normalized. For example, Figure 3.4 shows two candidate locations of the feasible placement positions for a trapezium part- A and B. In the case of A, f_y is zero, whereas case B has f_y as much as the diagonal line area under B. Therefore, placing the part in A would give the best result from the point of f_y only in this example. f_{xy} is the scrap area which is the number of the pixel having no parts in the region which is from the origin until the location of the current candidate piece. It contributes to make the nesting layout similar to rectangle. Figure 3.5 shows the difference of f_{xy} marked as diagonal lined in the case that the candidate piece is located in the position A and B.

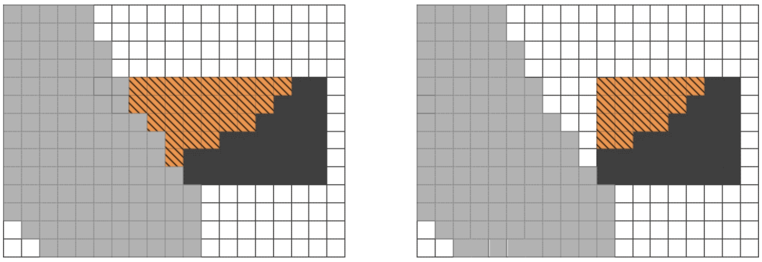

Figure 3.3 Difference between f_x of Kang (left) and of this research (right)

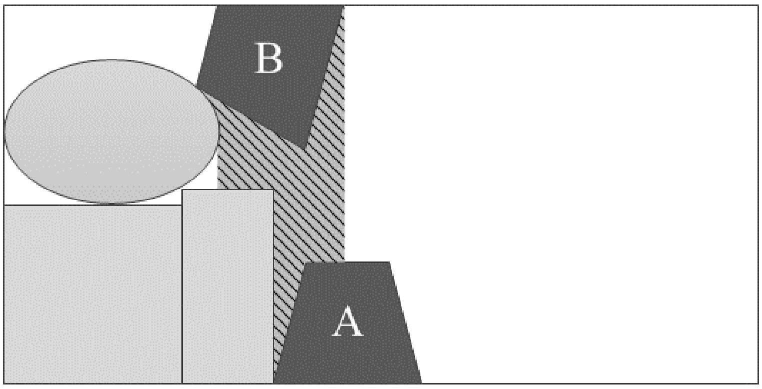

Figure 3.4 Comparison of f_y in the case of A and B

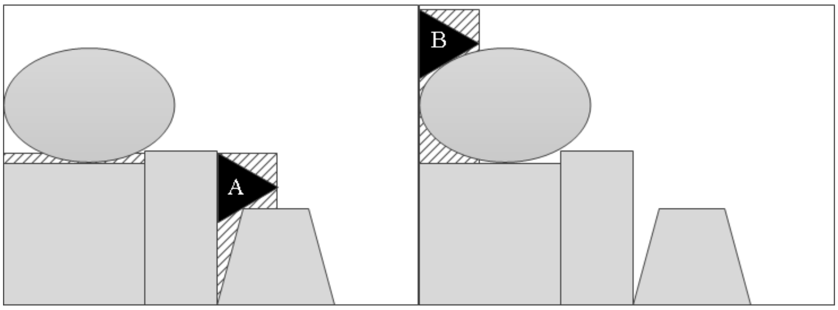

Figure 3.5 Comparison of f_{xy} in the case of A and B

$Usage\ Length$ and $Usage\ Density$ suggested by Weng and Kuo [7] are calculated as Equation 3 and 4. The parameters are shown in Figure 3.6. Firstly, $Usage\ Length$ is the ratio between the total length and the occupied length by the parts on a plate. It contributes to make the parts locate on the left side of a plate so that the remnant produced after the nesting has a long length as much as possible. This is for raising efficiency of the remnant by making the shape of it efficient. For this reason, the smaller $Usage\ Length$ is, the better efficiency of the remnant is. $Usage\ Density$ is defined as the ratio of total area size of all positioned pieces over the rectangle area occupied by the pieces.

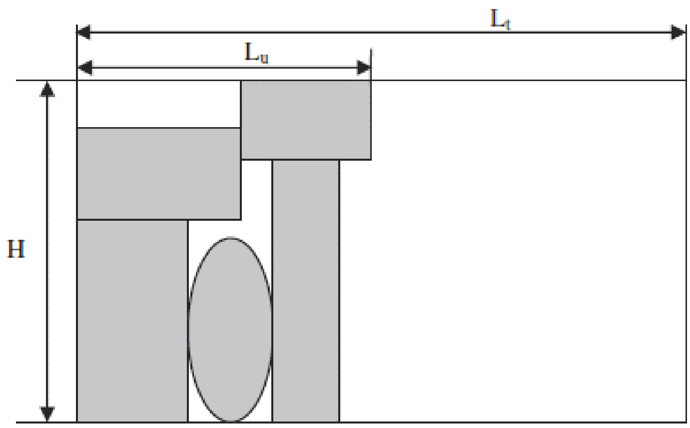

Figure 3.6 Parameters for evolution (Weng and Kuo, 2011)

$$Usage\ Length = \frac{L_u}{L_t} \tag{3}$$

$$Usage\ Density = \frac{Total\ area\ size\ of\ all\ positioned\ pieces}{L_u \times H} \tag{4}$$

3.3 Results

To assess efficiency of the objective function, the abundant numerical experiments were carried out for the stern, mid-ship and engine room blocks of the ship. The characteristics of each block could be different depending on the types or design of the ship, but, in general, the stern block has the characteristics as follows: many of the parts consist of not only lines but also arcs a lot comparing to other blocks' such as parallel blocks. Furthermore, the size of the parts is relatively similar to each other. In the case of the engine room of the ship, there are generally many parts in a block compared with any other blocks, but the size of the parts is nearly similar. However, the nesting using these parts is more difficult than those from other blocks since the geometry of the parts is much various and complex. In contrast, the mid-ship block which is the parallel block has the parts whose size is relatively huge and whose shape is close to rectangle, so simple. Therefore, the nesting of the parallel block tends to be rather easier.

