

## 一种自动化集装箱码头出口箱进箱选位算法

李隋凯<sup>1,2</sup>, 励益韬<sup>1,2</sup>, 孙未未<sup>1,2</sup>, 秦 涛<sup>3</sup>, 黄秀松<sup>3</sup>

(1. 复旦大学 计算机科学技术学院, 上海 201203;

2. 上海智能电子与系统研究院, 上海 200433; 3. 上海海勃物流软件公司, 上海 200080)

**摘 要:** 自动化集装箱码头的出口箱进箱选位是自动化码头运营的难点之一。针对带中转排的自动化码头出口箱进箱选位问题, 提出一种两倍状态多阶段动态规划算法。在阶段划分时, 将阶段数量增加至原有的2倍, 以适应2种状态转移, 而在进行状态转移时, 对2种状态分别计算来获得更优的结果。实验结果表明, 与现有的动态规划方法相比, 该算法通过应用更精确的状态转移方程, 可使翻箱数量降低20%。

**关键词:** 自动化集装箱码头; 堆场; 概率分布; 动态规划; 遗传算法

**中文引用格式:** 李隋凯, 励益韬, 孙未未, 等. 一种自动化集装箱码头出口箱进箱选位算法[J]. 计算机工程, 2019, 45(5): 272-278, 284.

**英文引用格式:** LI Suikai, LI Yitao, SUN Weiwei, et al. A storage space allocation algorithm for export containers in automated container terminals[J]. Computer Engineering, 2019, 45(5): 272-278, 284.

## A Storage Space Allocation Algorithm for Export Containers in Automated Container Terminals

LI Suikai<sup>1,2</sup>, LI Yitao<sup>1,2</sup>, SUN Weiwei<sup>1,2</sup>, QIN Tao<sup>3</sup>, HUANG Xiusong<sup>3</sup>

(1. School of Computer Science, Fudan University, Shanghai 201203, China;

2. Shanghai Institute of Intelligent Electronics and Systems, Shanghai 200433, China;

3. Shanghai Harbor Logistics Software Co., Ltd., Shanghai 200080, China)

**【Abstract】** Storage space allocation of export containers in an automated container terminal is a hard problem in terminal operation. To address the problem of space allocation for export containers in terminals with the transfer row, a double-state multi-stage planning algorithm is proposed. In the state dividing phase, the number of stages is doubled, to fit the two types of state transition. In the state transition phase, two types of states are calculated to get a better result. Experimental results show that compared with the existing dynamic planning method, the algorithm reduces the times container flips occur by 20%.

**【Key words】** automated container terminal; yard; probability distribution; dynamic planning; Genetic Algorithm( GA)

**DOI:** 10.19678/j.issn.1000-3428.0050566

## 0 概述

自动化集装箱码头具有降低人力成本、节约能源等优势, 已经成为世界港口的主要建设与发展方向。自从1993年荷兰鹿特丹港ECT自动化码头建成以来, 世界范围内已经有三十多个自动化集装箱码头。

本文研究主要基于洋山四期自动化集装箱码头, 该码头于2017年12月10日正式开港运营, 是全球规模最大的自动化集装箱码头<sup>[1]</sup>。由于箱区个数比其他自动化码头有大幅提升, 这使得对于每条船均摊到各个箱区内的集装箱数量变少; 考虑到出口箱在堆场中需要根据船的配载信息来进行分组, 使得每组中的

数量就会更少, 一般每个箱区内只有20箱左右。此外, 洋山四期自动化码头采用了高性能的轨道吊进行作业, 绝大多数出口箱在进箱作业时会采用接力方式, 即由陆侧轨道吊将集装箱吊至中间位置, 再由中间位置移动到海侧的最终位置。对中间位置的集装箱堆存进行有效管理, 可以减少未来在装船时的出口箱翻箱数量, 有效利用这一作业流程是提升码头整体性能的关键之一。本文针对该情况, 提出对应的模型和算法, 并且进行了真实数据的实验对比。

