Study on the Layout Optimization of Platform Based on Simulated Annealing Algorithm

Dequan Zheng
School of Management, Harbin Institute of Technology, Harbin, China
Email: zhengdequan@hit.edu.cn

Abstract—**Aiming at the production scheduling problem in the hull block production, this study investigates the maximum use of the assembly blocks working platform. By transforming the scheduling problem into two-dimensional bin packing problem, this study explores the effective block spatial layout under the condition of spatial constraints. In order to improve the drawbacks of the lowest horizontal line algorithm, this study proposes the lowest horizontal line-rotate searching algorithm. By combing the simulated annealing algorithm and the lowest horizontal line-rotate searching algorithm, the simulated results show that such scheduling approach can produce much better planning results. Finally, the study carries out a number of simulation experiments to optimize the parameters in practical application.**

Index Terms—**Block assembly planning, Two-dimensional Packing, Rotating search, Simulated Annealing Algorithm**

I. INTRODUCTION

According to the idea of block assembly shipbuilding, the hull blocks manufactured in different parts and phases are viewed as modules. The spatial scheduling of modules is related to plane layout. If the time factor is not considered, the idea on scheduling problem is similar to that in two-dimensional plane layout problem, i.e., the two-dimensional bin packing problem. If an effective algorithm is found for the two-dimensional bin packing problem, the goal of saving resources and promoting production efficiency will be achieved. Moreover, the approximation algorithm for solving two-dimensional bin packing problem can be applied to the solution of three-dimensional bin packing problem. Hence, it possesses important theory significance and practical value for the research on the two-dimensional bin packing problem.

For algorithms solving two-dimensional bin packing problem effectively, there are three major categories: exact algorithm, heuristic algorithm and modern heuristic algorithm. Heuristic algorithm can get desirable scheduling results within a reasonable time, even for large-scale planning problems. But for most heuristic algorithms, it is usually difficult to analyze their approximation ratio. Baker et al. proposed a different classical bin packing algorithm BL (Bottom - Left) [1]: the items are ranked by decreasing order of width; the current item is put in the lowest and leftmost location which can accommodate it. BLF (Bottom - Left - Fill) algorithm [2] is an improvement of the BL algorithm. E. K. Burke proposed a new heuristic algorithm BF (Best - Fit) [3]: the optimal item is chosen dynamically and a post-processing strategy is added, which enables a good solution within a very short time. Zhang Defu proposed two rapid and effective heuristic algorithms to solve two-dimensional bin packing problem with rotatable items, namely, recursive heuristic algorithm [4] and fitting heuristic algorithm [5]. Huang Wenqi et al [6, 7] proposed some effective algorithms for two-dimensional bin packing problem, in which the priority of minimum flexibility and corner occupation are ingeniously designed.

Modern heuristic algorithm is a kind of hybrid algorithm which combines heuristic algorithm and intelligent optimization algorithms (simulated annealing algorithm, genetic algorithm, tabu searching algorithm, ant colony algorithm and neural network). Lai et al [8] and Faina [9] applied simulated annealing algorithm to material cutting problems. Artificial neural network methods have been applied to these questions by Dagli and Poshyanonda [10]. Kroger[11] proposed a genetic algorithm to solve the two-dimensional bin packing problem (2BP). Bortfeldt [12] proposed a genetic algorithm without any coding and decoding. Lodi et al [13] proposed a tabu searching algorithm for 2BP. Burke [14] applied several modern heuristic algorithms based on BF and achieved satisfactory results. Hopper and Turton [15] conducted experimental analysis on heuristic algorithms and modern heuristic algorithms in solving two-dimensional bin packing problem. Generally speaking, modern heuristic algorithm is a more complicated form of heuristic algorithm. With this algorithm, multiple distinctive solutions may be obtained after many operations for one problem. All the algorithms have parameters which need adjusting for many times to achieve the optimal performance.

The main contents of this research are as follows: a simplification algorithm model is carried out to analyze the features of block scheduling and the working platform in hull production under certain constraint conditions; the simulated results are shown based on the combination of the simulated annealing algorithm with the lowest horizontal line-rotate searching algorithm; on this basis, the hull block spatial scheduling algorithm is designed to

doi:10.4304/jsw.8.7.1793-1800

optimize the key parameters through massive simulation experiments.

