A Constructive Heuristic for Discreet Berth Allocation Problems

Masri Ayob, Mohd Zakree Ahmad Nazri, Tri Basuki Kurniawan, Yang Xiao Fei, Esam Taha Yassen

Abstract

Berth allocation problem (BAP) is one of the most studied problems in the maritime transportation. BAP aims to allocate for a set of vessels a berthing time and position on a given terminal. BAP is a combinatorial optimization problem, which is difficult to find an optimal solution. Many heuristic and meta-heuristic methods have been proposed to deal with this problem. Most of these works are improvement heuristics, which require a constructive heuristic to generate an initial solution. Therefore, this work proposes a randomized greedy constructive heuristic (GR-BAP) for BAP. Instead of using a fully random and/or a greedy method, a part of the solution is randomly generated. Whilst, the rest of solution is constructed in a greedy manner. This approach will avoid generating low quality solutions, as the case with fully random. Furthermore, it is also different from the greedy method, which always generate the same solution, because GR-BAP can generate different solution for each trial. The proposed GR-BAP is applied on the well-known BAP benchmark instances. Results show that GR-BAP outperforms a fully random method on all tested instances. This indicates that combining greedy and random components helps the heuristic to produce good quality solution.

Keywords: Constructive heuristic, berth allocation problem, greedy heuristic

1. Introduction

Berth allocation problem (BAP) is a well-known optimization problems in the field of maritime transportation system [1][2]. BAP are classified into two types depending on berth type: discrete and continuous variants [1]. The discrete BAP allows only one vessel to be berthing at a time at each berth, regardless of vessel size. Whilst, continuous BAP allows several vessels to be berth at one berth at the same time. Optimizing the usage of berth is crucial for a cost-effective container terminal management, because a good quality of berth allocation can enhance vessel’s owner satisfaction and subsequently increase the terminal productivity, leading to higher revenues [3].

Recently, BAP has attracted researcher’s attention due to the increase in the containers usage in global transportation. BAP is a combinatorial optimization problem [2] that is difficult to solve for optimality. In the BAP, a set of the berth layout of a container terminal and a set of vessels that have to be served within the planning horizon is given. The aim of solving BAP is to maximize the usage of berth while simultaneously determine the time and berthing position for a give set of vessels in such a way that no more than one vessel is allocated to the same berth at the same time. The objective is to minimize the total handling time of all vessels [4]. To allocate vessel to the berth, additional data such as the vessel’s length, clearance, and draft; and the expected arrival and handling time are required for each vessel [1]. Vessels need to be moored within the boundaries of the quay. There are other constraints involved in berth allocation, which leads to a multitude of BAP formulations. These constraints restrict the feasible berthing positions of vessels according to the partitioning of the quay into berths and several other restrictions.

Since exact methods are only suitable for solving small size problems [1], a variety of heuristics and meta-heuristic methods have been developed to tackle BAP. Examples of these methods are tabu search [2][5][15], variable neighbourhood search [6], clustering search [7], simulated annealing [8], genetic
algorithm [9][10] and particle swam optimization [14]. Most of these works are improvement heuristics, which usually start with a given initial solution. A constructive heuristic is usually used to generate an initial solution. Good initial solution usually helps meta-heuristic to find a better quality solution. Unfortunately, many studies such as [6] simply used a random approach to construct the initial solution, which usually fail to generate good quality solution or even feasible solution. If the generated solution is not feasible, then a repair mechanism is usually applied in order to obtain a feasible solution. Since the BAP deal with several mandatory constraints that must be satisfied in order to obtain a feasible solution, the random approach will struggle to obtain a feasible solution.

In order to obtain an acceptable quality of initial solution, this work proposes a randomized greedy constructive heuristic (denoted as GR-BAP) for BAP. The proposed GR-BAP constructs a portion of the solution using a random selection method while, the rest of the solution is constructed in a greedy fashion. This approach will avoid generating low quality solutions; and by having a random mechanism, the generated solution will be different every time the GR-BAP is executed. This is different from the greedy method where the generated solution is always the same even when executed several times. The aim of this work is to generate an acceptable quality of initial solution by combing the benefit of both random and greedy constructive methods. GR-BAP is applied on the well-known BAP benchmark instances and compared with the fully random method. Results show that the proposed GR-BAP produced a better quality solutions compared to the fully random method, for all BAP instances tested in this work.

2. The Proposed Method: Greedy Randomized Constructive Heuristic

Constructive heuristics are used by many researchers to generate an initial solution for the iterative improvement methods such as simulated annealing and tabu search. Well-known examples of constructive heuristics are random and greedy methods. Random method is the most easy and simple one to be applied. Random method constructs the initial solution in iterative manner by selecting one element at random and adds it into a given partial solution. The process will stop once a complete solution is constructed. Although it is widely used, the generated solution is usually not in good quality and far from optimal solution. Furthermore, a low quality solution might need a lot of computational time to be improved. This case is not preferred especially when the solution should be improved within a fixed amount of time. Greedy method, on the other hand, constructs the initial solution in a greedy manner. It sorts elements in a solution using a pre-defined sorting criterion and adds them to a given partial solution according to the given orders. Greedy method can generate a better quality solution compared to a random method. However, the quality of the generated solution by a greedy method might be within attraction basin area. Hence, is very difficult to further improve it by the improvement base methods. Furthermore, a greedy method always returns the same quality solution even when it is applied many times.