## 1 背景知识

图1是洋山四期自动化码头堆场平面布局,

**基金项目:** 国家自然科学基金(61772138)。

**作者简介:** 李隋凯(1993—), 男, 硕士研究生, 主研方向为智能物流; 励益韬, 硕士研究生; 孙未未, 教授、博士; 秦 涛、黄秀松, 工程师。

**收稿日期:** 2018-03-02 **修回日期:** 2018-04-10 **E-mail:** lisuikai@fudan.edu.cn

61个箱区垂直于岸线排列,每个箱区贝位数范围为33~53,文献[1]中有详细介绍。

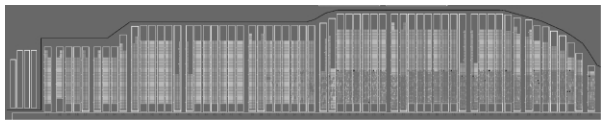


图1 洋山四期自动化码头堆场平面布局

### 1.1 相关术语

本节列出和本文工作相关的集装箱码头领域的术语。

**堆场:** 码头上用来堆放集装箱的场地。一个堆场由多个箱区组成。

**箱区:** 箱区由多个贝位组成。每个箱区有2台轨道吊用于运送集装箱到箱区内部。

**贝位/排:** 箱区的空间由贝位组成,每个贝位由排组成,每个排由场箱位组成,如图2所示。

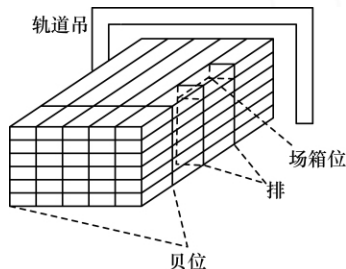


图2 箱区示意图

**限高:** 每个排上都有一个可堆叠高度的上限,不同的码头有不同的限高。在自动化码头中,一般限高为5 m或6 m。

**重量等级:** 根据集装箱的不同重量,可划分集装箱的重量等级,在通常情况下划分为6个重量等级。

**场箱位:** 集装箱在堆场中的位置。在图3中,总共有24个场箱位可用,其中虚线部分代表可以堆放集装箱的位置,但是必须保证其下面的位置有集装箱;而数字则表示当前集装箱的重量等级。

3				
2	5	4		
1	3	2	1	

图3 排与场箱位示例

**轨道吊:** 指箱区内用来堆放集装箱的机械。集装箱由轨道吊从堆场外堆放至堆场。

**翻箱:** 指将箱子临时移动到堆场其他位置的动作。该动作数量越少越好。

### 1.2 出口箱进箱特点

以洋山四期自动化码头为例,堆场的出口箱进箱满足以下特点:

1) 按照集卡到达堆场的时间先后,依次完成进箱。

2) 在装船时,重量等级高的箱子会先从堆场中运出,比其重量等级低的箱子需要翻箱,按照此规则产生的翻箱数量作为最终结果进行统计,该数量越小越好。具体如图4所示,图4中标注下划线的重量等级为2和3的集装箱需要翻箱,因为其下面的集装箱重量等级更高,需要优先从堆场中取出。

<u>2</u>		
3		<u>3</u>
2	5	4
1	3	2

图4 堆场翻箱示意图

3) 不同重量等级的集装箱到达港口的顺序符合随机平均分布。本文根据实际生产情况,重量等级分为6个等级。

4) 如图5所示,所有进入的箱子,均会先进箱至中转排,然后再由中转排接力移动到最终位置。

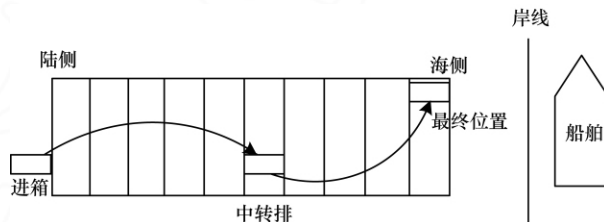


图5 自动化集装箱码头示例

5) 根据实际作业需要,中转排比最终位置堆存高度要低,并且允许堆放的中转排一般只有1排。

6) 由于自动化箱区的轨道吊专属于对应箱区,每个箱区的发箱能力有限,因此同一类型的出口箱要尽可能地分散到多个箱区,以避免装船发箱时拥堵。

表1列出对集装箱进行分类的属性,同一船全部属性都相同的集装箱视为同一类,例如:“6.096 m,高箱,釜山港,吨位等级1”的所有集装箱。

表1 集装箱类型分类

属性	特点说明
尺寸/m	6.096, 12.192, 13.716
高平	高箱, 平箱
目的港	同一条船有多个目的港的集装箱
吨位等级	按照集装箱重量划分为多个等级,一般分6级

## 2 自动化集装箱码头堆场进箱选位问题

根据以上出口箱进箱特点,定义本文研究的问题。

**问题(带中转排的自动化集装箱码头堆场进箱选位问题)** 给定 $K$ 个排,每个排都有固定限高 $H$ ,组成 $K \times H$ 个最终位置。现在需要连续出口箱进箱,有一个中转排用于中转,该排的高度限制为 $H_m < H$ ,

所有集装箱必须先进箱至中转排,之后再从中转排移动到某个最终位置,集装箱均可在中转排上等待后来的其他箱先放入最终位置后,再将此集装箱放入最终位置。最后所有集装箱都必须放到最终位置,并且翻箱数量最小。

翻箱数量的计算满足以下条件:每次选取最终位置内重量等级最高且高度最高的集装箱,移除其上面的所有集装箱并且将其数量加入翻箱数量,之后将此箱移除,直到所有  $N$  个集装箱全部被移除。集装箱的重量等级为 1 至重量等级上限  $W$  中的一个整数。

图 6 所示是一个进箱流程。其中,  $N=15, K=3, H=5$ 。图 6(a) 表示进箱前的状态;图 6(b) 表示当前进箱至中转排时,中转排上多了吨位等级为 4 的集装箱;图 6(c) 表示该箱移动到最终位置时整体的堆放情况。每次进箱都会重复该过程,直到所有集装箱全部进入最终位置,图 6(c) 的状态为仅差最后一个集装箱即完成进箱至中转排,因为中转排中的集装箱需要全部移动到最终位置。同样,图 6(a) 的状态也可直接变化为图 6(d) 的状态,之后再进箱。