II. Simplification Models and Lowest Horizontal-Line Rotate Searching Algorithm

A. Simplification Model of Hull Block Assembly Planning

As the semi-finished product of hull construction, the block assembly is characterized by massive production and complex process. To complete the whole manufacturing process of hull block, it is necessary to arrange the corresponding module to the fixed working platform. On the working platform, the height of the hull block is negligible in relation to the lifting height of the crane. Hence in the process of spatial scheduling, the hull block is regarded as two-dimensional, and the real height of the hull block is not considered in the mathematical models and algorithm solutions. The maximum projection of hull block on the working platform is considered as the geometric shape of the hull block. Moreover, statistical analysis is performed for the size of contact surfaces of all hull blocks. The geometry of the contact surface of each hull block is reasonably simplified.

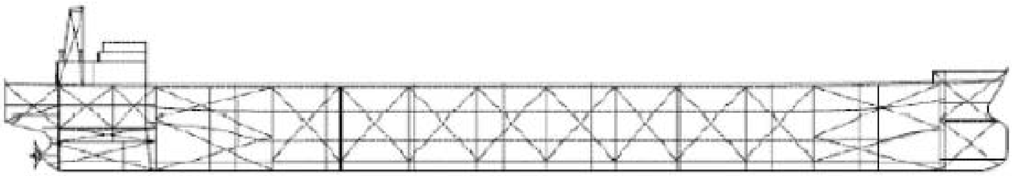

Figure1 All hull blocks

Figure 1 shows all hull blocks of a bulk carrier at a shipyard. Among all hull block geometries, most of the blocks in the middle hull are approximated as rectangular. The projected plane of the hull blocks of bow and stern are approximated as isosceles trapezoid, right trapezoid, parallelogram and triangle.

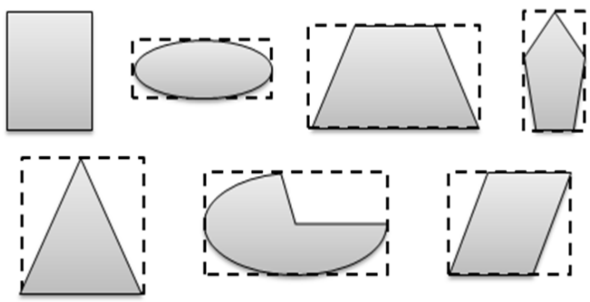

Figure 2 Module Projections Approximated as Rectangles

During actual production, the hull blocks are viewed as several types of convex polygon. The geometries of some hull blocks have local concavity and these concavity regions cannot be utilized by other blocks. For the sake of convenience, the geometry is fitted as a rectangle with minimum area whether it is a convex or a concave polygon (shown in Figure 2). The length and width of hull block working platform are limited, so it is consequently simplified the platform into a rectangle platform with length and width of fixed values.

In the process of spatial scheduling of hull block, the constraint conditions include the size of block working platform, blocking time order, lifting capacity and the correlation order of block-building. The goal of the spatial scheduling problem of hull blocks is to maximize the utilization efficiency of hull block working platform. The constraint conditions are as follows:

(1) Boundary constraint of block placement: the hull blocks must be placed within the working platform. It cannot exceed the boundary of the working platform.

(2) Direction constraint of block placement: the hull blocks must be capable for rotating within certain range of angle inside the working platform.

(3) Not overlap between blocks.

B. Lowest Horizontal-Line Rotate Searching Algorithm

The lowest horizontal-line searching algorithm is a commonly used method in rectangle strip packing problem (RSPP) [16]. In the lowest horizontal-line algorithm, if the width of the lowest horizontal line is smaller than the width of the rectangle to be arranged, then the lowest horizontal line will be reselected. The second lowest one will be taken as the new lowest horizontal line, and the original lowest horizontal will be deleted. Such treatment causes the waste of the spare space above the original lowest horizontal line.

The lowest horizontal-line rotate searching algorithm is an improvement of lowest horizontal line algorithm, but with an additional rotate searching process. The specific steps are as follows:

When placing a rectangular object S_i,

Step1: The lowest horizontal line L_{i} is chosen in the set of the highest contour lines. The leftmost one is selected when the number is larger than 1.

Step2: If the width of the lowest horizontal line L_{i} is larger than that of S_i, and the surplus height h_i of the working platform is larger than that of S_i, S_i will be placed here, and the highest contour line is updated.

Step3: If the width of L_i is less than the width of S_i, then S_i will be rotated by 90 degrees (exchanging the width and length), and step 2 is executed; if the width of Li is still smaller than that of *Si*, then the lowest horizontal line is updated. The second lowest horizontal line will become the new lowest horizontal line L_{j}, and the original lowest horizontal line L_{i} will be deleted; step2 is executed repeatedly, otherwise the placement is finished.

