Experimental Study of Rectangle Packing in VLSI

Rishabh Kaur Dingra[1], Dr.Bharthi S.H[2]

[1]*ECE Department ,REVA University*
[2]*ECE Department, REVA University*

---***---

Abstract - Rectangle packing problem is a NP-Hard problem. Its solution space is infinite and many algorithms and techniques has been proposed for the optimal packing of rectangles. One finite set of solution space is P-admissible where every solution is feasible, evaluation of each code is possible in polynomial time and the packing of best evaluated code in the space coincides with an optimal solution of rectangle packing. In this type of packing every packing corresponds to a pair of code called sequence-pair .Sequence-pair is a pair of positive loci and negative loci of blocks through which the geometrical information of each module is defined which helps to know the relative location of each modules and depending on which vertex directed acyclic graph is drawn. Horizontal constraint graph (HCG) is based on right-to constraint while vertical constraint graph (VCG) is based on below constraint. Finally based on Longest path calculation the width of the module is derived from HCG and the height of module is derived from VCG. Since the width and height of the chip is independently minimum, the resulting packing is optimal..

***Key Words*:** NP-hard , polynomial time , Directed acyclic graph

1.INTRODUCTION *(Size 11, Times New roman)*

VLSI Placement is a field of arranging or placing devices in a defined area following the constraints for the good placement or proper functionality of the chip.It is an important step of physical design cycle as it improves the performance of the circuit ,improves the routing(interconnection between the blocks) also it distributes the heat properly to avoid the malfunctioning of the circuit.VLSI placement also determines how much of the power will be consumed by the circuit.If the performance of the circuit is high having low power consumption with uniform heat distribution and having no congestion results into a good placement otherwise results into a bad placement. There are different styles of packing the modules or blocks(each module or device are reffered as a block) in a defined area.For example circle packing, rectangle packing,square packing, triangle packing,bin packing,strip packing,orthogonal packing into a specified area. According to the need or demand the packing style is chosen. Packing are classified into various categories like packing according to dimension is classified as 1D packing , 2D packing, 3D packing. In 1D packing one dimensional objects are placed in a unit box .2D packing deals with the packing of two dimensional objects like rectangle, square, triangle, circle etc. While 3D packing deals with packing of three dimensional objects like cube, cuboid, hypercube etc. According to shape packing are classified as square packing, Rectangle packing, Circle packing. In square packing square of different or same size are packed in a big square. In Rectangle packing rectangles of different or same size are packed in a big rectangle. Circle packing deals with the packing of circles into a big circle.According to the information of objects the packing are classified as online packing, offline packing, almost online packing. In offline packing the information of the objects are known i.e. the shape or size of the object and the packing arrangement can be changed further but in online and almost online packing the information about the object is not known in advance so once the packing is done it cannot be arranged further. According to packing problem it is classified as Bin packing, Strip packing, pallet loading, Knapsack problem. In bin packing the containers(bins) of fixed size are packed into containers so that the bins are packed in a minimum number of containers. In strip packing the objects having fixed width are placed into a big object of fixed width such that the height of the big object is minimized. Pallet loading deals with packing of large container into a fixed sized container orthogonally. In Knapsack problem each object is associated with some cost so that the object are to be packed with minimum cost.

Rectangle Packing Problem(RPP) is considered to be a NP hard or NP complete since there are many solutions i.e solution space is infinite and many techniques or optimum algorithms are proposed .

Input : Sets of rectangle of different size.
: Enclosing Rectangle of width W and height H.
(orientation can be change like 90 degree).

Objective : To place small rectangles into a big rectangle such that there will be no overlapping and minimum area is used.

Technique : Many techniques or optimum algorithms are Proposed like Simulated Algorithm(SA) , Genetic Algorithm(GA),Sequence Pair(SP) Technique.

.

2. Application

Rectangle packing concept is used in different industries like in wood or glass or fabrication industry rectangular sized pieces are to be cut from a big rectangular sheet so that maximum pieces can be obtained. In newspaper also rectangular sized columns or advertisements are to be placed in a fixed size of paper. In field of construction also maximum number of plots need to be defined within in a specified area. In placing of tiles also this concept is used to place the tiles in a fixed rectangular area. In keyboard designing also the small keys are to placed in a compact rectangular surface.

3. Literature Review

According to [34] Two dimension rectangle packing problem is a subset of cutting and packing problem. Many algorithms have been proposed for the solution of rectangle packing problem.The traditional heuristic algorithm and the meta-heuristic algorithms are the two categories of these algorithm[44]. Mostly used heuristics are Bottom-left(BL) and Bottom-left filled methods. Simulated annealing , genetic algorithm are the meta heuristic approaches. These two heuristic and meta-heuristic approach were proposed by Lodi et al.(1999).Simulated annealing and genetic algorithms are also effective for combinatorial optimization problem.

