

集装箱码头装卸系统集成调度的建模与优化

梁 亮, 陆志强

(上海交通大学 工业工程与物流工程系, 上海 200240)

摘 要 为提高集装箱码头作业效率, 在码头集装箱卡车作业能力充足的条件下, 对集装箱码头作业装卸系统集成调度问题进行了研究. 针对集装箱码头装卸作业的特点, 提出了装卸作业的预定义顺序约束、作业次序依赖的设备调整时间及岸桥碰撞干涉约束等, 并在此基础上建立了整数规划模型. 针对这一模型设计了两阶段启发式算法对问题进行求解. 通过在数值实验中将结果与低界相比较, 证明该算法效果良好. 特别对于规模较大的问题, 算法也可以进行有效的求解.

关键词 物流工程; 集装箱装卸系统; 集成调度; 启发式算法

Modeling and optimization of the integrated scheduling problem of container handling system

LIANG Liang, LU Zhi-qiang

(Department of Industrial Engineering & Logistics Management, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract On the assumption that the capacity of inner trucks was sufficient, the integrated scheduling problem of the container handling system was studied to improve the operation efficiency of container terminals. Based on the characteristics of container terminal operations, pre-defined operation constraints, sequence-dependent setup time and interference constraints for quay cranes were defined and an integer programming model was proposed for the problem. A two-stage heuristic algorithm was developed to solve the problem. The numerical experiments show the algorithm effective and efficient comparing to the lower bound, particularly for cases with large scales.

Keywords logistic engineering, container handling system; integrated scheduling; heuristic algorithm

1 引言

随着国际间贸易日益活跃, 集装箱化运输逐渐成为世界范围内主要的物流运输手段. 与之相适应, 集装箱港口也迫切需要提高其集装箱作业能力, 以形成流畅的集装箱进出港运输链. 从世界范围来看, 码头的运作效率很大程度上取决于码头装卸系统的作业效率^[1], 对集装箱码头装卸设备调度进行优化, 从而缩短对船作业时间, 一方面可以减少集装箱船舶的滞港时间, 提高码头的作业效率和竞争力; 另一方面可以提高集装箱船舶的周转率, 有助于船公司降低运营成本. 因而良好的调度计划对集装箱码头而言至关重要.

集装箱码头作为连接不同运输模式的枢纽节点, 是通过码头装卸系统的搬运作业来完成集装箱的转运功能的, 集装箱码头装卸通常由岸边吊桥 (Quay cranes)、集装箱卡车 (Internal container trucks) 和场地吊车 (Yard cranes) 等主要作业设备, 以及码头泊位和集装箱堆场等硬件设施构成, 如图 1 所示.

收稿日期: 2008-12-01

资助项目: 国家自然科学基金 (70771065); 上海市浦江人才计划 (07PJ14052)

作者简介: 梁亮 (1985-), 硕士研究生, 主要研究方向为集装箱码头作业系统调度优化; 陆志强, 男, 工学博士, 副教授, E-mail: zhiqianglu@sjtu.edu.cn.

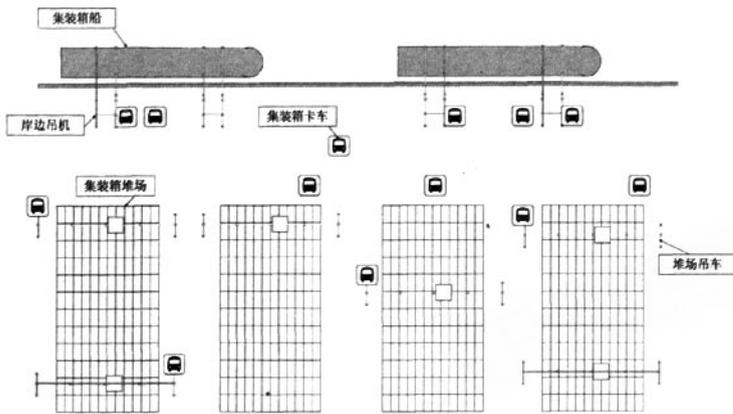


图 1 集装箱码头布局示意图

通常可以认为: 码头装卸系统调度问题是多阶段平行设备 Flow shop 生产调度基本问题(又称混合 Flow shop 问题)的延伸和扩展. 考虑到集装箱码头装卸系统调度问题所涉及到的研究领域, 本文就如下两个方面分别进行综述: 1) 关于集装箱码头装卸设备调度的研究; 2) 设备调整时间依赖于作业序列的柔性流水车间调度问题.

