

一种基于多目标优化的混合在线集卡调度方法

李 凡, 杜玉越

(山东科技大学 计算机科学与工程学院, 山东 青岛 266590)

摘 要: 为了提高集装箱港口工作效率, 在“大作业面”前提下, 结合任务触发型和集卡触发型两种集卡在线调度方法, 综合考虑系统总作业时间、岸吊等待时间以及集卡空载率问题, 建立多目标优化模型, 并运用启发式方法提出一种混合在线集卡调度算法。最后, 以港口实际数据为例, 通过实验仿真例证了模型优化效果和算法工作效率。

关键词: 在线集卡调度; 集装箱港口; 多目标优化; 混合; 仿真

中图分类号: U691.3

文献标志码: A

文章编号: 1672-3767(2017)02-0107-08

A Hybrid Online Scheduling Method of Truck Based on Multi-objective Optimization

LI Fan, DU Yuyue

(College of Computer Science and Engineering, Shandong University of
Science and Technology, Qingdao, Shandong 266590, China)

Abstract: In order to improve the efficiency of container ports, a multi-objective optimization model was constructed under the premise of the “full working-area” by combining the task trigger online scheduling and truck trigger online scheduling and by comprehensively considering the total operating time of system, the waiting time of quay cranes, and the no-load rate of trucks. Then a hybrid online scheduling algorithm of trucks was proposed by using a heuristic method. Finally, the optimization effect of the proposed model and the efficiency of the proposed algorithm were verified by simulation experiment based on the actual data of a port.

Key words: truck online scheduling; container terminal; multi-objective optimization; hybrid; simulation

集装箱港口(Container Terminal)是用来停靠运输集装箱的船舶、集装箱装卸作业的场所, 是在集装箱运输过程中水路和陆路运输的连接点, 也是集装箱多式联运的枢纽。随着世界经济的飞速发展, 全球化趋势日益加快, 远洋运输已在各国贸易中起到举足轻重的作用, 集装箱港口的建设与发展也成为竞争的焦点。港口在线调度策略的研究对自动化港口建设具有重要意义。为了提高集装箱港口的工作效率, 国内外学者针对集装箱调度优化做了大量研究。在时间优化方面, 谢晨等^[1]使用基于改进的 johnson 法则的启发式算法研究了集卡调度问题, 但只考虑了最小完成时间这一个因素, 缺乏对其他影响港口运营成本因素的考虑; CAO 等^[2]运用混合整数规划模型对岸吊与堆场运输问题进行研究, 并用遗传算法和 johnson 法则对模型进行求解, 但是优化目标考虑的因素不够全面。在路径优化方面, 蔡寒^[3]改进了拖车调度模型, 运用在线模式对集卡调度

收稿日期: 2016-07-31

基金项目: 国家自然科学基金项目(61170078, 61472228); 泰山学者建设工程项目; 山东省自然科学基金项目(ZR2014FM009); 山东省优秀中青年科学家科研奖励基金项目(BS2015DX010)

作者简介: 李 凡(1992—), 女, 山东淄博人, 硕士研究生, 主要从事港口调度与仿真研究。E-mail: 1321136177@qq.com

杜玉越(1960—), 男, 山东聊城人, 教授, 博士生导师, 主要从事 Petri 网、工作流、过程挖掘等方面的研究, 本文通信作者。E-mail: yydu001@163.com

问题进行研究,为自动化码头研究提供了思路,但是仅仅在“作业路”的基础上进行研究,优化效果不够理想;Kim 等^[4]讨论在装船时集卡和场桥的取箱路径优化问题,并采用了整数规划方法进行建模和求解,但没有考虑卸船时的情况,优化不够全面,效果不够理想。

目前国内集装箱港口大多采用人工驾驶机械,港口工作效率较低,系统的稳定性较差,建设自动化港口,研究在线集卡调度策略势在必行。清华大学蔡寒等^[3]提出了一种混合在线调度方法的框架,是基于“作业路”模式的调度,集卡的利用率较低,作业成本较高。很多研究以岸吊对集卡的总等待时间为主要优化目标^[5-7],而没有考虑集卡的空载率以及集卡与任务地点的距离问题。

