Metaheuristic for Selecting Lower Bound Applied to the Problem of Bin Packing

Joaquín Pérez, ¹Hilda Castillo, ¹Darnes Vilariño, ¹Rafael de la Rosa, ¹Adriana Mexican, ¹José C. Zavala, ¹René Santaolaya and ¹Hugo Estrada

¹Centro Nacional de Investigación y Desarrollo Tecnológico, Morelos, Mexico
²Centro Nacional de Investigación y Desarrollo Tecnológico, Benemérita Universidad Autónoma de Puebla, México
³Benemérita Universidad Autónoma de Puebla, Puebla, México
⁴Centro Nacional de Investigación y Desarrollo Tecnológico, Universidad Autónoma del Estado de Morelos, Morelos, México
⁵INFOTEC, México

Abstract: The theoretical lower bounds have been proposed to solve the Bin Packing problem, yet their optimal approximation of the number of bins is limited. In this sense, this paper presents a metaheuristic method to select a lower bound at a low cost. Getting lower bound more precise improves on convergence to the optimal solution methods; it helps to improve the quality of solutions. This paper proposes a new limit; this is a metaheuristic that is based on existing boundaries and characterization of instances of the Bin Packing problem. The characterization of instances provides information about their behavior. To validate this proposal, a benchmark of 16413 instances is used. For each instance the proposed limit and five known limits are calculated. The results of the metaheuristic proposed (LBMH) were compared with 1392 optimal known and this was higher by 34.47% compared to LB1. The results were matched with the optimal value of 1279 instances out of a total of 1392.

Key words: Bin Packing · Lower bounds · Metaheuristic · Benchmark · Characterization · Metrics

INTRODUCTION

The Bin Packing Problem (BPP) has been a topic discussed worldwide by many researchers; this problem can be defined as follows: given n items with weights \( w = \{w_1, w_2, ..., w_n\} \), where \( w_i \) is the weight of the \( i \)th item and such that \( 0 < w_i \leq C \), to allocate the items to bins with capacity \( C \) such that the number of bins used is minimized. This problem is classified as NP-hard [1]. Some problems such as the allocation of processors, the package delivery in computer networks [2], the allocation of allowances, among others, can be mapped to the BPP.

It would be very important to solve the problem, to determine the minimum number of bins that are necessary to accommodate the \( n \) items. To carry out this task different methods have been developed that allow to obtain a lower bound regarding the number of optimal bins. According to the experimental results obtained in this work, it was observed that the values obtained with these limits, in many cases, are far from the optimal number of bins. Another important element that was found in this analysis is the time consumed to calculate the limits and it was not possible to obtain some instances. So it was found what features exhibited this type of instances.

This paper presents a new lower bound which is called LBMH; this is based on a metaheuristic proposed that selects the best method to calculate the lower bound. The main contribution of this work is a heuristic method to select a lower bound at a low cost.

Corresponding Author: Joaquín Pérez, Departamento de Ciencias Computacionales, Centro Nacional de Investigación y Desarrollo Tecnológico, Morelos, México.
Hilda Castillo, Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Puebla, México.
Related Works: In this section the most relevant works that calculate the lower bound for the BPP of a dimension are analyzed.

In 1990 researchers S. Martello and P. Toth [3] in their book "Knapsack problems: Algorithms and Computer Implementations" presented three lower bounds, they are: LB1, LB2 and LB3. The first one calculates the sum of all weights of the items and it is divided between the capacity of the bin. This lower bound tries to find out how many bins are needed to place the \( n \) items. Note that the items may be broken into smaller pieces to obtain the minimum number of bins. The LB2 limit separates the items into groups and considers those items whose weights exceeds half the capacity of the bin. Therefore, the items are placed according to their size beginning with the biggest items. The LB3 limit is based on the dominance criterion proposed by S. Martello and P. Toth [3], this method uses an algorithm that places in the best possible way the items. The results are very good, but when the number of items grows the computation time is impractical.

Scholl [4] used a method that solves the BPP called Bison. This work included the LB4 limit, this limit uses three subsets of items that meet certain conditions in order to optimally store the items in the bins and this method is similar to the LB2 limit proposed by S. Martello and P. Toth [3].

In 2010 Jarboui [5] presented a method to calculate the LB1 or LBJ limit. The LBJ limit starts with a notional value and if there is no feasible solution whose value is less than or equal to the initial value, then it is assumed than the lower bound is the initial value plus one.