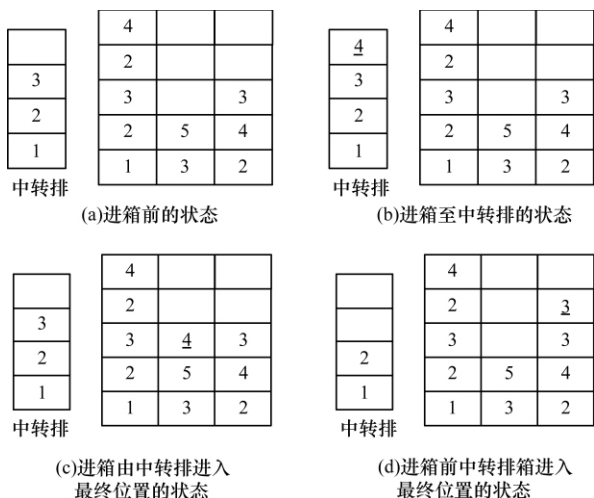


图 6 进箱流程示例

### 3 两倍阶段多状态动态规划算法

自动化堆场在作业时利用中转排暂存集装箱来提高效率,针对这一特点本文提出一种新的两倍阶段多状态动态规划算法 (Double Time Phases Multi-status Dynamic Programming, DMDP) 来减少翻箱数量。和现有相关工作不同,本文的动态规划算法在每次决策中都会计算“进箱至中转排”,或“将中转排的集装箱移动到最终位置”2 种状态转移,因此能够获得更优的结果。为了支持这种决策,首先,将总箱数为  $N$  的出口箱进箱过程划分成  $2N+1$  个阶段,使得每次无论进行哪种决策,都会进入下一阶段;其次,在构建每个阶段内的状态时,将中转排的信息也纳入状态表示中。因此,在做动态规划决策时就能够进行 2 种状态转移的最优选择。

#### 3.1 算法流程

算法流程包括以下 5 个阶段:

1) 确定决策对象。当每次决策时,都会选择以下 2 种情况之一:保持当前状态不变,允许下一个出口箱进入中转排,即“进箱至中转排”;或者在下一个出口箱进入中转排前,先将中转排上的集装箱移动到某个最终位置,即“将中转排的集装箱移动到最终位置”。

2) 阶段划分。划分为  $2N+1$  个阶段,规定 0 阶段的所有状态均为所有出口箱进箱完成的状态,即  $N$  个集装箱全部位于最终位置,0 阶段的所有期望值为 0,即 0 阶段没有任何预计需要的翻箱产生。任意  $n$  阶段内的可行状态,均可以由  $n+1$  阶段的某种可行状态变化而得到,该变化必为以下两者之一:

(1) 在中转排未达到限高的情况下,有某一个集装箱进箱至中转排。

(2) 中转排在至少有一个集装箱的情况下,有某一个集装箱由中转排移动到最终位置。

**定理 1** (阶段划分的正确性) 按照该阶段划分的方法,其每个阶段和出口箱进箱流程内每次集装箱位置发生变化后的情况是一一对应的,且  $N$  个集装箱恰巧在  $2N$  次阶段变化后,全部移动到最终位置。

**证明** 对于每个集装箱,其只会进行以下 2 种操作各 1 次:先进入中转排;之后从中转排移动到最终位置。由于总共有  $N$  个集装箱,那么总共有  $2N$  次操作数量,在这些操作全部完成后所有集装箱必然被全部移动到最终位置。而这个阶段划分的方法每次必然有一个操作被执行,因此,恰好和进箱过程中的所有情况一一对应,并且这  $2N$  个操作完成后,全部集装箱都会被移动到最终位置。

3) 状态构建。采用如下的 3 段表示方法和排序来进行状态构建:

第 1 段:长度为  $K$ ,每一位表示当前排的剩余高度,即可移动到该排的集装箱的数量。

第 2 段:长度为  $K$ ,每一位表示当前排的重量等级最高的集装箱的重量等级,第 2 段中的每一位和第 1 段一一对应,如果该排的剩余高度为 0 或  $H$ ,则该排的值为 0。

第 3 段:长度为  $H_m$ ,每一位表示中转排当前集装箱的重量等级。

在状态表示完成后,将第 1 段和第 2 段内的数值组成二元组,对二元组进行按照剩余高度从大到小的顺序排序,之后完成状态构建。

具体构建过程如图 7 所示,先获得每一排的剩余可堆叠高度 (0, 3, 2, 5)、重量等级 (0, 5, 4, 0) 和中转排重量等级 (1, 2, 3, 0),然后按照剩余可堆叠高度,对由剩余高度和重量等级组成的二元组序列 (0-0, 3-5, 2-4, 5-0) 从大到小进行排序,得到 (5-0, 3-5, 2-4, 0-0),由此得到最终的状态 (5, 3, 2, 0, 0, 5, 4, 0, 1, 2, 3, 0),长度为  $2K+H_m$ 。

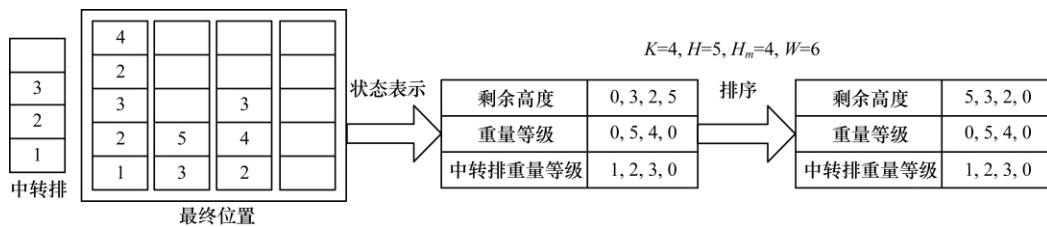


图7 状态构建过程

**引理 1** (状态构建的空间复杂度) 按照上述构建方法进行状态构建, 对应的空间复杂度为  $O(C_{H+K}^K \cdot (W+1)^{K+H_m})$ , 其中  $C_{H+K}^K$  为组合数。

**证明** 分两步证明: 1) 状态总数上限为  $C_{H+K}^K \cdot (W+1)^{K+H_m}$ ; 2) 对于每一种状态, 本文均只需要记录其哈希值来反向求解其具体状态, 而不需要存储具体长度为  $2K+H_m$  的状态信息。