本研究将以集卡为中心,在“大作业面”前提下,综合考虑集卡执行完当前任务所需的时间和岸桥延迟时间,以及执行完当前任务后集卡前往下一个任务地点的行驶时间,建立多目标优化模型,并提出一种混合在线集卡调度算法。最后,通过仿真实验验证提出的模型和算法的有效性。

1 问题描述

集装箱港口卸载作业流程可以简单描述为:为进港船舶分配泊位,由岸桥将集装箱从船舶卸到集卡,再由集卡将集装箱运输到相应堆场,然后场桥将集装箱卸下放置在堆场。集装箱装船作业流程恰好相反。集装箱港口的在线调度策略指根据港口实时数据,对各设备及场区的状态做出及时判断,然后给出实时调度指令。

在港口装卸作业过程中,集卡的调度策略主要有两种^[8]:“作业路”模式和“大作业面”模式。两种模式集卡调度路线分别如图 1 和图 2 所示。

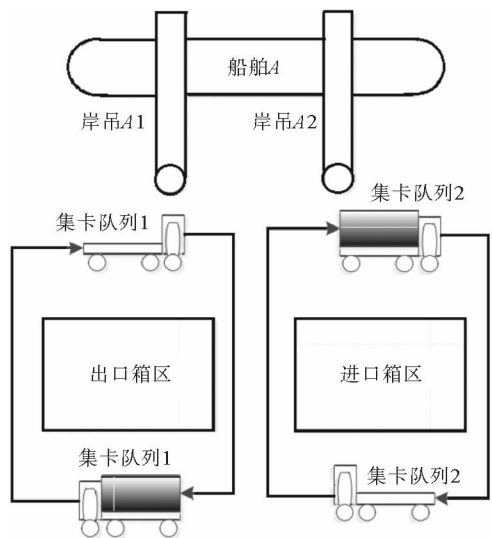


图 1 “作业路”模式调度路线

Fig. 1 “Working-line” mode schedule route

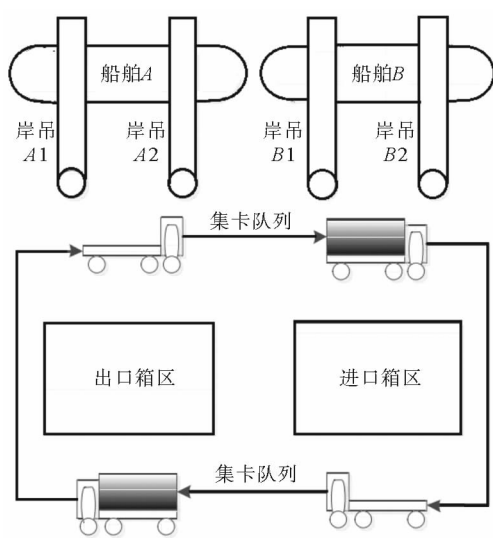


图 2 “大作业面”模式调度路线

Fig. 2 “Full working-area” mode schedule route

在“作业路”模式中,集卡队列 1 在出口箱区装载集装箱,运输到岸吊 A1 处,由岸吊 A1 卸载集装箱到船舶 A,然后集卡空载回到出口箱区,循环此过程;集卡队列 2 在岸吊 A2 处装载集装箱,运输到进口箱区,由对应的场桥将集装箱卸载,然后集卡回到岸吊 A2 处,循环此过程。此模式下每辆集卡固定为某一岸吊服务,行驶路线也相对固定,在集卡循环行驶过程中,一半的路程为空载,造成了集卡资源的浪费。在图 2“大作业面”模式中,集卡灵活地服务于所有岸吊,集卡可以在岸吊 B1 或 B2 处装载集装箱,运输到进口箱区,卸载集装箱后,行驶到出口箱区装载集装箱,然后运输到岸吊 A1 或 A2 处,由岸吊卸载集装箱到船舶 A。此模式集卡行驶路线相对灵活,集卡的空载率相对较低。

