BULETINUL ACADEMIEI DE ŞTIINŢE
A REPUBLICII MOLDOVA. MATEMATICA
Number 3(64), 2010, Pages 23–28
ISSN 1024–7696

A heuristic algorithm for the two-dimensional single large bin packing problem

V. M. Kotov*, Dayong Cao

Abstract. In this paper, we propose a heuristic algorithm based on concave corner (BCC) for the two-dimensional rectangular single large packing problem (2D-SLBPP), and compare it against some heuristic and metaheuristic algorithms from the literature. The experiments show that our algorithm is highly competitive and could be considered as a viable alternative, for 2D-SLBPP. Especially for large test problems, the algorithm could get satisfied results more quickly than other approaches in literature.

Mathematics subject classification: 34C05.
Keywords and phrases: Rectangular packing, heuristic, best-fit, concave corner, fitness value.

1 Introduction

Packing problem involves many industrial applications. For example, wood or class industries, ship building, textile and leather industry etc. All of these applications can be formalized as packing problem [1], for more extensive and detailed descriptions of packing problems, please refer to [1–4].

In this paper, we discuss the two-dimensional single large bin packing problem (2D-SLBPP). The problem could be described as follows:

Given a rectangular board with fixed size and a set of rectangular pieces. The research of 2D-SLBPP is how to pack rectangular pieces orthogonally on the board, in the meantime, try to decrease the worst of the board with no two pieces overlap.

2 A new heuristic packing algorithm for single bin packing

2.1 Placement strategy based on Concave Corner (BCC)

Before the description of our algorithm, suppose the width and height of rectangular board are W and H. Without loss of generality, all parameters are regarded as integer. The pieces should be packed with edges parallel to the edges of the board and couldn't be rotated by 90^{o}.

For constructing our heuristic algorithm, we propose some definitions and rules.

*This author is corresponding author

Some definitions

1. Let C_i denote the "Concave corner" (CC), see Figure 1, the CC is composed by two edges, and the size of the angle is 90^o, at the same time, the CC does not belong to any $piece_i$.

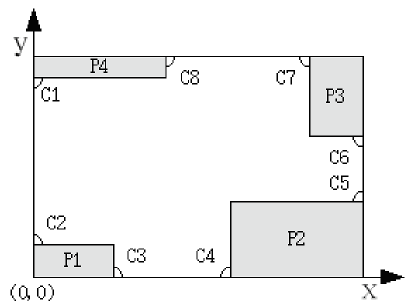

Figure 1. The example of *Concave corner*

2. Define the U, which is stated as formula (1):

$$U = \{U_1, U_2, \dots, U_k\}; \quad U_i \cap U_j = \varnothing, \quad i \neq j, \tag{1}$$

where the U_i is a set of the CC, and k is the number of non-connected domains in the board, for example, see Figure 2, before the P_5 is packed onto the board, we have $U = U_1$, $U_1 = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\}$, after packing the P_5, the U_1 should be divided into two areas, then we have: $U = \{U_1^{'}, U_2^{'}\}$, $U_1^{'} = \{C_1, C_7, C_8, C_9, C_{12}\}$, $U_2^{'} = \{C_2, C_3, C_4, C_{10}, C_{11}\}$. Obviously, before packing any piece onto the board, there exists 4 CC and $k = 1$.

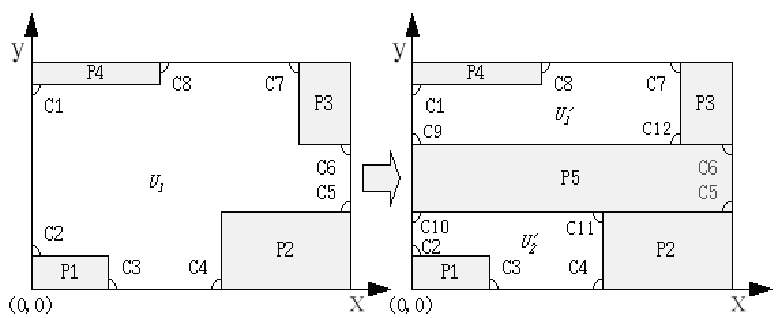

Figure 2. Dividing U_1 into $U_1^{'}$ and $U_2^{'}$

3. Define the edge of the U_i: Before any piece is packed onto the board, let l_U_1 denote the left edge of the U_1, and $r\_U_1 = W$ denote the right edge of the U_1. Similarly, we could define the t_U_1 and b_U_1 to denote the top edge and bottom edge of the U_1. So if U_i was divided into $U_e^{'}$ and $U_f^{'}$, we should update these parameters using the formulae (2-5):

