Experimenting Genetic Approach to Extend Rectangular Packing Heuristic Solution

Kawaljeet Singh
University Computer Center
Punjabi University,
Patiala-147 002, Punjab

Leena Jain
RIMT, Mandi Gobindgarh,
Ph.D Research Scholar,
Punjabi University, Patiala-147 002, Punjab

ABSTRACT

Nesting (Cutting and Packing) problems are optimization problem encountered in many areas of business that look for good arrangement of multiple items in larger containing regions. The objective of this problem is to maximize the utilization of resource material. There is a large range of the applicability of these problems as there are many diverse instances of it that are encountered in the industries as paper, glass, plastic and foam, leather, sheet metal cutting, furniture, garments, ship-building, shoe-making, car production, building materials, packaging etc. Most of the standard problems related to Nesting are known to be NP-complete. The development of exact algorithms which are faster and produce near optimal solutions is still a major research issue in this area. Proliferation of sophisticated desktops and faith of researchers in meta-heuristics have further allowed them to look beyond the traditional optimization techniques to solve this hard problem. In this paper Authors have tried to explore further expansion in feasible patterns for rectangle packing by applying genetic operators on the initial population of feasible patterns generated by revised AYC Nee's Rectangle Packing heuristic.

Keywords

Genetic algorithm, NP-complete, Rectangle packing heuristic

1. INTRODUCTION

The rectangle-packing problem is the problem of placing several given rectangles of arbitrary width and height into a minimum area rectangle without overlapping. This rectangle packing problem (RPP) is investigated in a lot of publications [1, 2, 3, 4, and 5] because of its large practical relevance in both the two-dimensional and three-dimensional case. Rectangle packing Problem can be applied to loading a set of rectangular objects onto a cargo pallet, with/ without stacking these objects. In the floor-plan design of VLSI, various chips are assigned to the physical regions of the printed circuit boards. Another application is cutting a set of rectangles out of a rectangular piece of stock material. The wood, the glass and the paper industry are mainly concerned with the cutting of regular figures (Figure 1). On the other hand, in shipbuilding, textile and leather industry irregular and arbitrary shaped items need to be packed (Figure 2 & Fig. 3). The RPP falls in NP-complete problems class where computation time for an exact solution increases with N and become rapidly prohibitive in cost as N increases. The solution approach to these problems lies in reducing the exhaustive search of all possible arrangements of nesting the parts and subsequently checking upon the execution time. Usually, various heuristic rules are proposed to generate different patterns, which are generally the priority rules used to allocate patterns to the stock sheet sequentially.

In this paper author used revised AYC Nee's Rectangle packing heuristic [7] to generate 120 feasible patterns. These feasible solutions are then used to find the optimal answers for a given instance of RPP using genetic optimization techniques.

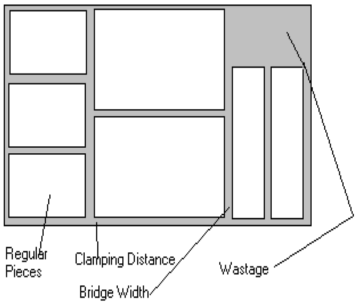

Figure 1Cutting of Regular Parts (All rectangular parts) from a Rectangular Sheet [6]

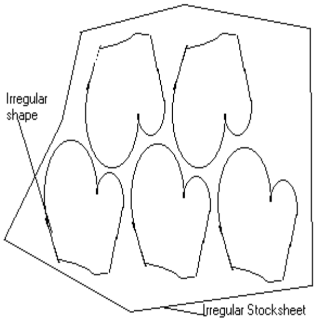

Figure 2 Cutting of Irregular Parts from an Irregular Stock Sheet [6]

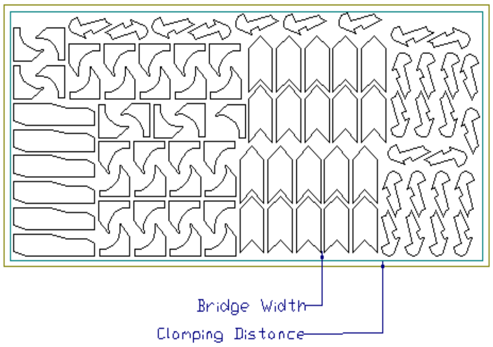

Figure 3 Cutting of Irregular Parts from a Rectangular Stock Sheet [6]

2. BRIEF SYNOPSIS OF THE REVISED AYC NEE'S RECTANGLE PACKING HEURISTIC

An heuristic study which is focused to 2D rectangular packing problem where rectangular items (relatively smaller in size) are packed on to a larger containing region, which is rectangular in shape, *say object* has been discussed in by the authors [7] The specific problem addressed is characterised by a set of rectangular items, which may contain identical items that can be rotated by 90° during packing process; packing arrangements are non-guillotine and orthogonal and the stock sheet is rectangular in shape, without any bad patches in it. Also, grain orientation of the item and object are trivial. The packing process has to ensure that there is no overlap of items while confining within the object. Brief description of the heuristic is as: -