2 集卡调度的数学模型

综合考虑系统总作业时间、岸吊的操作及等待时间、集卡空载时间问题以及任务地点与集卡的距离问题,建立混合在线集卡调度的多目标优化的数学模型。

2.1 模型假设与参数定义

根据港口集装箱调度的实际情况,结合有关文献中的场景设计^[2,3,9-10]及模型设计实际,给出下面的模型假设:

- 1) 所有船计划和堆场计划已知;
 - 2) 所有岸吊和场桥的装(卸)载操作时间及设备调整时间已知;
 - 3) 各任务地点之间的行驶时间已知;
 - 4) 所有岸吊和场桥的装卸箱任务序列已知;
 - 5) 集装箱为单一箱型,一辆集卡一次运输 1 TEU(Twenty-foot Equivalent Unit, 20 英尺标准箱)。
- 所涉及的模型与算法的参数定义如表 1 所示。

2.2 目标函数

基于 2.1 节的假设,考虑岸吊延迟时间、集卡行驶时间以及系统完成时间最短,建立多优化目标函数^[11-13]。

若同一岸吊当前执行的任务已经完成,可执行下一任务,但集卡未到就造成了岸吊等待,影响作业效率及经济成本。考虑岸吊的等待时间,得到目标函数:

$$f_1 = \sum_{i=1}^n \sum_{j=2}^{m_i} (t_{i,j}^{arr} - t_{i,j-1}^{dep})。$$
 (1)

减少集卡行驶路程可以降低集卡空驶率,考虑集卡位置与任务地点之间的空载行驶时间,得到目标函数

$$f_2 = \sum_{k=1}^n \left[d(s_{k,0}^{(0)}, s_{k,r}^{(2)}) + \sum_{r=2}^{n_k^R} d(s_{k,r-1}^{(1)}, s_{k,r}^{(2)}) \right]。$$
 (2)

在实际港口中,系统总作业完成时间直接反映港口的工作效率,考虑系统总作业完成时间,建立目标函数

$$f_3 = T = t - t_0。$$
 (3)

为了便于模型实现及算法比较,采用线性加权法将以上三个目标进行综合,式中各系数经相关专家验证得出。实际作业中,总是对系统总作业时间最为重视,所以设置 f_3 权重最大为 0.5, f_1 和 f_2 分别为 0.35 和 0.15。实际系统包含的岸吊数量是可以改变的,其数量也直接影响目标函数 f_1 值,因此,用 f_1 除以岸吊的数量 n^A ,取岸吊的平均延迟时间 $\frac{f_1}{n^A}$ 。同理,取集卡的平均空载行驶

表 1 参数定义

Tab.1 Parameter definition

符号	含义
i	工作点编号
j	某工作点的任务编号
k	集卡编号
r	某集卡的任务编号
n	为工作点数量
m_i	为工作点 i 的任务数量
n^K	集卡数量
n_k^R	集卡 k 的总任务数量
n^A	岸吊数量
t	当前时刻
t_0	系统开始时刻
$t_{i,j}^{arr}$	集卡到达工作点 i 执行任务 j 的时刻
$t_{i,j}^{dep}$	集卡离开工作点 i 执行任务 j 的时刻
t_i^h	工作点 i 的设备操作时间,即装(卸)车时间
t_j^h	集卡离开工作点 i 执行任务 j 的时刻
$s_{k,0}^{(0)}$	集卡 k 在停车场的位置
$s_{k,r}^{(1)}$	集卡 k 的第 r 个任务地点的位置
$s_{k,r}^{(2)}$	集卡 k 的第 r 个任务目的地的位置
$d(s_1, s_2)$	s_1 位置与 s_2 位置之间的行驶时间
L_i	任务地点的任务列表 L_i
T	系统任务完成花费时间
$A_{i,j} \in \{0,1\}$	表示工作点 i 的任务 j 是否被认领,未被认领则为 1,否则为 0
$B_{k,r,i,j} \in \{0,1\}$	工作点 i 的任务 j 是否被集卡 k 执行的第 r 个任务执行完毕,未执行完毕则为 1,否则为 0
$C_k \in \{0,1\}$	集卡 k 是否处于休眠状态,休眠状态则为 1,否则为 0