III. Solving Simulated Annealing Algorithm

Given the packing problem of n rectangular objects, if the objects could be rotated by 90 degrees, then the maximum number of solutions is 2n • n!. With n increasing, the searching range of solution space grows rapidly. Generally it cannot be solved with enumeration algorithms. Simulated annealing (SA) algorithm is exploited to search for solution space.

A Basic Idea of SA Algorithm

The starting point of SA algorithm is based on the similarity between physical annealing process and ordinary combinatorial optimization. SA starts from some high initial temperature, and a random searching is carried out in the solution space using the Metropolis sample strategy which has the feature of probability jump. With the continual dropping of temperature, the sampling process is repeated and the global optimal solution is finally obtained. The general steps of a standard simulated annealing algorithm can be described as follows:

(1) Giving an initial temperature $t=t_0$,randomly generated initial state $s=s_0$, set k=0;

(2) Repeating

(2.1) Repeating

(2.1.1) a new state s_j=*Genete*(s) ;

(2.1.2) If *min*

$$\{1, \exp[-(C(s_j)-C(s))/t_k]\} \geq \tan dom(0,1)$$

s=s_j

(2.1.3)Until Sampling stability criterion is satisfied

(2.2) Annealing temperature t_{k+1}=*update*(t_k),set k=k+1;

(3)Until the algorithm termination criterion being satisfied;

(4) Outputting results

The key parameters of SA algorithm include:

1. State Production Function

The starting point of designing the state production function (neighborhood function) is to guarantee that the generated candidate solutions are all over the solution space.

2. State Acceptance Function

The state acceptance function is usually given in terms of probability. A main difference between different acceptance functions lies in the form of acceptance probability. The following rules should be obeyed when designing a state acceptance probability:

(1) Under a constant temperature, the probability of accepting the candidate solution that makes the value of objective function decrease is larger than the probability of the candidate solution that makes the value of objective function increase[17];

(2) When the temperature is declining, the probability of accepting the solutions that makes the value of objective function increase would diminish.

(3) When the temperature is approaching zero, only the solutions that make the value of objective function to decrease are accepted.

The introduction of state function is the key factor for the SA algorithm to achieve a global searching. Usually, SA algorithm takes *min* [1, *exp* ($-\Delta C/t$)] as the state acceptance function.

3 Initial Temperatures

The higher the initial temperature, the greater the probability of getting high quality solution would be, though the computation time will increase accordingly. Hence, the determination of initial temperature should take both optimization quality and efficiency into consideration [17].

4. Temperature Updating Function

The way that the temperature decreases is used to modify the temperature value in the outer loop. The most commonly used temperature function is exponential annealing, $t_{i+1}=a*t_i$, $0<a<1$.

5 Termination Rule of Inner Loop

The Metropolis rule is used to determine whether the new solution is accepted or not.

Suppose the temperature is t. A new state j is produced based on the current state i, having energy of E_j and E_i respectively. If $E_j < E_i$, then the new state j is accepted as the current state; if the probability $p=exp[-(E_j-E_i)/kt]$ is larger than a random number in the interval [0, 1], the new state j is still accepted as the current state; otherwise state i would be retained as the current state, where k is the Boltzmann constant.

6 Termination Rule of Outer Loop

It is also known as the rule to terminate the algorithm. The steps consist: (1) setting a temperature threshold for the termination; (2) setting the iteration times of outer loop; (3) ensuring that the optimal value searched by the algorithm should keep constant for several steps; (4) checking whether the system entropy is stable.

B. Empirical Calculation for Spatial Scheduling of Hull Blocks

1) Mathematical Description of the Scheduling Problem

The empirical case assumes a rectangular working platform with the width of W and height of H and a rectangular object with the width of w_i and height of h_i . The origin of two-dimensional Cartesian coordinate system is taken as the lower left corner of the rectangular working platform, i.e. *(0,0)* is the coordinate of the lower left corner of the rectangular working platform. (X_{wi}, Y_{hi}) is the coordinate of upper left corner of rectangular object to be placed. The purpose is to find a solution consisting of the set of n elements to maximize the contact surface s between the rectangular object and the working platform. In other words, the coverage of the working platform should be the maximum, as calculated by $S/(W \times H) \times 100\%$.

The objects should be placed at right angle. That is, each side of the rectangle should be parallel to X-axis or Y-axis. No rectangles could be placed obliquely.

2) Description of Algorithm

Step1: Initial temperature t_0, iteration times K (iterations of temperature recovery), and iteration times L (iterations of the same temperature) are set.

Step2: A new state is randomly generated with the state production function.