1.1 集装箱码头装卸系统调度

截至目前为止, 国内外对于集装箱码头装卸系统的调度决策优化已经有了一定的研究, 但是多数研究都是围绕装卸系统中的一个阶段设备的作业优化问题而展开的. 在国际上, 对岸边吊桥的作业进行优化的研究如文献 [2-3], 等, 对集装箱卡车调度进行优化的如 [4-5], 等, 以及主要针对场地吊车进行调度优化的研究, 如 [6], 等. 目前将三个阶段作业作为一个整体进行优化的研究比较有限. Chen 等^[7] 对集装箱码头三阶段的整体操作进行了建模, 并且提出了一个禁忌搜索算法求解. Lau 等^[8] 研究了在装卸一体化情况下, 码头装卸设备操作的若干特点, 针对最小化 AGV 总移动距离, 最小化 AYC 总移动距离和最小化岸边吊桥延迟时间的多目标决策, 提出了多层次遗传算法 (MLGA) 和遗传-最大配对算法 (GAPM), 试验证明后者表现更加优于前者. 但是 [7-8] 均没有考虑岸边吊桥的碰撞约束, 而且上述文献所提出的算法都难以用于大规模问题的求解, 例如 Chen 等^[7] 提出的禁忌搜索算法适用于求解不超过 100 集装箱的问题, 而 [8] 中的 GAPM 算法所适用问题的规模也不超过 1000 个集装箱.

在国内, 对集装箱码头调度问题的研究中, 大多数采用系统仿真的手段对码头装卸设备的配置和调度策略进行优化, 如张姝妹^[9], 张海霖等^[10]; 相对而言使用数学规划的手段对集装箱码头装卸设备整体调度决策进行优化(即本文所采用的手段)的研究较少. 其中, 周鹏飞等^[11] 就集装箱码头调度中存在的确定性建立了泊位-岸桥分配模型并且设计了遗传算法加以优化, 但并未考虑码头装卸系统调度的后续阶段, 如集卡和场吊等; 曾庆成等^[12] 将集装箱码头调度问题划分为装卸桥和集卡两个层次, 建立了对应的双层数学模型并设计了遗传算法对该模型依次进行求解, 证明算法在减少岸桥和集卡的空载时间(设备调整时间)上比较有效. 该算法在求解时, 将多个集装箱集成为一个任务加以处理, 但是文中并未阐明此种处理方式的依据和将多个集装箱集成为一个任务的操作方法.

1.2 设备调整时间依赖于作业序列的柔性流水车间 (Flexible flow shop) 调度问题

集装箱装卸系统调度问题可看作是常规流水车间 (Flow shop) 调度问题的扩展, 考虑到装卸设备具有的设备调整时间依赖于作业序列的性质, 可将集装箱装卸设备调度问题视为具有作业序列相关的设备调整时间的柔性流水车间 (Flexible flow shop) 调度问题 (FFm|ST_{sd}). 针对该类问题, Liu^[13] 研究了目标函数为最小化设备调整时间(成本)总和的 FFm|ST_{sd}, SC_{sd}, r_j 问题. 他们将这类问题建模为可分离的整数规划问题, 然后采用拉格朗日松弛进行了求解, 并且设计了搜索启发式算法. Kurz 等^[14] 研究了 FFm|ST_{sd}|C_{max} 问题, 但他们假设其中一些工件不需要经过所有的 m 个阶段, 提出了三个启发式算法以求解该问题, 分别为插值算法, 基于 Johnson 规则的启发式算法和贪婪启发式算法, 其中基于 Johnson 规则的启发式算法效果较好. 进一步的, Kurz 等^[15] 比较了解决该类问题的四种启发式算法, 并计算了问题的下界, 利用下界对四种算法进行了评价. Zandieh 等^[16] 设计了一个免疫算法, 并认为该算法优于文献 [15] 中所提出的各种启发式算法算

法。但是,上述研究均从生产调度的角度出发,并没有考虑码头作业的特殊约束。同时,以上研究涉及的生产调度问题的规模也远远小于集装箱码头装卸设备调度问题。

2 整数规划模型

2.1 问题描述

码头的装卸作业从集装箱船舶进入指定泊位之后开始,集装箱搬运系统需要在计划期内完成进、出港集装箱的搬运作业。其中涉及到由岸边吊桥、集装箱卡车以及场地吊车所构成的一个多阶段一体化复杂操作系统,而每一个阶段又包含多个平行设备,其作业对象是靠泊船舶所有要求装/卸的集装箱集合,且装/卸作业都有其固定的流程(卸载作业:岸边卸载-集卡运输-场地吊放;装载作业:场地吊放-集卡运输-岸边装载)。可以认为,码头装卸系统调度问题是多阶段平行设备 Flow shop 生产调度基本问题(又称混合 Flow shop 问题)的延伸和扩展。不同于一般的混合 Flow shop 问题,码头装卸系统调度问题具有如下特点:

1) 设备可以移动且必须移动到指定位置之后才能够进行操作。而设备作业的调整时间取决于该设备作业任务安排的序列。其中设备作业的调整时间指的是在设备在结束上一个装卸作业之后到下一个装卸作业开始之间的设备移动时间;

2) 装卸系统的作业任务(Job,码头上一个集装箱的装卸作业定义为一个作业任务)之间存在一部分先后顺序限制,我们称之为预定义顺序约束集合。包括同一船舶作业区域(Bay,贝)中,集装箱卸载必须先于装载;同一舱盖板的甲板上集装箱和船舱内集装箱存在操作顺序约束,以及其他根据配载计划所确定的约束等。