In 2009 L. Wei [44] aims to maximize the packing rate of rectangular knapsack problem for which introduces a least wasted first strategy and a random local search.

In 2010 B.Wang uses Dynamic attractive factors presented by P.C Gilmore(1961) for two dimensional packing problem results into improved layout result.

K.Singh and Leena Jain [45] used AYC Nee's Rectangle packing heuristic presented by them in 2009 for the further feasible patterns.

Genetic algorithm(GA) proposed by J. Holland et al. (1975) has five main components presented by Vignaux et al.(1991).1)solution representation 2) strategy to create starting population 3)Evaluation function for the determination of solutions 4)Genetic operators and 5) values of parameters.

In 2011 Bitao Peng [46] aims to use the maximum area of bounding rectangle using Bottom Left algorithm and tree greedy search strategy for two dimensional Rectangle Packing Problems (RPP). Bottom Left(BL) algorithm introduced by Baker et al.[1] was the most popular approach further this approach was changed to a new version by various heuristics and an exact method using tree search techniques was proposed by Beasley JE in 1985.

In 2012 A.Bortfeldt[49] uses 2D Strip packing problem(SPP) for feasible placement of the rectangle in enclosing rectangle of fixed width and minimum length and 2D Knapsack Problem(K-P) for feasible placement of subset of rectangle in enclosing rectangle with given width and length.In two dimensional cutting and packing problem Orientation Constraint and Guillotine Cutting Constraint plays an important role.These constraints have following four subset proposed by Lodi et al.(1999).

Table -1: Constrained proposed by Lodi et al. in 1999

OF	Orientation of all pieces is fixed(O) and Guillotine cutting is not required(F)
OG	Orientation of all pieces is fixed(O) and Guillotine cutting is not required(G)
RF	Pieces may be rotated by 90 degree(R) and Guillotine cutting is not required(F)
RG	Pieces may be rotated by 90 degree(R) and Guillotine cutting is required(G)

In 2010 K.He et al. [48] proposed a Best fit algorithm (BFA)for two dimensional rectangle packing problem using algorithm of three dimensional rectangle packing problem.

In 2013 E.Huang and R.E.Korf [50] introduces a new approach by choosing Y-coordinates after X-coordinates of all rectangles and proposed three new benchmarks for optimal Rectangle Packing. Also the comparisons have been made with several competing search spaces for rectangle packing.

The Rectangle Packing Area Minimization Problem(RPAMP) has two main approach one is heuristic searching method based on layout representation second one is reduction approach for Strip Packing Problem(SPP) Other approaches are Branch and Bound technique and Linear Optimization technique. The most important approach for heuristic search is layout representation which is classified as slicing representation and non slicing representation. Non slicing representation are proposed by Murata et al.(1996) based on sequence pair, Guo et al. (1999) based on O-tree , Chang et al. (2000) based on B-tree and Lin et al. (2005) based on Transitive constraint graph(TCG).

In 2014 Kun He et al. [51] proposed a Dynamic Reduction Algorithm(DRA) and also mentioned that the layout representation is classified as slicing representation and non slicing representation and concluded that non slicing representation is better.According to packing zone Rectangle Packing Area Minimization Problem(RPAMP) is classified as Strip Packing Problem(SPP) and Rectangle Packing Problem(RPP).In 2016 Lei Wu et al.[53] introduces a concept of placing central rectangle in a center of final layout and proposed a heuristic algorithm called HACR for solving Central Rectangle- Rectangle Packing Area Minimization Problem(CR-RPAMP).In 2017 Lei Wu et al.[54] Further introduces a improved version of heuristic and named as IHACR. HACR [53] was having a disadvantage of consuming large time so to overcome the disadvantage IHACR[54] was proposed.IHACR uses three strategies: strategy of combining rectangles, leaving larger inner space and removing unnecessary comparisions.

4. Rectangle Packing Problem

Rectangle packing based on sequence pair [5] is studied and analyzed. H. Murata et al. [5] introduces a P-admissible solution space. It is a set of finite solution space where every solution is feasible and the code evaluation and packing is possible in polynomial time so the best code packing leads to optimal packing.

Let us consider a set of rectangles(blocks) r_i ={r_1, r_2, r_3, r_4, r_5} having width and height (w,h)={(2,2),(3,3),(2,1),(3,1),(2,1)} these rectangles are to packed in a big rectangle R (enclosing rectangle) having height H and fixed width W (W,H)=(6,5) such that there will be no overlapping and minimum area is used.