Step3: Rectangle placing algorithm is executed until the working platform can accommodate rectangles no more.

Step4: It is determined whether a new state should be accepted according to the Metropolis rule. Whether accepted or not, return to step 2 until iteration times L. The optimal state is identified under the current temperature. It is taken as the current state of the next temperature.

Step5: It is determined if the convergence criteria (the annealing termination criteria) are satisfied. If yes, go to step 6: if not, recover the temperature and let $ti=t0$. The optimal state at this time is taken to be the current state of the next moment. Step 2 is executed until iteration times K. The optimal solution is output.

Step6: The temperature update function is executed for temperature annealing, $t_{i+1}=a*t_i$ and a=0.9. Go to step 2.

3) Description of Empirical Calculation

The size of working platform is 10*8. A number of rectangles waiting for placing are given in the data Table1.

Step 1: Setting the initial temperature $t0$=1 and iteration times K=5, L=1000;

Step 2: Generating a new state to use the state production function under SA algorithm. Suppose the placing order of the new-state rectangles is {1, 2, 3, 4, 5, 6, 7, 8};

TABLE I.
RECTANGLES DATA

id	width	height
1	4	3
2	4	4
3	3	2
4	4	2
5	3	3
6	3	2
7	3	2
8	5	3

Step 3: Executing the rectangle placing algorithm:

(1) Selecting the lowest horizontal line; in the first time, the lowest horizontal line is initialized as the bottom side (0, 10, 0) of the working platform, respectively (abscissa of the starting point, the length w, the height h)

(2)Placing rectangle 1 (size 4*3), the width of the lowest horizontal line is larger than the width of rectangle 1 (w=10>4), and the surplus height of the working platform is larger than the height of rectangle 1 (h=8>3). Hence, rectangle 1 is placed here. The lowest horizontal line is updated as (4, 6, 0), as indicated by the dashed line in Figure3 (a).

(3) Rectangle 2 (size 4*4) is placed in the same way as rectangle 1, and the lowest horizontal line is updated to (8, 2, 0). When placing rectangle 3 (size 3*2), the width of the lowest horizontal line is smaller than the width of rectangle 3 (w=2<3), which means that rectangle 3 cannot be accommodated. Therefore, rectangle 3 is rotated for 90 degrees (exchanging width and height of rectangle 3). Then the width of the lowest horizontal line is equivalent to the width of rectangle 3 (w=2=2). Meanwhile, the surplus height of the platform is larger than the height of rectangle 3 (h=8>3). Thus, rectangle 3 can be placed here, as shown in Figure3 (b). The original lowest horizontal line is deleted, and the second lowest horizontal line is taken as a new lowest horizontal line.

(4) When placing rectangle 4 (size 4*2), that is as shown in Figure 3 (c). When placing rectangle 5 (size 3*3), that is as shown in Figure 3 (d). Rectangle 6 (size 3*2) and rectangle 7 (size 3*2) are placed normally.

(5) When placing rectangle 8 (size 5*3), the width of the lowest horizontal line is smaller than the width of rectangle 8 (w=1<5). Then rectangle 8 is rotated by 90 degrees, but the width of the lowest horizontal line is still smaller than the width of rectangle 8 (w=1<3). The lowest horizontal line (7, 3, 6) is updated, and the width of the lowest horizontal line now equals to the width of rectangle 8 (w=3=3). However, the surplus height of the work platform is smaller than the height of rectangle 8 (h=2<5), which means that rectangle 8 cannot be accommodated. The placing of rectangles is finished, and the output of the rectangle placing result is shown in Figure 3 (e).

Step 4: It is determined whether the new state is accepted according to Metropolis rule: The objective function is defined as the unoccupied rate of working platform C, C=(10 × 8-S)/10 × 8. The sum of the area of all rectangles which have been placed into the working platform is S. It is the first time to place the rectangles into the working platform. This new state is initialized as the current optimal state. When S=63, the objective function of current optimal state is C_0= (80-63)/80=0.2125.

Returning to step 2;

(1) Suppose the state production function generates a new state {5, 3, 2, 8, 6, 4, 7, 1};

The rectangle placing algorithm is executed, and the placing result is shown in Figure 3(f):

When S=66, the value of objective function of new state C_1 *is 0.175,* $C_1 < C_0$*. Hence the new state is accepted as the current optimal state. Let* $C_0 = C_1$.