To combine the benefit of both random and greedy methods, this work proposes a greedy randomized constructive heuristic (denoted as GR-BAP). GR-BAP takes the advantage of both random and greedy by constructing a portion of the solution at random and the rest using a greedy method. The pseudocode of GR-BAP is presented in Figure 1.

```
1. Set the random portion of the solution, R;
2. Set counter I=0;
3. While I<R do
   4. Randomly select one vessel from the unscheduled vessels queue
   5. Assign a possible berthing time and position with minimum cost for the selected vessel
   6. Delete the selected vessel from the vessels queue
   7. I=I+1
   8. End While
9. Sort the unscheduled vessels queue in ascending order based on their arrival time
10. While the unscheduled vessels queue is not empty
   11. Assign a possible berthing time and position with minimum cost for the first vessel in the vessels queue
   12. Delete the selected vessel for the vessels queue
13. End While
```

Figure 1. Greedy Randomized (GR-BAP) Pseudocode
GR-BAP works as follows: first it initializes the random portion parameter, $R$, (line 1) that represent the percentage of the solution that will be constructed in a random manner. At random stage (line 1-8), for each iteration, one vessel will be selected at random and will be assigned to a possible berthing position and time with minimum cost and without violating the constraints. The assignment is then added to the current partial solution. The process is repeated until the counter reach to $R$ value.

At greedy stage (line 9-12), the unscheduled vessels are then sorted in ascending order based on the arrival time (line 9); and will be scheduled starting with the one in the top list. For each vessel, the algorithm will assign a possible berthing position and time with minimum cost and without violating the constraints (lines 11-12). The assignment process is repeated until a complete solution is generated (lines 10-13). The GR-BAP will return a complete solution, once all vessels are scheduled.

3. Experimental Setup

The proposed GR-BAP is applied on the BAP benchmark instances introduced by [2][7]. In this work, the quality of the $s^{th}$ solution, $f(s)$ is calculated using equation (1) as in [7]. That is the quality of solution is evaluated based on the vessel waiting time. In solving this BAP, good quality solution has small waiting time and we will have an optimal solution if the waiting time is zero.

$$f(s) = m \sum_{i=1}^{n} \sum_{k=1}^{m} v_i \left[ T_i^k - a_i + t_i^k \sum_{j \in m} x_{i,j}^k \right] \quad (1)$$

Where,
- $n$: the number of vessels
- $m$: is a set of berth
- $v_i$: the priority of the $i^{th}$ vessel
- $T_i^k$: is the berthing time of vessel $i$ at berth $k$.
- $a_i$: the arrival time of vessel $i$
- $t_i^k$: handing time of vessel $i$ at berth $k$.
- $x_{i,j}^k$: a descion variable where $x_{i,j}^k = 1$ if vessel $j$ is serviced by berth $k$ after the vessel $i$

This work used six different instances with different size; and each one contains 60 vessels and 13 berths. These instances were generated and formulated by [2] based on Gioia Tauro port in Italy.

In this work, GR-BAP was implemented in Java and performed on PC with Core2 Quad with 2.66GH and 2.0 GB RAM. The proposed GR-BAP has two parameters: the termination creation of the search process and the percentage of the random portion, $R$. In this work, GR-BAP terminates when a complete solution is found, i.e., when all vessels have been allocated to the berth. Parameter $R$ represents the percentage of the random portion that will be allocated using the random method. The size of $R$ has an impact on the quality of the generated solution and thus it needs to be set carefully. In this work, three different values for $R$ is tested: 25%, 50% and 75% of the unscheduled vessels. To determine which values is the best (25%, 50% or 75%), a preliminary test was conducted using the six instances mentioned above. GR-BAP is executed 10 times.

4. Results

In the preliminary test, the best results out of 10 runs using the three different values of $R$ (25%, 50% and 75%) are presented in Table 1. The best results are shown in bold. As can be seen from Table 1, GR-BAP with $R=25\%$ produced the best results for all instances, while $R=50\%$ obtained the second best results followed by $R=75\%$. Therefore, GR-BAP with $R=25\%$ is used for comparing GR-BAP with random method. Since this is a constructive heuristic, and all the works in the literature reported on improvement heuristic (which use a solution generated by the constructive heuristic as an input/initial solution), it is not fair to make a comparison with those heuristics. Furthermore, the aim of this work is to prove that GR-BAP is capable of producing a good quality solution compared to random approach that usually used in the literature.
Table 1. The best results out of ten runs of the preliminary test for GR-BAP using different values of R (the percentage of the random portion)