时间 $\frac{f_2}{n^K}$ 。最后,得到总目标函数:

$$F = 0.35 \frac{f_1}{n^A} + 0.15 \frac{f_2}{n^K} + 0.5 f_3. \quad (4)$$

约束条件:

$$T \geq \max \{ t_{i,m_i}^{dep} + d(s_{i,m_i}^{(1)}, s_{i,m_i}^{(2)}) - t_0 \mid i \in \{1, 2, \dots, n\} \}; \quad (5)$$

$$t_{i,j}^{dep} \geq t_{i,j}^{arr} + t_i^{ch}, \forall i = 1, 2, \dots, n, j \in \{1, 2, \dots, m_i\}; \quad (6)$$

$$t_{i,j}^{arr} \geq t_{i,(j-1)}^{arr} + t_i^{ch}, \forall i = 1, 2, \dots, n, j \in \{1, 2, \dots, m_i\}; \quad (7)$$

$$C_k \cdot \sum_{i=1}^n \sum_{j=1}^{m_i} A_{i,j} = 0, \forall k = 1, 2, \dots, n^K; \quad (8)$$

$$C_k \cdot \sum_{r=1}^{n^K} B_{k,r,i,j} = 0, \forall i = 1, 2, \dots, n, j \in \{1, 2, \dots, m_i\}. \quad (9)$$

其中:式(5)表明系统总完成时间大于等于最后一个任务执行完毕的时间减去系统开始时间;式(6)表明同一任务的执行集卡离开时间与抵达时间至少间隔一个设备操作时间;式(7)表明同一工作点的相邻任务的执行集卡抵达时间至少间隔一个设备操作时间;式(8)表明集卡在所有工作点的任务都被其他集卡认领完毕后才可休眠;式(9)表明集卡只有完成所有其响应的任务后才可以休眠。

本节建立的混合在线集卡调度多目标优化模型,考虑了多个影响与评价集装箱港口作业效率和作业成本的因素,可用来评价港口的作业能力、港口调度算法的优劣。

3 混合在线集卡调度算法设计

集卡在线调度方法根据触发事件的不同,一般分为任务触发型和集卡触发型^[3]两种。任务触发型是指当有新任务产生而未被指派出去时,调用算法寻找合适集卡去完成。集卡触发型是指在集卡空闲时,调用算法寻找合适任务去执行。任务触发型能使每个任务都能选用最合适的集卡来执行,但是容易造成远处作业的集卡利用率低。集卡触发型能使每辆集卡均衡使用,提高集卡利用率,但容易造成岸吊或场桥等待时间过长。

在实际港口中希望集卡等待任务,而不是任务等待集卡,所以本研究以集卡触发型为基础,结合任务触发型策略的优点,运用启发式方法,建立基于“大作业面”的混合在线集卡调度算法。该算法每次执行都会遍历所有工作点,选取工作点的当前第一个未被集卡认领的任务,然后计算目标评价函数(函数所用某些变量定义已在第 2 节给出),选取合适的任务。考虑到设备的操作时间远大于算法的执行时间,为了最大限度减少任务等待时间和集卡等待时间,降低集卡空驶率,算法的主体 Select()函数是在集卡的当前任务结束前执行(即集卡卸载当前任务集装箱的同时,为其选好下一任务地点)。算法 1 描述了基于“大作业面”的混合在线集卡调度算法的基本步骤。

算法 1 基于“大作业面”的混合在线集卡调度算法

Input: 任务地点初始任务列表 L_i , 初始化集卡编号 k 。

Output: 目标函数 f_1, f_2, f_3 的值, 总目标函数值 F 。

Step1: 集卡 k 开始工作 $C_k = 0, r = 0, B_{k,r,i,j} = 1$; /* 初始化, 集卡 k 第 0 个任务执行完毕 */

Step2: **For each** r **do** /* 对每一任务 r 执行以下操作 */

Step2.1: 调用 Select() 函数; /* Select 函数在下文定义 */

Step2.2: **If** $B_{k,(r-1),i,j} = 0$ **then** Wait($B_{k,(r-1),i,j}$); /* 集卡 k 的第 $r-1$ 个任务未完成, 则等待 */

