Artificial Bee Colony Optimization for Yard Truck Scheduling and Storage Allocation Problem

Fangfang Zhang, Li Li(✉), Jing Liu, and Xianghua Chu

College of Management, Shenzhen University, Shenzhen, China
lli318@163.com, x.chu@szu.edu.cn

Abstract. The yard truck scheduling (YTS) and the storage allocation problem (SAP) are two significant sub-issues in container terminal operations. This paper takes them as a whole optimization problem (YTS-SAP) and analyzes the factor of different travel speeds of trucks based on different loads. The goal is to minimize the total time cost of the summation of the delay of requests and the travel time of yard trucks. Due to the simplicity and easy implementation of artificial bee colony (ABC), the algorithm is applied to address the issue. Computational experiment is employed to examine and analyze the problem solutions and the performance of ABC algorithm. Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) are chosen as contrastive algorithms. From the results of the computational experiment, it is found that ABC algorithm can achieve better solution for the YTS-ASP problem.

Keywords: Yard truck scheduling · Storage allocation problem · Artificial bee colony · Travel speed

1 Introduction

Statistics from the Chinese industry information network shows that the port transportation has been developing rapidly in the past decade [1]. As seen in Fig. 1, the throughput of the nationwide port shows an increasing trend year by year. It has grown from 117.67 million tons in 2013 to 124.52 million tons in 2014. The container throughput has grown from 1.9 twenty-foot equivalent unit (TEU) in 2013 to 2.02 TEU in 2014, with an increase of 6.3 % over the previous year. The throughput of coastal port and inland port are 1.7 TEU and 0.2 TEU in 2013, respectively. For such a speedy development, it is necessary to increase the terminal throughput. Unfortunately, in many countries, there is an important pressure from the political and business sectors. It is obvious that port handling capacity becomes more and more important for terminal executives to contend with that of other terminals [2].

More and more researchers have been attracted to do research on the container terminal operation problems [3–5]. As two significant sub-problems in terminal operations, Yard truck scheduling (YTS) refers to the scheduling of yard trucks to convey the containers between the seaside and the yard side, while the (SAP) relates to the optimizing of the storage space. It has shown its remarkable research value in both science and project [3, 5].

D.-S. Huang and K.-H. Jo (Eds.): ICIC 2016, Part II, LNCS 9772, pp. 908–917, 2016.
DOI: 10.1007/978-3-319-42294-7_81

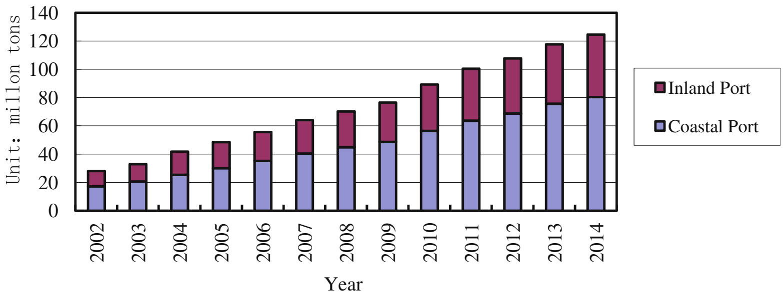

Fig. 1. The growth of the national port throughput from 2002 to 2014. (Color figure online)

Many researchers devote themselves to optimizing the YTS and SAP problems. Nishimura [6] et al. proposed single truck and multi-trucks scheduling models based on the number of work trucks, respectively. Taking the various processing times and ready times into consideration, Ng [7] et al. researched the issue of truck scheduling. For storage allocation problems, Kim [8] et al. considered how to distribute storage space for import containers. Lee [9] et al. considered both the charging and loading operations and developed a new heuristic method to find the scheduling scheme.

It is noticed that the YTS problems are highly correlated to the SAP problems. Bish [10] et al. was the first researcher that took these two issues into consideration as a whole. Lee [11] et al. also raised an integrated model for yard truck scheduling and storage allocation for import containers. Niu [12, 13] et al. used the particle swarm optimization algorithm and bacterial colony optimization algorithm to optimize the integrated YTS and SAP.

In 2005, a new optimization algorithm based on the intelligent foraging behavior of honey bee swarm was introduced by Karaboga, called ABC algorithm [14]. Since then, the ABC algorithm has been successfully applied in solving combinatorial optimization problems, such as job-shop scheduling problem [15], image segmentation [16], uninhabited combat air vehicle path planning [17], training neural networks [18], TSP problems [19], long-term prediction and feature selection. In this paper, we employ ABC algorithm to solve the yard truck scheduling and storage allocation problem.