Firstly, the nesting process was conducted in order to see the number of the parts allocated on a raw plate and the scrap ratio at the same time. Part group 1 which consists of 19 pieces from the stern block was used for this experiment. Next, the nesting process was carried out to compare the available length of the remnant with Part group 2 whose pieces are also from the stern. This part group is allocated onto a raw plate whose area is relatively larger than the total are of the pieces. Through those two nesting results, the experiments for the part groups of the other blocks-mid-ship and engine room- were implemented with some weighting factor sets which shows better nesting layout in order to see the differences of efficiency.

The size of raw plate was fixed at 20m×4m×13t (A Grade) which is width, length, thickness and quality, respectively. One pixel represented 20mm of width and length. The range to determine the rotation angle was 0 to 360 degree and the interval was 5 degree. The nesting sequence followed the traditional sequence which is fixed at the descendant order by the area of the parts. The experiment was carried out by a computer with an Intel® Core™ i3-2120 CPU @ 3.30GHz with 4.00GB RAM.

To examine the changes of the quality of the nesting layout, the various weighting factor sets were utilized. Table 3.1 shows some of the representative weighting factor sets from the numerical experiments. Table 3.2 is the numerical results of the part groups for each weighting factor sets.

Table 3.1 Weighting factor sets for each case

	α	β	γ	δ	ε
Case 1	0.33	0.33	0.33	-	-
Case 2	-	0.5	-	0.5	-
Case 3	-	0.33	0.33	0.33	-
Case 4	0.3	0.1	-	0.6	-
Case 5	0.1	0.3	-	0.6	-
Case 6	-	0.2	-	0.5	0.3
Case 7	-	0.2	-	0.3	0.5

Table 3.2 Numerical results of nesting process for part groups from stern block

	Part group 1		Part group 2	
	Number of nested parts on plate [pcs.]	Scrap ratio [%]	Available length of scrap	Scrap ratio [%]
Case 1	18	26.8	40	55.5
Case 2	18	28.7	348	〃
Case 3	19	25.9	315	〃
Case 4	19	25.9	341	〃
Case 5	18	28.7	344	〃
Case 6	18	27.8	341	〃
Case 7	18	27.7	332	〃

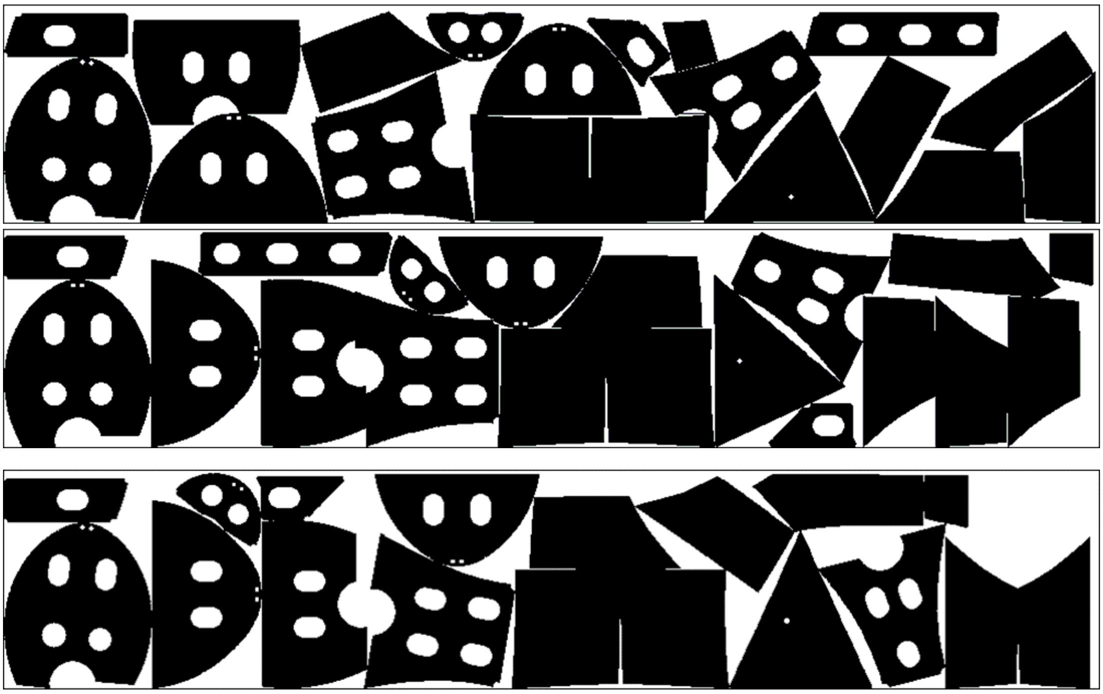

Figure 3.7 Case 3 (top), case 4 (middle), and case 5 (bottom) for part group 1

According to the nesting results for part group 1, all of the candidate pieces were allocated on a plate with case 3 and 4 and it led the smallest scrap ratio in both cases. The other cases could not place all pieces and the scrap ratio varied depending on the tendency of placement.

In the case of the nesting results for part group 2, all of the 34 pieces were arranged onto a raw plate with every weighting factor sets. Although all the cases showed the same scrap ratio, the resultant layout with the weighting factor set of case 2 had better quality than that with the weighting factor set of case 1 when it comes to the usefulness of the remaining stock. The other cases had the similar available length to case 2.

