

多式联运集装箱港口堆存箱位分配研究

唐璐瑶¹, 李俊^{1,2*}, 张凯霞¹, 严静雨¹

¹武汉科技大学汽车与交通工程学院, 湖北 武汉

²天津港集团有限公司, 天津

收稿日期: 2024年2月20日; 录用日期: 2024年3月11日; 发布日期: 2024年4月23日

摘要

集装箱多式联运港口作为联运网络的核心节点, 其作业效率与整个联运网络的通过效率直接相关, 而堆场作业效率对港口整体效率保证尤为重要。港口实际作业中, 基于船舶配载计划对堆场待装船集装箱进行合理的箱位分配, 可以有效提高堆场作业效率。在集装箱多式联运背景下, 考虑船舶配载计划, 以公路、铁路、水路集港的待装船集装箱为对象, 结合场桥作业研究堆场集装箱的箱位分配问题。为保证堆场整体作业效率, 建立集装箱堆场翻箱量最小和场桥作业时间最短的两阶段数学模型, 并改进人工蜂群算法进行求解。为避免两阶段求解对全局最优影响, 通过信息素与灵敏度协同来平衡两阶段最优解得到最优方案。算例研究表明: 改进人工蜂群算法求解得到的堆场翻箱量和场桥作业时间均优于人工蜂群算法和遗传算法; 随着算例规模的增大, 改进人工蜂群算法的优势更加凸显, 求解质量和效率均表现最好。以上结果论证了所构建模型与算法的有效性, 对于港口实际制定堆场箱位分配计划具有一定的参考意义。

关键词

集装箱多式联运, 箱位分配, 场桥调度, 两阶段模型, 改进人工蜂群算法

Study on Storage Space Allocation in Multimodal Container Ports

Luyao Tang¹, Jun Li^{1,2*}, Kaixia Zhang¹, Jingyu Yan¹

¹School of Automotive and Traffic Engineering, Wuhan University of Science and Technology, Wuhan Hubei

²Tianjin Port (Group) Co., Ltd., Tianjin

Received: Feb. 20th, 2024; accepted: Mar. 11th, 2024; published: Apr. 23rd, 2024

Abstract

As the core node of the intermodal container transport network, the operation efficiency of the

*通讯作者。

文章引用: 唐璐瑶, 李俊, 张凯霞, 严静雨. 多式联运集装箱港口堆存箱位分配研究[J]. 运筹与模糊学, 2024, 14(2): 702-713. DOI: 10.12677/orf.2024.142172

container multimodal port is directly related to the passing efficiency of the entire intermodal transport network, and the operation efficiency of the storage yard is particularly important to ensure the overall efficiency of the port. In the actual operation of the port, the reasonable space allocation of containers waiting for loading based on the ship stowage plan can effectively improve the operation efficiency of the storage yard. Under the background of container multimodal transport, the space allocation of container in storage yard is studied by considering the ship stowing plan, taking the containers waiting to be loaded in highway, railway and waterway port as the object and combining with yard bridge operation. In order to ensure the overall operation efficiency of the container yard, a two-stage mathematical model of minimum turnover and minimum operation time of the yard bridge was established, and the artificial bee colony algorithm was improved to solve the problem. In order to avoid the influence of two-stage solution on the global optimal, the optimal scheme is obtained by the coordination of pheromone and sensitivity to balance the two-stage optimal solution. The results show that the improved artificial bee colony algorithm is better than the general artificial bee colony algorithm and genetic algorithm in terms of yard turnover and bridge operation time. With the increase of the scale of the example, the advantages of the improved artificial bee colony algorithm are more prominent, and the solution quality and efficiency are the best. The results above demonstrate the effectiveness of the model and algorithm, and have a certain reference significance for the actual development of storage yard space allocation plan.

Keywords

Container Multimodal Transport, Slot Allocation, Field Bridge Scheduling, Two-Stage Model, Improved Artificial Bee Colony Algorithm

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

集装箱运输的高效、便捷和安全特性,使其在现代物流中占据重要地位。随着国家对集装箱运输行业的政策支持逐渐从基础设施的完善转向推动多式联运的发展,集装箱多式联运因融合了各种运输方式的优点而备受瞩目。交通运输部已经出台了一系列政策以推动和支持多式联运的发展,使得集装箱多式联运迎来了快速发展的新时期。为了提高多式联运网络的整体运输效率,铁路、公路和水路之间的协调与合作变得尤为重要。集装箱多式联运港口作为联运网络的核心,其作业效率直接影响整个网络的效率。而堆场作业效率又是港口效率的关键因素。因此,如何降低堆场中的无效翻箱作业、确保场桥等关键设备的高效运作,成为了行业内广泛关注和研究的重点。

由于集装箱堆场箱位分配的重要性,目前已有许多学者针对传统集装箱港口场景下的问题开展研究。Kim [1]等在考虑箱重的基础上,以最小化翻箱量为目标建立模型,并用动态规划算法求解。李燕晶[2]等遵循同一艘船的集装箱集中堆存的原则,以最小化压箱量为目标建立箱位分配模型。王曦[3]等在不同重量等级混堆的情况下,考虑翻箱量和运输距离,用改进的萤火虫算法求解模型。Gharehgozli [4]等为了提高堆栈利用率,提出一种不同类型集装箱混堆的堆存策略,以最小化翻箱时间为目标,并利用叠加启发式算法求解。常祎妹[5]等从到达时间切入,对公路集装箱的到达顺序进行预测,并以压箱量最小对铁路和公路集装箱进行箱位分配,利用启发式算法求解。计明军[6]等在同卸同装的前提下,以最小化压箱量为目标,提出了一种新的箱位分配策略,并与最矮分配策略进行对比。