Step2.3: 集卡 k 前往任务地点 i ;

Step2.4: 记录当前时刻 $t_{i,j}^{arr}$, 执行任务 j , 记录当前时刻 $t_{i,j}^{dep}$; /* 约束条件(5)(6) */

续表

算法 1 基于“大作业面”的混合在线集卡调度算法

Step2.5:将集装箱运至任务目的地,对应场桥或岸桥卸载集装箱;
Step2.6: $B_{k,r,i,j}=1,r++$; /* 集卡 k 的第 r 个任务完成,保证约束条件(8) */
Step2.7: $f_{k,r}^2+=f_{k,r}^2$; /* 每一个集卡的 $f_{k,r}^2$ 累加, $f_{k,r}^2$ 由函数 Select() 计算得出 */
Step2:Endfor /* 系统所有任务都被执行完毕 */
Step3:For($i=0;i>n;i++$) do
Step3.1:For($j=0;j>m;j++$) do $f_1+=f_{1,j}^1$; Endfor /* 每一个任务的 $f_{1,j}^1$ 累加, $f_{1,j}^1$ 由函数 Select() 计算得出 */
Step3:Endfor /* for 循环结束,得出目标函数 f_1 的值 */
Step4:集卡 k 驶入停车场,进入休眠状态 $C_k=1$; /* 保证约束条件(4)(8) */
Step5:For($k=0;k>n^K;k++$) do /* 对每一集卡执行以下操作 */
Step5.1:If $C_k!=1$ then Wait(C_k); /* 等待集卡执行完任务,进入休眠状态,保证约束条件(4) */
Step5.2: $f_2+=f_k^2$; /* 所有集卡的 f_k^2 累加,得计算目标函数 f_2 的值 */
Step5:Endfor /* for 循环结束,得出目标函数 f_2 的值 */
Step6:记录系统完成时间 $T=t-t_0$; /* 根据目标函数 f_3 得出 */
Step7:计算总目标函数 $F=0.35\frac{f_1}{n^A}+0.15\frac{f_2}{n^K}+0.5T$; /* 整合三个目标函数 */ END /* 算法结束 */

算法 1 的核心是计算目标函数寻找集卡下一个任务地点,将其单独编写为描述计算目标函数基本步骤的 Select() 函数

函数 Select()

Input: 各任务点的当前任务列表 L_i ,各地点之间的行驶时间 $d(s_1,s_2)$ 。
Output: 下一任务地点 i 。
Step1:For each $i\{1,2,\cdots,n\}$ do /* 对每一地点 i 执行以下操作 */
Step1.1:If 任务列表 L_i 为空 then 循环 $i++$,直到任务列表 L_i 不为空;Endif
Step1.2:取当前任务 j ;
Step1.3:If $A_{i,j}=0$ then /* 当前任务 j 已被其他集卡认领 */
Step1.3.2:For($j;A_{i,j}=0;j++$) do /* 对每一个已认领的任务 j 执行以下操作 */
Step1.3.2.1:If $j>m$ then $i++$,gotoStep1;Endif /* 地点 i 的所有任务都已被认领 */
Step1.3:Endif
Step1.4:根据目标函数 f_1 ,计算岸吊等待时间
$$f_{i,j}^1=[t_i^{\phi}-(t-t_{i,j-1}^{wx})]\cdot B_{k,r,i,j-1}+(t-t_{i,j-1}^{dp})(1-B_{k,r,i,j-1});$$

Step1.5:根据目标函数 f_2 ,计算集卡行驶时间
$$f_{k,r}^2=d(s_{k,i}^{(1)},s_{k,r-1}^{(2)})\cdot B_{k,r,i,j}+d(s_{k,0}^{(0)},s_{k,i}^{(1)})\cdot C_k;$$