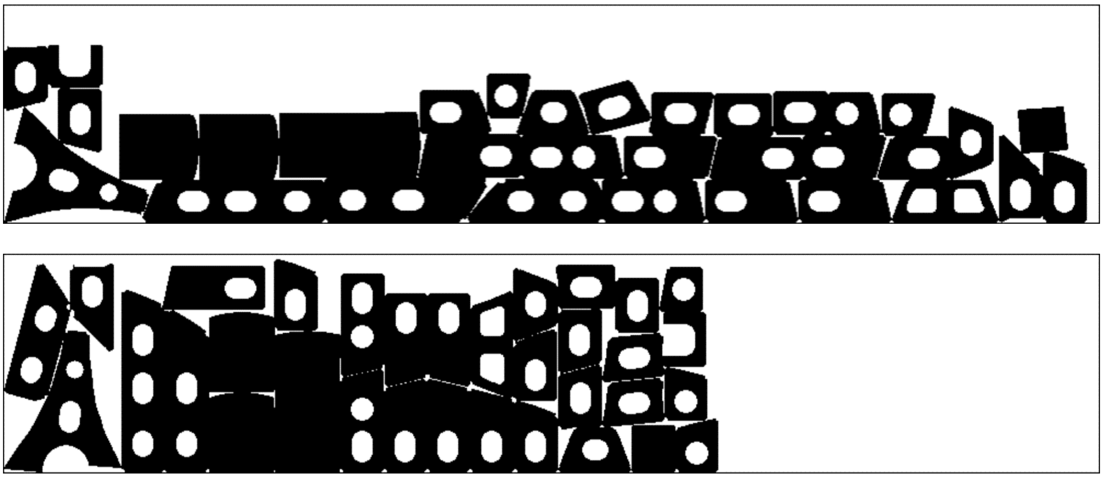

Figure 3.8 Case 1 (top) and case 2 (bottom) for part group 2

Following result is the nesting layout for a part group of the mid-ship block with the case 2 and 4 which result greater nesting efficiency in two previous experiences. Figure 3.9 and Figure 3.10 shows the nesting layout as images, a bitmap image above and a CAD drawing below. The CAD drawing is illustrated by drafting the numerical results of this nesting algorithm to Tribon M3. The layouts show that filling holes was successfully done and pairing and grouping of pieces were also adequate despite none of any considerations of the approaches about it.

Table 3.3 presents the number of the nested parts on a raw plate and the scrap ratio. Comparing the nesting results of two cases, the weighting factor set of case 2 led 6 more parts allocated onto a plate than case 4, while the weighting factor set of case 4 was more efficient in the nesting of part group 1. Hence, it is shown that the most efficient weighting factor sets are different depending on the characteristics of the candidate parts.

Table 3.3 Nesting results of parts from parallel block adopting weighting factor sets of case 2 and case 4

	Number of nested parts on plate [pcs.]	Scrap ratio [%]
Case 2	66	5.7
Case 4	60	5.8

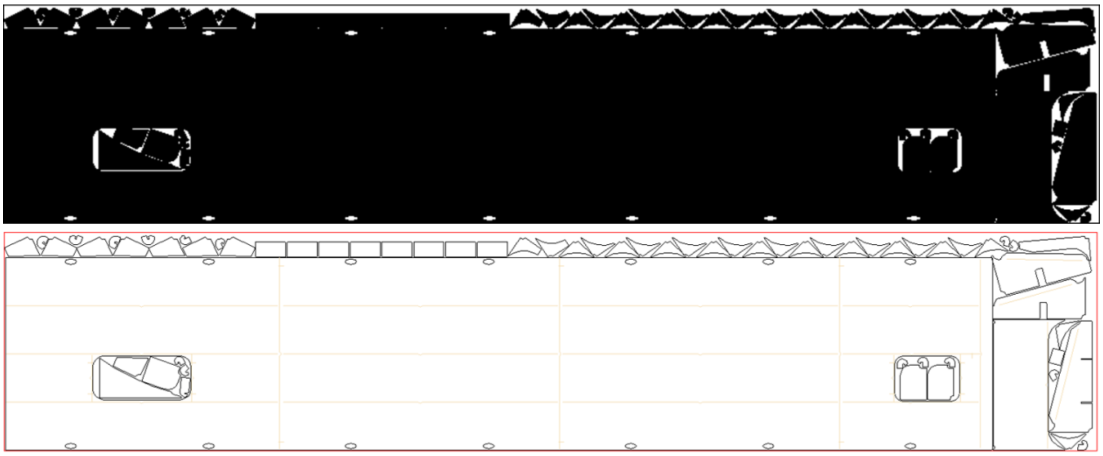

Figure 3.9 Bitmap image (top) and CAD drawing (bottom) from the result of case 2

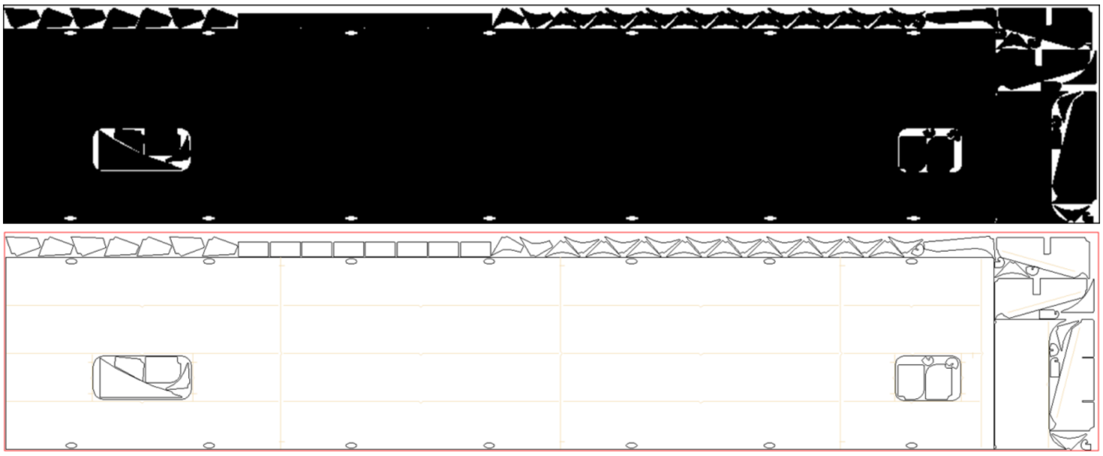

Figure 3.10 Bitmap image (top) and CAD drawing (bottom) from the result of case 4

Lastly, the results applying the case 2 and case 4 to the nesting for the parts of the engine room are shown in Figure 3.11 and Figure 3.12, and also in Table 3.4. There are 26 candidate parts in this part group and all of these pieces were allocated on a raw plate with the weighting factor set of case 2. With case 4, only 24 pieces were on a plate. In this case, the difference in the scrap ratio was about 4%.