本文主要研究在单装/卸情况下,具有以上两个作业特点的码头集装箱装卸系统的集成调度问题。同时,本文研究的码头搬运系统三个阶段设备中,集装箱卡车所组成的搬运系统不是整个搬运系统的瓶颈。

2.2 数学符号定义

定义一个集装箱作业任务(Job)为完成卸载或装载一个集装箱的过程。装/卸载一个集装箱必须依次经过三个阶段设备的操作,例如:对于一个卸载型的任务,上述操作分别是岸边吊机将集装箱从船上卸至集装箱卡车、集装箱卡车将集装箱运至场地吊车处和场地吊车将集装箱在堆场吊装。规定 k 为操作阶段索引, $k = 1, 2, 3$, 依次表示卸载集装箱所经过的岸桥作业阶段,集卡作业阶段和场吊作业阶段。对于一个装载型的任务, $k = 1, 2, 3$, 依次表示场吊阶段,集卡阶段和岸桥阶段。

因此,本文所研究的问题为一个三阶段混合 Flow shop 调度问题,集装箱作业任务(Job)集合为 $J = 1, 2, \dots, J; j \in J$ 为集装箱作业任务(Job)索引,规定 $j = 0$ 为虚拟集装箱作业任务,为所有设备作业的第一个任务; J^S 为所有集装箱作业任务的开始时间必须满足的预定义顺序约束集合,如果有 $(j, j') \in J^S$, 则表示预定义顺序约束要求集装箱 j 必须在集装箱 j' 之前开始装/卸。 J^S 包括的预定义顺序约束有如下几类:

- 1、在进行卸载作业时,为避免翻箱作业,任何集装箱必须先于堆放在其下方的集装箱开始作业;
- 2、船舱内的装/卸作业必须在其上方甲板上所有卸载作业完成后才可以开始;
- 3、甲板上方的装载作业必须在对应船舱内的装/卸作业完成后才可以开始;
- 4、在进行装载作业时,预配载计划中安排位置在下方的集装箱任务必须先于在其上方的集装箱任务开始。

M 为所有设备集合; $i \in M$ 为设备索引; M_k 为第 k 阶段设备集合, $k = 1, 2, 3$ 分别表示岸边吊桥、集卡和场地吊车设备子集。其中,岸边吊桥按照位置依次从小到大进行编号,且与集装箱船上贝的编号按照相同方向递增。集装箱 j 在第 k 阶段设备上的操作为 $O_{j,k}$, 规定操作 $O_{j,k}$ 的处理时间为 $p_{j,k}$, 为该设备的满载时间 ($p_{j,0} = 0$), 与对应集装箱任务 j 所处位置有关,其完成时刻为 $C_{j,k}$; 任一设备在连续作业时会产生相应的设备调整时间。规定该设备在连续作业之间的空载时间为该设备的设备调整时间 $s_{j,j',k}$ ($s_{0,j',k} = 0$), 对于岸桥和场地吊车而言,设备调整时间包括岸桥(场吊)小车空载移动的时间和岸桥(场吊)大车空载移动的时间。当岸桥和场地吊车完成某一贝的装/卸载作业,需要移动到其他贝进行作业时,所发生的设备调整时间就包括岸桥(场吊)大车空载移动时间。调度目标为 $\min C_{\max} = \max(C_{j,3})$ 。此外,我们规定 J_j^S 为操作必须在集装箱任务 j 之前完成的任务集合; B_j 为集装箱 j 在集装箱船上所在的贝的编号,且从船舶一端开始从小到大编号; H 为一个足够大的正数。

2.3 数学模型和决策变量

$x_{j,k,i}$ 1, 如果集装箱作业任务 j 是由处于阶段 k 的设备 i 完成的; 0, 其它;

$y_{j,j',k,i}$ 1, 如果集装箱作业任务 j 和集装箱作业任务 j' 都是由处于阶段 k 的设备 i 完成的; 0, 其它;

$z_{j,j',k,i}$ 1, 如果在阶段 k , 设备 i 上集装箱作业任务 j 在 j' 之前操作 (不一定连续); 0, 其它;

$u_{j,j',k,i}$ 1, 如果在阶段 k , 设备 i 上集装箱作业任务 j 为任务 j' 的紧前作业任务; 0, 其它;

$C_{j,k}$ 集装箱任务 j 在阶段 k 的操作完成时间, 规定 $C_{j,0} = 0$;

$v_{j,j',i,i'}$ 用来判断两个任务在岸吊吊机阶段是否同时操作的变量. 1, 当 $C_{j,1} \geq C_{j',1} \geq C_{j,1} - p_{j,1}$ 且同时 $x_{j,1,i} = x_{j',1,i'} = 1$; 0, 其它.