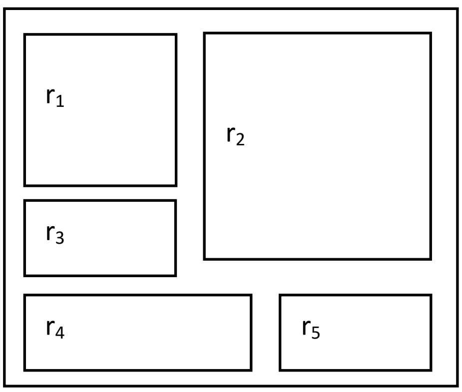

Fig -1: Figure

4.1 Getting a sequence pair from initial packing

Sequence pair is a pair of positive loci P+ and negative loci N-. Positive loci is the union of right up locus and left down locus of each block. Whereas negative loci is the union of up left locus and down right locus of each block.

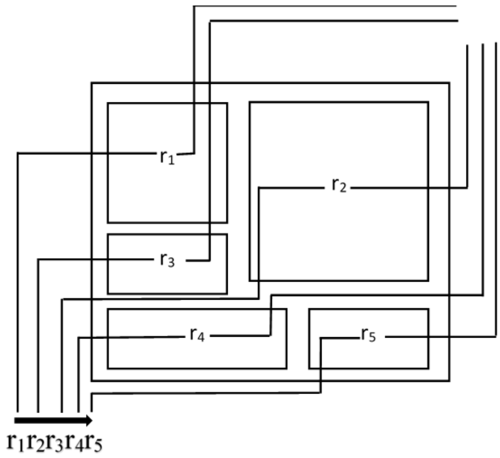

Fig -2: Positive loci P_+

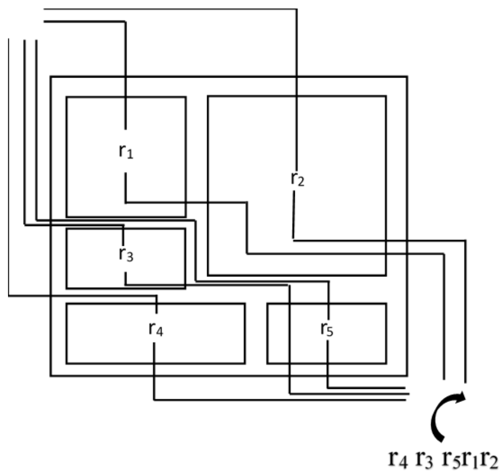

Fig -3: Negative loci N_-

Therefore the sequence pair is (P_+, N_-)=($r_1r_3r_2r_4r_5$, $r_4r_3r_5r_1r_2$).

4.2 Relative location of blocks

Each block is placed either right, left, above and below with respect to other blocks. To know the relation location a table is drawn below.

Ex: r2 is right to r1.

(Note: If no block is present that is indicated as null)

Table -2: Relative Location

Block Name	Right	Left	Above	Below
r_1	r_2	Null	Null	r_3, r_4, r_5
r_2	null	r_1, r_3	Null	r_4, r_5
r_3	r_2, r_5	Null	r_1	r_4
r_4	r_5	Null	r_1, r_2, r_3	Null
r_5	Null	r_3, r_4	r_1, r_2	Null

4.3 Horizontal and Vertical Constraint Graph

To get a optimal packing first we need to draw a weighted directed acyclic graph(DAG). The formation of DAG is explained below.

First draw a grid at an angle 45 degree. The size of grid is the total no. of blocks minus 1 i.e. if m no. of blocks are there then size of the grid is [m-1 x m-1]. As we have taken 5 blocks so our grid size will be [4 x 4]. Then write the P+ from up to down at left side of the grid and N- from down to up at right side of the grid.Now place the respective block at the horizontal and vertical intersection point of respective block.

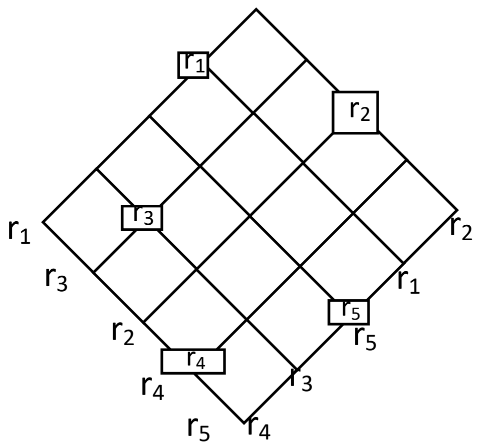

Fig -4: Figure

For Horizontal constraint graph (HCG) follow the algorithm A1 below and follow the Table3.1 to know which block is left to or right to.

Let S be the Sink(starting point having weight zero)

Let T be the target(end point having weigth zero)

Consider each block as a vertex Vi (i= 1 to 5) having weight Wi .