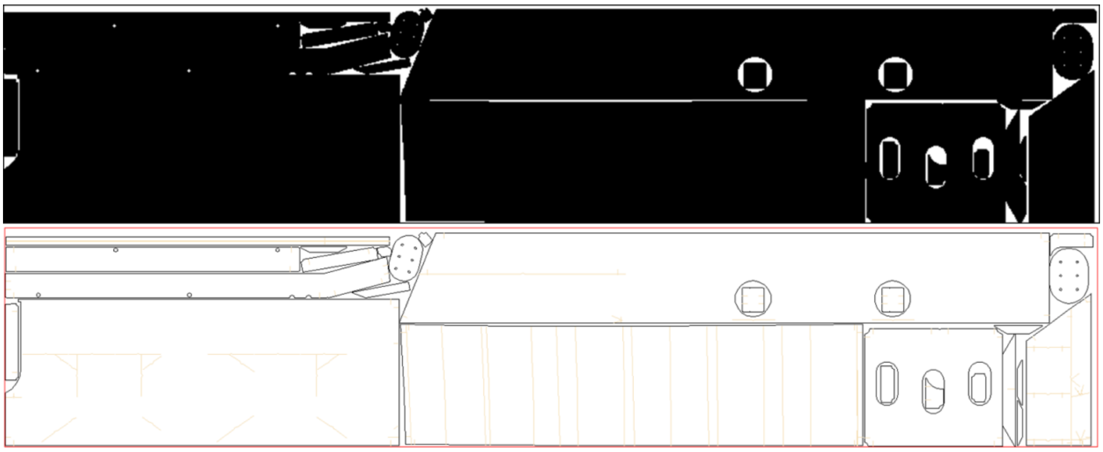

Figure 3.11 Case 2 for engine room parts

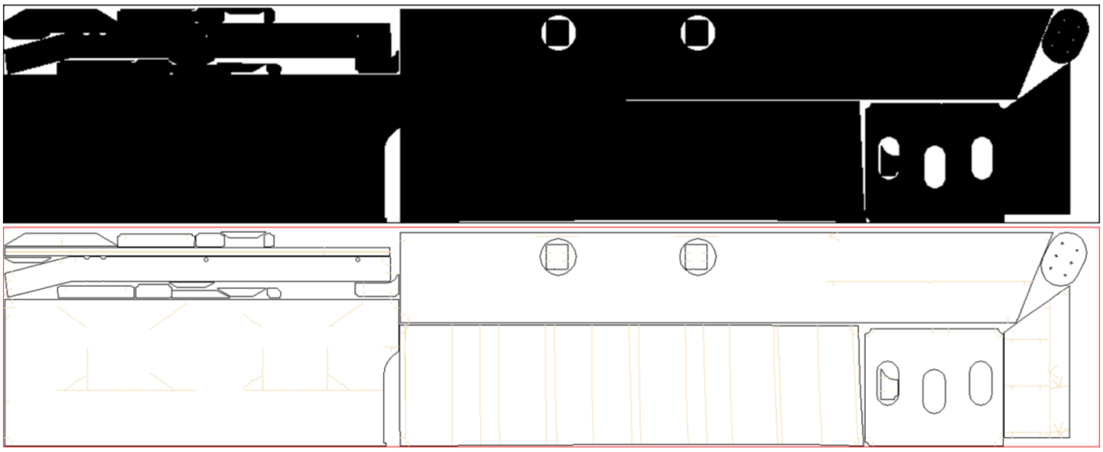

Figure 3.12 Case 4 for engine room parts

Table 3.4 Nesting results of parts from engine room utilizing weighting factor sets of case 2 and case 4

	Number of nested parts on plate [pcs.]	Scrap ratio [%]
Case 2	26	5.0
Case 4	24	8.9

3.4 Conclusion

In this chapter, we recomposed the objective function of placement algorithm using the previously suggested functions with the weighting factor method. Then, the resultant layouts were evaluated by two criteria. Firstly, we focused on the number of the nested parts on a raw sheet and scrap area. Second criterion was the compactness of resultant layout. Through the nesting results for part group 1 and 2 from the stern block with various weighting factor sets, it is known that case 2, 3, and 4 shows better efficiency with regard to available length of the remnant as well as scrap ratio. Those cases include f_y and $Usage\ Length$ and the variation of the weighting factor sets susceptibly affect the resultant layout. Comparing the nesting results of the part group from the stern, from the mid-ship block, and from the engine room, case 2 shows the efficient result in the case of the nesting for part group which includes relatively square-shaped geometries and has a great difference in size among the geometries. In contrast, the nesting for part group whose geometries have a slight difference in size has better efficiency by allocating more parts on a raw plate when the objective function carries the weighting factor set of case 4 than the other cases.

3.5 Application

The part of this algorithm is applied and used for the commercial nesting program (Figure 3.13).

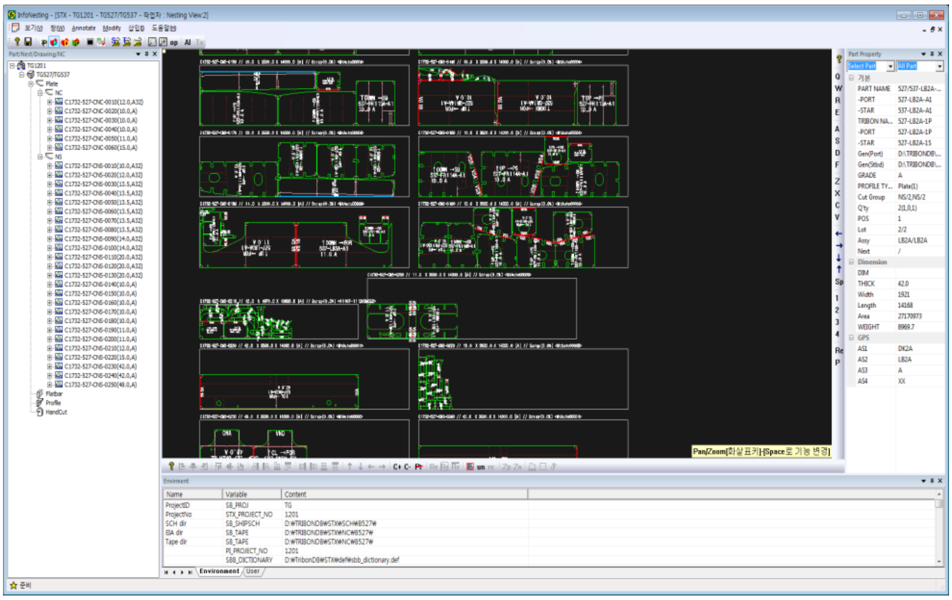

Figure 3.13 Application of the nesting algorithm on the commercial nesting program