<table>
<thead>
<tr>
<th>Instances</th>
<th>R=25%</th>
<th>R=50%</th>
<th>R=75%</th>
</tr>
</thead>
<tbody>
<tr>
<td>i01</td>
<td>2560</td>
<td>3172</td>
<td>3783</td>
</tr>
<tr>
<td>i02</td>
<td>2364</td>
<td>3053</td>
<td>4186</td>
</tr>
<tr>
<td>i03</td>
<td>2877</td>
<td>2962</td>
<td>2974</td>
</tr>
<tr>
<td>i04</td>
<td>3269</td>
<td>3435</td>
<td>3375</td>
</tr>
<tr>
<td>i05</td>
<td>2503</td>
<td>3269</td>
<td>3328</td>
</tr>
<tr>
<td>i06</td>
<td>2995</td>
<td>3034</td>
<td>3106</td>
</tr>
</tbody>
</table>

Note:
- All the values in column 2-4 are the quality of the best obtained solutions out of ten runs, calculated using equation (1).
- GR-BAP is a greedy randomized constructive heuristic for BAP.
- R is the random portion of the solution

Table 2 reports a comparison between GR-BAP with a fully random method. Both methods are tested on the same BAP instances mentioned in section 3. The best results of both methods over 10 runs are reported. The best obtained results are highlighted in bold font. Results in Table 2 shows that GR-BAP outperformed the fully random method on all six BAP instances. These results are not surprising because, in GR-BAP only 25% of the solution is randomly constructed. Other 75% of the solution is created by assigning each vessel to a possible berthing time and position with minimum cost. That is 75% of the assignments aim to reduce the waiting time of the vessel. Meanwhile, in the fully random method the whole solution is constructed at random regardless of the waiting time. This is the reason, why the fully random method is not better than GR-BAP. Furthermore, GR-BAP also shown a good performance compared to the fully random method, even when 75% of the solution is constructed at random.

Table 2. The results of GR-BAP with 25% of the solution is randomly constructed compared to random method.

<table>
<thead>
<tr>
<th>Instances</th>
<th>GR-BAP</th>
<th>Random method</th>
</tr>
</thead>
<tbody>
<tr>
<td>i01</td>
<td>2560</td>
<td>3971</td>
</tr>
<tr>
<td>i02</td>
<td>2364</td>
<td>4452</td>
</tr>
<tr>
<td>i03</td>
<td>2877</td>
<td>3068</td>
</tr>
<tr>
<td>i04</td>
<td>3269</td>
<td>3642</td>
</tr>
<tr>
<td>i05</td>
<td>2503</td>
<td>3412</td>
</tr>
<tr>
<td>i06</td>
<td>2995</td>
<td>3384</td>
</tr>
</tbody>
</table>

Note:
- All the values in column 2, 3 are the quality of the best obtained solutions out of ten runs, calculated using equation (1).
- GR-BAP is a greedy randomized constructive heuristic for BAP.

To conclude, the results shows that GR-BAP is very effective when compared to the fully random method. Therefore, combining randomized and greedy method can produce a good quality solutions compared to the random method. This is a constructive heuristic that only use one iteration to generate a solution, whilst other works in the literature such as [5] [2][7][6][11] are an improvement heuristic where thousands of iterations are performed to generate a final solution. Therefore, a fair comparison cannot be performed. Moreover, the proposed method (GR-BAP) is aimed to prove that a combination of randomized and greedy method can produce a good quality solution compared to the random method.
5. Conclusion

This work had proposed a constructive heuristic for the discrete berth allocation problems. The proposed method combines random method with greedy constructive heuristic. The random method is invoked first to construct a portion of the solution at random. This aims to ensure different solution is generated every time the algorithm is run. That is, for each run, the solution will be generated from different solution space, which is good to be used as an initial solution for an improvement heuristic. Then, the rest of the solution is constructed in a greedy manner with the aim is to minimize the total waiting time of the vessels. The proposed method is tested over the well-known berth allocation benchmark instances as in [2][7]. A preliminary test is conducted to determine the percentage of the solution that will be constructed at random. Results show that constructing 25% of the solution at random is better than 50% and 75%. A comparison between the proposed method and the fully random method showed that the proposed method obtained better quality solutions for all instances. This indicates that combining randomized and greedy method can produce an effective heuristic compared to the random heuristic method. Therefore, this prove that the solution generated by GR-BAP is more suitable to be applied as an initial solution for any improvement heuristic; compared to the solution generated by the fully random method because usually the improvement heuristic has a better chance or can quickly find a near optimal solution if they begin with a good quality initial solution.

6. Acknowledgments

The authors wish to thank the Universiti Kebangsaan Malaysia for supporting this work under the “Industrial and Community Research Grant Scheme” (INDUSTRI-2013-034) and Privasia Technology Bhd. (our industrial collaborator).

7. References


heuristic based on tabu search for the dynamic berth allocation problem”, Engineering Applications of
Artificial Intelligence, 25(6), pp.1132-1141. http://dx.doi.org/10.1016/j.engappai.2012.06.001

[12] Xin-She Yang, Zhihua Cui, Renbin Xiao, Amir Hossein Gandomi and Mehmet Karamanoglu. “Swarm


hybrid heuristic based on tabu search for the dynamic berth allocation problem”, Engineering
Applications of Artificial Intelligence, 25(6), pp.1132-1141.