Algorithm A1: Construction of HCG

START: If there exist a vertex V_i which is left to V_j
connect that two vertices
Else
connect the vertex V_j with S
Goto START until each vertex is visited
End If
If there exist no vertex V_j which is right to V_i
then connect V_j to T
End If
For HCG
W_i = width of the block
End for

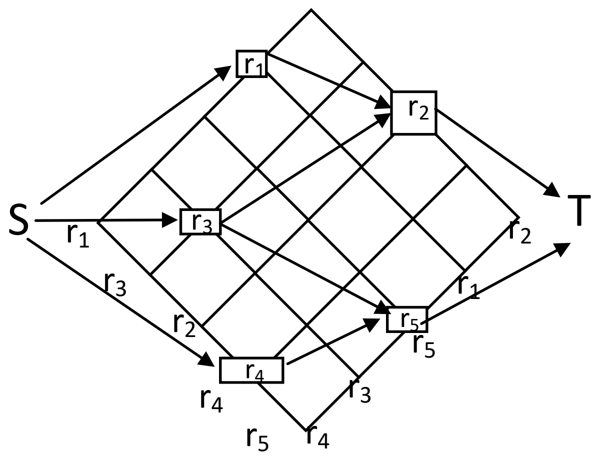

Fig -5: Horizontal Constraint Graph

For Vertical Constarint Graph(VCG) follow the algorithm A2 below and follow the Table3.1 to know which block is above to or below to.

Algorithm A2: Construction of VCG

START: If there exist a vertex V_i which is below to V_j
connect that two vertices
Else
connect the vertex V_j with S
Goto START until each vertex is visited
End If
If there exist no vertex V_j which is above to V_i
then connect V_j to T
End If
For VCG
W_i = height of the block
End for

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 07 | July - 2021 ISSN: 2582-3930

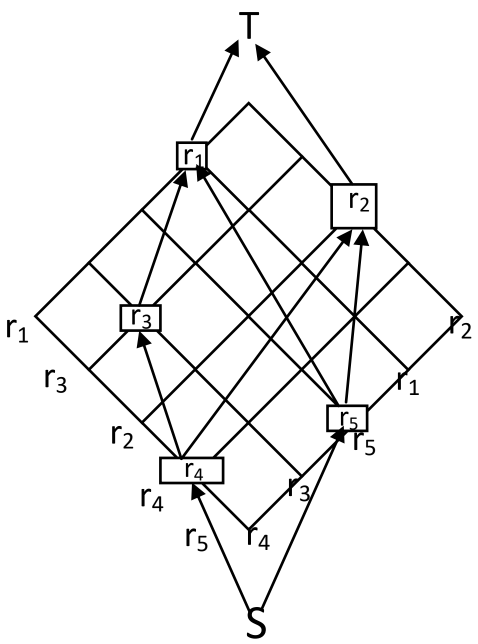

Fig -6: Vertical Constraint Graph

4.4 Optimal Packing

Based on the HCG and VCG we proposed a program using c language to get one of the optimal packing . Through this program giving the information of HCG the program calculates the x- coordinate of the block and giving the information of VCG the program calculates the Y-coordinate of the blocks.

```
#include<stdio.h>
#include<conio.h>
main()
{
      int n,i,edgecount,j,k,large=0,l,m;
      int p[20]={0};
      int pathvalue[10]={0};
      int x[20]={0};
      printf("Enter how many blocks are there :");
      scanf("%d",&n);
      for(i=1;i<=n;i++)
{
printf("\n How many edges incoming to vertex%d : ",i);
scanf("%d",&edgecount);
printf("\n Enter the vertex index value of all incoming
vertex to vertex%d : ",i);
              for(j=1;j<=edgecount;j++)
              {
               scanf("%d",&p[j]);
              }
                     for(k=1;k<=edgecount;k++)
                     {
                      printf("\nEnter the path value of vertex%d
                      via  vertex%d from source : ",i,p[k]);
                      scanf("%d",&pathvalue[k]);
                      }
                     large= pathvalue[0];
                              for(l=1;l<=edgecount;l++)
                              {
                               if(large< pathvalue[l])
                               large= pathvalue[l];
                               }
                               x[i]=large;
 }
 printf("\n  Coordinates are:\n");
 for(m=1;m<=n;m++)
{
           printf("  %d  ",x[m]);
 }
 return 0;
}
```

Table -3: X and Y coordinate

Block name	r_1	r_2	r_3	r_4	r_5
X-coordinate	0	2	0	0	3
Y-coordinate	2	1	1	0	0

Now place the blocks of respective (w,h) w.r.t the X-coordinate and Y-coordinate we will get one of the optimal packing.

Now place the blocks of respective (w,h) w.r.t the X-coordinate and Y-coordinate we will get one of the optimal packing.