1. The items are first sequenced. The criteria for sequencing is based on the all possible aspects related to a rectangle e.g., length, width, area, perimeter, length/ width ratio, or priority assigned to items. This priority is expressed in terms of urgency for an ordered piece or profit associated with it. The lists are sorted in increasing order and in decreasing order.
2. The item thus picked first is placed horizontally and then vertically at the lower left corner of the object, referred to as its reference *point* (Figure 4*)*. Also, it is important to define here *Sheet Utilization Ratio* as the sum of total area of all items placed on the stock sheet to total stock sheet area. The item placed at reference point of the object gives rise to two *pivot points*. Pivot points are the top-left and bottom-right corner of the ordered piece placed and the top-left and bottom-right corners of enclosing rectangle that encloses all ordered pieces placed so far. Essentially, pivot points are the only probable positions where next item in sequence can be placed. The next item is then placed at each of the pivot points, both length-wise and breadth-wise for all feasible results. Orientations that result in minimum wastage are retained. New pivot points are defined and used ones are deleted. Systematic Search for the Best Position of the Second Rectangle is illustrated in Figure 5. In the algorithm, pivot points are sequenced in the following three ways: -
   a. minimum radial distance (PPD)
   b. minimum x-distance, in case of tie, point with minimum y-distance is to be served first (PPL)
   c. minimum y-distance, in case of tie, point with minimum x-distance is to be served first (PPB)
3. This process is repeated until every rectangle has been placed.
4. The object is also placed in both the orientations.

Total number of feasible layouts thus possible are the product of possible orientations of the Object (2) X possible orientations of the first Item placed on the Object (2) X possible sequencing patterns (increasing and decreasing) (2) X different basis for sequencing of items considered (5) X different sequencings of pivot points (along) considered (3) = 2 X 2 X 2 X 5 X 3 = 120; Thus OL-IL-I-SL-PPD stands for pattern obtained when Object is oriented Length-wise; first-Item placed on the reference point is also oriented Length-wise; items are sorted in Increasing order and are Sequenced on the basis of Length; Pivot Points are arranged in increasing order along the Diagonal of the object.

3. SYNOPSIS OF GENETIC ALGORITHMS

Genetic Algorithms (GA) were developed by John Holland [8] and since then have been used in various fields of engineering. GA has been used quite successfully for combinatorial problems that are NP-complete. GAs have been used in a wide variety of optimisation tasks, including numerical optimisation, and combinatorial optimisation problems such as travelling salesman problem (TSP) [9] packing problem [1, 10, 11], job shop scheduling [12] and video & sound quality optimisation. A genetic algorithm is a randomized parallel search method modelled on natural selection and genetics [13]. In contrast to more standards search algorithms, GA bases their progress on the performance of a population of candidate solutions, rather than on a single candidate solution. The motivation behind this is that by simultaneously searching many areas of the design space the risk of getting stuck at local optima is greatly reduced. GA are probabilistic in nature and start off with a population of randomly generated candidates and evolve toward better solutions by applying genetic operators, modelled on the natural genetic process. For solving any problem, Genetic Algorithm is started with a **set of solutions** (represented by **chromosomes**) called **population**. A member of the population is a genotype, a chromosome, a string or a permutation. In our case, a chromosome is feasible pattern of Rectangle packing obtained by applying the modified heuristic. Solutions from one population are taken and used to form a new population. This is motivated with a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (**offspring**) are selected according to their fitness - the more suitable they are the more chances they have to reproduce. When a genotype is decoded, a packing pattern, called a phenotype, is formed. We may calculate the fitness function even to restrict the mutating parents. A GA generally has five components [14]

1. A representation for solutions to the problem
2. A method to create initial population
3. An evaluation function to determine the relative fitness of the solutions
4. Genetic operators that effect the composition of the offspring during reproduction and
5. Values for parameters that the GA uses (e.g. population size, probabilities of applying the genetic operators, etc.)

4. HYPOTHESIS

The solutions obtained by revised heuristic are exhaustive/ complete and need not be mutated so as to look for even better feasible solutions.

5. METHOD TO CREATE INITIAL POPULATION

An initial population of chromosomes is generated at the start of the GA by using either heuristic or randomly. In this paper revised rectangle packing heuristic [7] is used to generate the initial population comprising of 120 feasible solutions.

5.1 Encoding of a chromosome

A chromosome represents a solution to the problem and is encoded as a vector of integer number that represents the order of the rectangle which is present in the given data set. Each solution chromosome is made of n genes where each gene is a two tuple comprising, the rectangle number and total piece placed.

Chromosome = ($gene_1$; $gene_2$; . . . ; $gene_{n\_1}$; $gene_n$)

The n genes are used to obtain the Rectangle Packing Sequence, RPS, which is going to be used by the placement strategy. For example Figure 6 displays a chromosome for solving a five-rectangle packing problem. The decoding (mapping) of each chromosome into a RPS (Rectangle packing sequences) is accomplished by packing the rectangles in specific order. It is interpreted as that feasible solution involves packing of 3 pieces of item 1, 1 piece of item 2, 2 pieces of item 4, 1 piece if item 5 and 3 pieces of item 3 using the rectangle packing heuristic.

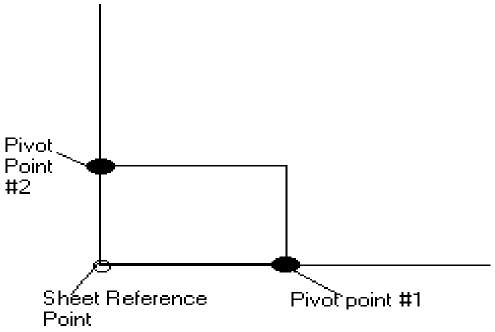

Figure 4 Placing first piece at sheet reference point.