Step1.6:计算评价函数 $f_{k,i}^{2,i}=0.35f_{1,j}^1+0.15f_{k,r}^2$; /* 通过优化岸吊等待时间和集卡行驶时间,来减少系统总作业时间,先对目标函数 f_1 和 f_2 整合 */
Step1:Endfor /* 遍历完所有任务地点 */
Step2:选择评价函数 $f_{k,i}^{2,i}$ 的值最小的任务地点;
Step3:记录下一任务地点 $S_{k,r}=i$,记录下一任务地点的此刻岸吊等待时间为 $f_{i,j}^1$,记录下一任务地点的集卡行驶时间为 $f_{k,r}^2$; /* 为计算目标函数 f_1 、 f_2 做准备 */
Step4: $A_{i,j}=0,B_{k,r,i,j}=0$; /* 标记地点 i 的任务 j 被认领,设置集卡 k 的任务 r 未完成 */
END /* 函数 Select() 结束 */

算法 1 中设计的任务以集装箱为中心,指卸船集装箱从离船到被运至堆场箱区安放(或装船集装箱从离开箱区到安放至货船)的全过程。算法核心是判断任务执行状态,并准确计算目标评价函数值,选择集卡的下一任务地点。与基于“作业路”的在线集卡调度算法^[3](以下简称“算法 2”)相比,算法 1 以“大作业面”为基础,扩大了集卡的服务范围,综合考虑了系统作业时间、岸吊等待时间以及集卡空载行驶时间。

4 仿真实验及结果分析

为了更好地验证算法 1 的有效性,通过 FlexTerm 软件进行仿真,运用第 2 节的目标函数,对算法 1 和算法 2 分别进行比较分析。

4.1 港口基本情况及案例设计

以青岛某集装箱港口为例,每个泊位有 2 台岸吊为其服务(1~2 号岸吊服务泊位 1,3~4 号岸吊服务泊位 2),泊位后方有 4 个堆场共 16 个箱区(1~4 号箱区为出口堆场 1,5~8 号箱区为进口堆场 1,9~12 号箱区为出口堆场 2,13~16 号箱区为进口堆场 2),每个箱区有 1 台场桥。岸吊操作一个集装箱所需时间为 1.5 min,场桥操作一个集装箱所需时间为 2 min,集卡平均行驶速度为 18 km/h。两泊位与箱区的行驶距离如表 2 所示,各箱区间的行驶距离如表 3 所示(只列出本实验用到的数据)。

表 2 箱区与泊位间的行驶距离
Tab. 2 The travel distance between the blocks and the berths m

箱区	1	2	3	4	13	14	15	16
泊位 1	150	150	200	200	470	660	520	720
泊位 2	560	370	610	420	250	250	300	300

表 3 各箱区间的行驶距离
Tab. 3 The travel distance between the blocks m

箱区	1	2	3	4	13	14	15	16
1	0	190	50	240	510	710	560	760
2	190	0	240	50	320	520	370	570
3	50	240	0	190	460	660	510	710
4	240	50	190	0	270	570	320	620
13	510	320	460	270	0	200	50	250
14	710	520	660	570	200	0	250	50
15	560	370	510	320	50	250	0	200
16	760	570	710	620	250	50	200	0