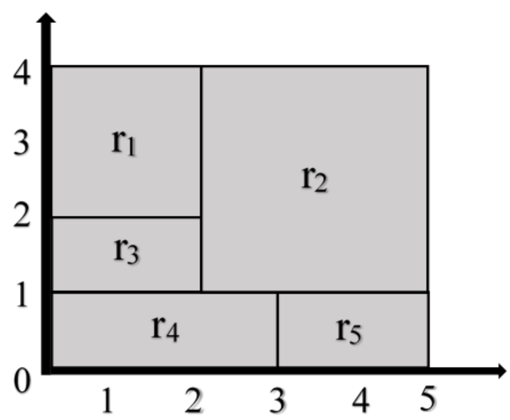

Fig -7: Optimal Packing

4.5 Packing for 8 blocks

Similarly we tried packing for 8 blocks of

(h,w) ={(2,3),(8,5),(4,3),(2,3),(3,3),(3,6)(3,3),(7,2)} in a big rectangle of (H,W) =(12,13).

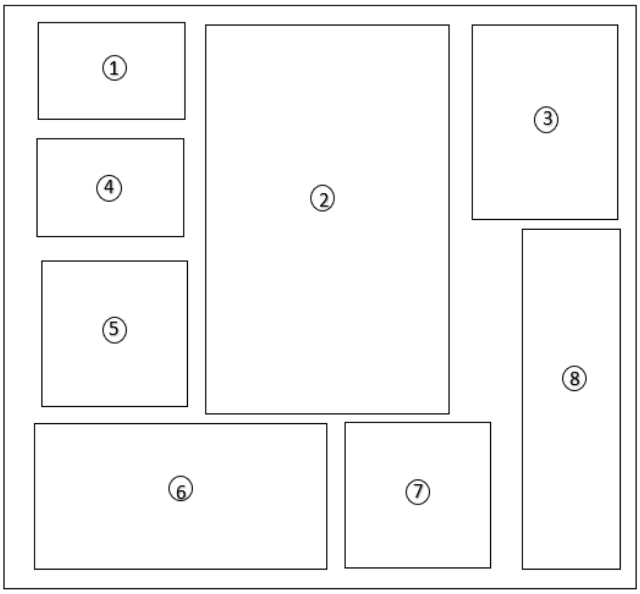

Fig -8: Initial packing of 8 blocks

Sequence-pair (P_+, N_-) =(14523678 , 67548123)

Module name	Right-to	Left-to	below	above
1	{2,3}	NULL	{6,7,5,4,8}	NULL
2	{3}	{1,4,5}	{6,7,8}	NULL
3	NULL	{1,4,5,2}	{6,7,8}	NULL
4	{2,3,8}	NULL	{6,7,5}	{1}
5	{2,3,8}	NULL	{6,7}	{1,4}
6	{7,8}	NULL	NULL	{1,4,5,2,3}
7	{8}	{6}	NULL	{1,4,5,2,3}
8	NULL	{6,7,5,4}	NULL	{1,2,3}

Fig -9: Relative location of 8 blocks

module	1	2	3	4	5	6	7	8
X	0	3	8	0	0	0	6	9
Y	8	7	7	6	3	0	3	0

Fig -10: Co-ordinates of 8 blocks

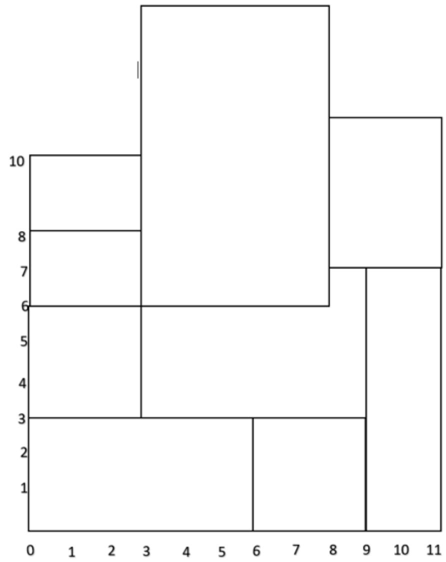

Fig -11: Final Packing of 8 blocks

5. CONCLUSIONS

We successfully packed 4 and 5 blocks using sequence-pair which results into optimal packing but 8 block packing results into large wastage of space which shows that every sequence-pair packing does not corresponds to optimal packing. A packing can have $(m!)^2$ Sequence-pair out of which atleast one corresponds to optimal packing.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my advisor Dr. Bharthi S.H for the continuous support of my study and research , for her patience, motivation, enthusiasm, and immense knowledge. Her guidance helped me in all the time of research.