状态信息第1段内由于排序过, 因此该段的不同状态数上限为  $C_{H+K}^K$ , 文献[3]的附录部分有相同子问题的证明。第2段内每一位的值不会超过  $W$ , 而第3段内的每一位的值也不会超过  $W$ , 两段状态的长度分别为  $K$  和  $H_m$ , 因此第2段、第3段的总状态数不会超过  $(W+1)^{K+H_m}$ 。所以, 状态总数不超过第1段、第2段、第3段的状态数的乘积, 即  $C_{H+K}^K \cdot (W+1)^{K+H_m}$ 。

哈希值的构建如下: 可以将第2段、第3段的状态看做  $W+1$  进制数, 则可以直接通过按位分解的方法将一个小于等于  $(W+1)^{K+H_m}$  的数字转化为第2段、第3段的状态信息; 通过预先生成第1段状态信息内的

所有状态进行存储并且进行编号, 使得每个具体状态和1到  $C_{H+K}^K$  的每个数字一一对应, 之后将该数字乘以  $(W+1)^{K+H_m}$  再加上第2段、第3段的状态对应的数字, 即可获得一个能够表示当前状态的哈希值。在此过程中, 预先生成的第1段信息的空间复杂度为  $C_{H+K}^K \cdot K$ , 但在本文问题假设条件下  $W>1$ , 则  $K < (W+1)^{K+H_m}$ , 因此, 不需将该部分加入空间复杂度表示。

4) 目标函数和转移代价。有2种方程转移方式, 分别对应出口箱进箱至中转排和从中转排移动到最终位置2种情况。

第1种情况: 出口箱进箱至中转排的过程不会产生转移代价, 但是必须满足中转排高度小于限定高度。在转移过程中, 根据吨位均匀分布, 对于不同重量等级的集装箱生成的预期结果均进行计算, 然后乘以该等级的概率, 来求解决定当前状态的翻箱数量期望最小。图8所示是一个状态变化的例子, 当一个新的集装箱进入中转排时, 其中仅有“中转排重量等级”信息发生了变化。

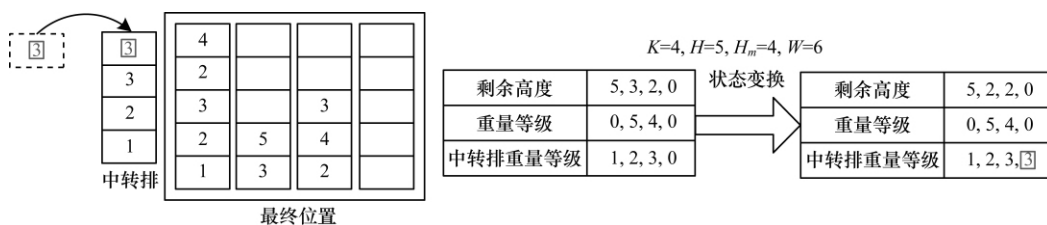


图8 状态变化(第1种情况)

第2种情况: 从中转排移动到最终位置时, 选取翻箱数量期望最小的一排进行堆叠, 其转移代价即为该箱放到上面引起的翻箱数量计算, 但是该过程

必须满足中转排高度不为0。图9所示是一个状态变化的例子, 当一个集装箱从中转排移动到最终位置时, 所有的状态信息都可能发生变化。



图9 状态变化(第2种情况)

5) 状态转移方程。公式如下:

$$F(X_n) = \begin{cases} 0, n=0 \\ \min_w \left( \sum_{w=1}^W P(w) (F(\text{Mid}(X_n, w)) + \text{CostM}(X_n, w)) \right), \\ \min_k (F(\text{Last}(X_n, D_k)) + \text{CostL}^k(X_n, D_k)) \end{cases}, 1 \leq n \leq 2N$$

其中,  $\min$  表示取较小值,  $Mid(X_n, w)$  表示当前状态为  $X_n$ 、重量等级为  $w$  的集装箱进入到中转排后的状态,  $Last(X_n, w)$  表示当前状态为  $X_n$ 、重量等级为  $w$  的集装箱进入到中转排后的状态,  $CostM$  表示进箱至中转排的代价, 对本文所讨论的问题中它是一个恒定值,  $CostL$  表示中转排移动到最终位置的代价,  $D_k$  表示选择的最终位置的第  $k$  排,  $w$  表示假设当前进箱运往中间位置的集装箱的等级,  $P(w)$  表示重量等级为  $w$  的概率, 此模型下  $P(w) = 1/W$ 。全部进箱到最终位置时的状态为  $X_0$ , 定义  $F(X_0) = 0$ 。