In general, there are many container transportation related problems, like berth allocation problem, quay crane scheduling problem, yard truck scheduling problem, yard crane scheduling problem and storage allocation problem. This paper commits to solving the combination of yard truck scheduling and storage allocation problem using ABC.

The remaining sections of this paper are organized as follows. Section 2 provides the problem description and formulation. ABC algorithm is presented for problem solutions in Sect. 3. Some computational experiments are illustrated in Sect. 4. Finally, Sect. 5 provides concluding remarks of this paper.

2 Problem Description and Formulation

In former research, the YTS-SAP model was introduced in detail in [9]. However, the travel speeds of various yard trucks are assumed to be 11.1 m/s, which are not in accordance with reality. In practice, the travel speed of the empty yard trucks is faster than that of the loaded yard trucks, obviously. Based on previous studies, we take the speed-difference factor into consideration. The notations and mathematical model are given as follows. The object is to schedule the yard trucks in order to minimize the summation of the travel time of the yard trucks as modelled in Eq. (2.1). The model is shown as follows.

$$Minimize: Z = \alpha_1 \sum_{i \in J} d_i + \alpha_2 (\sum_{i \in J} t_i + \sum_{i,j \in J} s_{ij} y_{ij}) \tag{2.1}$$

$$\sum_{i \in J^-} x_{ik} = 1 \quad \forall k \in K \tag{2.2}$$

$$\sum_{k \in K} x_{ik} = 1 \quad \forall i \in J^- \tag{2.3}$$

$$\sum_{i \in J''} y_{ij} = 1 \quad \forall i \in J' \tag{2.4}$$

$$\sum_{i \in J'} y_{ij} = 1 \quad \forall j \in J'' \tag{2.5}$$

$$w_i \geq a_i \quad \forall i \in J' \cup J'' \tag{2.6}$$

$$d_i \geq w_i + t_i - b_i \quad \forall i \in J' \cup J'' \tag{2.7}$$

$$w_j + M(1 - y_{ij}) \geq w_i + t_i + S_{ij} \quad \forall i \in J' \quad and \quad \forall j \in J'' \tag{2.8}$$

$$t_i = \tau_{o_i,e_i} \qquad \forall \in J^+ \tag{2.9}$$

$$t_i = \sum_{k \in K} \tau_{o_i,\varsigma_k} x_{ik} \qquad \forall i \in J^- \tag{2.10}$$

$$s_{ij} = \tau_{e_i,o_j} \quad \forall i \in J^+ \quad and \quad j \in J \tag{2.11}$$

$$s_{ij} = \sum_{k \in K} \tau_{o_i,\varsigma_i} x_{ik} \quad \forall i \in J^- \ and \ \forall j \in J \tag{2.12}$$

$$\tau_{o_i,e_i} = dis_{o_i,e_i} \big/ V_{load} \qquad \forall i \in J^+ \tag{2.13}$$

$$\tau_{o_i,\varsigma_k} = dis_{o_i,\varsigma_i} \big/ V_{load} \quad \forall i \in J^- \quad and \ \ k \in SL \tag{2.14}$$

$$\tau_{e_i,o_j} = dis_{e_i,o_j} / V_{empty} \quad \forall i \in J^+ \quad and \ j \in J \tag{2.15}$$

$$\tau_{\varsigma_k,o_i} = dis_{\varsigma_k,o_i} / V_{empty} \quad \forall j \in J \quad and \ \ k \in SL \tag{2.16}$$

$$x_{ik}, y_{ij} \in \{0,1\}, \forall i \in J', \forall j \in J'' \quad and \ \ \forall k \in K \tag{2.17}$$

$$w_i \in R \ \ \forall i \in J' \cup J'' \tag{2.18}$$

$$t_i \in R \ \forall i \in J \tag{2.19}$$

$$S_{ij} \in R \ \ \forall i \in J \ \ and \ \forall j \in J \tag{2.20}$$

$$d_i \geq 0 \ \ \forall i \in J' \cup J'' \tag{2.21}$$

The movement of a container from its origin to destination is defined as a job, denoted by i and j. Two types of requests are considered in this paper, loading requests and discharging requests. Let J^+ and J^- donate the set of loading requests and the set of discharging requests, respectively. V_{load} and V_{empty} donates the travel speed of loaded trucks and empty trucks, respectively. A soft time window $[a_i, b_i)$ for each job is given as a constant.