REFERENCES

1. Baker, Brenda S., et al. “Orthogonal Packings in Two Dimensions.” SIAM Journal on Computing, vol. 9, no. 4, 1980, pp. 846–855., doi:10.1137/0209064.
2. Sha, Lu, and R.w. Dutton. “An Analytical Algorithm for Placement of Arbitrarily Sized Rectangular Blocks.” 22nd ACM/IEEE Design Automation Conference, 1985, doi:10.1109/dac.1985.1586005.
3. Alon, A., and U. Ascher. “Model and Solution Strategy for Placement of Rectangular Blocks in the Euclidean Plane.” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 7, no. 3, 1988, pp. 378–386., doi:10.1109/43.3171.
4. Schiermeyer, Ingo. “Reverse-Fit: A 2-Optimal Algorithm for Packing Rectangles.” Algorithms — ESA 94 Lecture Notes in Computer Science, 1994, pp. 290–299., doi:10.1007/bfb0049416.
5. H. Murata, K. Fujiyoshi, S. Nakatake and Y. Kajitani, "Rectangle-packing-based module placement," *Proceedings of IEEE International Conference on Computer Aided Design (ICCAD)*, San Jose, CA, USA, 1995, pp. 472-479. doi: 10.1109/ICCAD.1995.480159.

6. Murata, H., et al. "VLSI Module Placement Based on Rectangle-Packing by the Sequence-Pair." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 15, no. 12, 1996, pp. 1518–1524., doi:10.1109/43.552084.
7. Murata, H., et al. "VLSI/PCB Placement with Obstacles Based on Sequence-Pair." Proceedings of the 1997 International Symposium on Physical Design - ISPD 97, 1997, doi:10.1145/267665.267675.
8. Murata, Hiroshi, and Ernest S. Kuh. "Sequence-Pair Based Placement Method for Hard/Soft/Pre-Placed Modules." Proceedings of the 1998 International Symposium on Physical Design - ISPD 98, 1998, doi:10.1145/274535.274560.
9. Scheithauer, Gubtram, and Uta Sommerweiß. "4-Block Heuristic for the Rectangle Packing Problem." European Journal of Operational Research, vol. 108, no. 3, 1998, pp. 509–526., doi:10.1016/s0377-2217(96)00359-1.
10. Healy, Patrick, et al. "An Optimal Algorithm for Rectangle Placement." Operations Research Letters, vol. 24, no. 1-2, 1999, pp. 73–80., doi:10.1016/s0167-6377(98)00048-0.
11. K. Hatta, S. Wakabayashi and T. Koide, "Solving the rectangular packing problem by an adaptive GA based on sequence-pair," *Proceedings of the ASP-DAC '99 Asia and South Pacific Design Automation Conference 1999 (Cat. No.99EX198)*, Wanchai, Hong Kong, 1999, pp. 181-184 vol.1.
12. Nagao, Akira & Sawa, Takashi & Shigehiro, Yuji & Shirakawa, I & Kambe, Takashi. (2000). A new approach to rectangle-packing. Electronics and Communications in Japan (Part III: Fundamental Electronic Science). 83. 10.1002/1520-6440(200012)83:12<94::AID-ECJC11>3.0.CO;2-D.
13. Area Optimization of packing represented by sequence-pair,2000
14. Sheqin Dong, Xianlong Hong, Youliang Wu, Yizhou Lin and Jun Gu, "VLSI block placement using less flexibility first principles," *Proceedings of the ASP-DAC 2001. Asia and South Pacific Design Automation Conference 2001 (Cat. No.01EX455)*, Yokohama, Japan, 2001, pp. 601-604.
15. Y. Shigehiro, S. Koshiyama and T. Masuda, "Stochastic tabu search for rectangle packing," *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)*, Tucson, AZ, USA, 2001, pp. 2753-2758 vol.4.
16. Lipnitskii, A. A. Cybernetics and Systems Analysis, vol. 38, no. 6, 2002, pp. 943–946., doi:10.1023/a:1022916710264.
17. Lodi, Andrea, et al. "Two-Dimensional Packing Problems: A Survey." European Journal of Operational Research, vol. 141, no. 2, 2002, pp. 241–252., doi:10.1016/s0377-2217(02)00123-6.
18. Imahori, S., et al. "Local Search Algorithms for the Rectangle Packing Problem with General Spatial Costs." Mathematical Programming, vol. 97, no. 3, 2003, pp. 543–569., doi:10.1007/s10107-003-0427-1.
19. Watanabe, Shinya, et al. "Multi-Objective Rectangular Packing Problem and Its Applications." Lecture Notes in Computer Science Evolutionary Multi-Criterion Optimization, 2003, pp. 565–577., doi:10.1007/3-540-36970-8_40.
20. Optimal rectangle packing:initial result,2003
21. Itoh, T., et al. "Hierarchical Data Visualization Using a Fast Rectangle-Packing Algorithm." IEEE Transactions on Visualization and Computer Graphics, vol. 10, no. 3, 2004, pp. 302–313., doi:10.1109/tvcg.2004.1272729.
22. E. Korf, Richard. (2004). Optimal Rectangle Packing: New Results.. Proceedings of the 14th International Conference on Automated Planning and Scheduling, ICAPS 2004. 142-149.
