Towards a Matheuristic Approach for the Berth Allocation Problem

Eduardo Aníbal Lalla-Ruiz[1(✉)] and Stefan Voß[2]

[1] Department of Computer Engineering, University of La Laguna, Santa Cruz de Tenerife, Spain
elalla@ull.es
[2] Institute of Information Systems, University of Hamburg, Hamburg, Germany
stefan.voss@uni-hamburg.de

Abstract. The Berth Allocation Problem aims at assigning and scheduling incoming vessels to berthing positions along the quay of a container terminal. This problem is a well-known optimization problem within maritime shipping. For solving it, we propose two POPMUSIC (Partial Optimization Metaheuristic Under Special Intensification Conditions) approaches that incorporate an existing mathematical programming formulation. POPMUSIC is an efficient metaheuristic that may serve as blueprint for matheuristics approaches once hybridized with mathematical programming. In this regard, the use of exact methods for solving the sub-problems defined in the POPMUSIC template highlight an interoperation between metaheuristics and mathematical programming techniques, which provide a new type of approach for this problem. Computational experiments reveal excellent results.

1 Introduction

Large optimization problems usually require significant computational effort. A natural way to solve these problems is by decomposing them into independent sub-problems that are treated with an appropriate procedure. In doing so, [9] propose the POPMUSIC framework. Its basic idea is to locally optimize sub-parts of a solution, 'a posteriori,' once a solution to the problem is available. These local optimizations are repeated until a local optimum is found. POPMUSIC may be viewed as a local search working with a special, large neighbourhood.

In this paper, we study the application of POPMUSIC for solving the discrete Dynamic Berth Allocation Problem (DBAP) proposed by [4]; for a general survey on berth allocation problems see [1]. In the DBAP, we are given a set of incoming ships N and a set of berths M. Each ship $i \in N$ has to be assigned to an empty berth $j \in M$ within their (berth and ship) time windows. The main goal of this problem is to minimize the sum of the ships service times, i.e. the time required to serve a ship from its arrival. This problem has been modeled as a Generalized Set-Partitioning Problem (GSPP) [3], its implementation in CPLEX allows to solve small-sized problem instances within reasonable computational times [6]. However, as the size of the instances becomes larger, it runs out of memory.

P.M. Pardalos et al. (Eds.): LION 2014, LNCS 8426, pp. 218–222, 2014.
DOI: 10.1007/978-3-319-09584-4_20

The GSPP formulation is as follows. A column represents a feasible assignment of a ship to a berth. The set of columns is denoted by Ω. Two matrices A and B are defined, both containing $|\Omega|$ columns. Matrix $A = (A_{i\omega})$ contains a row for each ship, and $A_{i\omega} = 1$, if and only if column ω represents an assignment of ship $i \in N$ to a berth. Each column of A contains exactly one non-zero element. Matrix $B = (B_{p\omega})$ contains a row per (berth, time) position. The rows of B are indexed by the set P, with $|P| = \sum_{k \in M}(e^k - s^k)$, where s^k and e^k are the start and end of the availability of berth k, respectively. The entry $B_{p\omega}$ is equal to 1, if and only if, position $p \in P$ is contained in the assignment that column ω represents. The cost c_ω of any column $\omega \in \Omega$ is the service time of the respective position assignment. With these definitions the GSPP formulation of the DBAP presented in [2] is as follows.

$$\min \sum_{w \in \Omega} c_w x_w \tag{1}$$

$$\sum_{w \in \Omega} A_{iw} x_w = 1, \ \forall i \in N \tag{2}$$

$$\sum_{w \in \Omega} B_{pw} x_w \leq 1, \ \forall p \in P \tag{3}$$

$$x_w \in \{0, 1\}, \ \forall w \in \Omega \tag{4}$$

The objective function (1) minimizes the service time of the vessels. The set of constraints (2) ensures that all vessels are served. Finally, the constraints (3) guarantee that at a time interval, in a berth, only one vessel can be served.

2 POPMUSIC Approach for the DBAP

The POPMUSIC approach for the DBAP considers a solution S by means of its scheduling order, where the solution is represented as an integer string and each berth is delimited by a 0. An example of a solution structure for 3 berths and 6 ships is as follows, $S = \{1, 0, 2, 4, 6, 0, 3, 5\}$. In this case, ship 3 is the first ship to be served at berth 3; once it departs, the next ship to be served is ship 5.