则集装箱码头装卸系统集成调度问题的整数规划模型如下:

$$\min C_{\max} = \max(C_{j,3}) \quad (1)$$

subject to:

$$\sum_{i \in M_k} x_{j,k,i} = 1, \forall j \in J, \forall k \in \{1, 2, 3\} \quad (2)$$

$$C_{j,k} - p_{j,k} \geq C_{j,k-1}, \forall j \in J, \forall k \in \{1, 2, 3\} \quad (3)$$

$$y_{j,j',k,i} \leq 0.5(x_{j,k,i} + x_{j',k,i}), \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (4)$$

$$0.5(x_{j,k,i} + x_{j',k,i}) \leq y_{j,j',k,i} + 0.5, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (5)$$

$$y_{j,j',k,i} = y_{j',j,k,i}, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (6)$$

$$z_{j,j',k,i} + z_{j',j,k,i} = y_{j',j,k,i}, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (7)$$

$$z_{j,j',k,i} - u_{j,j',k,i} \geq 0, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (8)$$

$$v_{j,j',i,i'} + y_{j,j',1,i} \leq 1, \forall j, j' \in J, \forall i, i' \in M_1 \quad (9)$$

$$C_{j,k} - (C_{j',k} + p_{j',k} + s_{j',j,k}) + (1 - u_{j,j',k,i})H \geq 0, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (10)$$

$$C_{j,k} - \left(C_{j',k} - p_{j',k} - \min_{j'' \in J} (s_{j'',j',k}) \right) + (1 - z_{j',j,k,i})H \geq 0, \forall j, j' \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (11)$$

$$C_{j,1} - C_{j',1} + (1 - v_{j,j',i,i'})H \geq 0, \forall j, j' \in J, \forall i \in M_1 \quad (12)$$

$$C_{j',1} - (C_{j,1} - p_{j,1}) + (1 - v_{j,j',i,i'})H \geq 0, \forall j, j' \in J, \forall i \in M_1 \quad (13)$$

$$x_{j,1,i} + x_{j',1,i'} \geq 2v_{j,j',i,i'}, \forall j, j' \in J, \forall i \in M_1 \quad (14)$$

$$\sum_{j' \in J} u_{j,j',k,i} \leq 1, \forall j \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (15)$$

$$\sum_{j' \in J} u_{j',j,k,i} \leq 1, \forall j \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (16)$$

$$x_{j,k,i} \geq 0.5 \left(\sum_{j' \in J} u_{j,j',k,i} + \sum_{j' \in J} u_{j',j,k,i} \right), \forall j \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (17)$$

$$0.5 \left(\sum_{j' \in J} u_{j,j',k,i} + \sum_{j' \in J} u_{j',j,k,i} \right) \geq x_{j,k,i} - 0.5, \forall j \in J, \forall k \in \{1, 2, 3\}, \forall i \in M_k \quad (18)$$

$$(C_{j,k+1} - p_{j,k+1}) + s_{j,j',k} \leq (C_{j',k} - p_{j',k}) + H(1 - u_{j,j',k,i}), \forall j, j' \in J, \forall k \in \{1, 2\}, \forall i \in M_k \quad (19)$$

$$C_{j,k} - p_{j,k} \leq C_{j',k} - p_{j',k}, \forall j, j' \in J^S, \forall k \in \{1, 2, 3\} \quad (20)$$

$$(1 - v_{j,j',i,i'})H \geq B_j - B_{j'} + 1, \forall B_j, B_{j'} \in B, \forall j, j' \in J, \forall i \geq i', i, i' \in M_1 \quad (21)$$

约束 (2) 保证每个任务的每一步操作都分别由且只能由对应阶段的一台设备进行. (3) 保证每个 Job 在各个阶段的操作顺序符合规定. (4), (5), (6) 规定了当 $x_{j,k,i} = x_{j',k,i} = 1$ 时, 有 $y_{j,j',k,i} = y_{j',j,k,i} = 1$. (7) 规定当 $y_{j,j',k,i} = 1$ 时, $z_{j,j',k,i}, z_{j',j,k,i}$ 中有一个变量为 1. (8) 式规定当两个任务 j 和 j' 在一台设备上连续操作时满足 $z_{j,j',k,i} = 1$. (9) 规定了一台岸吊上不能同时操作两个集装箱. (10), (11) 规定了 $C_{j,k}$ 与 $u_{j,j',k,i}$ 和