同时, 也有学者针对堆场场桥作业调度开展了研究。任刚[7]等在海铁联运背景下, 考虑班列入场顺序和离港时间, 以压箱量最小和场桥移动贝位数最小为目标, 并设计启发式算法求解。范厚明等在考虑场桥安全距离和场地分区域平衡的前提下, 以场桥均衡作业量和场桥非装卸时间最小[8]为目标, 建立混合整数规划模型求解。随后以作业量均衡为前提, 以最小化场桥移动成本和空闲成本[9]为目标建立模型求解。之后为研究送箱集卡到达不确定对出口箱堆存效率的影响[10], 将场桥调度与预翻箱结合, 建立双层混合整数规划模型进行求解。郑云峰[11]等在考虑多种因素之后, 以翻箱成本和场桥作业成本最小为目标, 建立模型求解并通过仿真实验验证。

综上所述, 当前研究主要聚焦于公路集装箱港口作业, 探讨其在港口堆场中的箱位分配问题。部分研究还将箱位分配与设备作业调度相结合进行深入探讨。然而, 针对多式联运集装箱港口场景的研究相对较少, 鲜有学者关注并开展集装箱堆存箱位分配与场桥设备联合优化问题的研究。因此, 本文立足于集装箱多式联运港口堆场作业的实际需求, 以船舶配载计划为基础, 结合场桥作业, 探讨多路径集港集装箱的堆存箱位分配问题。为解决此问题, 本文运用数学规划和双层建模方法构建了旨在最小化场桥作业时间和翻箱量的两阶段优化模型, 并采用改进的人工蜂群算法进行求解。

2. 问题描述

多式联运是由两种及以上的交通工具相互衔接、转运、共同完成运输任务的复杂运输, 典型的集装箱多式联运港口的作业流程图见图 1。对于集装箱多式联运港口的出口集装箱业务, 集装箱的来源多种

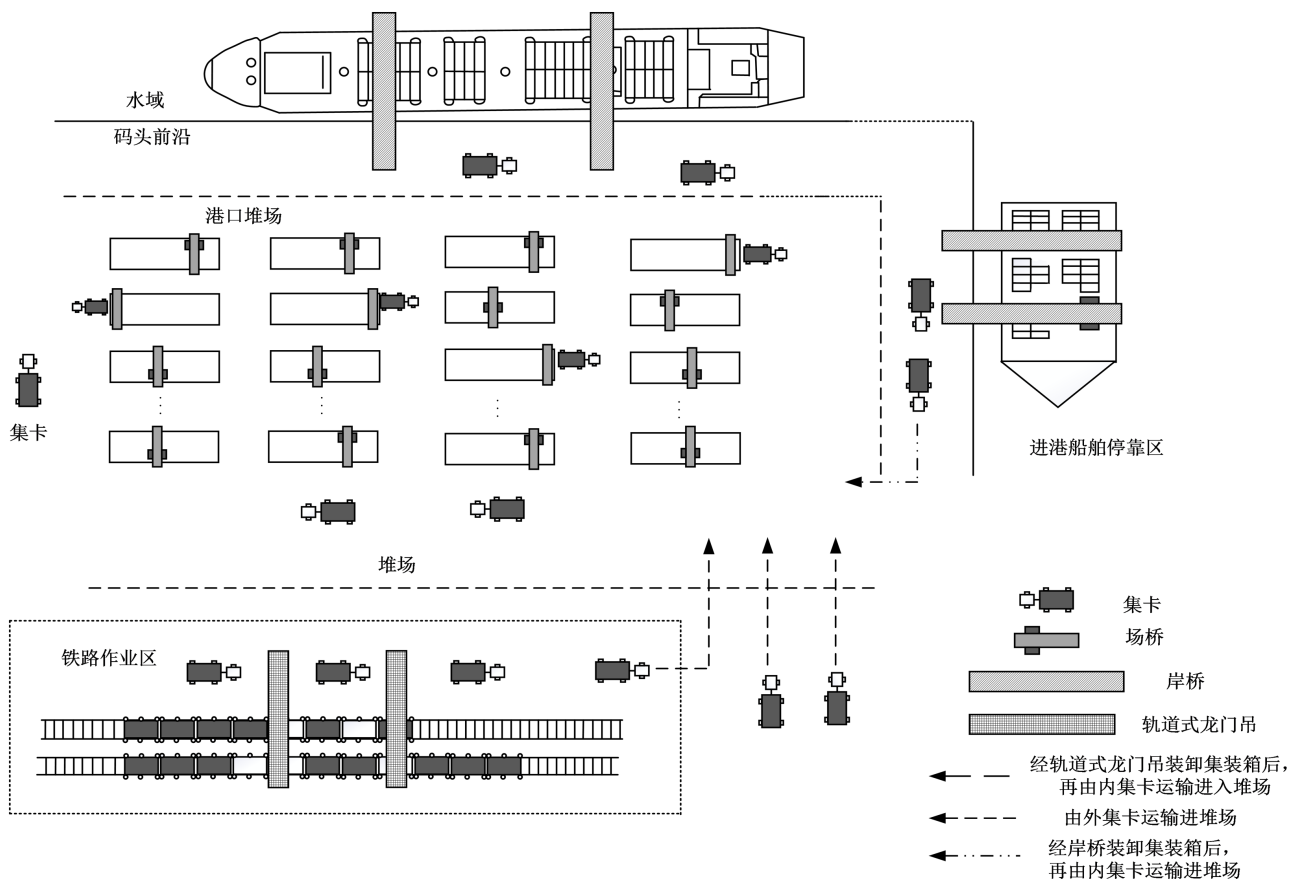


Figure 1. Schematic diagram of container multimodal port collection operation

图 1. 集装箱多式联运港口集港作业运作示意

多样, 公路集装箱由外集卡运输进堆场; 铁路集装箱由轨道式龙门吊卸下后, 由内集卡运输至堆场; 水路集装箱由岸桥卸下后, 由内集卡运输至堆场。无论是来自公路、铁路还是水路的集装箱, 在出口过程中都需要进入堆场进行箱位分配, 并且都需通过场桥作业来最终确定其在堆场中的存储位置。相对于传统集装箱港口, 布局更加复杂, 整体作业需要更多的协调性。