Algorithm 1 shows the POPMUSIC approach for the DBAP. The initial solution S is randomly generated by applying a random-greedy method (R-G) proposed by [4]. The solution is divided into h parts depending on the number of berths. The seed part, s_{seed}, is selected at random from the set of parts, H. Once a solution part is selected, the sub-problem R is established by joining the s_{seed} and its r neighbour parts according to the id of the part. Two parts are at distance 1, if they are consecutive, e.g. $part_1$ and $part_2$. The GSPP mathematical formulation of R is solved using CPLEX. The POPMUSIC-G differs from the original in the way the set of parts O is fulfilled when there is an improvement. That is, if a sub-problem is improved, all its composing parts (s_{seed} and its neighbour parts) are included in O.

Once the POPMUSIC process is over, all the solution parts are joined. The information obtained from them is used for determining a reduced problem

Algorithm 1. POPMUSIC framework

```
1  Generate an initial solution s at random using R-G
2  Decompose S in M parts according to the number of berths,
   H = {part_1, ..., part_M}
3  Set O = ∅
4  while O ≠ {part_1, ..., part_M} do
5      Select a seed part s_seed ∈ H at random
6      Build a sub-problem R composed of s_seed and its r nearest parts
7      Optimize R through solving its GSPP mathematical formulation
8      if R has been improved then
9          Update solution S
10         O ← ∅
11     else
12         Include s_seed in O
```

instance that will be provided to CPLEX. Similarly to the corridor method [7] this narrow problem allows CPLEX to solve the complete problem to optimality.

The computational experiments carried out in this work are conducted on a computer equipped with an Intel 3.16 GHz and 4 GB of RAM. The problem instances used for evaluating the proposed algorithm are a representative set of the largest instances provided by [4] and a representative set of instances proposed by [6]. For the latter set of instances the GSPP implemented in CPLEX using a standard computer runs out of memory. Moreover, we make a comparison among the best approximate approaches for each set of instances, namely, (i) Clustering-Search with Simulated-Annealing (CS-SA) [5], (ii) Particle Swarm Optimization (PSO) [8], (iii) T^2S^*+PR Tabu Search with Path-Relinking [6].

Table 1 illustrates the results for the instances of [4]. Regardless of the selection of the parameter r, POPMUSIC and POPMUSIC-G provide optimal solutions in all cases. This characteristic points out that recognizing 'useless' or 'time-consuming' parameters of the problem parameter space can be narrowed through using the information provided by exactly solving the sub-problems. POPMUSIC-G running times are meaningful compared to those of the approximate solution approaches. Note that, although CS-SA and PSO are able to provide optimal solutions for these cases, they cannot guarantee optimality.

Table 2 shows the results obtained for some large instances proposed by [6]. As can be seen, CPLEX runs out of memory as the size of the instances is larger. In this sense, thanks to the POPMUSIC template, the problem can be narrowed and solved to optimality. This feature is relevant when assessing the behaviour of the best solution approach employed for these instances, T^2S^*+PR, where the evaluation of its performance could not be done because CPLEX runs out of memory without providing an upper bound. Evidently, for some of the instances we are able to improve the best solution results to date.

Table 1. Results for the instances provided by [4]

	GSPP [3]		POPMUSIC		POPMUSIC-G		T^2S^*+PR [6]			CS-SA [5]			PSO [8]		
	r=1, 2, 3, 4		r=1, 2, 3, 4		r=1, 2, 3, 4										
	opt.	t(s.)	obj. val	t(s.)	obj. val	t(s.)	obj. val	gap (%)	t(s.)	obj. val	gap (%)	t(s.)	obj. val	gap (%)	t(s.)
i01	1409	33.20	1409	34.2	1409	11.47	1410	0.07	1.41	1409	0.00	12.47	1409	0.00	11.11
i02	1261	29.18	1261	54.21	1261	12.61	1261	0.00	1.26	1261	0.00	12.59	1261	0.00	7.89
i03	1129	28.17	1129	33.79	1129	13.89	1129	0.00	1.13	1129	0.00	12.64	1129	0.00	7.48
i04	1302	29.20	1302	35.68	1302	13.65	1302	0.00	1.30	1302	0.00	12.59	1302	0.00	6.03
i05	1207	27.93	1207	28.14	1207	11.57	1207	0.00	1.21	1207	0.00	12.68	1207	0.00	5.84
i06	1261	29.75	1261	36.6	1261	15.15	1261	0.00	1.26	1261	0.00	12.56	1261	0.00	7.67
i07	1279	32.89	1279	26.73	1279	12.2	1279	0.00	1.28	1279	0.00	12.63	1279	0.00	7.5
i08	1299	30.19	1299	57.12	1299	15.56	1299	0.00	1.30	1299	0.00	12.57	1299	0.00	9.94
i09	1444	30.89	1444	54.2	1444	13.59	1444	0.00	1.45	1444	0.00	12.58	1444	0.00	4.25
i10	1213	29.14	1213	26.57	1213	12.29	1213	0.00	1.21	1213	0.00	12.61	1213	0.00	5.2