### 3.2 算法伪代码

$F(X_n)$  函数、 $Mid(X_n, w)$  变换和  $Last(X_n, k)$  变换的伪代码如算法 1~算法 3 所示。

#### 算法 1 $F(X_n)$ 函数

输入  $X_n; f[]$  是一个全局实数数组, 记录  $F(X_n)$  的值, 初始时  $f[X_0] = 0$  且其他所有的  $f[] = MAX$ , 其中,  $MAX$  表示一个极大值, 在算法实现中取  $10^9$ ;  $\min(x, y)$  表示  $x, y$  2 个数取较小值

输出  $F(X_n)$

$F(X_n)$

1 if ( $f[X_n] \neq MAX$ ) return  $f[X_n]$ ;

2  $t_1 = MAX$ ;

3 if ( $X_n$  中转排不为满)

4  $t_1 = 0$ ;

5 for  $i = 1$  to  $W$

6  $CostM, X_{n-1} = Mid(X_n, i)$ ;

7  $t_1 = t_1 + CostM + (F(X_{n-1}))$ ;

8  $t_1 = t_1 / W$ ;

9  $t_2 = MAX$ ;

10 if ( $X_n$  中转排不为空)

11 for  $k = 1$  to  $K$

12  $CostL^k, X_{n-1}^k = Last(X_n, k)$ ;

13  $t_2 = \min(t_2, F(X_{n-1}^k) + CostL^k)$ ;

14  $f[X_n] = \min(t_1, t_2)$ ;

15 return  $f[X_n]$ ;

#### 算法 2 $Mid(X_n, i)$ 变换

输入  $X_n, i$

输出  $X_{n-1}$

$Mid(X_n, i)$

1  $X_{n-1} = X_n$  将第  $i$  个箱置于中转排顶层箱上方;

2 return  $X_{n-1}$ ;

#### 算法 3 $Last(X_n, k)$ 变换

输入  $X_n, k$

输出  $X_{n-1}^k, CostL^k$

$Last(X_n, k)$

1  $CostL^k = 0$ ;

2 if ( $X_n$  的中转排顶层箱重量小于  $k$  排最大重量)

3  $CostL^k = 1$ ;

4  $X_{n-1}^k = X_n$  第  $k$  排重量更新, 中转排移除顶层箱;

5 return  $X_{n-1}^k, CostL^k$ ;

### 3.3 时空复杂度分析

本节主要对该算法的时间复杂度和空间复杂度进行分析。

**定理 2**(DMDP 算法的时间复杂度) 两倍阶段

多状态动态规划算法 (DMDP) 的时间复杂度为  $O(C_{H+K}^K \cdot (W+1)^{K+H_m} \cdot (W+K \cdot (H+K)))$ 。

**证明** 使用引理 1 获得的结论, 其状态数为  $C_{H+K}^K \cdot (W+1)^{K+H_m}$ 。在转移时, 前半部分  $Mid$  的式子最多执行  $W$  次, 而后半部分的  $Last(X_n, w)$  求解中需要计算  $K$  次哈希值, 时间复杂度为  $O(K \cdot (H+K))$ , 则每次状态转移时间复杂度为  $O(W+K \cdot (H+K))$ 。而对于每一个状态都会执行状态转移方程一次, 因此, 总时间复杂度为  $O(C_{H+K}^K \cdot (W+1)^{K+H_m} \cdot (W+K \cdot (H+K)))$ 。

**定理 3**(DMDP 算法的空间复杂度) 两倍阶段多状态动态规划算法 (DMDP) 的空间复杂度为  $O(C_{H+K}^K \cdot (W+1)^{K+H_m})$ 。

**证明** 空间的使用除了状态的构建和存储, 其空间使用都小于  $O((W+1)^{K+H_m})$ , 根据引理 1 得证, 空间复杂度为  $O(C_{H+K}^K \cdot (W+1)^{K+H_m})$ 。

## 4 实验结果与分析

### 4.1 实验设置

实验采用的数据集是 1 000 组洋山港集装箱码头的真实作业数据, 每组数据是一个长度为  $K \times H$  的重量等级序列。按照尺寸、高平和目的港对集装箱进行分类, 每类平均到各个箱区的集装箱数量约为 18 个。为了提高堆场的利用率, 要求堆放出口箱的所有排最终都尽可能堆满, 即  $K \times H$  尽可能接近 18。由于自动化集装箱码头的堆叠高度要求为高箱允许最多堆叠到 5 层, 因此高箱所占用的排数至少为 4, 即  $H=5, K=4$ ; 平箱允许堆叠高度为 6 层, 占用排数为 3, 即  $H=6, K=3$ 。

所有算法均由 C++ 实现, 编译器版本号为 g++ (tdm64-2) 4.8.1, 运行在一台 CPU 为 Intel Core i5 的计算机上。

### 4.2 对比算法

现有工作集中在传统堆场应用场景, 都不考虑中转排的情况。实验中把本网的 DMDP 算法和 4 个现有算法的典型代表做效果和性能对比。5 个算法的简要描述如下。

1) 贪心算法: 该算法来自于码头作业工人在实际操作时的策略, 这也是当前大部分码头的实际情况, 该算法由文献 [4] 提出。其策略为: 对于每一个出口箱进箱, 每次选取该排内所有集装箱重量等级都比此箱低的排的位置, 除非所有可选排内都有比此箱重量的等级高的集装箱; 在这个前提下, 每次选取排内重量等级最大值最高的一排的位置。

2) 动态规划算法: 文献 [2] 提出的动态规划算法具有最小翻箱数量期望, 是这一领域的经典工作。其计算公式在文献 [5] 中有修正, 对比实验时采用了修正后的方法。