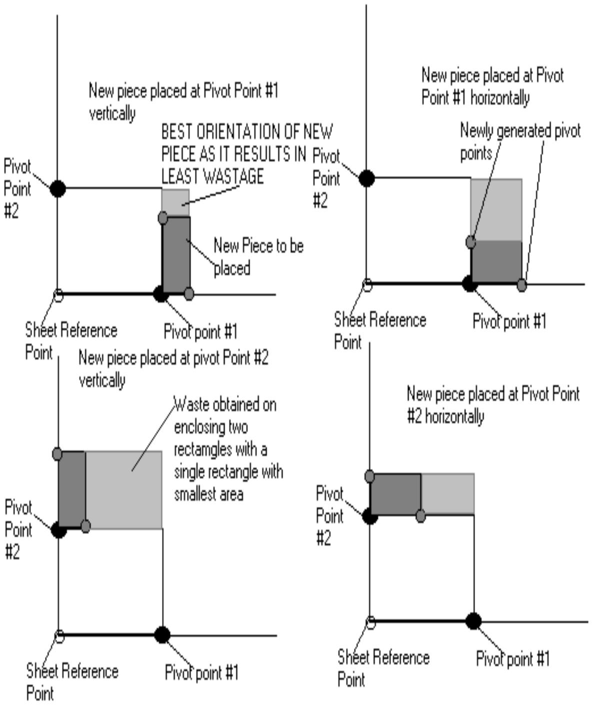
New piece placed at Pivot Point #1 vertically
BEST ORIENTATION OF NEW PIECE AS IT RESULTS IN LEAST WASTAGE
Pivot Point #2
New Piece to be placed
Sheet Reference Point
Pivot point #1
New piece placed at Pivot Point #1 horizontally
Newly generated pivot points
Pivot Point #2
Sheet Reference Point
Pivot point #1
New piece placed at pivot Point #2 vertically
Waste obtained on enclosing two rectamgles with a single rectangle with smallest area
Pivot Point #2
Sheet Reference Point
Pivot point #1
New piece placed at Pivot Point #2 horizontally
Pivot Point #2
Sheet Reference Point
Pivot point #1

Figure 5 Systematic search for the best position of the second rectangle

(1,3)	(2,1)	(4,2)	(5,1)	(3,3)

Figure 6 Chromosome Example

5.2 Evaluation function and relative fitness of the solutions

The fitness function is defined over the genetic representation and measures the *quality* of the represented solution. The fitness function is always problem dependent. A **fitness function** is a particular type of objective function that prescribes the optimality of a solution (that is, a chromosome) in a genetic algorithm so that that particular chromosome may be ranked against all the other chromosomes. The natural fitness function (measure of quality) for this type of problem is the value of the %Trim Loss or % Utilization factor given by

%Trim Loss = (Unused area/ Stock rectangle area)*100

%Utilization factor = (Used area/ Stock rectangle area)*100

In this paper Utilization factor consider as a fitness function.

6. GENETIC OPERATORS DEFINED THAT ARE USED FOR PRODUCING THE OFFSPRING

The chromosome is a permutation of the rectangles and requires specialized operators. Many genetic algorithms for solving sequencing optimization problems, such as the traveling salesman problem, machine scheduling, vehicle routing, and others, use such a permutation representation. Several crossover operators have been proposed for the permutation representation, such as the partial-mapped crossover [15], the heuristic crossover [16], the order crossover [17], the cycle crossover [18], the position-based crossover [19], the order-based crossover [20] and the sub-tour exchange crossover [20]. Also, a number of mutation operators have been proposed, such as inversion mutation, insertion mutation, displacement mutation, reciprocal exchange mutation [21] and heuristic mutation [22]. In this study genetic algorithm used only crossover operator that is order based crossover (OBX).The OBX operates as follows. First a cut is determined. The left hand portion of the first chromosome will then be copied to the offspring the other position will then filled by unused genes of second chromosome (with respect to same order). The cut point lies after the 50% of TPP (Total piece placed) of first chromosome. Figure 7 illustrates the crossover operator of genetic algorithm.

Chromosome: 1

(10,3)	(5,2)	(9,3)	(8,3)	(4,1)	(3,3)	(7,0)	(6,0)	(2,0)	(1,0)

Chromosome: 2

(1,1)	(2,1)	(3,3)	(4,1)	(5,0)	(6,1)	(7,0)	(8,2)	(9,1)	(10,1)

TPP of first chromosome=15 Cut point recommended=15/2=8

(10,3)	(5,2)	(9,3)	(1,1)	(2,1)	(3,3)	(4,1)	(6,1)	(7,0)	(8,2)

New generated chromosomes

Figure 7 Crossover operator of genetic algorithm

7. EXPERIMENTAL SETUP

So as to apply the genetic algorithm operator on the initial population generated by revised rectangle packing heuristic following three data sets (Table 1) were considered.