Chapter 4

ORDER DECISION ALGORITHM

4.1 General

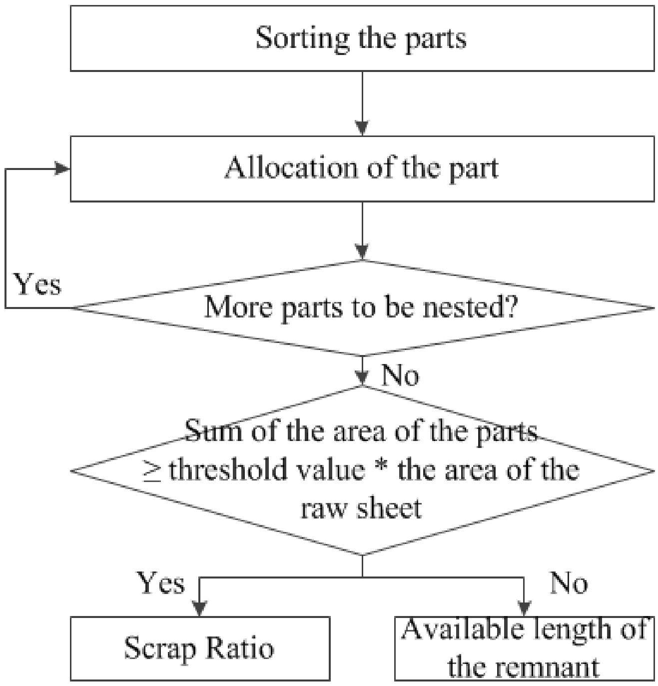

Figure 4.1 Flow chart

The whole algorithm follows the process shown in Figure 4.1. In this algorithm, firstly, we have to sort the parts according to the nesting sequence. Whereas traditional sequence has usually followed the descendant order of the parts' area as mentioned in the introduction, we adopted Simulated Annealing. After sorting the pieces, they are allocated one by one by placement algorithm. Once allocation process of all of the parts finishes, the resultant layout is evaluated by the objective function. The objective function here is set in accordance with those purposes- either the scrap ratio or the available length of the remnant.

Following sections introduce Simulated Annealing and the objective function.

4.2 Simulated Annealing

To determine the placing sequence which gives the best result, Simulated Annealing (SA) is utilized in this paper. There has been an enormous quantity of work where SA has been applied to various combinatorial optimization problems. Pure local search approaches frequently end up entrapped in local optimal solutions, since they can only move to better solutions in the neighborhood. SA can be seen as an evolution of those approaches, by allowing some controlled uphill movements, in order to achieve global optimality. Accepting a movement to a worse solution depends on a control parameter (the temperature) and on the magnitude of the variation of the objective function. Many real-life problems have been successfully tackled by approaches based on SA. The main reasons for this success are the high quality of the solutions, the easy inclusion of real-life constraints and the robustness of these approaches. The main drawback is the large computational effort needed [19]. Hence, Kim and Lee [10] set the termination condition to reduce the computing time. However, this research finds the nesting sequence which gives the most efficient result after all iteration. The comparison of the results of the initial sequence, the best and the worst during the iteration history will be shown in Implementation.

4.3 Objective functions

As aforementioned in the introduction, the characteristics of relationship among the parts in each group are different from the others, so the nesting process has to be carried out considering these characteristics. In this paper, the process has two main purposes which are the decrease of scrap ratio and the increase of available length of remnants. In order to obtain the appropriate result depending on the situation, this algorithm is codified to separate the nesting problems into following two types. If the area ratio of the all candidate parts to the raw plate in the given problem is over the user defined threshold, the process focuses on the decrease of scrap ratio. If the ratio is smaller than the threshold, the process tends to increase available length of the scrap shortening the occupied length by the parts being nested.

The objective function of the placement algorithm consists of f_y and $Usage\ Length$ which led more efficient nesting results (see section 3.3 Results).

$$\text{min.}$$

$$\mathrm{F} = 0.5 f_y + 0.5 Usage\ Length$$

Chapter 5

IMPLEMENTATION

5.1 Outline and Environment of Experiments

In this paper, problems have been divided into two types focusing on how long the usage length of the remnants is formed after finishing the nest process and how much the scrap is generated. Also, this research sets the user defined threshold as predefined value of the raw sheet area. If the sum of all part area is over the limit, it focuses on the waste ratio. If not, the usage length is spotlighted. The initial arrangement sequence for SA follows the descendant order by area of the candidate parts. Each group was approximately run about 1000 iteration times.

Since ship parts have the irregular shape and various characteristics depending on the block which they belong to, part groups from the stern and engine room of the ship are used for the numerical experiments in order to verify the quality of irregular ship part nesting. The general characteristics of each block of the ship are introduced in section 3.3 Results. We randomly chose 26 parts from the stern and 33 parts from the engine room which are relatively similar in size to focus on the scrap ratio. To examine the effect of the nesting sequence on efficiency of the remnants, another 34 parts from the stern block of the ship within the limit which is that the sum of the area of the parts is under the threshold value of that of a raw sheet.

The size of raw plate was fixed at 20m×4m×13t (A Grade) which is width, length, thickness and quality, respectively. One pixel represented 20mm of width and length. The range to determine the rotation angle was 0 to 360 degree and the interval was 15 degree to reduce the computing time because it takes too much time to get the results after all iteration times. The initial nesting sequence for SA followed the descendant order by the area of the parts. The experiment was carried out by a computer with an Intel® Core™ i3-2120 CPU @ 3.30GHz with 4.00GB RAM.

Table 1 shows the results for the number of the parts arranged on a raw plate and the scrap ratio. Table 2 presents the results for the available length of remnants evaluated in the problem. All the figures below were obtained through TRIBON M3 by drafting the result from our nesting process.