23. Lesh, N., et al. "Exhaustive Approaches to 2D Rectangular Perfect Packings." Information Processing Letters, vol. 90, no. 1, 2004, pp. 7–14., doi:10.1016/j.ipl.2004.01.006.
24. Mukhacheva, E. A., and A. S. Mukhacheva. "The Rectangular Packing Problem: Local Optimum Search Methods Based on Block Structures." Automation and Remote Control, vol. 65, no. 2, 2004, pp. 248–257., doi:10.1023/b:aurc.0000014721.38981.bb.
25. on rectangle packing,2004.
26. Shigehiro, Yuji, et al. "A New Approach to Rectangle Packing Problem Based on Stochastic Tabu Search." Transactions of the Society of Instrument and Control Engineers, vol. 40, no. 7, 2004, pp. 747–754., doi:10.9746/sicetr1965.40.747.
27. Lesh, N., et al. "New Heuristic and Interactive Approaches to 2D Rectangular Strip Packing." Journal of Experimental Algorithmics, vol. 10, 2005, doi:10.1145/1064546.1083322.
28. Imahori, S., et al. "Improved Local Search Algorithms for the Rectangle Packing Problem with General Spatial Costs." European Journal of Operational Research, vol. 167, no. 1, 2005, pp. 48–67., doi:10.1016/j.ejor.2004.02.020.
29. Zhang, Defu, et al. "A Hybrid Heuristic Algorithm for the Rectangular Packing Problem." Lecture Notes in Computer Science Computational Science – ICCS 2005, 2005, pp. 783–791., doi:10.1007/11428831_97.
30. Ibaraki, Toshihide, and Kouji Nakamura. "Packing Problems with Soft Rectangles." Hybrid Metaheuristics Lecture Notes in Computer Science, 2006, pp. 13–27., doi:10.1007/11890584_2.
31. Moffitt, Michael D. and Martha E. Pollack. "Optimal Rectangle Packing: A Meta-CSP Approach." *ICAPS* (2006).
32. Bortfeldt, Andreas. "A Genetic Algorithm for the Two-Dimensional Strip Packing Problem with Rectangular Pieces." European Journal of Operational Research, vol. 172, no. 3, 2006, pp. 814–837., doi:10.1016/j.ejor.2004.11.016.
33. Drakidis, A., et al. "Packing-Based VLSI Module Placement Using Genetic Algorithm with Sequence-Pair Representation." IEE Proceedings - Circuits, Devices and Systems, vol. 153, no. 6, 2006, p. 545., doi:10.1049/ip-cds:20050134.
34. Packing based vlsi module placement using genetic algorithm with sequence-pair representation,2006
35. Huang, Wenqi, and Duanbing Chen. "An Efficient Heuristic Algorithm for Rectangle-Packing Problem." Simulation Modelling Practice and Theory, vol. 15, no. 10, 2007, pp. 1356–1365., doi:10.1016/j.simpat.2007.09.004.
36. Zhang, De-Fu, et al. "An Improved Heuristic Recursive Strategy Based on Genetic Algorithm for the Strip Rectangular Packing Problem." Acta Automatica Sinica, vol. 33, no. 9, 2007, pp. 911–916., doi:10.1360/aas-007-0911.
37. Yano, Yuuki, and Mineo Kaneko. "Solution Space Reduction of Sequence Pairs Using Model Placement." 2007 50th Midwest Symposium on Circuits and Systems, 2007, doi:10.1109/mwscas.2007.4488756.
38. Clautiaux, François, et al. "A New Exact Method for the Two-Dimensional Orthogonal Packing Problem." European Journal of Operational Research, vol. 183, no. 3, 2007, pp. 1196–1211., doi:10.1016/j.ejor.2005.12.048.
39. Huang, Wenqi, et al. "A New Heuristic Algorithm for Rectangle Packing." Computers & Operations Research, vol. 34, no. 11, 2007, pp. 3270–3280., doi:10.1016/j.cor.2005.12.005.
40. Korf, Richard E., et al. "Optimal Rectangle Packing." Annals of Operations Research, vol. 179, no. 1, 2008, pp. 261–295., doi:10.1007/s10479-008-0463-6.
41. Huang, Wenqi, and Duanbing Che. "An Efficient Quasi-Human Heuristic Algorithm for Solving the Rectangle-Packing Problem." Simulated Annealing, 2008, doi:10.5772/5562.
42. Search strategies for rectangle packing,2008
43. Beltrán-Cano, David, et al. "Solving the Rectangle Packing Problem by an Iterative Hybrid Heuristic." Computer Aided Systems Theory - EUROCAST 2009 Lecture Notes in Computer Science, 2009, pp. 673–680., doi:10.1007/978-3-642-04772-5_87.
44. Aşık, Önder Barış, and Ender Özcan. "Bidirectional Best-Fit Heuristic for Orthogonal Rectangular Strip Packing." Annals of Operations Research, vol. 172, no. 1, 2009, pp. 405–427., doi:10.1007/s10479-009-0642-0.
45. Wei, Lijun, et al. "A Least Wasted First Heuristic Algorithm for the Rectangular Packing Problem." Computers & Operations Research, vol. 36, no. 5, 2009, pp. 1608–1614., doi:10.1016/j.cor.2008.03.004.