Table 1 Data sets

<table>
<tr><td colspan="11">SET-I MOTHER SHEET_LENGTH=70 WIDTH=40</td></tr>
<tr><td>S.No</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr>
<tr><td>L</td><td>22</td><td>31</td><td>35</td><td>24</td><td>30</td><td>13</td><td>14</td><td>14</td><td>12</td><td>13</td></tr>
<tr><td>W</td><td>21</td><td>13</td><td>9</td><td>9</td><td>7</td><td>11</td><td>10</td><td>8</td><td>8</td><td>7</td></tr>
<tr><td>Qty</td><td>1</td><td>1</td><td>3</td><td>3</td><td>2</td><td>3</td><td>1</td><td>3</td><td>3</td><td>3</td></tr>
<tr><td colspan="11">SET-II MOTHER SHEET_LENGTH=593 WIDTH=100</td></tr>
<tr><td>S.No</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr>
<tr><td>L</td><td>96.87</td><td>120.8</td><td>125.5</td><td>120.4</td><td>105.7</td><td>114.7</td><td>119.2</td><td>119.2</td><td>150.4</td><td>161.1</td></tr>
<tr><td>W</td><td>13.97</td><td>18.06</td><td>19.39</td><td>19.85</td><td>20.02</td><td>20.39</td><td>21.1</td><td>21.39</td><td>26.02</td><td>30.98</td></tr>
<tr><td>Qty</td><td>3</td><td>1</td><td>2</td><td>2</td><td>4</td><td>1</td><td>3</td><td>3</td><td>1</td><td>2</td></tr>
<tr><td colspan="11">SET-III MOTHER SHEET_LENGTH=593 WIDTH=100</td></tr>
<tr><td>S.No</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr>
<tr><td>L</td><td>80.41</td><td>105.9</td><td>129</td><td>122.5</td><td>107.4</td><td>116.5</td><td>120.8</td><td>124.5</td><td>147.7</td><td>194.1</td></tr>
<tr><td>W</td><td>11.6</td><td>15.84</td><td>19.94</td><td>20.21</td><td>20.34</td><td>20.69</td><td>21.38</td><td>22.29</td><td>25.56</td><td>37.32</td></tr>
<tr><td>Qty</td><td>3</td><td>1</td><td>2</td><td>2</td><td>4</td><td>1</td><td>3</td><td>3</td><td>1</td><td>2</td></tr>
</table>

Note: L indicates Length, W is width and Qty is Quantity

8. RESULTS AND DISCUSSIONS

The revised heuristic is used to generate the 120 feasible patterns which are shown in Table 2, 3 and 4 for Set I, Set II and Set III respectively. Where Heuristic **OL**, **OB** indicates Orientation of **O**bject in Length-wise and Breadth-wise direction. Heuristic **IL**, **IB** indicates Orientation of first item in Length-wise and Breadth-wise direction. **PPL, PPB, PPD** indicates sequencing Pivot Points along the **L**ength, **B**readth and **D**iagonal of the sheet and Heuristic OL-IL-D-SL-PPL indicates Orientation of **O**bject is Length-wise and that of first **I**tem placed is **L**ength-wise; pieces are sequenced in **D**ecreasing order on the basis of **S**equencing basis **L**ength and **P**ivot **P**oints are sequenced in increasing order along the **L**ength of the sheet; TPP is Total Piece Placed; UF is Sheet Utilization factor. To start with the genetic algorithm, we considered the healthy chromosomes that have best utilization factor. Such healthy chromosomes with their respective utilization factor and total piece placed are listed in Table 5, 6 and 7 for Set I, Set II and Set III respectively. OBX operator was applied to generate new chromosome (Figure 8). The resulting feasible pattern is obtained by laying items on the object by applying revised heuristic algorithm. Accordingly layout for chromosome 1, Chromosome 2 and new generated chromosome is given in Figure 9, 10 and 11 respectively. Orientation of the object and item are now only important aspects to obtain different layouts as the sequence is not be altered at all. A number of off-springs were mutated. The genetically produced chromosome was observed to be even better [94.6 % (Set-I)] than the participating parents [86.80%, 92.10 % (Set-I)].

Table 2 Performance of different heuristics for dataset 1

			OL-IL		OB-IL		OL-IB		OB-IB	
#	ORDER	SPP	TPP	UF(%)	TPP	UF(%)	TPP	UF(%)	TPP	UF(%)
I.	D-SA	PPB	9	85.9	10	89.9	**11**	**92.1**	9	85.9
II.	D-SA	PPD	9	85.9	10	89.9	**11**	**92.1**	9	85.9
III.	D-SA	PPL	9	85.9	10	89.9	**11**	**92.1**	9	85.9
IV.	D-SAR	PPB	12	85.6	11	81.6	11	81.6	11	81.6
V.	D-SAR	PPD	12	85.6	11	81.6	11	81.6	11	81.6
VI.	D-SAR	PPL	12	85.6	11	81.6	11	81.6	11	81.6
VII.	D-SB	PPB	11	78.6	15	91.8	14	88.7	14	88.7
VIII.	D-SB	PPD	11	78.6	15	91.8	14	88.7	14	88.7
IX.	D-SB	PPL	11	78.6	15	91.8	14	88.7	14	88.7
X.	D-SL	PPB	9	86.3	10	86.8	10	86.8	9	82.6
XI.	D-SL	PPD	9	86.3	10	86.8	10	86.8	9	82.6
XII.	D-SL	PPL	9	86.3	10	86.8	10	86.8	9	82.6
XIII.	D-SP	PPB	7	79.6	9	85.7	10	87.9	7	79.6
XIV.	D-SP	PPD	7	79.6	9	85.7	10	87.9	7	79.6
XV.	D-SP	PPL	7	79.6	9	85.7	10	87.9	7	79.6
XVI.	I-SA	PPB	16	75.1	16	75.1	17	82.8	15	67.4
XVII.	I-SA	PPD	16	75.1	16	75.1	17	82.8	15	67.4
XVIII.	I-SA	PPL	16	75.1	16	75.1	17	82.8	15	67.4
XIX.	I-SAR	PPB	15	76.6	15	76.6	15	76.6	16	84.3
XX.	I-SAR	PPD	15	76.6	15	76.6	15	76.6	16	84.3
XXI.	I-SAR	PPL	15	76.6	15	76.6	15	76.6	16	84.3
XXII.	I-SB	PPB	15	88.5	14	80.8	14	80.8	15	85.8
XXIII.	I-SB	PPD	15	88.5	14	80.8	14	80.8	15	85.8
XXIV.	I-SB	PPL	15	88.5	14	80.8	14	80.8	15	85.8
XXV.	I-SL	PPB	15	76.6	15	76.6	15	76.6	15	76.6
XXVI.	I-SL	PPD	15	76.6	15	67.9	15	76.6	15	76.6
XXVII.	I-SL	PPL	15	76.6	15	67.9	15	76.6	15	76.6
XXVIII.	I-SP	PPB	17	83	16	75.3	16	75.5	16	75.3
XXIX.	I-SP	PPD	17	83	16	75.3	16	75.5	16	75.3
XXX.	I-SP	PPL	17	83	16	75.3	16	75.5	16	75.3