3) 遗传算法: 可以达到理论最优的翻箱数量, 但其必须在已知进箱序列的前提下才有效, 且运行时间较长, 因此, 很少在实际工作中应用。参照文

献[6-7]提出的算法来进行对比实验。本文实验中采用的具体参数为:

(1) 每个个体都代表  $N$  个箱子进箱的最终结果。每个个体都有  $N$  位,染色体用数字序列表示,染色体上的数字对应基因位,属于  $1 \sim K$  中的数字,且  $1 \sim K$  每个数字恰好出现  $H$  次,代表这个箱子将要落到的排的编号。

(2) 适应度即为箱总数减去翻箱数量的值,翻箱数量越少,适应度越高。

(3) 选择和交叉过程如下:每次随机选取当前种群一定比例的个体(如 10%),挑选其中适应度最高的个体作为交叉父代。交叉采用 OX 交叉方法,2 个父代产生 2 个子代,子代加入构建“新种群”。不断循环选择交叉操作,直至新种群个体数等于其父代种群数。

(4) 变异时随机选择“新种群”一定比例(如 1%)

的个体,每个个体随机交换一次 2 个值不同的基因位。

(5) 当代数到达设定的最大遗传代数(如 100)或种群最优适应度等于箱总数时终止。

4) 顺序算法:是一个朴素的基准算法,用来衡量各种算法的优化程度。对于每一个出口箱进箱,如果有还没堆叠到高度上限的排,那么堆存位置选择至该排顶端;如果只有堆叠到高度上限的排和空的排,则选择任意一个空排,堆叠至该排的第一层。

5) 本文算法 DMDP:和上述对比算法不同,DMDP 算法利用了中转排,不同的中转排高度  $H_m$  会有不同的性能,本文对  $H_m$  的不同取值分别进行实验。

#### 4.3 结果分析

实验结果见表 2,当中转排的高度限制  $H_m = 1$  时,DMDP 算法会退化为不利用中转排的算法,结果和文献[2]的算法一致。因此,表 2 中只展示  $H > H_m > 1$  的结果。

表 2 算法实验结果

算法及数据类型	高箱( $K=4, H=5$ )			平箱( $K=3, H=6$ )		
	翻箱数量	优化率/%	运行时间/s	翻箱数量	优化率/%	运行时间/s
顺序算法	6 418	0.00	0.83	6 401	0.00	0.81
贪心算法	2 152	66.47	0.85	2 948	53.94	0.93
动态规划算法	1 834	71.42	1.88	2 676	58.19	1.56
遗传算法	2 717	57.55	774.00	2 836	55.69	701.00
DMDP( $H_m=2$ )	870	86.44	3.28	1 373	78.55	1.97
DMDP( $H_m=3$ )	476	92.58	15.75	834	86.97	3.15
DMDP( $H_m=4$ )	307	95.22	2 259.00	587	90.83	13.71
DMDP( $H_m=5$ )	—	—	—	383	94.02	11 715.00

1) 算法结果比较。表 2 展示了翻箱数量和优化率 2 项指标,其中翻箱数量直接代表了算法结果的优劣,优化率为其他算法相比顺序算法减少的翻箱数的百分比。从实验数据可以看出,对比动态规划算法,当  $H_m$  取值为  $H-1$  或  $H-2$  时,其优化率均在 90% 以上。对于高箱,DMDP 算法的最高优化率相比动态规划算法提升了 20%,而动态规划算法相比贪心算法仅提升了 5%;对于平箱,DMDP 算法的最高优化率相比动态规划算法提升了 35%,而动态规划算法相比贪心算法仅提升了 5%。这说明 DMDP 算法对结果的提升效果十分显著。相比之下,贪心算法、遗传算法和动态规划算法的优化率差别均在 10% 左右,其整体优化率都距 DMDP 算法差距较大。

2) 算法运行效率比较。遗传算法的运行时间较长,其主要原因是对于每一组数据都需要运行获得结果,而 DMDP 算法对于每种状态只需要求解 1 次,从而可以在状态数量较少时较快地获得结果。当代数到达 1 000 代时,平均每组数据的运行时间在 1 s 左右,如果再采取 5 次运行取最优值的方法,其每组解的运行时间将达到 5 s,而这会给实际生产带来较大影响。

从表 2 中还可以看出,DMDP 算法的运行时间在  $H_m = H-1$  时大幅增长,其运行时间在  $H_m < H-1$  的情况下都在 20 s 以内。据此推荐在实际生产中使用  $H_m = H-2$  的模型来进行快速求解,在保证较快计算速度的前提下得到最好的优化结果。

#### 5 相关工作

出口箱进箱选位算法是码头信息化的关键组成部分,在传统人工作业码头中已经有大量研究。对于此类问题,文献[8]提出,选位问题可以归结为一类满足约束条件的优化问题,而文献[9]对集装箱移动过程中的问题进行了总结,出口箱选位属于其中的一类问题。文献[10]提出跟随时间变更制定滚动计划,将选位问题分为多个阶段进行求解。文献[11]提出利用集装箱的在场时间信息来进行选位决策。文献[12]提出对集装箱进行分类来进行选位。文献[13]对集装箱使用分类策略会使堆存结果更优。文献[2]提出一种基于动态规划的算法,该算法通过计算每种选择可能带来的翻箱数量的期望值来进行位置选择,以使得翻箱数量期望值最小。文献[3]根据决策树模型,将算法的效率进行了大幅的