5.2 Results focused on scrap ratio

Table 5.1 Experimental results on scrap ratio

Order	Part group from stern		Part group from engine room	
	Number of the nested parts on a plate [pcs.]	Scrap ratio [%]	Number of the nested parts on a plate [pcs.]	Scrap ratio [%]
Initial	20	23.1	25	22.2
Best	22	22.4	29	20.1
worst	20	37.4	28	33.1

The results of nesting process adopting SA showed the importance of the nesting sequence in the process as shown in Table 5.1. At first, a part group from stern was considered. When the parts are arranged in descendant order by the area, 20 pieces of parts could be nested on a raw plate and the scrap ratio was 23.1%. After the 11^{th} iteration, the process gave the best result having two more parts on the plate and showing a slight decrease in the scrap ratio to 22.4%. With the worst nesting sequence which was occurred by the characteristics of SA, the scrap ratio showed an increase of 14.3% more than the initial sequence although the same number of pieces was arranged. (Figure 5.1)

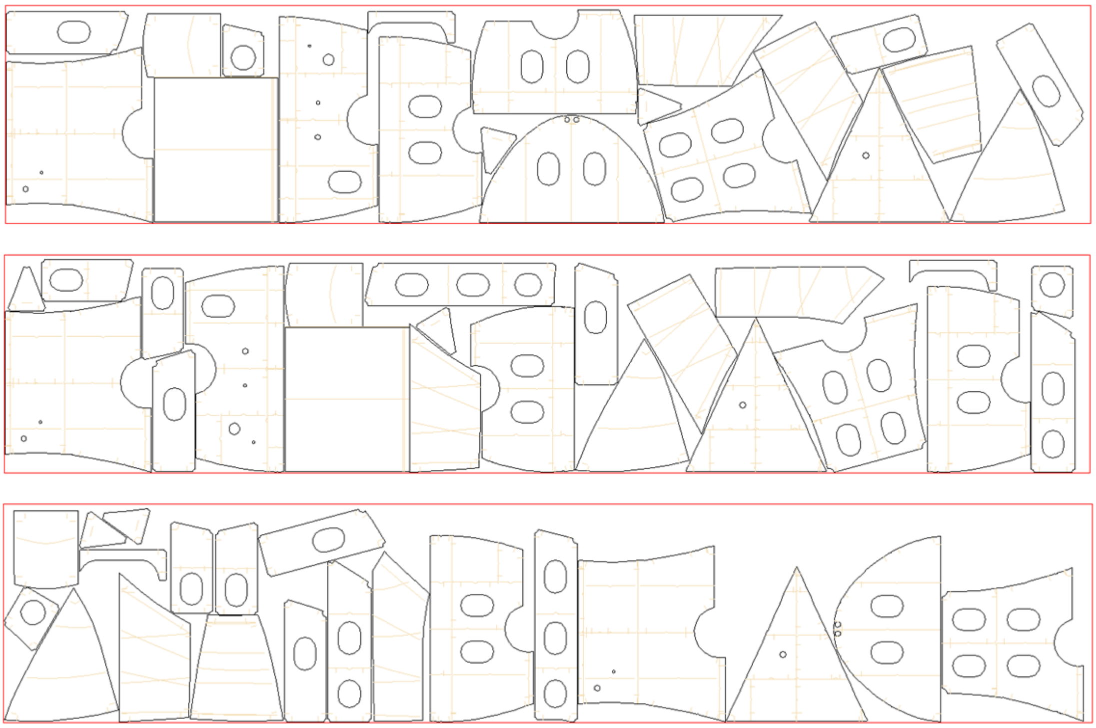

Figure 5.1 The resultant layout of the stern block

(Top: initial, Middle: the best, Bottom: the worst)

Secondly, the nesting of a part group from the engine room initially resulted that 25 pieces out of 33 were placed on a plate and the scrap ratio was 22.2%. The process showed the improved result after 828th iteration nesting 4 more parts and leading the scrap ratio to 20.1%. In the case of the worst results, the scrap ratio was 33.1%, increased by 10.9% compared to the result by the initial sequence (Figure 5.2).

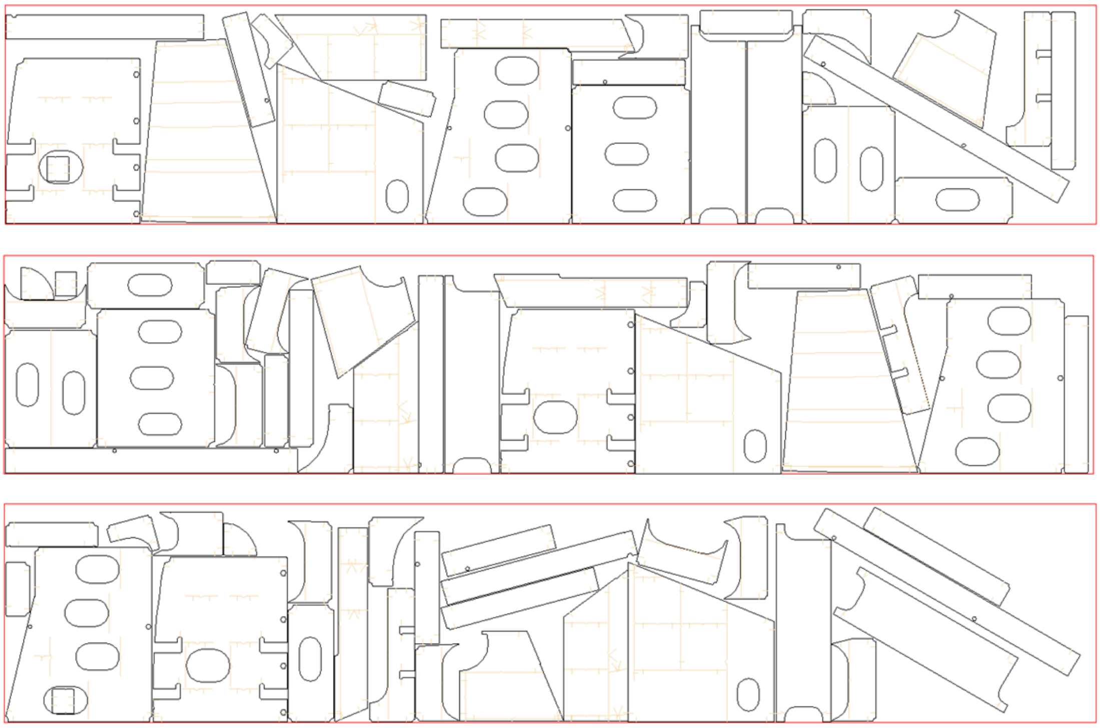

Figure 5.2 The resultant layout of the engine room

(Top: initial, Middle: the best, Bottom: the worst)

Although the parts might be conveniently nested following descendant order by the area, it is yielded that the nesting process with SA would give the developed results improving the nesting density.