Note: Heuristic OL-IL-D-SL-PPL indicates Orientation of **O**bject is Length-wise and that of first **I**tem placed is **L**ength-wise; pieces are sequenced in **D**ecreasing order on the basis of **S**equencing basis **L**ength and **P**ivot **P**oints are sequenced in increasing order along the **L**ength of the sheet; TPP is Total Piece Placed; UF is Sheet Utilization factor;

Table 3 Performance of different heuristics for dataset II

			OL-IL		OB-IL		OL-IB		OB-IB	
#	ORDER	SPP	TPP	UF(%)	TPP	UF(%)	TPP	UF(%)	TPP	UF(%)
I.	D-SA	PPB	17	75.81	3	6.85	3	6.85	17	78.76

II.	D-SA	PPD	17	75.81	3	6.85	3	6.85	17	78.76
III.	D-SA	PPL	17	75.81	3	6.85	3	6.85	17	78.76
IV.	D-SAR	PPB	17	75.81	3	6.85	3	6.85	16	76.48
V.	D-SAR	PPD	17	75.81	3	6.85	3	6.85	16	76.48
VI.	D-SAR	PPL	17	75.81	3	6.85	3	6.85	16	76.48
VII.	D-SB	PPB	17	74.19	3	6.85	3	6.85	**18**	**81.68**
VIII.	D-SB	PPD	17	74.19	3	6.85	3	6.85	**18**	**81.68**
IX.	D-SB	PPL	17	74.19	3	6.85	3	6.85	**18**	**81.68**
X.	D-SL	PPB	16	71.57	3	6.85	3	6.85	16	76.48
XI.	D-SL	PPD	16	71.57	3	6.85	3	6.85	16	76.48
XII.	D-SL	PPL	16	71.57	3	6.85	3	6.85	16	76.48
XIII.	D-SP	PPB	17	75.81	3	6.85	3	6.85	16	76.48
XIV.	D-SP	PPD	17	75.81	3	6.85	3	6.85	16	76.48
XV.	D-SP	PPL	17	75.81	3	6.85	3	6.85	16	76.48
XVI.	I-SA	PPB	19	70.6	17	62.02	17	62.02	19	70.6
XVII.	I-SA	PPD	19	70.6	17	62.02	17	62.02	19	70.6
XVIII.	I-SA	PPL	19	70.6	17	62.02	17	62.02	19	70.6
XIX.	I-SAR	PPB	18	75.2	3	6.85	3	6.85	17	78.97
XX.	I-SAR	PPD	18	75.2	3	6.85	3	6.85	17	78.97
XXI.	I-SAR	PPL	18	75.2	3	6.85	3	6.85	17	78.97
XXII.	I-SB	PPB	17	62.02	17	62.02	17	62.02	19	70.6
XXIII.	I-SB	PPD	17	62.02	17	62.02	17	62.02	19	70.6
XXIV.	I-SB	PPL	17	62.02	17	62.02	17	62.02	19	70.6
XXV.	I-SL	PPB	19	70.6	19	70.6	19	70.6	18	66.49
XXVI.	I-SL	PPD	19	70.6	19	70.6	19	70.6	18	66.49
XXVII.	I-SL	PPL	19	70.6	19	70.6	19	70.6	18	66.49
XXVIII.	I-SP	PPB	19	70.6	19	70.6	19	70.6	19	70.6
XXIX.	I-SP	PPD	19	70.6	19	70.6	19	70.6	19	70.6
XXX.	I-SP	PPL	19	70.6	19	70.6	19	70.6	19	70.6

Note: Heuristic OL-IL-D-SL-PPL indicates Orientation of **O**bject is Length-wise and that of first **I**tem placed is **L**ength-wise; pieces are sequenced in **D**ecreasing order on the basis of **S**equencing basis **L**ength and **P**ivot **P**oints are sequenced in increasing order along the **L**ength of the sheet; TPP is Total Piece Placed; UF is Sheet Utilization factor;