$z_{j,j',k,i}$ 之间的关系. (12), (13), (14) 则是对 $v_{j,j',i,i'}$ 的定义. 约束 (15), (16) 规定任意一个 Job 在每个阶段作业期间最多有一个紧前的 Job 和一个紧后的 Job. 约束 (17), (18) 的含义为, 如果一个 Job 被指派给某一阶段的一台设备进行作业, 则该 Job 不可能与指派给同一阶段其他设备的 Job 构成紧前或者紧后的作业关系 (约束 (17)); 反之, 两个 Job 在某一阶段的作业关系确定为紧前或者紧后, 则这两个 Job 必然是在同一台设备上作业的 (约束 (18)). 约束 (19) 反映了阻塞的情况, 即一台设备上的两个操作是连续的时候, 只有当前一个 Job 的后续操作开始后, 这台设备才能继续加工下一个 Job. 同时, 这两个约束同时避免了死锁情况的发生. 约束 (20) 规定所有的操作需要满足预定义的顺序. (21) 为岸吊阶段的碰撞约束, 其含义是: 在任何时刻, 如果有两台岸吊同时进行作业, 那么编号较小的岸吊所处的贝编号也较小, 而且两台岸吊的作业间距至少为 1 个贝宽 (安全距离). 由于岸吊和船舶的贝都是按照相同方向从大到小进行编号的, 因此岸吊之间不会发生碰撞.

3 基于 Johnson 法则的启发式算法 (JLS 算法)

由于装载作业与卸载作业互为逆向作业, 在解决单卸作业调度问题的前提下, 单装作业调度问题也可以通过转化为一个单卸作业调度问题加以解决. 转换的具体方法是: 首先为所有的待装载的集装箱构造唯一对应的假想集装箱, 将假想的集装箱按照预配载计划布置在集装箱船舶上, 然后按照单卸作业的调度算法将上述假想集装箱卸载至对应真实集装箱在堆场上的位置. 这样得到的各台设备的“卸载”作业的集装箱作业序列为实际装载作业序列的逆序列. 本文中所述的算法为单卸作业调度算法, 同时单装作业的调度问题可以通过采取上述方法加以解决, 在此不再赘述.

基于集装箱卡车运输系统不是整个码头运输系统瓶颈的假设, 我们可以将集装箱卡车运输阶段从整个码头三阶段调度问题中剥离出来, 从而将原来的三阶段问题分解成为两个问题依次加以解决:

- 1) 不考虑集装箱卡车运输条件下的岸吊-场吊两阶段调度问题;
- 2) 在已决定岸吊和场吊处理顺序的条件下的集装箱卡车的指派问题.

由于集装箱卡车的能力可以保证充足, 因此第二个问题必然存在对第一阶段问题不产生影响的优化解. 我们分两阶段求解原问题: 首先利用基于 Johnson 规则的启发式算法求解岸吊-场吊两阶段调度问题; 然后再解决集装箱卡车的指派问题.

3.1 算法第一阶段

Step 1 将长度为 L 的作业区域按照作业岸边吊机的数量平均划分成连续的子作业区域, 岸边吊机 i 负责的作业区域内的所有任务组成该岸边吊机的作业集 J_i , 参见图 2.

Step 2 对每一台岸边吊机 i , 取所有的 $j \in J_i$, 分别计算其修正的处理时间 $\tilde{p}_{j,1}$ 和 $\tilde{p}_{j,3}$, 其中, $\tilde{p}_{j,1} = p_{j,1} + \min_{j' \in J_i} (s_{j,j',1})$, $\tilde{p}_{j,3} = p_{j,3} + \min_{j' \in J_i} (s_{j,j',3})$.

Step 3 令 $U_i = \{j \in J_i | \tilde{p}_{j,1} \leq \tilde{p}_{j,3}\}$, $V_i = \{j \in J_i | \tilde{p}_{j,1} \geq \tilde{p}_{j,3}\}$.

Step 4 将 U_i 中的任务按照 $\tilde{p}_{j,1}$ 递增的顺序排列, 将 V_i 中的任务按照 $\tilde{p}_{j,3}$ 的递减顺序排列, 将 V_i 排至 U_i 的末端, 得到卸载作业序列 $S_{D,i}$.

Step 5 取完成时间最早的吊机 i , 执行 $S_{D,i}$ 中第一个任务 j , 检验是否违反预定义顺序和岸边吊机碰撞约束, 若违反则从 $S_{D,i}$ 中取第二个任务 j' , 直至选中不违反上述约束的任务为止. 若不违反则在作业完成后从 $S_{D,i}$ 中删除 j . 如果该吊机的卸载作业序列 $S_{D,i}$ 中无可执行的任务 (所有任务都违反上述约束), 则选择完成时间第二早的吊机 i' 进行指派.

Step 6 重复 Step 5, 直到所有 $S_{D,i}$ 为空集.

Step 7 计算当前所有岸边吊机的完成时间, 记最大值为 C_{max} , 最小值为 C_{min} , 计算系数 $\beta = (C_{max} - C_{min})/C_{max}$ 和初始作业集合的移动距离 $\Delta l = \alpha\beta$, 其中 α 为常数. 如果 $\beta \geq 0.05$, 则将完成时间最晚的岸边吊机的初始作业集合缩小距离 Δl , 对应将其临近的完成时间相对较小的岸边吊机的初始作业集合扩大 Δl , 并返回第二步. 如果连续 3 次循环中 β 值未减小, 则令 $\Delta l = l$ 进行迭代. 当 $\beta \leq 0.05$ 或者循环达到 40 次

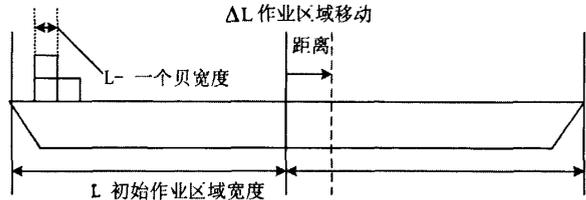


图 2 岸边吊机作业区域调整示意图

时结束, 记录当前最优分配结果.