$$l\_U_e^{'} = Min\{x_j^{'}\},\ 1 \leqq j \leqq s,\ \ C_j^{'} \subset U_e^{'},\ \ x_j^{'}\ \ is\ \ x - coordinate\ of\ the\ C_j^{'}; \tag{2}$$

$$r\_U_e^{'} = Max\{x_j^{'}\},\ 1 \leqq j \leqq s,\ \ C_j^{'} \subset U_e^{'},\ \ x_j^{'}\ \ is\ \ x - coordinate\ of\ the\ C_j^{'}; \quad (3)$$

$$b\_U_e^{'} = Min\{x_j^{'}\},\ 1 \leqq j \leqq s,\ \ C_j^{'} \subset U_e^{'},\ \ y_j^{'}\ \ is\ \ y - coordinate\ of\ the\ C_j^{'}; \quad (4)$$

$$t\_U_e^{'} = Max\{x_j^{'}\},\ 1 \leqq j \leqq s,\ \ C_j^{'} \subset U_e^{'},\ \ y_j^{'}\ \ is\ \ y - coordinate\ of\ the\ C_j^{'}; \quad (5)$$

where s is the number of CC in $U_e^{'}$.

Note. After a new piece is packed onto the board, if no U_i was divided, the edges of U_i should not be changed.

4. When a new piece $piece_i$ is packed onto the board, let s denote the number of edges which is touched with some packed $piece_h$ for the position of one C_k, if the $piece_i$ could be packed, then we compute the parameter $pFit\_C_k$ using formula (6):

$$pFit\_C_k = \sum_{j=1}^{s} p_j, \quad (6)$$

if the $piece_i$ is packed onto C_k with corner of the piece (query every U_m), the $piece_i$ touches one edge of the U_m ($p_j = 2$) and the $piece_i$ touches one edge of the other packed piece ($p_j = 1$), see Figure 3.

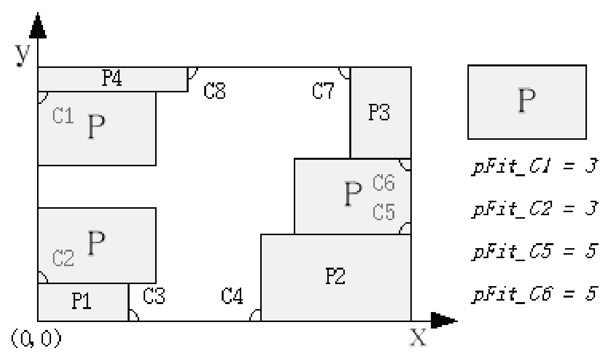

Figure 3. Computation of $pFit\_C_i$ in every U_k

5. Define the edge distance of C_k:

$$ed\_C_k = Min\{the\ distance\ between\ vertex\ of\ C_k\ with\ the\ edge\ of\ U_e\} \quad (7)$$

if $C_k \subset U_e$.

Packing rules. When $piece_i$ is packed onto the board, compute the $pFit\_C_k$ of every C_k, and select the packing position with maximal $pFit\_C_k$, if $pFit\_C_k$ are equal to each other, then select the position with ed_C_k is the shortest, then if the ed_C_k is the same, select the position randomly.

After completing the definitions and packing rules, we construct the heuristic algorithm based on the concave corner (BCC) as algorithm 1:

Algorithm 1 heuristic Packing (packing sequence)

```
s ⇐ 0;
i ⇐ 0;
while packing sequence is not null do
   get piece_i from the packing sequence;
   using the packing rules mentioned above to get good position for piece_i;
   if good position exists then
      pack the piece_i into the board at the good position;
      remove the piece_i from packing sequence;
      s ⇐ s + area of picec_i;
      continue;
   end if
   i ⇐ i + 1;
end while
return s;
```

2.2 Random search

Since the result of the heuristic packing (BCC) depends on the order of the packing sequence, so we import a random search to enhance the quality of the solution, which is described as follows:

Algorithm 2 middle Heuristic (origin data of all pieces, maxcall)

```
produce a packing sequence according to the area of all pieces from big to small;
best ⇐ 0;
area ⇐ 0;
swapLimit ⇐ pieces number × 1 / 3;
for i = 0 to maxcall do
   area = heuristicPacking(packing sequence);
   if area equals to the total area of all pieces then
      break;
   end if
   if area > best then
      best ⇐ area;
   end if
   for j = 0 to swapLimit do
      swap the pieces order of packing sequence randomly;
   end for
end for
```

3 Experiments

We implemented our algorithm by C++ programming language, and the 21 rectangular packing instances coming from [5] are used. For evaluating the algorithm more reasonably, we set the maxcall is 1000 and run the program 100 times. our experiments are run on a IBM T400 notebook PC with 2.26 GHZ CPU, GRASP is introduced in [6] and tabu search algorithm is presented in [7] (TABU), both GRASP and TABU were run on a Pentium III at 800 MHz, which is almost thrice as slow as ours. The test results are listed in Table 1.