为验证本文模型和算法的有效性,结合港口实际,取 3 个不同规模的案例进行模拟仿真。

案例 1:某时刻泊位 1 停靠一艘货轮需卸载进口集装箱 2 000 个,2 h 后泊位 2 停靠一艘货轮需装载出口集装箱 2 000 个。

案例 2:某时刻泊位 1 停靠一艘货轮需卸载进口集装箱 1 000 个,同时泊位 2 停靠一艘货轮需装载出口集装箱 800 个。

案例 3:某时刻泊位 1 停靠一艘货轮需卸载出口集装箱 5 000 个,2 h 后泊位 1 停靠一艘货轮需装载进口集装箱 3 000 个。

4.2 实验结果与分析

根据以上数据,用 FlexTerm 仿真软件,分别嵌入算法 1 和算法 2 对模型进行求解,软件仿真截图如图 3 所示。

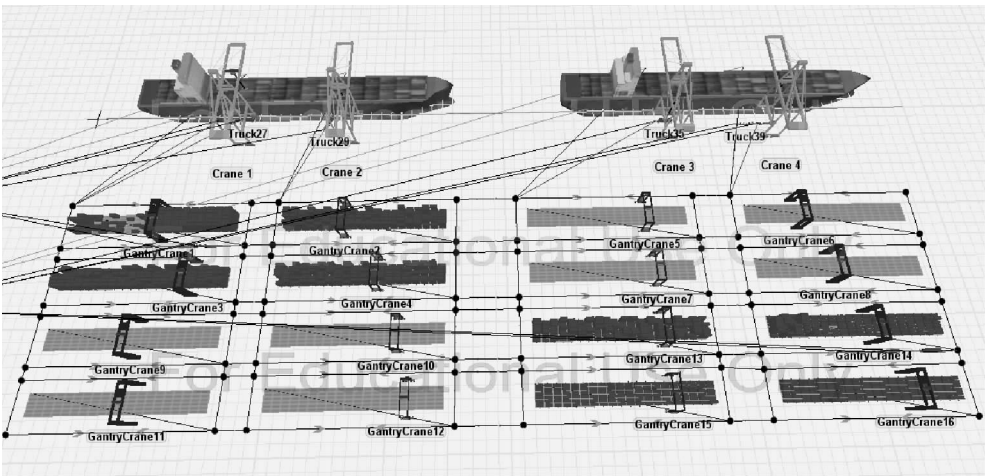


图 3 软件仿真截图
Fig. 3 The print-screen of software simulation

根据该港口基本情况,设定岸吊的数量 n^A 为 4。根据相关专家经验及相关文献^[8-9,14],得出在系统配置 9 辆集卡时效率较高。因此,取系统集卡的数量 n^K 为 9。

通过仿真,得到不同案例中基于算法 1 和算法 2 的各目标函数值(如表 4)。

表 4 不同案例中两种算法运行结果

Tab. 4 The results of the two algorithms running in different cases

案例/算法		岸吊等待时间 (f_1)	集卡空载行驶 时间(f_2)	系统总作业完成 时间(f_3)	总目标函数值 (F)
案例 1	算法 1	1 142	38 791	958	1 225.44
	算法 2	1 476	44 372	1 214	1 475.68
案例 2	算法 1	845	26 341	663	844.45
	算法 2	1 104	29 867	859	1 023.88
案例 3	算法 1	2 328	72 253	1 858	2 336.92
	算法 2	2 957	83 042	2 427	2 856.27

表 5 给出了算法 1 与算法 2 在各目标函数指标上的比较,其中, f_1 、 f_2 、 f_3 、 F 表示算法 1 得出的各指标值, f_1^* 、 f_2^* 、 f_3^* 、 F^* 表示算法 2 得出的各指标值。 $(f_1^* - f_1)/f_1^*$ 、 $(f_2^* - f_2)/f_2^*$ 、 $(f_3^* - f_3)/f_3^*$ 、 $(F^* - F)/F^*$ 表示算法 1 得出的各指标相对算法 2 的优化程度^[15]。由表 5 可以看出,所提出的算法 1 在岸吊等待时

表 5 不同案例中算法优化效果

Tab. 5 Algorithm optimization in different cases %

案例	优化效果			
	$(f_1^* - f_1)/f_1^*$	$(f_2^* - f_2)/f_2^*$	$(f_3^* - f_3)/f_3^*$	$(F^* - F)/F^*$
案例 1	22.6	12.6	21.1	17.0
案例 2	23.5	11.8	22.8	17.5
案例 3	21.3	13.0	23.4	18.2
平均	22.5	12.5	22.4	17.6

间和系统总作业时间方面,优化效果较好,对集卡空载行驶时间的优化次之,其中,岸吊的等待时间平均减少 22.5%,系统总作业时间平均减少了 22.4%,集卡的空载行驶时间平均减少 12.5%。综合以上各指标,总目标函数 F 优化效果也较好,综合指标平均降低了 17.6%。