Several works have been proposed for the characterization of instances; the most relevant is M. Quiroz [6], who carried out an analysis of the metrics proposed by Cruz [7] and Alvarez [8]. In this paper were implemented and analyzed 31 metrics, of which only 14 were suitable for to discern whether an instance is easy or difficult to solve. In addition, some metrics are proposed to evaluate the performance of a genetic algorithm for BPP.

Heuristic Proposed: In recent years to find the optimal solution of BPP, it has been developed various models, which have introduced the concept of lower limit. The lower limit expresses the number of containers that must at least be used to place \( n \) objects. Compute the lower bound of an instance does not mean finding the number of containers that are really necessary to place \( n \) objects and therefore this value may not match the optimal value of the problem.

However, by being a lower bound, it is impossible to find a solution where the number of bins is less than the lower bound. If it is possible to get the lower bound of an instance and the value obtained is the optimal or close to the number of bins of the optimal solution, then it will be more feasible than the heuristics can determine if they have reached the optimal. Given the above, then exacts and metaheuristics methods may restrict the number of bins in the solution and may obtain the optimal solution so the quality of the solution is ensured.

Therefore it can be said that getting the maximum value resulting from LB1, LB2, LB3, LB4 and LBJ limits will be the closest or even equal to the number of bins in the optimal solution value.

However, calculating all these lower bounds can be counterproductive for some instances; this is because in some cases it is required much computational time. When performing experiments with 16,413 instances, it was noted that the lower bound, LB3 is the most computing time consuming. It was also noted that in most instances, the obtained lower bounds (LB1, LB2, LB3, LB4, LBJ) have the same value, but there is a considerable number of instances whose best lower bound was between LB1 and LBJ, so it would be sufficient to calculate the LB1 or LBJ limit.

From this knowledge, it is possible to design a heuristic method to filter those instances whose value is the lower bound LB1 or LBJ. The metrics proposed by M. Quiroz [6] were calculated and the results were analyzed; this allowed locating those metrics to characterize the instances. In particular for this work, three metrics were relevant; they are kurtosis, higher and MaxRepe [6]. From this analysis, some patterns were identified, which were transformed into conditions of the metaheuristic LBMH. For example, according to the results obtained, it was observed that those instances whose lower bound is LB1 or LBJ have the feature that the largest object does not exceed 57% of the bin capacity. A more detailed explanation can be found in the next section.

Considering the above stated, the proposed metaheuristic is shown in Figure 1. The description of each of the variables used in the metaheuristic is located in Table 1. In general, for those instances that do not meet any of the conditions (rules) of the metaheuristic, all limits are calculated and the maximum is obtained, it will be the new lower limit.
Table 1: Description of the Variables Used in the Pseudo Code.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>porcItem1</td>
<td>Percentage of the space that occupies the biggest item of the instance in the bin.</td>
</tr>
<tr>
<td>porcMaxRepe</td>
<td>Percentage associated with the metric</td>
</tr>
<tr>
<td>Max Repe</td>
<td>Maximum number of repetitions</td>
</tr>
<tr>
<td>Kurt</td>
<td>Kurtosis associated with the instance.</td>
</tr>
<tr>
<td>w1</td>
<td>Largest weight value of the instance</td>
</tr>
<tr>
<td>lower Bound</td>
<td>Final value of lower bound for the instance</td>
</tr>
<tr>
<td>N</td>
<td>Number of elements</td>
</tr>
<tr>
<td>C</td>
<td>Bin capacity</td>
</tr>
<tr>
<td>LB1, LB2, LB3, LBJ</td>
<td>Calculated limits</td>
</tr>
</tbody>
</table>

Fig 1: Pseudo code of metaheuristic LBMH

```
Input: w = \{w_1, w_2, ..., w_n\} set of weights of items. W bin size
Output: lowerBound calculated value of the lower bound
1  porcItem1 = \sum_{i=1}^{n} w_i / 100 \%
2  porcMaxRepe = \max_i \sum_{j=1}^{n} w_j \times \text{MaxRepe} \times 100 / n
3  Kurt = Kurtosis()
4  if
5   4.1 (porcItem1 < 25 and porcMaxRepe > 20) or
6      4.2 (porcItem1 > 57 and porcMaxRepe > 10)
7   or
8   4.3 (porcItem1 < 57 and porcMaxRepe <= 59 and Kurt > 3.0) or
9   4.4 (porcItem1 < 59 and porcMaxRepe <= 65 and Kurt > 3.0) then
10  lowerBound = \max\{LB1, LBJ\}
11  else
12  7.1 (LB1 = LB2 and Kurt <= 8000) or (Kurt > 15000) then
13    9  LB3 = 0
14  10  lowerBound = \max\{LB1, LB2, LB3, LBJ\}
11  return lowerBound
```

Fig 2: Average number of times in which a limit was selected by the heuristic.