提升,使其能够在规模更大的数据下快速解出,其方法和文献[2]的模型较为相似。文献[14]提出遗传算法来求出期望值,选择最优期望值来进行启发式算法的决策。文献[4]提出使用一些策略的启发式方法来进行搜索。然而这些工作研究的进箱问题,其问题模型和本文研究,这些方法都是针对没有中转排的情况,其问题模型较为简单。

本文所研究的出口箱进箱选位问题,可以利用中转排暂存集装箱,然后等待其他集装箱先进场后再进入最终位置,其利用某些场箱位暂存集装箱的特征类似于翻箱问题。对于翻箱问题已经有大量研究成果。文献[15]提出预翻箱问题,对于这类问题,文献[16]具体地对预翻箱问题进行了定义,即在进箱结束后,为了满足装船时的需求,进行预翻箱,使得装船时不需翻箱,此过程所消耗的代价最小。文献[17]提出利用邻域搜索来进行求解获得结果。另外,还有一类CRP问题(Container Relocation Problem),该类问题主要是求解集装箱翻箱的最少移动数量,而不是求解进箱位置选择,例如文献[18]提出利用树搜索方法进行求解,文献[19]提出一种优化后的启发式算法来进行求解,文献[20]提出利用对搜索空间的压缩来提升性能。然而,这些工作研究的翻箱问题,其问题模型和本文所要研究的不同,这些工作都不涉及对后续进箱的情况进行推测和预计。

本文研究的自动化码头进箱问题中进箱和接力同时存在,现有工作主要以传统码头为应用背景,只单纯研究进箱或翻箱问题,目前为止尚未发现和本文研究问题相同的工作。

本文的具体贡献如下:

1) 针对实际作业过程,提出一个新的出口箱进箱选位的问题。该问题基于进箱过程,所有集装箱必须先从海侧运送到中间位置进行接力,之后再运送到最终位置。针对该问题,本文提出一个两倍阶段多状态动态规划算法,其时间复杂度能够满足实际运行需求,且比起现有的动态规划方法<sup>[2]</sup>,其采用了更加精确的状态转移方程,能将问题中的进箱情况进行更好的表示,从而获得更优的结果。

2) 本文实验部分采用了1 000组真实数据进行实验,其每组数据均来自于实际进港集装箱作业。实验结果表明,采用针对此模型的新算法,优化效果比现有算法有显著提升。该算法也是洋山四期自动化集装箱码头项目堆场选位系统的核心组成部分之一。

## 6 结束语

本文提出带中转排的自动化码头出口箱进箱选位问题,并且针对该问题提出了两倍状态多阶段动态规划算法。该算法根据中转排的特点设计了新的

状态表示方法,使用了更为精确的状态转移方程。实验结果表明,与现有算法相比,该算法可使翻箱数量减少20%。下一步将该算法应用到自动化集装箱码头的进出箱问题,以提高码头作业效率。

## 参考文献

- [1] 刘广红,程泽坤,罗勋杰,等. 洋山四期工程全自动化集装箱码头总体布置[J]. 水运工程,2016(9):46-51.
- [2] KIM K H, PARK Y M, RYU K R. Deriving decision rules to locate export containers in container yards [J]. European Journal of Operational Research, 2000, 124(1): 89-101.
- [3] GHAREHGOZLI A H, YU Y, DE K R, et al. A decision-tree stacking heuristic minimising the expected number of reshuffles at a container terminal [J]. International Journal of Production Research, 2014, 52(9): 2592-2611.
- [4] GUERRA-OLIVARES R, SMITH N R. A heuristic procedure for the outbound container space assignment problem for small and midsize maritime terminals [J]. International Journal of Machine Learning and Cybernetics, 2018, 9(10): 1719-1732.
- [5] ZHANG Canrong, CHEN Weiwei, SHI Leyuan, et al. A note on deriving decision rules to locate export containers in container yards [J]. European Journal of Operational Research, 2010, 205(2): 483-485.
- [6] 周明,孙树栋. 遗传算法原理及应用[M]. 北京: 国防工业出版社, 1999.
- [7] 欧阳超,陈志泊,孙国栋. Web 服务组合 QoS 优化中的改进遗传算法[J]. 计算机工程, 2017, 34(8): 231-235.
- [8] RASHIDI H, TSANG E P K. Novel constraints satisfaction models for optimization problems in container terminals [J]. Applied Mathematical Modelling, 2013, 37(6): 3601-3634.
- [9] VIS I F A, DE K R. Transshipment of containers at a container terminal: an overview [J]. European Journal of Operational Research, 2003, 147(1): 1-16.
- [10] ZHANG Chuqian, LIU Jiyin, WAN Y, et al. Storage space allocation in container terminals [J]. Transportation Research Part B: Methodological, 2003, 37(10): 883-903.
- [11] KIM K H, PARK K T. A note on a dynamic space-allocation method for outbound containers [J]. European Journal of Operational Research, 2003, 148(1): 92-101.
- [12] 张艳伟,石来德,宓为建,等. 集装箱码头出口箱集港堆存模型研究[J]. 中国工程机械学报, 2007, 5(1): 32-38.
- [13] DEKKER R, VOOGD P, VAN A E. Advanced methods for container stacking [J]. OR Spectrum, 2006, 28(4): 563-586.
- [14] YANG Huiyun, SHAO Qiangqian, CHANG Xin, et al. Storage optimization for export containers based on generalized delivery sequence [C]//Proceedings of 2016 International Conference on Logistics, Informatics and Service Sciences. Sydney, Australia [s. n.], 2016: 1-5.