Constraints (2.2) insure that each unloading container can be located in a storage location. Constraints (2.3) mean each storage location can save one and only one unloading container. Constraints (2.4) and (2.5) guarantee each route is a one-to-one assignment. Constraints (2.6) imply that the requests can only be performed after the earliest given time. Constraints (2.7) compute the delay time of requests. Constraints (2.8) limit the relationship of the start time of two neighboring requests in the same route. Constraints (2.9) and (2.10) represent the processing time of job i. Constraints (2.11) and (2.12) obtain the setup time from the destination of job i to the origin of job j. Constraints (2.13)–(2.16) limit the speed of the trucks based on different load. Constraints (2.17)–(2.21) define the range for w_i, t_i, s_{ij} and d_i.

3 Artificial Bee Colony Optimization

In ABC algorithm, three types of bees are included, which are employed bees, onlookers and scouts. The process of looking for food sources in ABC is equal to the process of finding the optimum solution. Specifically, the activities for each role of bees are demonstrated as follows.

- Employed bees
  This type of bees go to the food source found by themselves previously. The mount of employed bees and food sources are often set to be equal.
- Onlooker bees
  Onlookers are placed on the food sources by using a probability based selection process. With the growing amount of the nectar, the probability value for the food source is preferred by onlooker's increases.

- Scout bees
  Scouts, which are mainly concerned with finding any kind of food source, are used to doing random search. The average number of scouts is 5 %–20 % of bees.

The Pseudo-code of the ABC algorithm is presented in Fig. 2.

```
ABC Algorithm
Initialize operation;
WHILE ((Iter < MaxCycle))
//Stage 1: Employed Bees
FOR(i = 1:(FoodNumber))
  Form a new food source;
  Calculate the fitness of the new food source;
  Greedy selection;
  END FOR
  Calculate the probability p;
//Stage 2:Onlooker Bees
FOR(i = 1:(FoodNumber))
  Parameter P is set randomly;
  Onlooker bees find food sources depending on P;
  Form a new food source;
  Evaluate the fitness of the new food  source;
  Greedy selection;
END FOR
//Stage 3:Scout Bees
IF(any employed bee turns to scout bee)
  Parameter p is set randomly;
  The scout bees find food sources depending on p
  ;
END IF
Record the best solution;
Iter = Iter + 1;
END WHILE
```

Fig. 2. Pseudo-code of the ABC algorithm

ABC was initially used to solve continuous problems, while the YTS-SAP problem is a typical discrete combinatorial optimization problem. It is necessary to make modifications according to YTS-SAP problem by encoding and decoding of the individuals. To transfer continuous variables to discrete scheduling ones effectively, the random key representation and the smallest position value (SPV) rule are employed in this study. The application of ABC to YTS-SAP is given in the following section.

4 Computational Experiment

In this part, we examine the problem proposed in Sect. 2 by illustrating a set of computational experiments. In the computation test, we assume that 20 containers should be operated. The travel speed of empty yard trucks is assumed to be 11.1 m/s and the weight parameters α1 and α2 are 0.6, 0.4, respectively [9]. But the travel speed of loaded yard trucks is assumed to be 5.5 m/s. Each instance is run for 500 iterations and 20 replications for each instance to collect the results for statistical analysis.

A set of computational experiments are conducted to examine the problems we proposed. To show the optimization performance and effectiveness of YTS-SAP problem, three representative instances are selected, as shown in Table 1. We can see that the solution is feasible, given arbitrary subset of jobs, there is always a feasible assignment of jobs to yard trucks. It is obvious that the routes of trucks are one-to-one assignment.

Table 1. The scheduling result of three test instances

Case	Job	Truck	Scheduling	Value
1	10	1	Route: 6 → 9 → 10 → 3 → 7 → 2 → 8 → 5 → 1 → 4	368.2
2	8	2	Route1: 6 → 3 → 5 Route2: 7 → 2 → 8 → 1 → 4	243.5
3	7	3	Route1: 1 → 4 Route2: 6 → 3 → 5 Route3: 7 → 2	224.1

To verify the optimization performance of ABC algorithm on YTS-SAP problem, five representative instances with different scales are selected as test problems and the PSO, GA algorithms are chosen as comparable algorithms, as shown in Table 2. The amount of job and truck are denoted by n, m, respectively. From the mean values of the objectives, we can observe that the ABC algorithm performs better than the PSO and GA algorithms. This is not meant to be exhaustive, but rather indicative of the different scale instances related to the problem analyzed in the current paper.