Analysis of the Behavior of the Results: When you are seeking a solution for an instance, it is desirable to know in advance the number of bins to be used in the final solution; this can be carried out using any method to obtain the lower bound. The problem is to know which of the limits is appropriate. After the analysis of the methods to calculate the lower bound, it was observed that the time required to calculate the LB1 limit is almost negligible compared to the time of the other limits. However, the value obtained, in many cases, does not match the optimal. The LB2 and LB4 limits require a longer computation than LB1. In the case of LBJ limit, its time is longer than LB1 but less than LB2 and LB4 and its values are closest of the value optimal. The LB3 limit requires much computation time, especially when the number of items is large, although this is not definite because there are instances with a large number of items whose computation time is reasonable. Those instances, where the calculation of LB3 requires a very high time of calculation (hours), were characterized and a pattern was found. This condition is shown in Figure 1, on line 7.1. If an instance meets this condition, the LB3 limit is set to zero. Subsequently, the rest of the boundaries are calculated and the new lower bound is chosen from the maximum of the boundaries.

For the experimental part a benchmark that consists of 13 datasets was used. Grouped into the following sets: Uniform and Triplets (Falkenauer [9]), Data Set 1, Data set 2 and Data set 3 (Scholl [4]) which we called bin1data, bin2data and bin3data, Was 1 and Was 2 (Schwerin [10]), Dual_distribution (Burke [11]), Hard 28 and 53 NIRUP (Schoenfield [12]), of a total of 16413 instances.

The LB1, LB2, LB3, LB4 and LBJ limits were calculated and it was found that the best limits are LB3 and LBJ. Since the calculation time of LB3 is expensive for many instances then it was decided not to calculate this limit for this type of instances. The behavior of LB3 limit can be summarized as follows: the LB3 limit method uses at the MTRP procedure, who makes use of the dominance criterion, which attempts to place the majority of objects in bins in the best way possible. If the LB3 limit does not places all items, then is uses the LB2 limit and is and in the same way the LB2 limit uses the LB1 limit. Therefore, the LB3 limit has the best performance, because LBJ limit is the most appropriate limit. The LB4 limit is never selected as the appropriate limit. Therefore, it is not used in the heuristic performance, because its values are more approximate to the optimal. It is found that for the 10.5%
Table 2: Average Time Used in Calculating the Lower Limit.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>LB1</th>
<th>LB2</th>
<th>LB3</th>
<th>LB4</th>
<th>LBJ</th>
</tr>
</thead>
<tbody>
<tr>
<td>53nirup</td>
<td>0</td>
<td>0.013</td>
<td>0.602</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>bin1data</td>
<td>0</td>
<td>0.002</td>
<td>2.074</td>
<td>0.002</td>
<td>0.00004</td>
</tr>
<tr>
<td>bin2data</td>
<td>0</td>
<td>0.014</td>
<td>79.315</td>
<td>0.005</td>
<td>0</td>
</tr>
<tr>
<td>bin3data</td>
<td>0</td>
<td>1.49</td>
<td>75.38</td>
<td>0.005</td>
<td>0</td>
</tr>
<tr>
<td>dualdistribution</td>
<td>0.00006</td>
<td>0.035</td>
<td>77.236</td>
<td>0.007</td>
<td>0.00004</td>
</tr>
<tr>
<td>Falkenauer</td>
<td>0</td>
<td>0.039</td>
<td>108.446</td>
<td>0.007</td>
<td>0.00004</td>
</tr>
<tr>
<td>ffd-hardA</td>
<td>0</td>
<td>0.016</td>
<td>2.75</td>
<td>0.002</td>
<td>0.0001</td>
</tr>
<tr>
<td>ffd-hardB</td>
<td>0.000004</td>
<td>0.014</td>
<td>3.568</td>
<td>0.002</td>
<td>0.0001</td>
</tr>
<tr>
<td>ffd-hardC</td>
<td>0.000007</td>
<td>0.014</td>
<td>2.651</td>
<td>0.001</td>
<td>0.00003</td>
</tr>
<tr>
<td>ffd-hardD</td>
<td>0</td>
<td>0.013</td>
<td>1.715</td>
<td>0.001</td>
<td>0.000035</td>
</tr>
<tr>
<td>hard28</td>
<td>0</td>
<td>0.036</td>
<td>3.924</td>
<td>0.005</td>
<td>0</td>
</tr>
<tr>
<td>schwerin</td>
<td>0</td>
<td>0.026</td>
<td>3.552</td>
<td>0.004</td>
<td>0</td>
</tr>
<tr>
<td>waescher</td>
<td>0</td>
<td>0.107</td>
<td>12.61</td>
<td>0.004</td>
<td>0</td>
</tr>
</tbody>
</table>

Average: 0.14 \(\text{time units}/\text{dataset}\) 28.756 0.05 0.0022

of the test instances, the LBJ limit is the most appropriate. The LB4 limit is never selected as the appropriate limit. Therefore, it is not used in the heuristic.