因此, 本文紧密结合集装箱多式联运港口的实际作业情况, 以船舶配载计划为基础, 针对通过待装船的公路、铁路和水路集装箱, 结合场桥作业计划, 深入探讨多式联运港口堆场的箱位分配问题。为了更有效地分解和简化问题, 本文参考了港口作业流程以及已有的研究成果[12], 将研究内容划分为两个阶段。在第一阶段, 主要目标是优化集港集装箱在堆存过程中的场桥作业计划, 以最小化场桥作业时间; 而在第二阶段, 则聚焦于翻箱量的优化。这样的研究策略有助于我们更全面、系统地理解和解决多式联运集装箱港口的箱位分配问题。

3. 模型构建

3.1. 前提与假设

为了更好地解决本文问题, 从实际场景出发, 提出了以下几点假设:

- 1) 集港集装箱的船舶配载计划已知;
- 2) 港口堆场中进口集装箱和出口集装箱分开堆存;
- 3) 铁路、公路和水路集装箱的先后到达顺序已知;
- 4) 集装箱为统一标准尺寸。

3.2. 符号说明

1) 集合

N : 到达港口集港的待装船集装箱总数;

N_A : 到达港口的铁路集装箱的总数;

N_B : 到达港口的公路集装箱的总数;

N_W : 到达港口的水路集装箱的总数;

N_m : m 时段到达港口的集装箱总数;

C : 到达港口所有集装箱的集合, $i, j, v \in C$;

e, f : m 时段内需要堆存的集装箱;

C_A : 铁路集装箱的集合;

C_B : 公路集装箱的集合;

C_W : 水路集装箱的集合;

M : 集装箱装船过程所需要的总时间, 可分为 m 个时间段, $m, m' \in M$;

B : 贝位的集合;

S : 栈的集合;

T : 层的集合。

2) 参数

b, s, t : 堆场内贝位、栈、层的编号;

H_{bs} : 堆场堆存的额定高度, b 为堆存集装箱的贝位, s 为堆存集装箱的栈;

p_i : 集装箱的配载计划;

G ：一个无穷大的数；

T^m ： m 时段场桥作业集装箱所需时间；

t_1 ：场桥移动一个贝位的距离所需的时间；

t_2 ：场桥移动一个栈的距离所需的时间；

t_3 ：场桥作业一个集装箱所需的时间；

t_f^m ： m 个时段场桥作业集装箱 f 所需的时间；

t_{ef}^m ： m 个时段场桥将集装箱从 (b_e^m, s_e^m) 移动到 (b_f^m, s_f^m) 所需的时间。

3) 变量

η_{ibstm} ：为 0~1 变量，在 m 时段，当集装箱对下层集装箱造成压箱，产生翻箱量时为 1，否则为 0；

φ_{ibstm} ：为 0~1 变量，在 m 时段，如果集装箱在箱位 (b, s, t) 上时为 1，否则为 0；

α_{im} ：为 0~1 变量，在 m 时段，如果集装箱到达港口为 1，否则为 0；

β_{im} ：为 0~1 变量，在 m 时段，如果集装箱通过铁路运达港口则为 1，否则为 0；

θ_{im} ：在 m 时段，如果集装箱通过公路运达港口则为 1，否则为 0；

δ_{im} ：在 m 时段，如果集装箱通过水路运达港口则为 1，否则为 0；

$\varphi_{ijbst'm}$ ：为 0~1 变量，在 m 时段，如果集装箱 i 在箱位 (b, s, t) 上，集装箱 j 在 (b, s, t) 上则为 1，否则为 0；

D_{ef}^m ：为 0~1 变量，如果场桥将集装箱从 e 移动到 f 则为 1，否则为 0。

3.3. 模型建立

结合多式联运集装箱港口的实际作业流程，将出口集装箱的堆场作业过程分为两个阶段：第一阶段集装箱集港并进入堆场堆存，对场桥完成集港集装箱的堆存作业时间进行优化；第二阶段截港后装船作业，对场桥完成集装箱装船作业产生的翻箱量进行优化。

结合问题描述，本文构建的第一阶段模型以集港集装箱堆存时的场桥作业时间最短为目标[13]，其中场桥作业时间为场桥在不同贝位、栈之间水平移动所需时间以及在不同层垂直移动所需时间之和。

$$Z_1 = \min T^m \tag{1}$$

$$\sum_{f=1}^{N_m} D_{ef}^m - \sum_{f=1}^{N_m} D_{fe}^m = \begin{cases} 1, & e = 0, 1 \leq m \leq M \\ -1, & e = N_m + 1, 1 \leq m \leq M \\ 0, & e \neq 0, 1 \leq m \leq M \end{cases} \tag{2}$$

$$t_{ef}^m = \begin{cases} \max = \{t_1 \times |b_f^m - b_0|, t_2 \times |s_f^m - s_0|\}, & 1 \leq m \leq M, 1 \leq f \leq N_m \\ \max = \{t_1 \times |b_e^m - b_f^m|, t_2 \times |s_e^m - s_f^m|\}, & 1 \leq m \leq M, 1 \leq e, f \leq N_m \\ \max = \{t_1 \times |b_e^m - b_0|, t_2 \times |s_e^m - s_0|\}, & 1 \leq m \leq M, 1 \leq e \leq N_m, f = N_m + 1 \end{cases} \tag{3}$$

$$t_f^m = \begin{cases} t_3 + \sum_{b=1}^{N_m} t_{ef}^m, & 1 \leq m \leq M \\ G \\ t_{ef}^m, & 1 \leq m \leq M, 1 \leq e \leq N_m, f = N_m + 1 \end{cases} \tag{4}$$

$$T^m = \sum_{f=1}^{N_m} t_f^m, 1 \leq m \leq M \tag{5}$$

式(1)为目标函数,表示 m 时段集港集装箱所需的场桥作业时间。式(2)表示场桥将集装箱从 e 搬运到 f 时,场桥移动路线的起点和中点只有一个。式(3)表示场桥对出口集装箱 f 进行水平移动所需要的时间。式(4)表示 m 时段场桥作业集装箱 f 所需的时间。式(5)表示 m 时段场桥作业待堆存集装箱所需的时间。