46. A steplike stacking heuristic algorithm for solving rectangle packing problem,2009
47. Singh, Kawaljeet, and Leena Jain. “Experimenting Genetic Approach to Extend Rectangular Packing Heuristic Solution.” International Journal of Computer Applications, ecot, no. 1, 2010, pp. 1–7., doi:10.5120/1534-137.
48. Hoshi, Fumiya, et al. “A New Deterministic Algorithm for Two-Dimensional Rectangular Packing Problems Based on Polyomino Packing Models.” 2010 IEEE International Conference on Systems, Man and Cybernetics, 2010, doi:10.1109/icsmc.2010.5641803.
49. He, Kun, and Wenqi Huang. “An Efficient Placement Heuristic for Three-Dimensional Rectangular Packing.” Computers & Operations Research, vol. 38, no. 1, 2011,pp.227233.,doi:10.1016/j.cor.2010.04.015.
50. Peng, Bitao. “A Effective Heuristic Algorithm for Two-Dimensional Rectangle-Packing Problems.” Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), 2011, doi:10.1109/tmee.2011.6199553.
51. He, Kun, et al. “An Efficient Deterministic Heuristic for Two-Dimensional Rectangular Packing.” Computers & Operations Research, vol. 39, no. 7, 2012, pp. 1355–1363., doi:10.1016/j.cor.2011.08.005.
52. Bortfeldt, Andreas. “A Reduction Approach for Solving the Rectangle Packing Area Minimization Problem.” European Journal of Operational Research, vol. 224, no. 3, 2013, pp. 486–496., doi:10.1016/j.ejor.2012.08.006.
53. Peng, Bi-Tao, and Yong-Wu Zhou. “Recursive Heuristic Algorithm for the 2D Rectangular Strip Packing Problem.” Journal of Software, vol. 23, no. 10, 2012, pp. 2600–2611., doi:10.3724/sp.j.1001.2012.04187.
54. research status and trends of rectangular packing problem,2012
55. Huang, E., and R. E. Korf. “Optimal Rectangle Packing: An Absolute Placement Approach.” Journal of Artificial Intelligence Research, vol. 46, 2013, pp. 47–87., doi:10.1613/jair.3735.
56. optimization algorithm for rectangle packing problem based on varied factor genetic algorithm and lowest front line technique,2014
57. He, Kun, et al. “Dynamic Reduction Heuristics for the Rectangle Packing Area Minimization Problem.” European Journal of Operational Research, vol. 241, no. 3, 2015, pp. 674–685.
58. Martello, Silvano, and Michele Monaci. “Models and Algorithms for Packing Rectangles into the Smallest Square.” Computers & Operations Research, vol. 63, 2015, pp. 161–171., doi:10.1016/j.cor.2015.04.024.
59. I. L. Markov, J. Hu and M. Kim, "Progress and Challenges in VLSI Placement Research," in *Proceedings of the IEEE*, vol. 103, no. 11, pp. 1985-2003, Nov. 2015. doi: 10.1109/JPROC.2015.2478963
60. Wu, Lei, et al. “A Novel Heuristic Algorithm for Two-Dimensional Rectangle Packing Area Minimization Problem with Central Rectangle.” Computers & Industrial Engineering, vol. 102, 2016, pp. 208–218.
61. Wu, Lei, et al. “An Improved Heuristic Algorithm for 2D Rectangle Packing Area Minimization Problems with Central Rectangles.” Engineering Applications of Artificial Intelligence, vol. 66, 2017, pp. 1–16.,
62. Laverty, Brian, and Thomas Murphy. “Optimal Rectangle Packing for the Square.” Recreational Mathematics Magazine, vol. 5, no. 9, 2018, pp. 5–47., doi:10.2478/rmm-2018-0001.