Returning to Step 2;

(2) Suppose the state production function generates a new state {6, 3, 2, 1, 4, 8, 7, 5};

The rectangle placing algorithm is executed, and the placing result is shown in Figure 3(g):

When $S=48$, $C_1=0.4$. $C_1>C_0$; and

$exp[-(C_1—C_0)/t_0]=exp[-(0.4—0.175)/1]$
$=exp(-0.225)$

Then, $Min[1,exp(-0.225)]=exp(-0.225)$

Situation one:

Suppose *random* (0, 1) =0.999999,

Then *Min* [1, exp(-0.225)]< *random*(0, 1),

This does not satisfy the Metropolis Acceptance Criterion. Hence, the new state is not accepted, and the current state is kept unchanged.

Situation two:

Suppose *random*(0,1)=0.000001,

Then *Min*[1,exp(-0.225)]< *random*(0,1),

This satisfies the Metropolis Acceptance Criterion. Therefore, the new state is probably accepted as the new optimal state.

Let $C_0=C_1$;

Returning to Step 2;

(3)The operations above are repeated until iteration times L=1000 to find the optimal state when the temperature t_0=1.

Step5: It is determined whether the annealing termination criterion is satisfied. (The annealing termination criterion is as t_i<0.001)

It is clear that t_0=1>0.001, which means that the annealing termination criterion is not satisfied.

Turing to Step 6;(When t_0<0.001, let i=0, t_0=1, and $C_i=C_0$. The optimal state at t_i is taken as the current state at t0, and goes to Step 2. This procedure is repeated until iteration times K=5, then the global optimal state is output);

Step6:The temperature update function is executed for temperature annealing; $t_1=a*t_0=0.9*1=0.9$, i++;

The optimal state when t_0=1 is taken as the current state when t_1=0.9;

Returning to Step 2;

(1) Suppose the new state generated by the state production function is {8, 4, 3, 1, 2, 6, 5, 7};

The rectangle placing algorithm is executed. The result is shown in Figure 3 (h).

$S=63$, $C_1=0.2125$. $C_1>C_0$. At this time,

$exp[-(C_1—C_0)/t_0]=exp[-(0.2125—0.175)/1]$
$=exp(-0.0375)$

Then, $Min[1,exp(-0.0375)]=exp(-0.0375)$

Situation one:

Suppose *random* (0, 1)=0.999999,

Then *Min*[1, exp(-0.0375)]< *random*(0, 1),

This does not satisfy the Metropolis Acceptance Criterion. Therefore, the new state is not accepted, and the current state is kept unchanged;

Situation two:

Suppose *random*(0, 1)=0.000001,

Then *Min*[1, exp(-0.0375)]> *random*(0, 1),

This satisfies the Metropolis Acceptance Criterion. Then the new state is accepted as the current optimal state.

Let $C_0=C_1$;

Returning to Step 2;