5.3 Results focused on available length

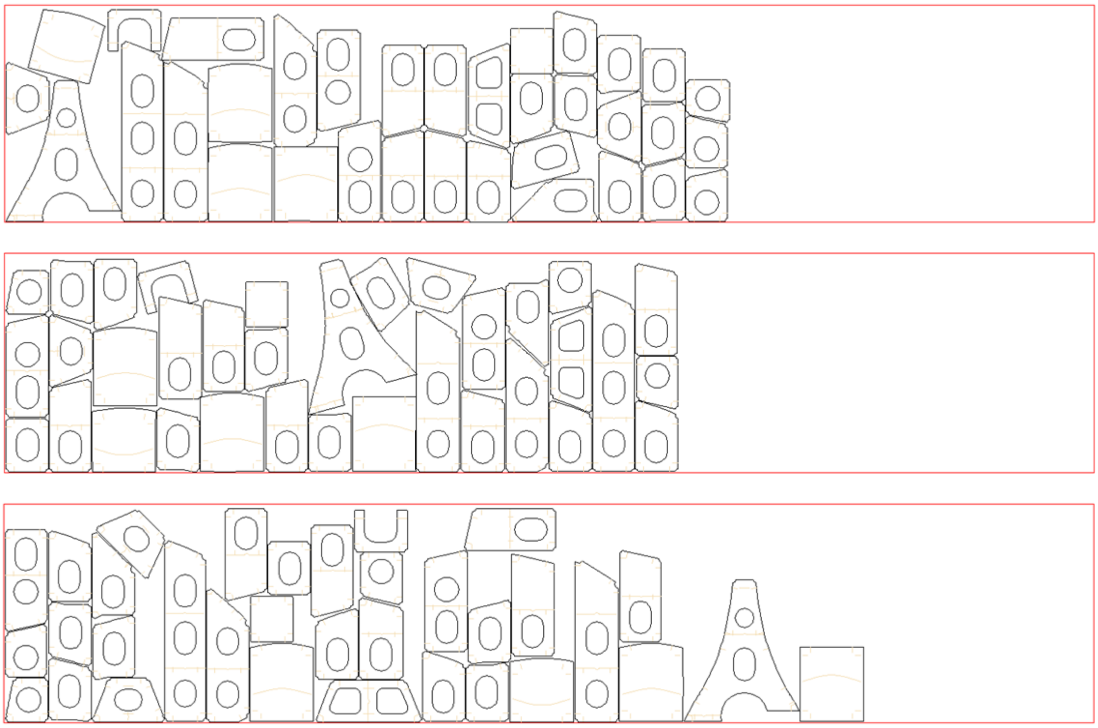

Figure 5.3 The resultant layout of the part group from stern

(Top: initial, Middle: the best, Bottom: the worst)

Table 5.2 Experimental results on available length

Order	Available length of remnant [%]
Initial	33.4
Best	38.1
Worst	21.1

Figure 5.3 shows the nesting layout of the parts on the sheet and Table 5.2 is the numerical results on the available length which indicates efficiency of the scrap. Adopting the initial sequence which is in descending order by area of the pieces resulted 33.4% of the

available length of the scrap. In the case that the sequence showed the best result after the 683th iteration, the available length increased by about 4.7% by comparison with that of the initial, whereas the worst sequence resulted the decrease of 12.3% in the available length.

Chapter 6

CONCLUSIONS AND SUGGESTIONS

6.1 Conclusions

When it comes to economic efficiency for production in many manufacturing industries, it is required that the nesting algorithm satisfies the various purposes demanded in each industry. In the shipbuilding, the nesting is processed with each block of the ship which has its own characteristics. The nesting algorithm of this paper considers the characteristics of the shipbuilding industry and consists of two main algorithms, placement algorithm and order decision algorithm. Although the placement sequence in the shipbuilding nesting generally follows the order of the area or aspect ratio of the parts, Simulated Annealing is used in this paper to find the best combination which is decided by an arrangement sequence for improvement of the quality.

Each algorithm has its own objective function and the interaction of these two functions affects efficiency of the resultant layout. The objective function for the placement contributes the translation of each part to be nested and decides its appropriate rotation angle. This function here is constructed with the terms from the previously suggested researches and the weighting factor method is adopted for the function. As the results of the numerical experiments with various weighting factor sets, it is founded that efficiency of each set might be different depending on the characteristics and the combination of the candidate parts.

Furthermore, a term contributing to allocate the pieces at the bottom of a raw plate as below as possible and another one making the nesting layout compact in x-direction are included in the weighting factor sets which have more efficient results.

Considering the function of the order decision, it operates in accordance with the main purposes of this nesting algorithm which are to reduce the scrap ratio and to compact the resultant layout. Depending on the criterion of the judgment, this nesting algorithm focuses on the one of those purposes. This research set the criterion as the area ratio of the raw plate to all candidate parts and selects the appropriate objective function. Comparing with the common placement sequence, the results of the experiments yields that it is more efficient to be able to select the objective function depending on the area ratio and there exists the arrangement sequence and correspondent placement showing the more improved qualities of the resultant layout for each purpose.

Consequently, the appropriate determination of the objective functions in the order decision algorithm and placement algorithm with a focus on the scrap ratio and available length as considered in this paper, and the interaction between those two functions lead to more improved efficiency in the shipbuilding-customized nesting algorithm.

6.2 Suggestions for future studies

In relation to time efficiency, the nesting process here still has the limitation. In order to obtain the arrangement sequence, the enough iteration time is required for SA. Whereas the time required by the process of representation of geometry and placement is relatively short, SA causes tremendous time with numerous pieces in a ship block. For this reason, the effort to reduce time will be needed by using the minimum information of geometry or other methodology. The geometrical error caused by the Pixel Representation and the resolution is also the problem to be solved. Although the compensation process is done, more improved solution to the fundamental causes will be required for more complex and irregular geometry.

BIBLIOGRAPHY

[1] M. Adamowicz and A. Albano, "Nesting two-dimensional shapes in rectangular modules," *Computer-Aided Design,* vol. 8, pp. 27-33, 1976.