Table 4 Performance of different heuristics for dataset III

#	ORDER	SPP	OL-IL		OB-IL		OL-IB		OB-IB	
			TPP	UF(%)	TPP	UF(%)	TPP	UF(%)	TPP	UF(%)
I.	D-SA	PPL	17	85.82	3	4.71	3	4.71	17	82.47
II.	D-SA	PPW	17	85.82	3	4.71	3	4.71	17	82.47
III.	D-SA	PPD	17	85.82	3	4.71	3	4.71	17	82.47
IV.	D-SAR	PPL	19	73.16	17	65.79	17	65.79	18	69.48
V.	D-SAR	PPW	19	73.16	17	65.79	17	65.79	18	69.48
VI.	D-SAR	PPD	19	73.16	17	65.79	17	65.79	18	69.48
VII.	D-SB	PPL	16	82.44	3	4.71	3	4.71	17	81.22
VIII.	D-SB	PPW	16	82.44	3	4.71	3	4.71	17	81.22
IX.	D-SB	PPD	16	82.44	3	4.71	3	4.71	17	81.22
X.	D-SL	PPL	17	82.47	3	4.71	3	4.71	17	82.47
XI.	D-SL	PPW	17	82.47	3	4.71	3	4.71	17	82.47
XII.	D-SL	PPD	17	82.47	3	4.71	3	4.71	17	82.47
XIII.	D-SP	PPL	17	82.47	3	4.71	3	4.71	17	82.47
XIV.	D-SP	PPW	17	82.47	3	4.71	3	4.71	17	82.47
XV.	D-SP	PPD	17	82.47	3	4.71	3	4.71	17	82.47
XVI.	I-SA	PPL	19	70.48	17	61.12	17	61.12	19	72.17

XVII.	I-SA	PPW	19	70.48	17	61.12	17	61.12	19	72.17
XVIII.	I-SA	PPD	19	70.48	17	61.12	17	61.12	19	72.17
XIX.	I-SAR	PPL	18	82.06	3	4.71	3	4.71	18	73.52
XX.	I-SAR	PPW	18	82.06	3	4.71	3	4.71	18	73.52
XXI.	I-SAR	PPD	18	82.06	3	4.71	3	4.71	18	73.52
XXII.	I-SB	PPL	19	70.48	17	61.12	17	61.12	20	76.85
XXIII.	I-SB	PPW	19	70.48	17	61.12	17	61.12	20	76.85
XXIV.	I-SB	PPD	19	70.48	17	61.12	17	61.12	20	76.85
XXV.	I-SL	PPL	19	70.48	18	65.8	18	65.8	19	70.48
XXVI.	I-SL	PPW	19	70.48	18	65.8	18	65.8	19	70.48
XXVII.	I-SL	PPD	19	70.48	18	65.8	18	65.8	19	70.48
XXVIII.	I-SP	PPL	19	70.48	18	65.8	18	65.8	19	70.48
XXIX.	I-SP	PPW	19	70.48	18	65.8	18	65.8	19	70.48
XXX.	I-SP	PPD	19	70.48	18	65.8	18	65.8	19	70.48

Note: Heuristic OL-IL-D-SL-PPL indicates Orientation of **O**bject is Length-wise and that of first **I**tem placed is **L**ength-wise; pieces are sequenced in **D**ecreasing order on the basis of **S**equencing basis **L**ength and **P**ivot **P**oints are sequenced in increasing order along the **L**ength of the sheet; TPP is Total Piece Placed; UF is Sheet Utilization factor;

Table 5 Chromosomes considered for crossover on the basis of relative fitness (SET-I)

Order	Chromosome	UF (%)	TPP
D-SA	(1,1) (2,1) (3,3) (4,1) (5,0) (6,1) (7,0) (8,2) (9,1) (10,1)	92.10%	11
D-SB	(1,1) (2,1) (6,3) (7,1) (4,2) (3,0) (9,1) (8,3) (10,1) (5,0)	91.80%	15
I-SB	(10,3) (5,2) (9,3) (8,3) (4,1) (3,3) (7,0) (6,0) (2,0) (1,0)	88.50%	15
D-SP	(2,1) (3,3) (1,1) (5,1) (4,0) (6,1) (7,0) (8,1) (9,1) (10,1)	87.90%	10
D-SL	(3,3) (2,1) (5,2) (4,2) (1,0) (8,0) (7,1) (10,1) (6,0) (9,0)	86.80%	10
D-SAR	(5,2) (3,3) (4,3) (2,0) (10,3) (8,1) (9,0) (7,0) (6,0) (1,0)	85.60%	12
I-SAR	(1,1) (6,3) (7,3) (9,3) (8,1) (10,3) (2,0) (4,2) (3,0) (5,0)	84.30%	16
I-SP	(9,3) (10,3) (8,3) (6,3) (7,1) (4,3) (5,1) (1,0) (2,0) (3,0)	83.00%	17
I-SA	(10,3) (9,3) (8,3) (7,1) (6,3) (5,2) (4,2) (3,0) (2,0),(1,0)	82.80%	17

Table 6 chromosomes considered for crossover on the basis of relative fitness (SET-II)