(下转第284页)

由图 7 可以看出,智能干扰机实施最佳干扰后,本文算法相比于 FRSA、ORSA、RRSA 算法可以得到更高的平均回报,而且随着迭代次数的增加,本文算法逐渐接近 OOSA 最佳选择算法。

## 5 结束语

本文研究在智能干扰下用户路由及信道的联合选择问题,并将用户与干扰机之间的路由及信道选择问题建模为分层 Stackelberg 博弈模型,提出基于分层 Q 学习的网络层与 MAC 层联合优化抗干扰学习算法,在智能干扰下学习最佳路由选择及相应的信道分配策略。仿真结果表明,用户应用本文算法可以根据当前环境做出最佳抗干扰决策,实现高效可靠的信息传输。下一步将通过降低算法复杂度,使其应用于更大规模的通信网络。

## 参考文献

- [1] JIA Luliang, XU Yuhua, SUN Youming. A hierarchical learning solution for anti-jamming Stackelberg game with discrete power strategies [J]. IEEE Wireless Communications Letters, 2017, 6(6): 818-821.
- [2] YAO Fuqiang, JIA Luliang, SUN Youming. A hierarchical learning approach to anti-jamming channel selection strategies [J]. Wireless Networks, 2019, 25(1): 1-13.
- [3] WANG Beibei, WU Yongle. An anti-jamming stochastic game for cognitive radio networks [J]. IEEE Journal on Selected Areas in Communications, 2011, 29(4): 877-889.
- [4] ZHANG Yu, FU Fangwen, SCHAAAR M V D. On-line learning and optimization for wireless video transmission [J]. IEEE Transactions on Signal Processing, 2010, 58(6): 3108-3124.
- [5] 赵彪, 李鸥, 栾红志. Q 学习算法在机会频谱接入信道选择中的应用 [J]. 信号处理, 2014, 30(3): 298-305.
- [6] GHAFARI A. Real-time routing algorithm for mobile ad hoc networks using reinforcement learning and heuristic algorithms [J]. Wireless Networks, 2017, 23(3): 703-714.
- [7] LAM K, KRICHENE W, BAYEN A. On learning how players learn: estimation of learning dynamics in the routing game [C]//Proceedings of the 7th International Conference on Cyber-Physical Systems. Washington D. C., USA: IEEE Press, 2016: 11-14.
- [8] GHORBEL M B, HAMDAOUI B, GUIZANI M, et al. Distributed learning-based cross-layer technique for energy-efficient multicarrier dynamic spectrum access with adaptive power allocation [J]. IEEE Transactions on Wireless Communications, 2016, 15(3): 1665-1674.
- [9] NIE Junhong, HAYKIN S. A Q-learning-based dynamic channel assignment technique for mobile communication systems [J]. IEEE Transactions on Vehicular Technology, 1997, 48(5): 1676-1687.
- [10] WANG Wenbo, KWASINSKI A, NIYATO D, et al. A survey on applications of model-free strategy learning in cognitive wireless networks [J]. IEEE Communications Surveys and Tutorials, 2016, 18(3): 1717-1757.
- [11] 冯陈伟, 张璘. 一种基于 Q 学习的网络接入控制算法 [J]. 计算机工程, 2015, 41(10): 99-104.
- [12] CHEN Yihshen, CHANG Chungju, REN Fangchin. Q-learning-based multirate transmission control scheme for RRM in multimedia WCDMA systems [J]. IEEE Transactions on Vehicular Technology, 2007, 53(1): 38-48.
- [13] 黄云霞. 基于改进 Q 学习的认知无线网络动态频谱接入算法研究 [D]. 成都: 电子科技大学, 2012.
- [14] 张磊, 周金和, 张元. 基于 Stackelberg 博弈的云 CDN 缓存资源分配算法 [J]. 计算机工程, 2017, 43(8): 15-20.
- [15] 姜永, 陈山枝, 胡博. 异构无线网络中基于 Stackelberg 博弈的分布式定价和资源分配算法 [J]. 通信学报, 2013, 34(1): 61-68.
- [16] KIM K H, BAE J W. Re-marshaling export containers in port container terminals [J]. Computers & Industrial Engineering, 1998, 35(3/4): 655-658.
- [17] LEE Y, HSU N Y. An optimization model for the container pre-marshalling problem [J]. Computers and Operations Research, 2007, 34(11): 3295-3313.
- [18] HUANG Shan Huen, LIN Tsan Hwan. Heuristic algorithms for container pre-marshalling problems [J]. Computers and Industrial Engineering, 2012, 62(1): 13-20.
- [19] FORSTER F, BORTFELDT A. A tree search procedure for the container relocation problem [J]. Computers and Operations Research, 2012, 39(2): 299-309.
- [20] JIN Bo, ZHU Wenbin, LIM A. Solving the container relocation problem by an improved greedy look-ahead heuristic [J]. European Journal of Operational Research, 2015, 240(3): 837-847.
- [21] KU D, ARTHANARI T S. On the abstraction method for the container relocation problem [J]. Computers and Operations Research, 2016, 68(1): 110-122.

编辑 陆燕菲

编辑 索书志

(上接第 278 页)