第二阶段的模型基于已知的船舶配载计划和堆场作业规则,对提箱过程中的堆场翻箱量进行优化,以翻箱量[14]最小为目标,在提升堆场作业效率的同时,缩短装船作业时间。

$$Z_2 = \min \sum_{m=1}^{|M|} \left(\sum_{i=1}^{N_A} N_{Aim} + \sum_{i=1}^{N_B} N_{Bjm} + \sum_{i=1}^{N_W} N_{Wvm} \right) \quad (6)$$

$$N_{Aim} = \sum_{bst} \sum_{e=1}^{t-1} \eta_{ibst(t-e)m} \varphi_{ibstm} \beta_{im} \alpha_{im} \quad (7)$$

$$N_{Bjm} = \sum_{bst} \sum_{e=1}^{t-1} \eta_{ibst(t-e)m} \varphi_{ibstm} \theta_{im} \alpha_{im} \quad (8)$$

$$N_{Wvm} = \sum_{bst} \sum_{e=1}^{t-1} \eta_{ibst(t-e)m} \varphi_{ibstm} \delta_{im} \alpha_{im} \quad (9)$$

$$\sum_i \varphi_{ibstm} \leq 1, \forall i \in C, \forall b \in B, \forall s \in S, \forall t \in T, \forall m \in M \quad (10)$$

$$\sum_{bst} \varphi_{ibstm} = 1, \forall i \in C, \forall b \in B, \forall s \in S, \forall t \in T, \forall m \in M \quad (11)$$

$$\sum_i \varphi_{ibst(t+1)m} \leq \sum_i \varphi_{ibstm}, \forall i \in C, \forall b \in B, \forall s \in S, \forall t, t+1 \in T, \forall m \in M \quad (12)$$

$$\varphi_{ibstm} = \varphi_{ibst(m-1)} + 1, \forall i \in C, \forall b \in B, \forall s \in S, \forall t \in T, \forall m, m-1 \in M \quad (13)$$

$$\sum_{it} \varphi_{ibstm} \leq H_{bs}, \forall i \in C, \forall b \in B, \forall s \in S, \forall t \in T, \forall m \in M \quad (14)$$

$$\sum_{ibstm} \varphi_{ibstm} \theta_{im} \alpha_{im} + \sum_{i'b's't'm} \varphi_{i'b's't'm} \beta_{i'm} \alpha_{i'm} + \sum_{i''b''s''t''m} \varphi_{i''b''s''t''m} \delta_{i''m} \alpha_{i''m} = N_A + N_B + N_W \quad (15)$$

$$\forall i, i', i'' \in C, \forall b, b', b'' \in B, \forall s, s', s'' \in S, \forall t, t', t'' \in T, \forall m \in M$$

$$N_B = \sum_{ibstm} \varphi_{ibstm} \theta_{im} \alpha_{im}, \forall i \in C, \forall b \in B, \forall s \in S, \forall t \in T, \forall m \in M \quad (16)$$

$$N_A = \sum_{i'b's't'm} \varphi_{i'b's't'm} \beta_{i'm} \alpha_{i'm}, \forall i' \in C, \forall b' \in B, \forall s' \in S, \forall t' \in T, \forall m \in M \quad (17)$$

$$N_W = \sum_{i''b''s''t''m} \varphi_{i''b''s''t''m} \delta_{i''m} \alpha_{i''m}, \forall i'' \in C, \forall b'' \in B, \forall s'' \in S, \forall t'' \in T, \forall m \in M \quad (18)$$

$$p_i \varphi_{ibstm} - p_j \varphi_{ibst'm} + G(1 - \varphi_{ijbst'm}) \leq 0, \forall i, j \in C, \forall b \in B, \forall s \in S, \forall t, t' \in T, \forall m \in M \quad (19)$$

$$\varphi_{ibstm} \alpha_{im} > \varphi_{jbs(t-1)m'} \alpha_{jm'}, m > m', \forall i, j \in C, \forall b \in B, \forall s \in S, \forall t, t-1 \in T, \forall m, m' \in M \quad (20)$$

式(6)表示在堆存过程中铁路集装箱、公路集装箱和水路集装箱的总翻箱量最小;式(7)表示在 m 时段铁路集装箱产生的翻箱量;式(8)表示在 m 时段公路集装箱产生的翻箱量;式(9)表示在 m 时段水路集装箱产生的翻箱量;式(10)表示堆场中一个箱位最多可以存放一个集装箱;式(11)表示堆场中一个集装箱只能占用一个箱位;式(12)表示堆场中的集装箱必须从下往上依次堆存;式(13)表示集装箱在被放进箱位时要确保箱位空缺;式(14)表示堆场堆存的集装箱的高度不能超过额定高度;式(15)表示堆场堆存的集装箱数等于铁路、公路和水路到达的总集装箱数;式(16)~(18)表示公路、铁路、水路集港所有集装箱均按照配载计划完成装船;式(19)中 G 是很大的一个整数,表示装船顺序在后的集装箱要堆在装船顺序在前的集装箱下面;式(20)表示堆存在下面的集装箱的到达时间早于其上方的集装箱的到达时间。