由以上结果分析可知,算法 1 与基于“作业路”的在线集卡调度算法(算法 2)相比优化效果较好,具有良好的实用性。

5 总结

综合考虑系统作业完成时间、岸吊等待时间(即利用率问题)、集卡空载行驶时间(即空载率问题)等因素,建立了多目标优化模型,并结合传统的任务触发型集卡调度算法和集卡触发型集卡调度算法,对基于“作业路”的在线集卡调度算法进行改进,提出一种基于“大作业面”的混合在线集卡调度算法,且通过实验仿真验证了算法的有效性。除了本研究所考虑的上述 3 个因素外,实际港口运作中,岸桥分配、船舶的在港时间、港口前后方作业的衔接等,也是影响港口成本的重要因素,将作为下一步的研究内容。

参考文献:

- [1]谢晨,梁承姬,徐德洪.基于 HFSS 的集装箱码头集成调度模型的建立与求解[J].华中师范大学学报(自然科学版),2016,50(3):369-375.
- XIE Chen,LIANG Chengji,XU Dehong. A integrated scheduling model based on the HFSS for container terminal[J]. Journal of Central China Normal University (Natural Science),2016,50(3):369-375.
- [2]CAO J X,SHI Q X,LEE D H. Integrated quay crane and yard truck schedule problem in container terminals[J]. Tsinghua Science and Technology,2010,15(4):467-474.
- [3]蔡寒.基于仿真的集装箱码头拖车调度研究[D].北京:清华大学,2007.
- [4]KIM K H,KIM K Y. An optimal routing algorithm for a transfer crane in port container terminals[J]. Transportation Science,1999,33(1):17-33.
- [5]CHENG Y L,SEN H C,NATARAJAN K,et al. Dispatching automated guided vehicles in a mega container terminal[M]. Springer US,2005,98:355-389.
- [6]MEISEL F,BIERWIRTH C. A unified approach for the evaluation of quay crane scheduling models and algorithms[J]. Computers & Operations Research,2011,38(3):683-693.
- [7]田亮.基于同步装卸的岸桥与集卡协同作业优化研究[D].大连:大连海事大学,2013.
- [8]吴名建.港口集装箱拖车调度优化研究[D].南京:南京航空航天大学,2010.
- [9]马慧娟,杜玉越.一种集装箱港口集卡的动态调度方法[J].山东科技大学学报(自然科学版),2016,35(4):99-105.
- MA Huijuan,DU Yuyue. A Dynamic scheduling method of truck in container terminal[J]. Journal of Shandong University of Science and Technology (Natural Science),2016,35(4):99-105.
- [10]梁承姬,翟点点,王丹丹.港口集装箱多场桥装卸动态调度模型研究[J].计算机仿真,2016(3):363-366.
- LIANG Chengji,ZHAI Diandian,WANG Dandan. Dynamic scheduling model research for multiple yard crane in container terminal[J]. Computer Simulation,2016(3):363-366.
- [11]黄发良,张师超,朱晓峰.基于多目标优化的网络社区发现方法[J].软件学报,2013,24(9):2062-2077.
- HUANG Faliang,ZHANG Shicao,ZHU Xiaofeng. Discovering network community based on multi-objective optimization [J]. Journal of Software,2013,24(9):2062-2077.
- [12]CAI B,HUANG S,LIU D,et al. Multiobjective optimization for autonomous straddle carrier scheduling at automated container terminals[J]. IEEE Transaction on Automation Science and Engineering,2013,10(3):711-725.
- [13]CARLO H J,VIS I F A,ROODBERGEN K J. Storage yard operations in container terminals:Literature overview,trends, and research directions[J]. Flexible Services and Manufacturing Journal,2015,27(2):224-262.
- [14]杜振,宫会丽,张蕾,等.港口物流集装箱调度效率优化仿真研究[J].计算机仿真,2016,33(1):340-343.
- DU Zhen,GONG Huili,ZHANG Lei,et al. Simulation study on container scheduling optimization in port logistics efficiency [J]. Computer Simulation,2016,33(1):340-343.
- [15]徐亚,陈秋双,龙磊等.基于多目标规划的堆场空间分配问题研究[J].系统工程学报,2009,24(3):365-369.
- XU Ya,CHEN Qiushaung,LONG Lei,et al. Yard space allocation based on multi-objective programming[J]. Journal of Systems Engineering,2009,24(3):365-369.

(责任编辑:傅 游)