Step 8 计算所有任务在 QC 阶段的实际完成时间 $C_{j,1}$, 计算 $C_{j,2} = C_{j,1} + p_{j,2}$. 对所有任务按照 $C_{j,2}$ 由小到大排序为 S_{IT} .

Step 9 选择 S_{IT} 中排位最靠前的任务 j' , 为任务 j' 依次指派作业的场地吊车, 并按照实际发生的设备调整时间计算所有场地吊车的 $C'_i = C_i + p_{j',1} + s_{j,j',1}$, 其中 j 为岸边吊机 i 当前作业的任务. 如果场地吊车 i 具有最小的 C'_i 值, 则将 j' 指派给 i 并将 j' 从 S_{IT} 中删除.

Step 10 重复 Step 8 直到 S_{IT} 为空.

Step 11 结束.

3.2 算法第二阶段

Step 1 把所有任务根据 $C_{j,1}$ 从小到大排列得到序列 S . 所有集卡当前操作完成时间 $C_i = 0$, 记所有集卡操作的第一个任务为 $j = 0$ (虚拟任务). $s_{0,j,1}$ 为集卡当前位置到达任务 j 位置所需的时间.

Step 2 取 S 中排位第一的任务 j , 计算所有集卡 $C'_i = C_i + p_{j,2} + s_{j,j',2}$, 其中 j 为集装箱卡车当前作业的任务. 将 j' 指派给 C'_i 值最小的集卡 i , 则将 j' 指派给 i 并将 j' 从 S_{IT} 中删除.

Step 3 重复 Step 2 直到 S 为空集.

4 低界

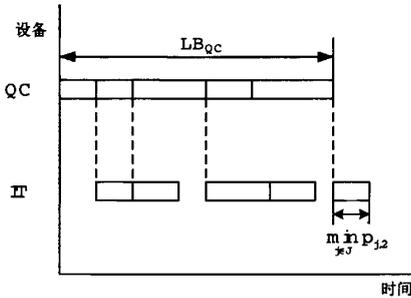


图 3 情况 1 (QC 为瓶颈)

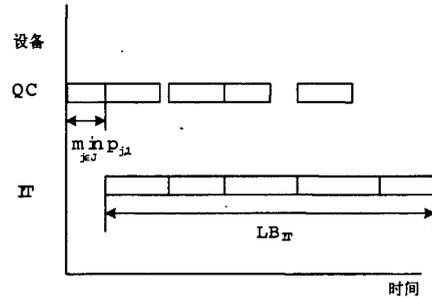


图 4 情况 2 (IT 为瓶颈)

在不考虑设备调整时间的情况下, 易得到岸边吊机阶段的低界 $LB_1 = \sum_{j \in J} p_{j,1} / N_{QC}$, 第二阶段下界 $LB_2 = \max \left(LB_1 + \min_{j \in J} p_{j,2}, \sum_{j \in J} p_{j,2} / N_{IT} + \min_{j \in J} p_{j,1} \right)$, 分别对应图 3 和图 4 甘特图所示的两种情况. 同理三阶段作业的下界为 $LB_3 = \max \left(LB_2 + \min_{j \in J} p_{j,3}, \sum_{j \in J} p_{j,3} / N_{YC} + \min_{j \in J} p_{j,2} \right)$. 其中 $N, N_{QC}, N_{IT}, N_{YC}$, 分别为作业任务数量, 岸边吊桥数量, 集装箱卡车数量和场地吊车数量. 考虑作业时存在设备的调整时间, 因此每个阶段实际消耗的总时间为总作业时间与该阶段设备总调整时间之和. 以岸边吊桥阶段为例, 当作业任务总数量为 N , 岸边吊桥数量为 N_{QC} 时, 在岸边吊机阶段至少发生 $N - N_{QC}$ 次设备调整, 因此岸边吊桥阶段设备调整总时间 S_1 满足:

$$S_1 \geq (N - N_{QC}) \min_{j,j' \in J} s_{j,j',1} \tag{22}$$

故 $LB(S_1) = (N - N_{QC}) \min_{j,j' \in J} s_{j,j',1}$, 因此可以得到修正后的岸边吊桥阶段的低界为:

$$LB_1 = \left(LB(S_1) + \sum_{j \in J} p_{j,1} \right) / N_{QC} = \left[(N - N_{QC}) \min_{j,j' \in J} (s_{j,j',1}) + \sum_{j \in J} p_{j,1} \right] / N_{QC} \tag{23}$$