4. 算法设计

4.1. 人工蜂群算法

人工蜂群算法(Artificial Bee Colony, ABC)是模仿蜜蜂的群体觅食行为提出的一种优化方法, 由雇佣蜂、观察蜂和侦察蜂三个基本部分组成。雇佣蜂对食物源进行发现, 并将发现的食物源信息与观察蜂共享, 观察蜂以较大的概率选择优秀食物源, 采用与雇佣蜂同样的食物源发现方式进行食物源搜索, 侦察蜂对旧的食物源丢弃并产生一个新的食物源。在蜂群进化过程中, 雇佣蜂和观察蜂的数量与食物源的数量相同, 并始终保持不变, 侦察蜂的数量只有一只, 通过 3 种蜜蜂的协同搜索, 从而发现搜索空间内的优秀食物源, 主要公式如下:

1) 蜜源初始化

初始化时, 随机生成 S_N 个可行解(等于雇佣蜂的数量)并计算适应度函数值。随机产生可行解的公式如下:

$$x_{ij} = x_{\min,j} + rand(0,1)(x_{\max,j} - x_{\min,j}) \quad (21)$$

式中 $x_i (i=1,2,\dots,S_N)$ 为 D 维向量, D 为优化参数的个数, $j \in \{1,2,\dots,D\}$ 。

2) 新蜜源的更新搜索公式

蜜蜂记录自己到目前为止的最优值, 并在当前蜜源领域内展开搜索, 基本 ABC 在蜜源附近搜索新蜜源的公式为:

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (22)$$

式中 $j \in \{1,2,\dots,D\}$, $k \in \{1,2,\dots,S_N\}$, k 为随机生成且 $k \neq i$, φ_{ik} 为 $[-1, 1]$ 之间的随机数。

3) 观察蜂选择雇佣蜂的概率

$$P_i = \frac{fit(x_i)}{\sum_{n=1}^{S_N} fit(x_n)} \quad (23)$$

式中 $fit(x_i)$ 为第 i 个解的适应值对应蜜源的丰富程度。蜜源越丰富, 被观察蜂选择的概率越大。

4) 侦察蜂的产生

如果蜜源经过 $trial$ 次迭代达到搜索阈值 L 后, 还没有更新为更好的蜜源, 该蜜源会被放弃, 同时该蜜源对应的雇佣蜂转变为侦察蜂按式(27)随机产生一个新的位置代替原蜜源。

$$x_i = \begin{cases} x_{\min,j} + rand(0,1)(x_{\max,j} - x_{\min,j}), & trial \geq L \\ x_i, & trial < L \end{cases} \quad (24)$$

在以上的算法设计中, 轮盘赌策略导致观察蜂选择适应度高的蜜源, 忽略适应度低的蜜源, 使得物种多样性降低, 也就将可行解的范围缩小。在本文中, 单一的选择方法对于两个目标的优化来说, 会将有可能的解筛选出去, 使算法过早陷入局部最优。

4.2. 算法改进

以上 ABC 算法中, 轮盘赌策略导致观察蜂选择适应度高的蜜源, 缩小了可行解的搜索范围。针对本文的两阶段问题模型, 单一蜜源选择的方法在求解两个阶段问题时, 会选出各自的最优方案, 阶段性的最优可能不是整体的最优, 无法保证两阶段目标的均衡优化。因此, 对于蜜源的选择应该扩大范围, 以实现两阶段问题的均衡求解。

为解决上述问题,提出改进人工蜂群(Improved Artificial Bee Colony, IABC)算法,在自由搜索中引入了灵敏度的概念,通过与蜜源的信息素配合来选择搜索区域,以扩大蜜源的选择范围。通过信息素与灵敏度的协同作用,使算法在双目标均衡优化方面具有一定的导向性。采用这种选择方式,任何适应度的蜜源都有可能被选中,从而增加了种群的多样性,防止算法陷入局部最优解,同时也提高了优秀蜜源被选中的概率,保证了观察蜂选择蜜源的方向。

具体改进方法为:计算蜜源的最大适应度值和最小适应度值,从而计算出第 i 个蜜源的信息素 $O(i)$ 并与随机生成的第 i 个观察蜂的灵敏度 $S(i)$ 进行比较。若 $S(i) \leq O(i)$, 则通过邻域搜索更优的蜜源,若 $S(i) \geq O(i)$, 则蜜源的位置不变。蜜源信息素的计算如式(25)所示:

$$O(i) = \begin{cases} \frac{fit(x_i) - fit(x_{\min})}{fit(x_{\max}) - fit(x_{\min})}, & fit(x_{\max}) \neq fit(x_{\min}) \\ 0, & \text{其他} \end{cases} \quad (25)$$

其中, $O(i)$ 为第 i 个蜜源的信息素; $fit(x_i)$ 为第 i 个蜜源的适应度值; $fit(x_{\max})$ 为最大适应度值; $fit(x_{\min})$ 为最小适应度值。

算法步骤如下:

步骤 1: 初始化参数, 设置人工蜂群算法的最大迭代次数 `max_itr` 和蜜蜂数量 `numb_bees`。

步骤 2: 初始化种群, 按照式(21)产生初始可行解。

步骤 3: 雇佣蜂按照式(22)对蜜源进行领域搜索产生新蜜源, 计算适应度值并依据贪婪法将新旧蜜源进行比较, 选出使得目标函数值最小的蜜源。

步骤 4: 观察蜂通过共享信息, 根据式(25)将两个目标进行选择, 同时产生新的蜜源, 通过对比新旧蜜源保留使得两个目标最优的蜜源。

步骤 5: 若蜜源达到开采限度, 雇佣蜂转化为侦察蜂, 并按照式(24)产生新的蜜源。若蜜源未达到开采限度, 则记录最好的蜜源。

步骤 6: 判断是否满足终止条件, 若不满足, 则返回步骤 3; 若满足, 则输出最优解, 算法结束。

5. 数值实验

5.1. 不同算法对比分析

为了验证本文算法的有效性,以北方某港实际入场集装箱为依据,生成了 19 组不同规模的算例。其中,算例规模“ $2 \times 6 \times 4$ ”表示该算例包含堆场堆存区域包含 2 个贝位,各贝位结构为 6 栈 4 层; N 表示包含的集港集装箱数量。