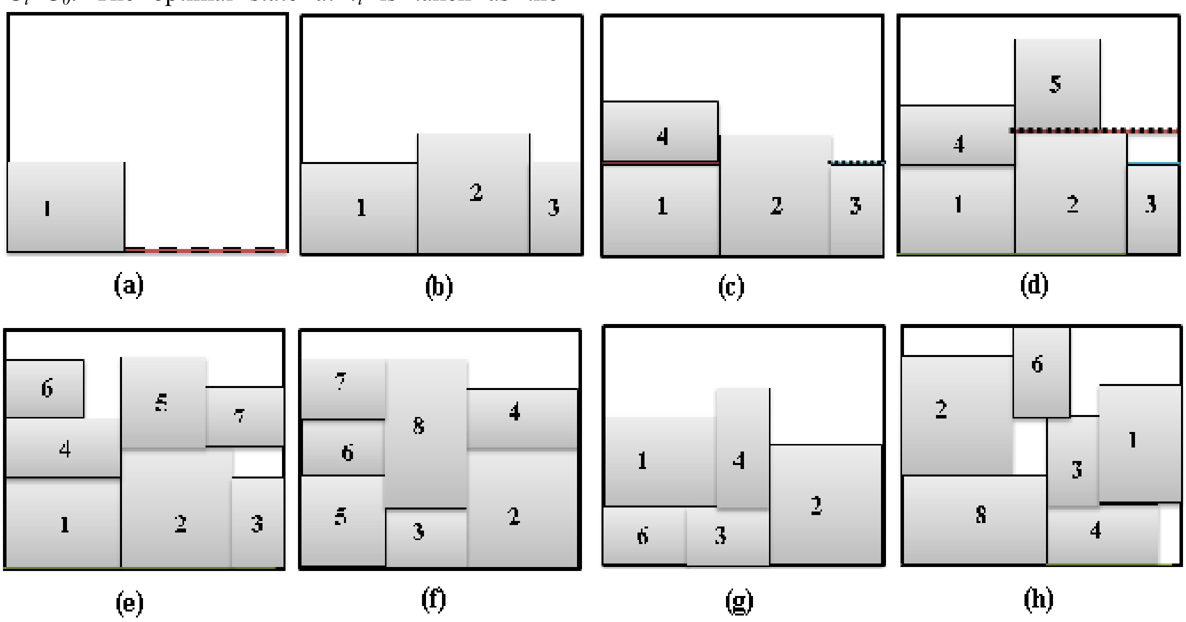

Figure 3 Module layout chart

IV. Empirical applications and Parameter Optimization

A. Examples of Applications

The manufacture area provided by the working platform of the shipyard is a rectangular area with length of 100 meter and width of 80 meter. Table2 shows the serial number and size of each hull block stored in the database.

The output result with the algorithm is shown in Figure5.

TABLE Ⅱ.
HULL MODULE DATA

id	width	height	id	width	height
No1	39	32	No17	40	20
No2	26	20	No18	37	32
No3	40	32	No19	20	19
No4	37	15	No20	20	20
No5	37	15	No21	20	20
No6	29	23	No22	20	19
No7	30	20	No23	37	15
No8	40	32	No24	37	15
No9	30	23	No25	32	9
No10	40	20	No26	20	15
No11	40	20	No27	20	19
No12	32	20	No28	31	20
No13	30	23	No29	38	15
No14	20	19	No30	32	9
No15	28	12	No31	37	15
No16	40	20			

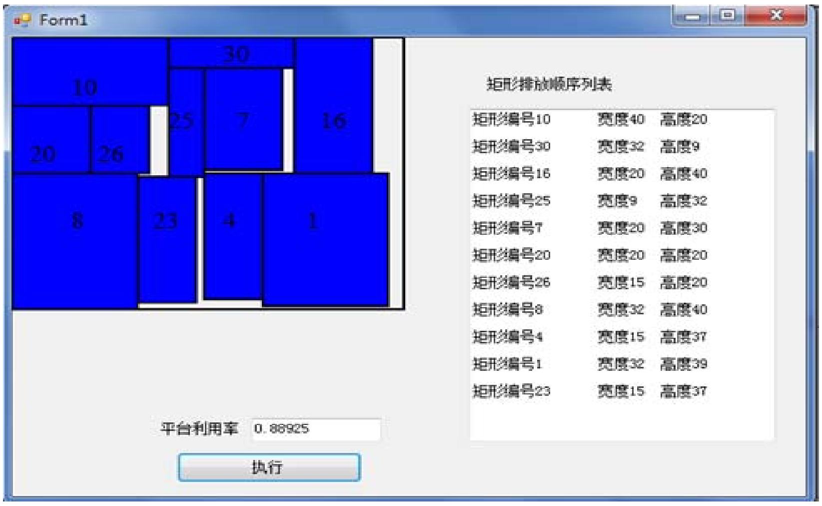

Figure 5 Result Output

Figure 5 shows the final placing state of hull blocks inside the working platform as well as the placing order. The utilization efficiency of the working platform is 0.88925.

B. Parameter Optimization

In the simulated annealing algorithm, several important parameter values might impact the experimental result. Those parameter values include initial temperature *t*, cooling parameter *a* and the number of sampling times *L* under each temperature. To investigate the influence on the result brought by the change of key parameters in the simulated annealing algorithm, the simulation experiments are carried out. Controlling variables, when a parameter is changed, the other parameters remain unchanged. Thus, the impact of the specific variable on the utilization efficiency of working platform *C* is estimated.

(1) The impact of initial temperature on the utilization efficiency of working platform is studied:

When the initial temperature *t* is chosen as the variable with other variables unchanged, the cooling parameter *a*=0.9, sampling times *L*= 100 and initial temperature *t* =

100, 50 and 10 in sequence. The imitation experiment is performed for 10 times under each temperature. The utilization efficiency *C* is recorded. Table 3 shows the utilization efficiencies of working platform *C* under different values of initial temperature, and the mean value is calculated.