We take the instance (n = 25, m = 7) as an example. The convergence progress of the mean fitness values is shown in Fig. 3. It is obvious that the ABC algorithm performs better than PSO and GA algorithms.

With the same amount of travel jobs and the same kind of discharging and loading requests, Table 3 shows the change of the objective values, according to the number of trucks. In term of the objective values, they are changed from 379.3 (m = 1) to 308.7 (m = 2), with the reduction of 18.6 %. Similarly, they are changed from 308.7 (m = 2) to 291.2 (m = 3), with the reduction of 5.7 %.

It is consistent with our experience that increasing the number of trucks can reduce the cost of the total time. But with the increasing of the number of trucks, the reduction of the objective becomes more and more slow. It may be because of the delay of setup time caused by truck congestion.

Table 2. Performance of three algorithms on five test instances

Case	(n, m)	Algorithm	Mean
1	(10, 3)	ABC	291
		PSO	314
		GA	335
2	(15, 5)	ABC	424
		PSO	442
		GA	471
3	(25, 7)	ABC	1834
		PSO	1839
		GA	1842
4	(30, 9)	ABC	1967
		PSO	2197
		GA	2201
5	(35, 10)	ABC	2055
		PSO	2364
		GA	2016

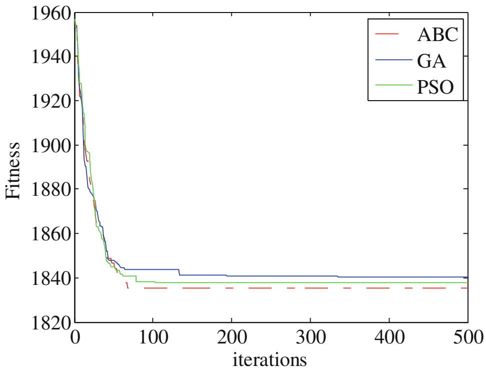

Fig. 3. Objective values at iterations. (Color figure online)

Figure 4 shows the objective values based on the different number of routes. The total time is closely related to the number of yard trucks distributed to work at the specified jobs. Namely, the cost time greatly depends on the ratio of the number of containers to the number of yard trucks. In this experiment, the reduction of the cost time will become less if the number of yard trucks is more than 3.

Table 3. The objective with different number of trucks

Case	Discharging	Loading	Job	Truck	Value
1	5	5	10	1	379.3
2	5	5	10	2	308.7
3	5	5	10	3	291.2
4	5	5	10	4	278.7
5	5	5	10	5	268.3
6	5	5	10	6	261.7
7	5	5	10	7	255.2
8	5	5	10	8	251.2
9	5	5	10	9	248.3
10	5	5	10	10	247

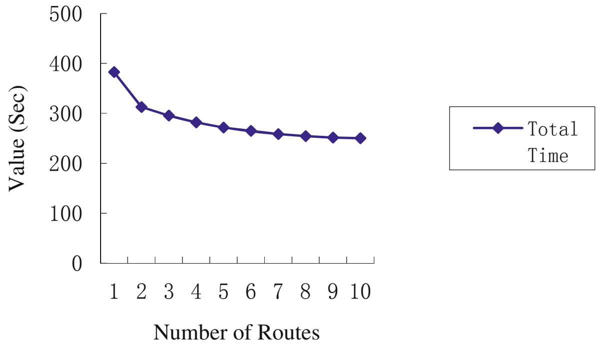

Fig. 4. Effect of number of routes

5 Conclusions and Future Work

Yard truck scheduling and storage allocation problem are two intractable issues in container terminal operation. This paper analyzes the yard truck scheduling and storage allocation problem as a whole. The objective is to minimize the total time cost of the summation of the delay of requests and the travel time of yard trucks.

Aiming to improve the efficiency of the container terminal operations, artificial bee colony algorithm is employed to address YTS-SAP due to its success in many other engineering problems.

To verify the optimization performance of ABC algorithm, we compare it with PSO and GA algorithms. Experimental results on five different scale instances show that ABC algorithm performs better than PSO and GA algorithms. Also we focus on how the number of trucks impacts the global objective. In future, other efficient algorithms need to probe into solving YTS-SAP problem by taking some other objectives into consideration, such as the uncertain factors, the cost of trucks and so on.