Table 2 shows the average computation time that each dataset used to calculate each of the studied lower limits. This table shows that the LB3 limit is computationally more expensive, based on this observation, LB3 is computed only when necessary.

To know when it is necessary to calculate the LB3 limit for an instance, it is a priority to find a metric or a parameter that allows the discrimination. To do this, the work of Alvarez [8] and Quiroz [6] was reviewed. Fifteen metrics were implanted and only two were used to select those instances for which LB3 limit should not be calculated, they are: kurtosis and MaxRepe. These metrics allowed to isolate instances whose kurtosis is greater than 15000 and other conditions which are shown in figure 1, on line 7.1, it is recommended to use LB1 or LBJ when the elements with very small weights, less than 25% of the capacitance of the bin and a maximum of repetitions that does not exceed 20% of all items of the instance and the only limits to be calculated are LB1 or LBJ. Also, if the objects do not exceed 57% and up to 65% of the capacity of the container and the repetitions (MaxRepe) are very few (at most 3% for one case and 1.5% for the other case), then only must be calculated LB1 and LBJ and the lower bound is the maximum of them.

To select the approximate limit to the optimal, the maximum value of the calculated limits of each instance is obtained and this value is selected as the most appropriate limit for the instance.

Other important and used metrics are: porcItem1 and porcMaxRepe. The first represents the largest percentage related to the capacity of the container object and the second is the percentage represented by MaxRepe regarding the number of objects in the instance (porcMaxRepe).

In summary, after evaluating the results obtained by the metrics, it is observed that it only is needed to calculate the LB1 limit to get the lower bound for those instances where the items are small. If the instances contain large and small items, it is necessary to calculate all limits and to get the maximum. But if the kurtosis is very large then it is not necessary to calculate the LB3 limit because of the computational cost.

Experimental Results: To validate the heuristic proposed, the limits mentioned in the previous section and the metaheuristic LBMH were implemented. The programming of these lower bounds was carried out in the programming language C++ using Windows 7 operating system.

With the implementation of the proposed metaheuristic, the lower bound was computed to all sets of instances. From the experiments, it was observed that

Table 3: Number of Instances of Each Data Set That Comply with Any the Rules of LBMH.

<table>
<thead>
<tr>
<th>Rules</th>
<th>4.1</th>
<th>4.2</th>
<th>4.3</th>
<th>4.4</th>
<th>7.1</th>
<th>neither (9)</th>
<th>Ninstancias</th>
</tr>
</thead>
<tbody>
<tr>
<td>53nirup</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>52</td>
<td>53</td>
<td></td>
</tr>
<tr>
<td>bin1data</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>720</td>
<td>720</td>
<td></td>
</tr>
<tr>
<td>bin2data</td>
<td>240</td>
<td>200</td>
<td>0</td>
<td>10</td>
<td>30</td>
<td>480</td>
<td></td>
</tr>
<tr>
<td>bin3data</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>dualdistribution</td>
<td>0</td>
<td>23</td>
<td>0</td>
<td>206</td>
<td>31</td>
<td>260</td>
<td></td>
</tr>
<tr>
<td>Falkenauer</td>
<td>80</td>
<td>0</td>
<td>0</td>
<td>60</td>
<td>140</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ffd-hardA</td>
<td>0</td>
<td>102</td>
<td>49</td>
<td>64</td>
<td>1285</td>
<td>1500</td>
<td></td>
</tr>
<tr>
<td>ffd-hardB</td>
<td>500</td>
<td>1123</td>
<td>46</td>
<td>50</td>
<td>1981</td>
<td>3700</td>
<td></td>
</tr>
<tr>
<td>ffd-hardC</td>
<td>900</td>
<td>2762</td>
<td>49</td>
<td>23</td>
<td>2066</td>
<td>5800</td>
<td></td>
</tr>
<tr>
<td>ffd-hardD</td>
<td>0</td>
<td>1875</td>
<td>24</td>
<td>9</td>
<td>1592</td>
<td>3500</td>
<td></td>
</tr>
<tr>
<td>hard28</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>28</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>schwerin</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td></td>
<td></td>
</tr>
<tr>
<td>waescher</td>
<td>11</td>
<td>9</td>
<td>0</td>
<td>2</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>1851</td>
<td>6185</td>
<td>168</td>
<td>156</td>
<td>206</td>
<td>7847</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Number of Instances That Matched the Optimal