Order	Chromosome	UF (%)	TPP
D-SA	(10,2) (9,1) (8,3) (7,3) (3,2) (4,2) (6,1) (2,1) (5,1) (1,1)	78.76%	17
D-SB	(10,2) (9,1) (8,3) (7,3) (6,1) (5,4) (4,2) (3,1) (2,0) (1,1)	81.68%	18
D-SL	(10,2) (9,1) (3,2) (2,1) (4,2) (8,3) (7,2) (6,0) (5,0) (1,3)	71.57%	16
D-SP	(10,2) (9,1) (3,2) (8,3) (7,3) (4,2) (2,1) (6,0) (5,0) (1,3)	75.81%	17
I-SA	(1,3) (5,4) (2,1) (6,1) (4,2) (3,2) (7,3) (8,3) (9,0) (10,0)	70.59%	19
I-SAR	(10,2) (5,4) (8,3) (6,1) (7,3) (9,1) (4,2) (3,0) (2,1) (1,0)	78.97%	17
I-SL	(1,3) (5,4) (6,1) (7,3) (8,3) (4,2) (2,1) (3,1) (9,0) (10.0)	66.49%	18

Table 7 chromosomes considered for crossover on the basis of relative fitness (SET-III)

Order	Chromosome	UF(%)	TPP
D-SA	(10,2) (9,1) (8,3) (7,3) (3,2) (4,2) (6,1) (5,1) (2,0) (1,2)	85.82%	17
D-SAR	(10,0) (9,1) (3,2) (8,3) (4,2) (7,3) (6,1) (5,3) (2,1) (1,3)	73.16%	19
D-SB	(10,2)(9,1) (8,3) (7,3) (6,1) (5,4) (4,1) (3,0) (2,0) (1,1)	82.45%	16
D-SL	(10,2) (9,1) (3,2) (8,3) (4,2) (7,3) (6,0) (5,0) (2,1) (1,3)	82.47%	17
I-SA	(1,2) (2,1) (5,4) (6,1) (4,2) (3,2) (7,3) (8,2) (9,1) (10,0)	72.17%	19
I-SAR	(10,2) (5,4) (8,3) (6,1) (7,3) (9,0) (4,1) (3,0) (2,1) (1,3)	82.06%	18
I-SL	(1,3) (2,1) (3,2) (4,2) (5,4) (6,1) (7,3) (8,3) (9,0) (10,0)	70.48%	19
I-SB	(1,3) (2,1) (3,2) (4,2) (5,4) (6,1) (7,3) (8,3) (9,1) (10,0)	76.85%	20
I-SP	(1,3) (2,1) (5,4) (6,1) (7,3) (4,2) (8,2) (3,2) (9,0) (10,0)	65.8%	18

Chromosome: 1 TPP=10 Utilization Factor:86.80%

(3,3) (2,1) (5,2) (4,2) (1,0) (8,0) (7,1) (10,1) (6,0) (9,0)

Chromosome: 2 TPP=11 Utilization Factor:92.10%

(1,1) (2,1) (3,3) (4,1) (5,0) (6,1) (7,0) (8,2) (9,1) (10,1)

TPP of first chromosome=10 Cut point lie=10/2=5

New Chromosome: Utilization Factor: 94.5%

(3,3) (2,1) (5,2) (1,1) (4,1) (6,1) (7,0) (8,2) (9,1) (10,1)

Figure 8 Ordered Based Crossover (OBX) Operator for Set I

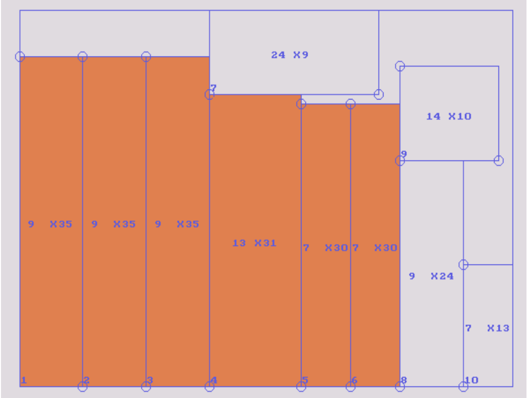

Figure 9 Layout (Chromosome 1) generated by Revised Rectangle Packing Algorithm UF= 86.8% TPP=10

9. CONCLUSIONS

The rectangle packing (other synonyms to packing are nesting and cutting) is a peculiar engineering problem with no mathematical descriptors for the process. There are infinite possible solutions for most of the problems but no unique, tightly fitting solution can normally be expected. Time needed to find the cutting pattern with the least amount of stock sheet wastage increases exponentially with the getting higher in number of different types of items used in the problem. This phenomenon is commonly known as combinatorial explosion. In this paper a revised heuristic rectangle packing algorithm is used to generate the different layout with different sheet Utilization factor.

Attempts have been successful to further expand set of feasible patterns for rectangle packing by applying genetic operators on the initial population of feasible patterns generated by revised AYC Nee's Rectangle Packing heuristic. However, empirical studies carried out though confirm paltry success stories only so as to reject the null hypothesis.