IABC 和 ABC 算法均采用 Python 3.8 编程实现,参数均设置初始种群为 20,迭代次数为 500。所有算例均在笔记本电脑(Intel(R) Core(TM)i5-1240P 1.70 GHz)上运行求解。为进一步说明 IABC 算法优势,引入箱位分配问题中常用的遗传算法进行对比分析。遗传算法为直接调用 Python 软件 `geatpy` 库中的增强精英保留的遗传算法,其初始种群为 20,迭代次数为 500,与其他算法保持一致。不同算法的求解结果见表 1,表中加粗部分表示不同算法求解当前算例的最优结果。

从表 1 结果可以看出,1) 所有算例求解中 IABC 算法均在更短的时间内找到了更好的结果;2) 对于场桥作业时间而言,算法 IABC、ABC、GA 求解得到的平均值依次为 6782 s、8417 s、10,413 s,且随着算例规模的增加,IABC 优势越发明显;3) 对于翻箱量而言,算法 IABC、ABC、GA 求解得到的平均值依次为 24 个、28 个、30 个,不同规模下,IABC 算法仍具有优势;4) 在求解时间方面,算法 IABC、ABC、GA 求解得到的平均值依次为 61 s、72 s、87 s,不同算法之间差异不是很明显,但 IABC 算法总

是耗时更短, 求解效率最高。以上结果表明, 通过信息素与灵敏度协同均衡双目标优化的改进策略, 有效提高了 IABC 算法的求解性能, 实现了问题的高效求解。

Table 1. Algorithm comparison table

表 1. 算法对比表

| 算例 | 场桥作业时间/s | | | 翻箱量/个 | | | 算法消耗时间/s | | | |
|------------|----------|--------|--------|--------|------|-----|----------|------|-----|-----|
| | N | IABC | ABC | GA | IABC | ABC | GA | IABC | ABC | GA |
| 2 × 6 × 4 | 30 | 443 | 477 | 492 | 4 | 5 | 5 | 10 | 15 | 20 |
| 2 × 6 × 4 | 40 | 732 | 1014 | 1072 | 5 | 7 | 8 | 16 | 19 | 25 |
| 3 × 6 × 4 | 50 | 1004 | 1068 | 1364 | 8 | 11 | 12 | 21 | 25 | 42 |
| 3 × 6 × 4 | 60 | 1354 | 1744 | 2221 | 10 | 12 | 13 | 27 | 30 | 35 |
| 4 × 6 × 4 | 70 | 1824 | 2684 | 2803 | 11 | 12 | 14 | 32 | 37 | 53 |
| 4 × 6 × 4 | 80 | 2123 | 2965 | 3396 | 12 | 15 | 15 | 33 | 39 | 46 |
| 5 × 6 × 4 | 90 | 2490 | 3125 | 4109 | 13 | 27 | 19 | 35 | 45 | 58 |
| 5 × 6 × 4 | 100 | 2998 | 4354 | 4801 | 17 | 20 | 21 | 38 | 47 | 63 |
| 6 × 6 × 4 | 110 | 4076 | 5108 | 6010 | 20 | 22 | 24 | 42 | 60 | 75 |
| 7 × 6 × 4 | 130 | 5903 | 8168 | 10,193 | 24 | 27 | 28 | 49 | 74 | 88 |
| 8 × 6 × 4 | 140 | 6134 | 7754 | 9007 | 26 | 27 | 29 | 50 | 73 | 108 |
| 8 × 6 × 4 | 150 | 8377 | 11,748 | 13,484 | 28 | 31 | 33 | 54 | 81 | 99 |
| 9 × 6 × 4 | 160 | 9024 | 13,222 | 14,191 | 30 | 33 | 37 | 67 | 78 | 103 |
| 9 × 6 × 4 | 170 | 9605 | 12,038 | 15,489 | 32 | 35 | 40 | 76 | 87 | 105 |
| 9 × 6 × 4 | 180 | 12,976 | 15,248 | 17,140 | 36 | 37 | 42 | 89 | 99 | 121 |
| 10 × 6 × 4 | 190 | 14,678 | 16,073 | 18,486 | 40 | 42 | 47 | 113 | 121 | 136 |
| 10 × 6 × 4 | 200 | 13,397 | 15,510 | 20,676 | 42 | 47 | 51 | 128 | 138 | 148 |
| 11 × 6 × 4 | 210 | 13,967 | 16,279 | 24,851 | 47 | 56 | 60 | 136 | 148 | 157 |
| 11 × 6 × 4 | 220 | 17,753 | 21,341 | 28,058 | 50 | 62 | 67 | 144 | 153 | 164 |
| 平均值 | | 6782 | 8417 | 10,413 | 24 | 28 | 30 | 61 | 72 | 87 |

为进一步直接展示 IABC 算法优势, 绘制了不同算法的求解结果对比图, 如图 2 所示。从图中结果可以直观看出, 在场桥作业时间方面, IABC 算法求解结果优于 ABC 与 GA 算法结果, 且随着算例规模的增加其优势更加明显, 见图 2(a); 在堆场翻箱量方面, IABC 算法表现最优, 但不同算法求解结果差距较小, 见图 2(b); 在算法消耗时间方面, IABC 算法对比 ABC 与 GA 明显具有更高效率, 见图 2(c)。

5.2. 翻箱策略影响分析

集装箱作为阻碍箱被提起时, 其落箱位置会决定再次翻箱的次数, 因此选择一个合适的落箱位置, 可以减少场桥的作业次数。已有的翻箱策略有最矮栈翻箱策略和邻近栈翻箱策略两种[15], 将对这两种策略进行对比分析, 结果见表 2。同样地, 表 2 中加粗部分表示求解当前算例的最优结果。