<table>
<thead>
<tr>
<th>Instancias</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
<th>L4</th>
<th>LJ</th>
<th>LBMH</th>
<th>Ninst</th>
</tr>
</thead>
<tbody>
<tr>
<td>53nirup</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>53</td>
</tr>
<tr>
<td>bin1data</td>
<td>255</td>
<td>532</td>
<td>665</td>
<td>322</td>
<td>255</td>
<td>665</td>
<td>720</td>
</tr>
<tr>
<td>bin2data</td>
<td>457</td>
<td>457</td>
<td>434</td>
<td>457</td>
<td>480</td>
<td>480</td>
<td>480</td>
</tr>
<tr>
<td>bin3data</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>dualdistribution</td>
<td>0</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Falkenauer</td>
<td>58</td>
<td>58</td>
<td>58</td>
<td>58</td>
<td>58</td>
<td>58</td>
<td>58</td>
</tr>
<tr>
<td>ffd-hardC</td>
<td>0</td>
<td>2</td>
<td>9</td>
<td>0</td>
<td>2</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>ffd-hardD</td>
<td>0</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>hard28</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>schwerin</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>798</td>
<td>1082</td>
<td>1208</td>
<td>866</td>
<td>843</td>
<td>1279</td>
<td>1392</td>
</tr>
</tbody>
</table>

Porcentaje: 57.33 77.73 86.78 62.21 60.56 91.88 100
the rule that more instances fulfill is the rule 4.2. This rule indicates that most of the instances studied have objects that occupy at most 57% of the capacity of the bin and the percentage of repetitions (MaxRepe) is less than 10. More details on how many instances fulfill the proposed rules can be seen in Table 3.

Further it did the comparison of the number of bins obtained in the optimum solution against the lower bound obtained with the heuristic. That was performed with 1392 instances. The optimal values for sets: dataset1, dataset2 and dataset3 were taken from an internet site and for the sets: hard28, 53nirup, dualdistribution, ffd-Hardc, ffd-hardd and Schwerin hard28, were obtained from the solutions found with the implementation of the model proposed by J. M. Valerio [13] and [14]. When the results were analyzed, it was found that for 1279 instances, the optimal number of bins coincided with the lower bound obtained with the proposed heuristic. If LB1 limit had been used, this bound would have coincided only with 798 optimal. Table IV shows the number of instances per dataset, coinciding with the optimal value. The percentage of the different boundaries that coincided with the optimal values is shown at the bottom of the table. It is relevant to note that the proposed metaheuristic LBMH exceeded by 34.57% the LB1 limit.

According to these results, the metaheuristic LBMH obtained an accurate lower bound than the other limits discussed.

CONCLUSIONS

Based on the results, we concluded that it is possible to improve the calculation of the lower bound of an instance using the techniques found in the literature and through the characterization of the instances. The characterization of the instances indicates the technique or techniques to be used to calculate the limit.

This heuristic can be used in metaheuristic systems to solve instances of the BPP. These systems typically perform a number of iterations and their stop conditions is to reach the lower bound value optimal or exceed a certain number of iterations. If the optimal lower bound is not reached, the system will stop because it exceeded the number of iterations, so you will not know if the optimal is reached or not. With this proposal, the quality of the lower limit is improved and these systems will know whether the solution is or not optimal.

In addition, the heuristic approach could be used in the exact and heuristic approach.

The methods for obtaining the lower bound of an instance that resulted closer to the number of bins of the optimal solution were LB3 and LBJ. However, the limit LB3 consumed more time than the limit LBJ.

The limit LB4 is rarely selected by the instances. For this sample, only one instance selected this limit, so we decided not to use it in the metaheuristic. Because, the limit LB2 exceeds the limit LB4 in the number of optimal found, see Table 4.

According to the results, we observed that for instances of objects whose weights do not exceed half of the capacity of the bin and where the number of repetitions, MaxRepe, is small enough, it is necessary only to compute LB1 and LBJ, so the bound lower will be the maximum of these values.

When an instance has items have large and small weights (greater than 65% than the capacity of the bin) and the number of repetitions is high (more than 3%), then in that case it was not possible to discern about what is more appropriate limit be calculated, all limits should be calculate in order to select the maximum value.

REFERENCES