Acknowledgments. This work is partially supported by the National Natural Science Foundation of China (Grants nos. 71571120, 71271140, 71461027, 71471158 and 71501132), the Natural Science Foundation of Guangdong Province (Grant nos. 1614050000376, 2016A030310067, 2016A030310074 and 2015A030313556) and the Innovation and Development Fund Project of Shenzhen University (Grant no. 16XSCX04).

References

1. The Chinese Industry Information Network. http://www.chyxx.com/ (2015)
2. Stahlbock, R., VoB, S.: Operations research at container terminals: a literature update. OR Spectr. **30**, 1–52 (2008)
3. Zhu, M., Fan, X., Cheng, H., He, Q.: Modeling and simulation of automated container terminal operation. J. Comput. **5**(6), 951–957 (2010)
4. Kim, K.H., Lee, H.: Container terminal operation: current trends and future challenge. In: Lee, C.-Y., Meng, Q. (eds.) Handbook of Ocean Container Transport Logistics. International Series in Operations Research & Management Science, vol. 220, pp. 43–73. Springer, Switzerland (2015)
5. Moorthy, R., Teo, C.P.: Berth management in container terminal: the template design problem. Oper. Research-Spektrum **28**(28), 495–518 (2006)
6. Nishimura, E., Imai, A., Papadimitriou, S.: Yard trailer routing at a maritime container terminal. Transp. Res. Part E Logist. Transp. Rev. **41**(1), 53–76 (2005)
7. Ng, W.C., Mak, K.L., Zhang, Y.X.: Scheduling trucks in container terminals using a genetic algorithm. Eng. Optim. **39**(1), 33–47 (2007)
8. Kim, K.H., Kim, H.B.: Segregating space allocation models for container inventories in port container terminals. Int. J. Prod. Econ. **59**, 415–423 (1999)
9. Lee, D.H., Cao, J.X., Shi, Q.X., Chen, J.H.: A heuristic algorithm for yard truck scheduling and storage allocation problems. Transp. Res. Part E Logist. Transp. Rev. **45**(5), 810–820 (2009)
10. Bish, E.K., Leong, T.Y., Li, C.L., Ng, J.W.C., Simchi-Levi, D.: Analysis of a new vehicle scheduling and location problem. Naval Res. Logist. **48**, 363–385 (2001)
11. Lee, D.H., Cao, J.X., Shi, Q.: Integrated model for truck scheduling and storage allocation problem at container terminals. In: The Proceedings of the TRB Meeting, pp. 211–216 (2008)
12. Niu, B., Xie, T., Bi, Y., Liu, J.: Bacterial colony optimization for integrated yard truck scheduling and storage allocation problem. In: Huang, D.-S., Han, K., Gromiha, M. (eds.) ICIC 2014. L:NCS, vol. 8590, pp. 431–437. Springer, Switzerland (2014)
13. Niu, B., Xie, T., Duan, Q.Q., Tan, L.J.: Particle swarm optimization for integrated yard truck scheduling and storage allocation problem. In: IEEE Congress on Evolutionary Computation, pp. 634–639 (2014)
14. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)
15. Zhou, G., Wang, L., Xu, Y., Wang, S.: An effective artificial bee colony algorithm for multi-objective flexible job-shop scheduling problem. In: Huang, D.-S., Gan, Y., Gupta, P., Michael Gromiha, M. (eds.) ICIC 2011. LNCS, vol. 6839, pp. 1–8. Springer, Heidelberg (2012)
16. Horng, M.H.: Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. Expert Syst. Appl. **38**(11), 13785–13791 (2011)

17. Xu, C., Duan, H., Liu, F.: Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning. Aerosp. Sci. Technol. **14**(8), 535–541 (2010)
18. Bullinaria, J.A., AlYahya, K.: Artificial bee colony training of neural networks. In: Terrazas, G., Otero, F.E.B., Masegosa, A.D. (eds.) Nature Inspired Cooperative Strategies for Optimization (NICSO 2013). Studies in Computational Intelligence, vol. 512, pp. 191–201. Springer, Switzerland (2014)
19. Li, L., Cheng, Y., Tan, L.J., Niu, B.: A Discrete Artificial Bee Colony Algorithm for TSP Problem, pp. 566–573. Springer, Heidelberg (2012)