[2] A. R. Babu and N. R. Babu, "A generic approach for nesting of 2-D parts in 2-D sheets using genetic and heuristic algorithms," *Computer-Aided Design,* vol. 33, pp. 879-891, 2001.

[3] M. Konopasek, "Mathematical treatments of some apparel marking and cutting problems," *US Department of Commerce Report,* vol. 99, pp. 90857-10, 1981.

[4] Y. Stoyan, G. Scheithauer, N. Gil, and T. Romanova, "Phi-functions for complex 2D-objects," *Quarterly Journal of the Belgian, French and Italian Operations Research Societies,* vol. 2, pp. 69-84, 2004.

[5] G. B. Bang, "Optimal Allocation of Two-Dimensional Irregular Shapes for Minimum Waste," Master, Seoul National University, 1990.

[6] H. G. Kang, "Optimal layout of irregular shape ship parts," Master, Department of Industrial Engineering, Chonnam National University, 1998.

[7] W. C. Weng and H. C. Kuo, "Irregular stock cutting system based on AutoCAD," *Advances in Engineering Software,* vol. 42, pp. 634-643, 2011.

[8] K. A. Dowsland, S. Vaid, and W. B. Dowsland, "An algorithm for polygon placement using a bottom-left strategy," *European Journal of Operational Research,* vol. 141, pp.

371-381, 2002.

[9] B. H. Ryu and D. J. Kim, "A Study on the Irregular Nesting Problem Using Genetic Algorithm and No Fit Polygon Methodology," *Journal of the Korean Society of Ocean Engineers*, vol. 18, pp. 77-82, 2004.

[10] D. S. Kim and C. S. Lee, "Determination of an optimal sequence for nesting with heuristic algorithm," in *CAD/CAM Conference*, 2009.

[11] E. K. Burke, R. S. Hellier, G. Kendall, and G. Whitwell, "Complete and robust no-fit polygon generation for the irregular stock cutting problem," *European Journal of Operational Research*, vol. 179, pp. 27-49, 2007.

[12] J. A. Bennell and J. F. Oliveira, "The geometry of nesting problems: A tutorial," *European Journal of Operational Research*, vol. 184, pp. 397-415, 2008.

[13] *Tribon M3 Program Manual*. Available: http://www.aveva.com

[14] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems journal*, vol. 4, pp. 25-30, 1965.

[15] A. Michael, "Graphics Programming Black Book," ed: Coriolis Group Books, 1997.

[16] Hexar. *Drawing Lines - The Bresenham Algorithm*. Available: http://www.cs.toronto.edu/~smalik/418/tutorial2_bresenham.pdf

[17] J. Burkardt. (2012). Available: http://people.sc.fsu.edu/~jburkardt/cpp_src/circle_arc_grid/circle_arc_grid.html

[18] H. B. Lee and W. S. Ruy, "Determination of Nesting Algorithm Fitness Function through

the Various Numerical Experiments," *Journal of Ocean Engineering and Technology,* vol. 27, pp. 28-35, 2013.

[19] A. M. Gomes and J. F. Oliveira, "Solving irregular strip packing problems by hybridising simulated annealing and linear programming," *European Journal of Operational Research,* vol. 171, pp. 811-829, 2006.

감사의 글

지나고 보니 지난 2 년이 얼마나 짧았던 시간이었는지 새삼 느끼게 됩니다. 하지만 막상 그 시간 안에서는 잘해오고 있는지 고민하고 순간 순간의 어려움에 부딪혀 힘들어했던 적도 많이 있었습니다. 그렇기 때문에 어느 때보다 기도를 열심히 했던 2 년이었습니다. 이렇게 마지막까지 잘 이끌어주신 하느님께 감사의 기도 드립니다.

대학원에 들어오면서 처음으로 새로운 분야를 접하게 되었고, 이 분야에 대해 전혀 지식이 없었던 저를 처음부터 끝까지 변함없이 잘 이끌어주신 유원선 교수님 감사합니다. 교수님이 하나하나, 저보다도 열심히 옆에서 지도를 해주셨기에 이렇게 유종의 미를 거둘 수 있었습니다. 또한 학부 때부터 좋은 가르침을 주셨고, 대학원에 진학할 수 있었던 계기를 주셨던 조일형 교수님 감사합니다. 마지막까지 여러모로 신경 많이 써주셨던 점 또한 감사합니다. 항상 관심 가져주시고, 소중한 조언 해주셨던 팽동국 교수님, 이종현 교수님, 배진호 교수님께도 감사 드립니다.

학부와 대학원에서의 시간 동안 함께 공부하고 생활했던 모든 선·후배 및 동기들에게도 감사의 말씀 전합니다. 함께 했던 시간만큼 감사할 분들이 너무 많아 개인적으로 감사를 전하겠습니다.

처음 뵈었을 때부터 지금까지 저에게 늘 칭찬과 격려 아끼지 않으시고 저를 위해 항상 기도 해주시는, 영적 아버지이신 이태수 미카엘 신부님, 감사합니다.

집에 항상 늦게 들어가고, 때로는 학교에서의 삶에 지쳐 집에 가면 무뚝뚝했었던 딸의 투정까지도 다 받아주셨던 어머니, 감사합니다. 어머니는 저에게 가장 아름답고, 훌륭한 분이십니다. 나에게 너무 소중하고 언제나 날 위해주는 착한 언니, 감사합니다. 자주 뵙지는 못하지만 항상 저를 아껴주시는 마리나 이모, 감사합니다.

만나면 모든 걸 다 잊을 수 있을 만큼 즐겁고 행복한, 옆에서 묵묵히 응원해주는 친구들, 지원, 송이, 나래, 소은, 소연, 그리고 늘 언니 같이 챙겨주는 고마운 현정, 같이 이야기하고 함께 웃었던 은영 언니, 모두 감사합니다. 이 외에도 가까운 곳에서, 먼 곳에서 응원해주시는 모든 분들 감사합니다.

앞으로 어떤 미래가 기다리고 있을지 모르겠지만, 어떠한 상황 속에서도 성실하게 발전해나가는 모습 보여드리겠습니다. 한 분 한 분 너무 감사하고, 사랑합니다.