从表中结果可以看出, 1) 对于场桥作业时间而言, 最矮栈策略和邻近栈策略求得平均值依次为 15,397 s、13,870 s。当作业箱量少于 110 时, 最矮栈策略所需的作业时间较少; 当作业箱量增加至 110

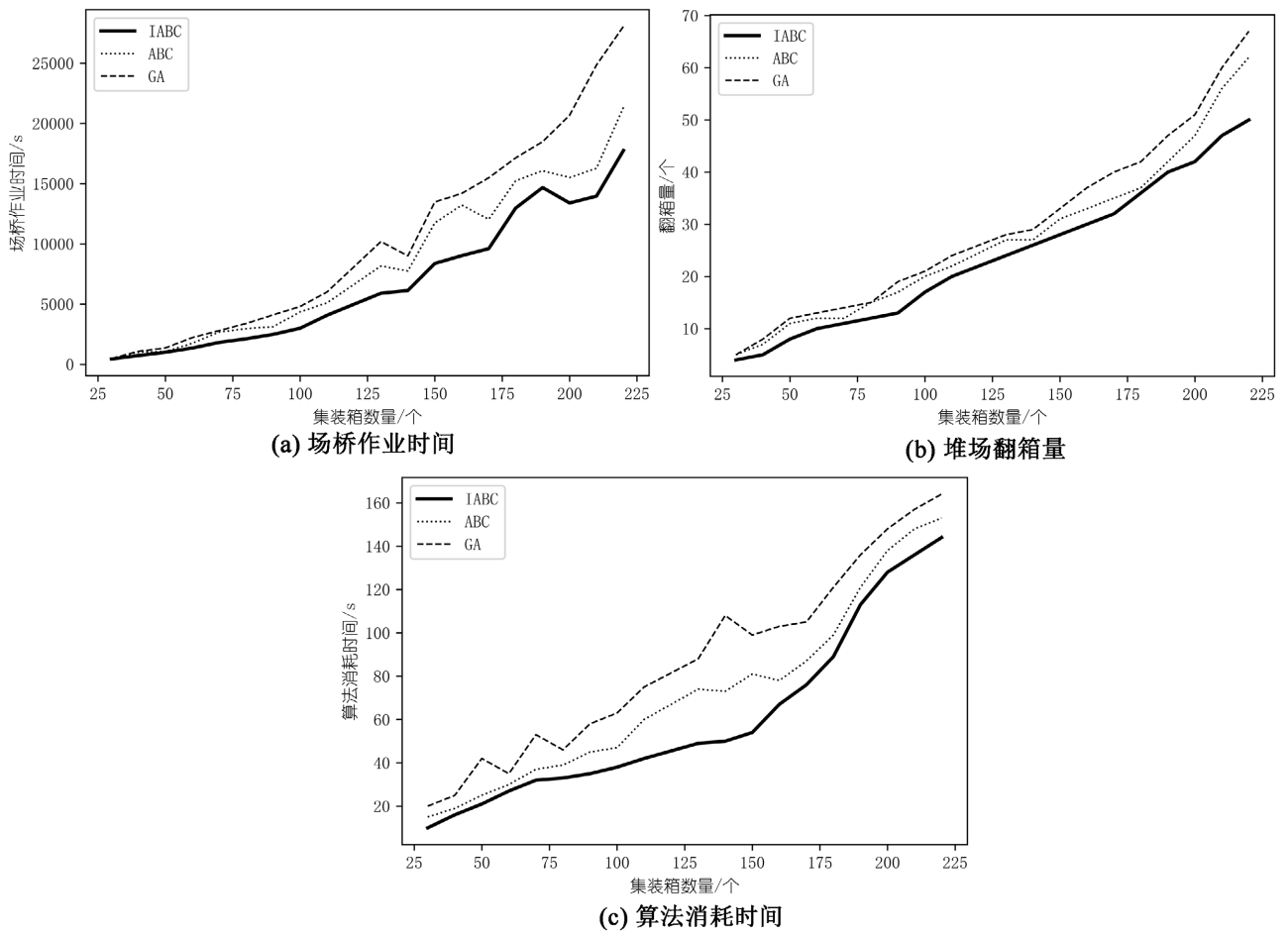


Figure 2. Comparison of the solution results of different algorithms

图 2. 不同算法求解结果对比

Table 2. Table of box turning strategy comparison

表 2. 翻箱策略对比表

| 算例 | | 场桥作业时间/s | | 翻箱量/个 | |
|-----------------------|-----|----------|-------|-------|-------|
| 算例规模 | N | 最矮栈策略 | 邻近栈策略 | 最矮栈策略 | 邻近栈策略 |
| $2 \times 6 \times 4$ | 30 | 443 | 455 | 4 | 5 |
| $2 \times 6 \times 4$ | 40 | 732 | 865 | 5 | 7 |
| $3 \times 6 \times 4$ | 50 | 1004 | 1187 | 8 | 10 |
| $3 \times 6 \times 4$ | 60 | 1454 | 1697 | 10 | 12 |
| $4 \times 6 \times 4$ | 70 | 1824 | 2038 | 11 | 13 |
| $4 \times 6 \times 4$ | 80 | 2123 | 2628 | 12 | 14 |
| $5 \times 6 \times 4$ | 90 | 2490 | 3067 | 13 | 15 |
| $5 \times 6 \times 4$ | 100 | 2998 | 3587 | 17 | 19 |
| $6 \times 6 \times 4$ | 110 | 4076 | 4389 | 20 | 21 |
| $7 \times 6 \times 4$ | 130 | 5903 | 5332 | 24 | 25 |