同理在集装箱卡车阶段和场地吊车阶段, 用 $LB(S_2) + \sum_{j \in J} p_{j,2}$ 和 $LB(S_3) + \sum_{j \in J} p_{j,3}$ 分别替换 $\sum_{j \in J} p_{j,2}$ 和 $\sum_{j \in J} p_{j,3}$, 可以得到在存在设备调整时间情况下的作业时间的低界, 如下式:

$$LB_2 = \max \left(LB_1 + \min_{j \in J} p_{j,2}, \left[(N - N_{IT}) \left(\min_{j,j' \in J} s_{j,j',2} \right) + \sum_{j \in J} p_{j,2} \right] / N_{IT} + \min_{j \in J} p_{j,1} \right) \tag{24}$$

$$LB_3 = \max \left(LB_2 + \min_{j \in J} p_{j,3}, \left[(N - N_{YC}) \left(\min_{j,j' \in J} s_{j,j',3} \right) + \sum_{j \in J} p_{j,3} \right] / N_{YC} + \min_{j \in J} p_{j,2} \right) \tag{25}$$

5 数值实验

根据在某港口调研的情况设置试验数据和参数,其中计划期内开放 20 片堆区,每个堆区指派一台场地吊车作业,岸边吊桥数量为 3,集装箱卡车数量为 10,具体设备参数参见表 1.其中 t 为作业时间, x 为运行距离,单位分别为秒和米.集装箱卡车速度设定为 15km/h.为不失一般性,每次试验中船舶上集装箱分布随机生成,舱盖板范围随机生成.通过编程实现 JLS 算法和下界,在每个问题规模下测试 20 组数据,计算结果的平均值,结果如表 2 所示.

表 1 数值试验设备运行经验公式

设备名称	作业类型	经验公式	设备名称	作业类型	经验公式
QC	垂直升降(空)	$t = 0.342x + 5.407$	YC	垂直升降(空)	$t = 1.002x + 5.883$
QC	垂直升降(满)	$t = 0.878x + 2.477$	YC	垂直升降(满)	$t = 2.000x + 3.986$
QC 小车	水平移动	$t = 0.262x + 5.195$	YC 小车	水平移动	$t = 0.501x + 2.921$
QC 大车	水平移动	$t = 1.335x + 5.898$	YC 大车	水平移动	$t = 0.604x + 5.694$

表 2 JLS 算法平均计算结果与下界的比较

问题规模 (计划期内卸载集装箱总数)	C_{max} (秒)	下界(秒)	差距	计算时间(秒)
1000	36621.77	33603.31	8.98%	74.90
1500	52094.61	47430.78	9.83%	164.66
2000	68556.85	62219.42	10.18%	199.45
2500	85698.69	78098.06	9.73%	249.86
3000	107978.31	98983.22	9.09%	318.05
3500	125774.02	114388.65	9.95%	340.47
4000	132336.99	124113.93	6.62%	398.18
4500	139518.07	131061.85	6.45%	422.72
5000	146926.09	141146.52	4.09%	479.33
5500	164301.02	157174.61	4.53%	525.17
6000	183250.17	174629.89	4.93%	607.13
6500	197429.09	192131.16	2.76%	667.43
7000	219633.71	209381.97	4.90%	720.34

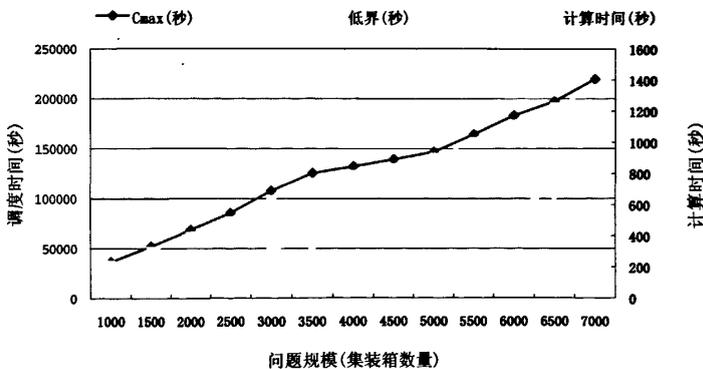


图 5 JLS 算法平均计算结果

由表 2 可看出,在问题规模较大的情况下,JLS 算法的性能较好,而且即便在较小的卸载规模下,JLS 算法所得到的调度结果与下界相比也基本控制在 10% 以内.随着卸载规模逐渐增大,JLS 算法所得结果与下界之差距缩小至 5% 以内.考虑到集装箱码头搬运系统作业系统的复杂性,很难推导出能够全面反映集装箱搬运系统作业的各种约束的原问题下界.事实上,本研究中所采用的下界是松弛预定义顺序约束和岸边吊机的碰撞约束以后的松弛问题的下界,即便如此,我们认为使用 JLS 算法所得到的解的质量(相对于我们的下界)还是较高的.由于 JLS 算法中在不同的计算情况下选取了不同迭代步长 Δl 进行计算,也比较有效的避免了计算结果的波动.考虑到 JLS 算法在岸边吊机分配阶段规定集装箱船上同一贝内的集装箱全部由一台岸吊进行作业,因此在小规模作业时,由于集中堆放等原因可能导致调度结果相对于下界出现一定幅度的波

动,但是在解决大规模问题时,采用上述的作业方式可以大规模的缩小搜索空间,节省大量的计算时间,而且随着问题规模的扩大,集中堆放等因素导致的算法性能波动也随之减弱,因此在立足于解决大规模调度问题的前提下,JLS 算法还是有效的保证了算法的精度和稳定性.这一点在表 2 中也得到了印证.