TABLEIII.
EFFECTS OF DIFFERENT INITIAL TEMPERATURE ON *C*

	C_1	C_2	C_3	C_4	…	C_{10}	C_{mean}
t=100	0.758	0.683	0.862	0.768	…	0.763	0.7738
t=50	0.776	0.587	0.748	0.833	…	0.734	0.7105
t=10	0.760	0.672	0.733	0.721	…	0.746	0.7023

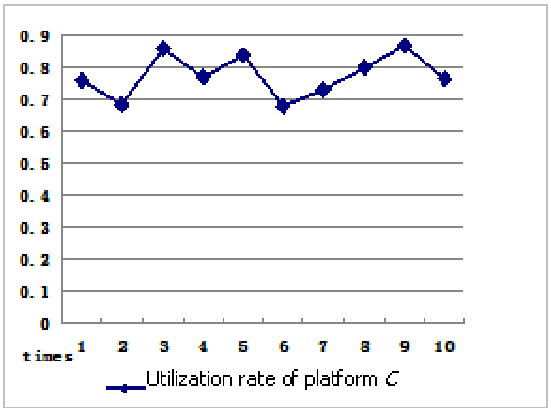

Figure 6 Line chart of *C at t*=100

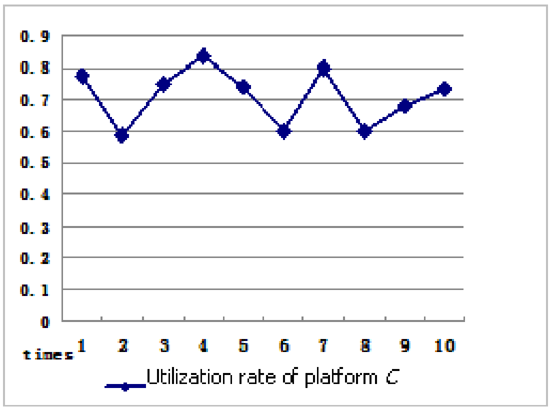

Figure7 Line chart of *C at t*=50

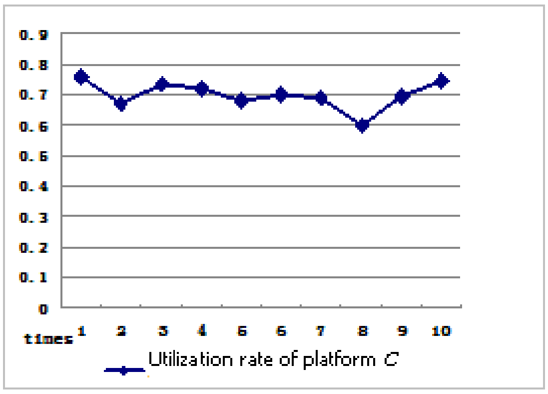

Figure8 Line chart of *C at t*=10

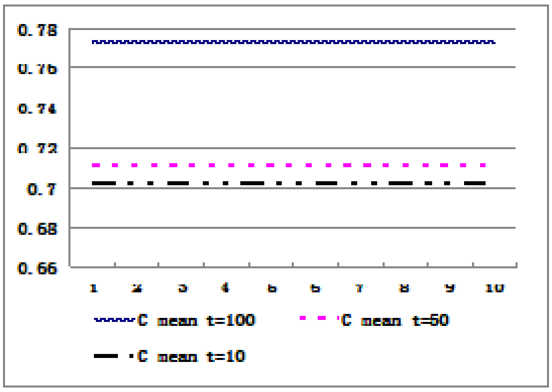

Figure9 *C* mean under different initial temperature

Figure 6, 7 and 8 show the scatter plot and line chart of the utilization efficiency of the working platform when the initial temperature is 100, 50 and 10 respectively. Figure 9 shows the mean value of utilization efficiency C at different temperatures. It is easy to see that the solution obtained by the simulated annealing algorithm is better if the initial temperature is set higher.

(2) The method of studying the influence of cooling parameter and sampling times on the objective function is the same as that for initial temperature. The presentation is simplified by using Table4 and Table5.

TABLE Ⅳ

THE DIFFERENT COOLING PARAMETERS EFFECT ON C

	C_1	C_2	C_3	C_4	…	C_{10}	C_{mean}
a=0.9	0.645	0.812	0.692	0.580	…	0.725	0.764
a=0.6	0.662	0.728	0.703	0.781	…	0.649	0.760
a=0.3	0.739	0.668	0.622	0.625	…	0.585	0.683

TABLE V.
THE DIFFERENT SAMPLING TIMES EFFECT ON C

	C_1	C_2	C_3	C_4	…	C_{10}	C_{mean}
L=100	0.649	0.753	0.638	0.795	…	0.746	0.775
L=60	0.680	0.712	0.671	0.754	…	0.784	0.732
L=20	0.691	0.627	0.631	0.718	…	0.697	0.694

Experiment results have proved that the SA algorithm will search a better solution if the initial temperature is higher, and the cooling speed is lower and the number of sampling times is enough. But the optimization with SA algorithm takes more time. Both the optimization quality and speed should be considered when choosing the values of key parameters. Finally, the parameters in the algorithm are selected as initial temperature t=50, cooling parameter a=0.9 and sampling times L=100.