续表

| | | | | | |
|------------------------|-----|--------|--------|----|----|
| $8 \times 6 \times 4$ | 140 | 7134 | 5936 | 26 | 27 |
| $8 \times 6 \times 4$ | 150 | 7534 | 6387 | 28 | 30 |
| $9 \times 6 \times 4$ | 160 | 8723 | 7298 | 30 | 33 |
| $9 \times 6 \times 4$ | 170 | 10,605 | 9075 | 32 | 36 |
| $9 \times 6 \times 4$ | 180 | 12,976 | 11,098 | 36 | 37 |
| $10 \times 6 \times 4$ | 190 | 14,678 | 12,876 | 40 | 40 |
| $10 \times 6 \times 4$ | 200 | 15,397 | 13,870 | 42 | 45 |
| $11 \times 6 \times 4$ | 210 | 16,327 | 14,973 | 47 | 49 |
| $11 \times 6 \times 4$ | 220 | 19,263 | 17,752 | 50 | 59 |
| 平均值 | | 15,397 | 13,870 | 42 | 45 |

及以上时, 邻近栈策略所需的作业时间较少; 2) 对于堆场翻箱量而言, 最矮栈策略和邻近栈策略求得的平均值依次为 42 个、45 个, 且最矮栈策略的翻箱量一直略小于邻近栈策略。以上结果表明, 翻箱策略对对场桥作业时间影响较大, 且随着作业箱量的增加, 邻近栈策略较优。

6. 结语

在集装箱多式联运港口实际背景下, 考虑船舶配载计划, 以公路、铁路、水路集港的待装船集装箱为对象, 研究了堆场集装箱箱位分配问题。为实现问题的分解降维, 基于双层建模方法构建最小化场桥作业时间和翻箱量的两阶段优化模型, 并改进的人工蜂群算法实现问题有效求解。最后, 通过算例研究论证了模型与改进算法的有效性, 得出的主要结论如下: 1) 引入信息素与灵敏度改进搜索方式有效改善了 IABC 算法的双目标协同优化能力, 其求解结果明显优于 ABC 和 GA 算法; 2) 翻箱策略对于翻箱量优化的影响并不突出, 但其对场桥作业时间影响较大, 当作业箱量较多时, 选择邻近栈策略可降低场桥作业时间。在多式联运港口的实际作业中, 堆场不同的计划周期、集港集装箱的失约或随机到达、场桥设备突发故障等不确定性扰动都带来了出口箱箱位分配的动态决策需求。后续研究中, 会将上述不确定性因素纳入决策考虑, 使研究更为贴近现实作业场景, 进一步提高集装箱多式联运港口的工作效率。

基金项目

湖北省自然科学基金计划项目(2023AFB071)。

参考文献

- [1] Kim, K.H., Park, Y.M. and Ryu, K.R. (2000) Deriving Decision Rules to Locate Export Containers in Container Yards. *European Journal of Operational Research*, **124**, 89-101. [https://doi.org/10.1016/S0377-2217\(99\)00116-2](https://doi.org/10.1016/S0377-2217(99)00116-2)
- [2] 李燕晶. 船舶积载图约束下铁水联运集装箱堆场空间资源优化方法[D]: [硕士学位论文]. 北京: 北京交通大学, 2017.
- [3] 王曦. 基于改进萤火虫算法的集装箱堆场出口箱箱位分配问题研究[D]: [硕士学位论文]. 大连: 大连海事大学, 2017.
- [4] Gharehgozli, A. and Zaerpour, N. (2018) Stacking Outbound Barge Containers in an Automated Deep-Sea Terminal. *European Journal of Operational Research*, **267**, 977-995. <https://doi.org/10.1016/j.ejor.2017.12.040>
- [5] 常祎妹, 朱晓宁. 考虑配载计划的铁路集港箱位指派[J]. 交通运输工程学报, 2020, 20(4): 205-216.
- [6] 计明军, 黄思佳, 郭文文. 海铁联运中心站堆场箱位指派优化[J]. 系统工程理论与实践, 2016, 36(6): 1555-1567.

-
- [7] 任刚, 靳志宏, 庞毛毛, 等. 基于班列抵港模式的海铁联运堆场箱位分配优化[J]. 大连海事大学学报, 2022, 48(1): 1-10
- [8] 范厚明, 姚茜, 马梦知. 多场桥分区域平衡策划下的集装箱堆场箱位分配问题[J]. 控制与决策, 2016, 31(9): 1603-1608.
- [9] 范厚明, 马梦知, 姚茜, 郭振峰. 集装箱堆场箱位分配及多场桥调度协同优化问题[J]. 上海交通大学学报, 2017, 51(11): 1367-1373.
- [10] 范厚明, 孔靓, 岳丽君. 考虑出口箱进出场及预翻箱的箱位分配与场桥调度协同优化[J]. 运筹与管理, 2021, 30(6): 26-34.
- [11] 郑云峰, 刘瑞麟, 饶本顺, 周校彬, 张国庆. 确定任务视角下集装箱堆场箱位分配研究[J]. 重庆交通大学学报(自然科学版), 2023, 42(7): 120-127.
- [12] 刘婵娟, 胡志华. 基于滚动计划的堆场出口箱存储位置分配两阶段模型[J]. 上海大学学报(自然科学版), 2017, 23(5): 789-800.
- [13] 范厚明, 孔靓, 岳丽君. 考虑出口箱进出场及预翻箱的箱位分配与场桥调度协同优化[J]. 运筹与管理, 2021, 30(6): 26-34.
- [14] 常祎妹. 铁水联运港口核心作业计划的优化[D]: [博士学位论文]. 北京: 北京交通大学, 2019.
- [15] 祝慧灵, 计明军, 郭文文, 等. 基于配载计划的集装箱提箱顺序和倒箱策略优化[J]. 交通运输系统工程与信息, 2016, 16(2): 191-199.