Computers & Operations Research 70 (2016) 140–154

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

An adaptive large neighborhood search for the discrete and continuous Berth allocation problem

Geraldo Regis Mauri [a,*], Glaydston Mattos Ribeiro [b], Luiz Antonio Nogueira Lorena [c], Gilbert Laporte [d]

[a] Federal University of Espírito Santo, Brazil
[b] Federal University of Rio de Janeiro, Brazil
[c] National Institute for Space Research, Brazil
[d] CIRRELT and HEC Montréal, Canada

ARTICLE INFO

Available online 12 January 2016

Keywords:
Berth allocation problem
Metaheuristic
Adaptive large neighborhood search

ABSTRACT

The Berth Allocation Problem (BAP) consists of assigning ships to berthing positions along a quay in a port. The choice of where and when the ships should move is the main decision to be made in this problem. Considering the berthing positions, there are restrictions related to the water depth and the size of the ships among others. There are also restrictions related to the berthing time of the ships which are modeled as time windows. In this work the ships are represented as rectangles to be placed into a space × time area, avoiding overlaps and satisfying time window constraints. We consider discrete and continuous models for the BAP and we propose an Adaptive Large Neighborhood Search heuristic to solve the problem. Computational experiments indicate that the proposed algorithm is capable of generating high-quality solutions and outperforms competing algorithms for the same problem. In most cases the improvements are statistically significant.

1. Introduction

The marine transportation of cargo has increased in the last decades due to the growth of the international trade. In order to satisfy this demand and to handle the intense flow of ships and cargo in ports (containers for example), the service time of the ships must be minimized while balancing the fast service requirements made by shipping companies and the economical use of available resources in the port. This concern gives rise to a logistical planning problem known as the Berth Allocation Problem (BAP). According to Imai et al. [1] the BAP has a primary impact on the efficiency of port operations. The BAP consists of determining where and when ships should berth along a quay, subject to side constraints. Its objective is usually to minimize the total time spent by the ships at the port.

There exist several modeling and solution approaches for the BAP, depending on the problem features. Here we consider two different versions of the problem called the discrete (BAP-D) and the continuous (BAP-C) cases, both of which are NP-hard [2]. The discrete case considers that the quay is divided into several specific berthing positions and only one ship can be serviced at a time in each berth, regardless of its size. In the continuous case, ships can berth anywhere but spatial restrictions must be considered, as for example the size of the ships. More details about this will be reported in Section 2.

The BAP is usually solved by means of heuristics. Thus Cordeau et al. [2] presented a Tabu Search (TS) heuristic to solve BAP-D and BAP-C. A Simulated Annealing (SA) heuristic was later proposed by Mauri et al. [3] for BAP-D using the same instances as those of [2]. Mauri et al. [4] proposed a Population Training Algorithm with Linear Programming (PTA/LP) to solve BAP-D. The results improved some of the solutions presented in [3]. Following the previous works, Lopes et al. [5] presented a Greedy Randomized Adaptive Search Procedure (GRASP) with Path-Relinking (PR) for BAP-D. All solutions reported by the authors have been proven to be optimal by an exact algorithm called GSPP proposed by Buhrkal et al. [6].

Lalla–Ruiz et al. [7] proposed a hybrid metaheuristic combining a Tabu Search (TS) based on the one reported in [2], with Path-Relinking (PR) to solve BAP-D. These authors have also generated a new set of instances with realistic features. For these instances, they applied the GSPP reported in [6], the TS proposed in [2], an improved TS and the TS-PR proposed by them. GSPP was able to prove optimality only on small instances, and the TS-PR found the best solutions.

---

[*] Corresponding author. Tel.: +55 28 3552 8902; fax: +55 28 3552 8903.
E-mail addresses: geraldo.mauri@ufes.br (G.R. Mauri), glaydston@pet.coppe.ufrj.br (G.M. Ribeiro), lorena@lac.inpe.br (L.A.N. Lorena), gilbert.laporte@cirrelt.ca (G. Laporte).