Table 2. Results for the instances provided by [6]

	GSPP [3]		POPMUSIC		POPMUSIC-G		T^2S^*+PR [6]		
	r = 1, 2, 3, 4				r = 1, 2, 3, 4				
	opt.	t(s.)	obj. val.	t(s.)	obj. val.	t(s.)	best	gap (%)	t(s.)
40x5-01	2301	41.51	2301	166.46	2301	53.07	2303	0.09	0.90
40x5-02	2829	59.89	2829	118.72	2829	55.23	2834	0.18	1.09
40x5-03	2880	99.20	2880	116.76	2880	59.06	2880	0.00	0.50
40x7-03	—	—	2119	122.78	2119	62.09	2119	0.00	1.17
55x5-03	—	—	5499	371.92	5499	106.71	5499	0.00	2.67
55x7-03	—	—	3825	196.18	3825	129.37	3833	0.21	5.57
55x7-05	—	—	3797	337.76	3797	151.00	3801	0.11	3.56

3 Conclusions

In this paper we have provided a POPMUSIC adaptation to the Discrete Dynamic Berth Allocation Problem. By using a given mathematical programming formulation together with the decomposition approach inherent to POPMUSIC we were able to solve large scale instances from the literature to optimality or close to optimality that had been out of reach for optimal solution before.

While additional experimentation is still needed, the results provided in this work highlight the application of POPMUSIC for solving large-sized problems. In this regard, the POPMUSIC approaches proposed in this work have a great potential for 'recognizing' relaxed constraints in the parameter space of the problem through leveraging the information obtained by solving the sub-problems. This also incorporates an explicit learning mechanism towards having an autoadaptive control of the size of the sub-problems to be solved.

References

1. Bierwirth, C., Meisel, F.: A survey of berth allocation and quay crane scheduling problems in container terminals. Eur. J. Oper. Res. **202**(3), 615–627 (2010)
2. Buhrkal, K., Zuglian, S., Ropke, S., Larsen, J., Lusby, R.: Models for the discrete berth allocation problem: a computational comparison. Transp. Res. Part E **47**(4), 461–473 (2011)

3. Christensen, C.G., Holst, C.T.: Berth allocation in container terminals. Master's thesis, Technical University of Denmark (2008)
4. Cordeau, J.F., Laporte, G., Legato, P., Moccia, L.: Models and tabu search heuristics for the berth-allocation problem. Transp. Sci. **39**, 526–538 (2005)
5. de Oliveira, R.M., Mauri, G.R., Lorena, L.A.N.: Clustering search for the berth allocation problem. Expert Syst. Appl. **39**(5), 5499–5505 (2012)
6. Lalla-Ruiz, E., Melián-Batista, B., Moreno-Vega, J.M.: Artificial intelligence hybrid heuristic based on tabu search for the dynamic berth allocation problem. Eng. Appl. Artif. Intell. **25**(6), 1132–1141 (2012)
7. Sniedovich, M., Voß, S.: The corridor method: a dynamic programming inspired metaheuristic. Control Cybern. **35**(3), 551–578 (2006)
8. Ching-Jung, T., Kun-Chih, W., Hao, C.: Particle swarm optimization algorithm for the berth allocation problem. Expert Syst. Appl. **41**, 1543–1550 (2014)
9. Taillard, É., Voß, S.: POPMUSIC - partial optimization metaheuristic under special intensification conditions. In: Ribeiro, C.C., Hansen, P. (eds.) Essays and Surveys in Metaheuristics, pp. 613–629. Kluwer, Boston (2002)