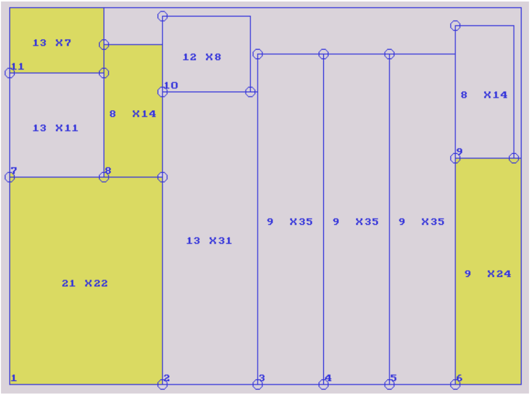

Figure 10 Layout (Chromosome 2) generated by Revised Rectangle Packing Algorithm UF= 92.1% TPP=11

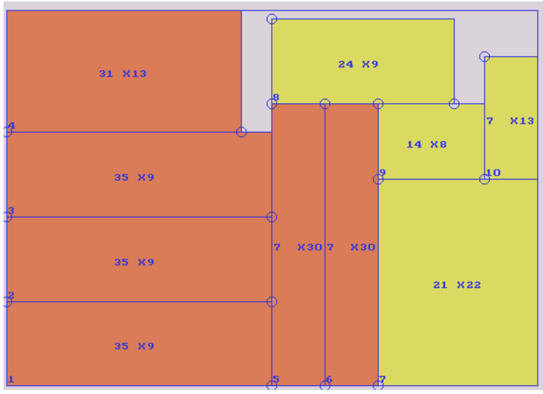

Figure 11 Layout generated by Genetic Algorithm UF= 94.6% TPP=10

REFERENCES

[1] Dowsland, K.A., and Dowsland, W.B. 1992. Packing problems. Eur. J. OR, 56, pp. 2-14.

[2] Dyckhoff, H. and Finke, U. 1992. Cutting and packing" production and distribution. Physica Verlag, Heidelberg..

[3] Dyckhoff, H., Scheithauer, G., and Terno, J. 1996. Cutting and packing: An annotated bibliography, Preprint MATH-NM-08-1996, Techn. Univ. Dresden, 1996. to appear in Combinatorial Optimization: An Annotated Bibliography, eds. Dell'Amico, Maffioli, Martello.

[4] Sweeney, P.E., and Ridenour, E.L. 1989. Cutting and packing problems: A categorized, application–oriented research bibliography, Technical report, Working Paper 610, School of Business Administration, University of Michigan.

[5] J. Terno, R. Lindemann, and G. Scheithauer. Zuschnittprobleme und ihre praktische L¨osung. Verlag Harri Deutsch, Thun und Frankfurt/Main, 1987.

[6] Singh, K. 2001. Designing Algorithms For Nesting Regular and Irregular Shapes, Ph. D. Thesis, Thapar University, Patiala (India).

[7] Singh, K and Jain, L. 2009. An empirical study of a modified Cheok-Nee's heuristic for 2d rectangular packing problem, Appejay J. Management Technology. 4, 53-64.

[8] Holland, J.H. 1975. Adaptation in Natural and Artificial Systems, The University of Michigan Press.

[9] Braun, H. On Solving Travelling Salesman Problem by Genetic Algorithm", in Parallel Problem-Solving from Nature, Lecture Notes in Computer Science 496, H.P. Schwefel and R. Manner Eds, Springer-Verlag, app. 129-133.

[10] Lipnitskii, A.A. 2002. Use of Genetic Algorithms for Solution of The Rectangle Packing Problem. Cybernetics And Systems Analysis. 38 (6), 943-946.

[11] Ismail, H.S. and Hon, K.K.B. 1995. Nesting of two-dimensional shapes using genetic algorithms, Proceedings of the Institution of Mechanical Engineers, Part B, 209, 115-124.

[12] Goldstein, 1991. Genetic Algorithm Simulation of the SHOP Scheduling Problem, by Jonathan M. Goldstein, in September 1991 published by An ICMS/Shell Oil Business Consultancy, 1991.

[13] Goldberg, D.E. 1989. Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading, Mass.

[14] Vignaux, G.A. and Michalewicz, Z. 1991. A Genetic Algorithm for the Linear Transportation Problem, IEEE Transactions on Systems, Man and Cybernetics 2, 21, 445-452.

[15] Goldberg, D. and Lingle, R. 1985. Alleles, loci and the traveling salesman problem. In Proceedings of the first international conference on genetic algorithms, 154–159, 1985.

[16] Grefenstette, J., Gopal, R., Rosmaita, B., and Gucht, D.,1985. Genetic algorithms for the traveling salesman problem.. In Proceedings of the first international conference on genetic algorithms, 160–168.

[17] Davis, L. 1995. Applying adaptive algorithms to domains. In Proceeding of the international joint conference on artificial intelligence, 162–164.

[18] Oliver, I.., Smith, D., Holland, J. 1987. A study of permutation crossover operators on the traveling salesman problem, In Proceedings of the second international conference on genetic algorithms. 224–230.

[19] Syswerda, G. 1991. Scheduling optimization using genetic algorithms. In L. Davis (Ed.), Handbook of genetic algorithms, 332–349. New York: Van Nostrand Reinhold,

[20] Yamamura, M. Ono, T. and Kobayashi, S. 1992. Character-preserving genetic algorithms for traveling salesman problem.. J. Japanese Society Arti. Intelligence, 6, 1049–1059.

[21] Gen, M. and R. Cheng, R. 1994. Evolution program for resource constrained project scheduling problem." In Proceeding of the first IEEE conference on evolutionary computation. 736–741.

[22] Gen, M. and R. Cheng, R. 1997. "Genetic algorithms and engineering design." New York: Wiley.