http://dx.doi.org/10.1016/j.cor.2016.01.002
0305-0548/

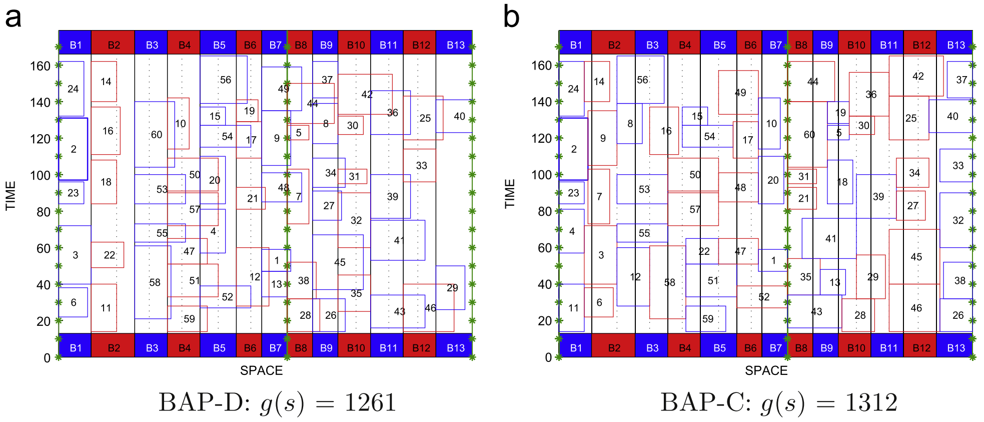

Fig. 1. Best-known solutions for the instance i02. (a) BAP-D: $g(s)=1261$. (b) BAP-C: $g(s)=1312$. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Mauri et al. [8] proposed a Memetic Algorithm (MA) and adapted the SA presented in [3] to solve BAP-C, outperforming the results found by TS proposed in [2]. A Clustering Search (CS) hybrid metaheuristic was used by Oliveira et al. [9,10] to solve BAP-D and BAP-C, respectively. The authors found the optimal solutions for all instances of BAP-D (proved by [6]). For BAP-C, CS improved the solutions found in [2] and [8]. Ting et al. [11] proposed a Particle Swarm Optimization (PSO) algorithm for BAP-D. The authors found the same optimal solutions reported in [5,6] and [9] within shorter computational times. However, for BAP-C, the optimal solutions for the instances proposed in [2] are unknown, and to the best of our knowledge, CS [10] has yielded the best-known solutions.

In order to solve BAP-D and BAP-C, we propose an Adaptive Large Neighborhood Search (ALNS) heuristic based on the principle of destruction and recreation of solutions. We believe we are the first researchers to use ALNS to solve the BAP. In addition, we must stress that only few authors have developed algorithms that can be applied to both BAP-D and BAP-C.

Our computational experiments show the relative superiority of our ALNS algorithm. It found all optimal solutions for BAP-D and new best solutions were obtained for BAP-C on the instances proposed in [2]. Our ALNS was also able to find new best solutions for the instances proposed by [7] for the BAP-D. In addition, we generated a new set of instances based on defining continuous features in the instances proposed by [7]. Again, our ALNS presented highly competitive results.

The remainder of this paper is organized as follows. Section 2 describes the BAP in details. Our ALNS heuristic is described in Section 3. Computational results are presented in Section 4, followed by conclusions in Section 5. An appendix depicts the new best solutions found by ALNS for BAP-C on the instances proposed by [2].

2. The Berth Allocation Problem

In the BAP, the aim is to assign ships to berths subject to several physical and technical restrictions. To define the allowable berth positions of a ship, one must take consideration constraints on the depth of the water (allowable draft) and on the maximum distance from the most favorable location along the quay, considering the ship's length and the location of the outbound and inbound cargo. Some ports have hard draft constraints since they are located in geographical regions subject to important tide variations. In such cases, the ships can only berth within time windows of about two hours around the high tide [12]. In addition to this time window constraint, there exist time windows for the completion time of ship servicing [2]. The handling time of a ship depends on its berthing point and is a function of the distance from the berth to the pick-up and delivery area of the cargo (containers for example) stored in the port yard. Each berth has therefore a different time period during which it can handle each ship, and depending on the availability of the equipment located in the berth, some ships cannot always be handled by it.

These features make it possible to model the BAP in different ways. As mentioned, considering the spatial aspects of the berths and ships, the BAP can be modeled as a discrete or as a continuous problem [13]. In BAP-D, the quay is divided into several berths and only one ship can be serviced at the same time in a berth. The length of the ships are not considered and no berth interferes with the other ones. In BAP-C, there are no divisions along the quay and the ships can therefore move at any position. In another version of BAP-C, the quay is divided into berths, but the large ships can use more than one position, thus allowing small ships to share their berth [2]. This version allows for a better representation of reality, especially when some berths have different equipments capable of operating specific types of cargo (ships and containers of various sizes, for example).

Cordeau et al. [2] and Mauri et al. [8] have shown that the BAP can be represented in a two-dimensional space (see Fig. 1(a) and (b)) considering the ships as rectangles whose two dimensions are the ship length, including a safety margin, and its handling time. The BAP can therefore be viewed as the problem of packing these rectangles into this two-dimensional space without overlaps and while satisfying the time window and availability constraints.

Following Cordeau et al. [2] and Mauri et al. [8], in this paper we consider a quay divided into berths with two parts, the left and the right. So, each berth k has two neighbors, the right part of the berth $k-1$ and the left part of the berth $k+1$. Discontinuous segments must be also considered, representing the initial and final berths, as well as natural obstacles such as sharp curves, for example. In these cases, the berths have only one neighbor and are not divided (see berths 1, 7, 8, 9 and 13 in Fig. 1(a) and (b)).

Fig. 1(a) and (b) depict the optimal solution (cost$=g(s)=1261$) and the best-known solution (cost$=g(s)=1312$) for instance i02 (also used in our computational experiments – see Section 4) considering the BAP-D and BAP-C representations, respectively. In these figures, we use the colors red and blue to enhance visualization. Each rectangle represents a ship, and the red ones are assigned to the nearest red berth. This is also true for the blue ones. The green lines indicate discontinuities and the solid and dotted black lines indicate the different berths and their parts (left and right), respectively. We can see that for BAP-D there are