Table 1. Comparisons of the average filling rate (FR) and the average running time (T)

Instance	Area	Number of items	GRASP		TABU		BCC	
			FR(%)	T(s)	FR(%)	T(s)	FR(%)	T(s)
1	400	16	100	0.94	100	0.42	100	0.13
2	400	17	96.5	9.28	100	4.23	96.5	1.10
3	400	16	100	0.06	100	0.95	100	0.24
4	600	25	98.33	19.44	100	0.44	96.83	2.47
5	600	25	99.5	17.36	100	4.16	99	2.27
6	600	25	100	0.71	100	0.0	99.33	1.97
7	1800	28	98.06	26.80	100	4.91	96.72	3.21
8	1800	29	97.5	37.35	100	10.11	95.56	3.33
9	1800	28	98.56	30.92	100	5.52	97.33	2.58
10	3600	49	98	102.05	99.44	45.27	96.83	9.43
11	3600	49	97.89	110.79	99	67.59	98.19	9.18
12	3600	49	98.44	94.41	99.44	51.11	98.47	8.72
13	5400	73	98.3	212.07	98.93	135.97	97.63	26.22
14	5400	73	98.39	231.56	99.28	96.80	97.39	28.38
15	5400	73	98.37	231.24	99.54	82.06	97.39	26.50
16	9600	97	98.65	480.44	99.46	240.39	98.06	53.76
17	9600	97	98.47	465.49	98.42	399.86	98.21	57.33
18	9600	97	98.44	478.02	99.64	206.78	98.02	52.89
19	38400	196	98.08	3760.14	99.03	3054.38	98.35	311.56
20	38400	197	98.8	2841.96	99.34	1990.70	98.80	360.42
21	38400	196	98.29	3700.99	98.61	5615.75	98.39	324.49

4 Conclusion

In this paper, a heuristic algorithm BCC based on the random search method for the two-dimensional single large bin packing problem is proposed, the experiments show that our algorithm is highly competitive and could be considered as a viable alternative for 2D-SLBPP. Especially for large test problems, the algorithm could get satisfied results more quickly than other approaches in the literature. Furthermore, if BCC could combine with some appropriate intelligent optimization methods, we think that it could get better optimal solutions in acceptable time.

References

[1] Lodi A., Martello S., Monaci M. *Two-dimensional packing problems: A survey.* European Journal of Operational Research, 2002, No. 141(2), 241–252.

[2] Dyckhoff H. *A typology of cutting and packing problems.* European Journal of Operational Research, 1990, No. 44(2), 145–159.

[3] Editorial. *Cutting and Packing.* European Journal of Operational Research, 2007, No. 183, 1106–1108.

[4] Wäscher G., Haussner H., Schumann H. *An improved typology of cutting and packing problems.* European Journal of Operational Research, 2007, No. 183(3), 1109–1130.

[5] Hopper E., Turton B. C. H. *An empirical investigation of meta-heuristic and heuristic algorithms for a 2D packing problem.* European Journal of Operational Research, 2001, No. 128(1), 34–57.

[6] Alvarez-Valdes R., Parreño F., Tamarit J. M. *A GRASP algorithm for constrained two-dimensional non-guillotine cutting problems.* Journal of Operational Research Society, 2005, No. 56(4), 414–425.

[7] Alvarez-Valdes R., Parreño F., Tamarit J. M. *A tabu search algorithm for two-dimensional non-guillotine cutting problems.* European Journal of Operational Research, 2007, No. 183(3), 1167–1182.

Received March 1, 2010

V. M. Kotov
Department of Applied Mathematics
and Computer Science
Belarusian State University
Minsk 220030, Belarus
E-mail: *kotovvm@yandex.ru*

Dayong Cao
Department of Applied Mathematics
Harbin University of Science and Technology
XueFu road, 52, Harbin 150080 China
E-mail: *caodayong@hrbust.edu.cn*