V.CONCLUSION

As an effective scheduling approach, block assembly planning helps more and more shipbuilding enterprises get benefits from manufacturing flexible and effective control. Aim at the maximum use of the working platform, this study identifies the model and the algorithm for block assembly efficient layout in the working platform during the hull block production. The spatial scheduling problem of hull blocks is transformed into a two-dimensional bin packing problem based on the factors analysis of hull block division. The feasible layout location is expressed in mathematics algorithm, and it creates the conditions for the computing the spatial block scheduling. In the empirical applications, the spatial module scheduling algorithm is upgraded by combining SA algorithm with the lowest horizontal-line rotate searching algorithm. By optimizing the parameters, the simulated results show the better planning resolution and practical application value.

REFERENCES

[1] Kumar A.A., Kazuhiro A. “Evaluation and management of the cost and risk of welded formations in modular ship construction”. *Journal of Ship Production*. 2005, 21(1), pp. 8-13.

[2] Hifi M,V.Zissimopolous. “Constrained two-dimensional cutting:An improvement of Christ fides and Whitlock’s exact algorithm”*J.Oper.Res.Soc*.1997,48,pp.324-331.

[3] E.K.Burke, G.Kendall, G.Whitwell. “A New Placement Heuristic for the Orthogonal Stock-Cutting Problem”. *Operations Research*, 2004, 52(4), pp. 655-671.

[4] Defu Zhang,Yan Kang,Ansheng Deng. “A new heuristic recursive algorithm for the strip rectangular packing problem”. *Computer& Operations Research*, 2006, 33(8), pp. 2209-2217.

[5] Yu-Liang Wu,Wenqi Huang,Siu-chung Lau,C.K.Wong. “An effective quasi-human based heuristic for solving the rectangle packing problem”. *European Journal of Operational Research*, 2002,141,pp.341-358.

[6] Wenqi Huang,Duanbing Chen,Ruchu Xu. “A new heuristic algorithm for rectangle packing”. *Computers&Operations Research*, In Press.

[7] Dagli C H,Hajakbari.A simulated annealing approach for solving stock cutting problem. In Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 1990, pp.221-223.

[8] Lai,K.K.,J.W.M.Chan.Developing a simulated annealing algorithm for the cutting stock problem.Comput.Indust.Engrg,1996,32(1),pp.115-127.

[9] Faina,L.An application of simulated annealing to the cutting stock problem Eur.J.Oper Res1999,114,pp.542-556.

[10] Burke E.K,Kendall G,Whitwell G. Metaheuristic enhancements of the best-fit heuristic for the orthogonal stock cutting problem[R].Computer science technical report no.NOTTCS-TR-SUB-0605091028-4370,University of Nottingham,2006.

[11] Dagli,C.H.,P.Poshyanonda. New approaches to nesting rectangular patterns. Intelligent Manufacturing, 1997,3(3),pp.177-190.

[12] Krager B. Guillotineable bin packing: a genetic approach.European Journal of Operational Research, 1995, 84, pp. 645-661.

[13] Bortfeldt A.A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces.European Journal of Operational Research,2006,172, pp. 814-837.

[14] Lodi A,Martello S,Vigo D .Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. INFORMS Journal on Computing, 1999, 11, pp. 345-357.

[15] Lodi A,Martello S,Vigo D.Neighborhood search algorithm for the guillotine non-oriented two-dimensional bin packing problem. In:Voss S,Martello S,Osman I H,Roucairol C.(Eds.), Meta-Heuristics:Advances and Trends in Local Search Paradigms for Optimization,Kluwer Academic Publishers:Boston,1998, pp. 125-139.

[16] Lodi A,Martello S,Vigo D.Approximation algorithms for the oriented two-dimensional bin packing problem.European Journal of Operational Research,1999,112,pp.158-166.

[17] E.Hopper,B.C.H.Turton.An empirical investigation of meta-heuristic and heuristic algorithms for a 2D packing problem[J].European Journal of Operational Research,2001,128,pp.34-57.

Dequan Zheng was born in Zhejiang province, China. He got the Bachelor degree of economics from Harbin Institute of Technology in 1997 Harbin, and Master of Laws from Xi’an Jiaotong University in 2000 Xi’an, China. And now he is also a PhD. candidate in public administration in Harbin Institute of Technology. He is Lecturer of Public Administration at Harbin Institute of Technology in Harbin.