6 结论

本文针对集装箱码头搬运系统调度问题建立了数学规划模型并设计了 JLS 算法,用以缩短集装箱码头对船作业时间. JLS 算法的创新之处在于针对性地解决带有预定义作业顺序约束和具有与作业次序相关的设备调整时间特点的集装箱码头搬运系统调度问题,并且能够在问题规模十分庞大的情况下具有较快运算速度.数值试验结果显示 JLS 算法在计算时间和计算精度上都具有良好的表现.考虑到在岸桥装卸一体化作业的条件下,各阶段设备作业序列之间的相互关系更加复杂,整体上优化的空间较单装/卸的作业模式也更大,因此未来可在岸桥装卸一体化作业的前提下,针对集装箱码头的集成调度问题进行更加深入的研究.

参考文献

- [1] Vis F A I, de Koster R. Transshipment of containers at a container terminal: An overview[J]. *European Journal of Operational Research*, 2003, 147(1): 1-16.
- [2] Pererofsky R I, Daganzo C F. A branch and bound solution method for the crane scheduling problem[J]. *Transportation Research Part B*, 1990, 3(24): 159-172.
- [3] Daganzo C F. The crane scheduling problem[J]. *Transportation Research Part B*, 1989, 23B(3): 159-175.
- [4] Ebru K Bish. A multiple-crane-constrained scheduling problem in a container terminal[J]. *European Journal of Operational Research*, 2003, 144(1): 409-417.
- [5] Kim K H, Kim Y K. Routing straddle carriers for the loading operation of containers using a beam search algorithm[J]. *Computers & Industrial Engineering*, 1999, 36(1): 55-68.
- [6] Ng W C, Mak K L. Yard crane scheduling in port container terminals[J]. *Applied Mathematical Modelling*, 2005, 29(3): 263-276.
- [7] Chen L, Bostel N, Dejax P, et al. A tabu search algorithm for the integrated scheduling problem of container handling systems in a maritime terminal[J]. *European Journal of Operational Research*, 2008, 181(1): 40-58.
- [8] Lau Y K H, Zhao Y. Integrated scheduling of handling equipment at automated container terminals[J]. *International Journal of Production Economics*, 2008, 121(2): 665-682.
- [9] 张婕姝. 港口生产调度仿真模型 [J]. *上海海事大学学报*, 2005, 26(2): 42-46.
Zhang J S. On the simulation model for port production scheduling[J]. *Journal of Shanghai Maritime University*, 2005, 26(2): 42-46.
- [10] 张海霖, 江志斌, 许泓. 集装箱港口集疏运调度系统作业模式的仿真分析 [J]. *上海交通大学学报*, 2006, 40(6): 1024-1030.
Zhang H L, Jiang Z B, Xu H. A simulation study of container terminal scheduling system[J]. *Journal of Shanghai Jiaotong University*, 2006, 40(6): 1024-1030.
- [11] 周鹏飞, 康海贵. 面向随机环境的集装箱码头泊位——岸桥分配方法 [J]. *系统工程理论与实践*, 2008, 28(1): 168-169.
Zhou P F, Kang H G. Study on berth and quay-crane allocation under stochastic environment in container terminal[J]. *Systems Engineering — Theory & Practice*, 2008, 28(1): 168-169.
- [12] 曾庆成, 杨忠振. 集装箱码头作业调度双层规划模型及求解算法 [J]. *哈尔滨工程大学学报*, 2007, 28(3): 277-281.
Zeng Q C, Yang Z Z. A bi-level programming model and its algorithm for operation scheduling in container terminal[J]. *Journal of Harbin Engineering University*, 2007, 28(3): 277-281.
- [13] Liu C Y, Chang S C. Scheduling flexible flow shops with sequence-dependent setup effects[J]. *IEEE Transactions on Robotics and Automation*, 2000, 16(4): 408-419.
- [14] Kurz M E, Askin R G. Comparing scheduling rules for flexible flow lines[J]. *International Journal of Production Economics*, 2003, 85(3): 371-388.
- [15] Kurz M E, Askin R G. Scheduling flexible flow lines with sequence-dependent setup times[J]. *European Journal of Operational Research*, 2004, 159(1): 66-82.
- [16] Zandieh M, Fatemi Ghomi S M T, Moattar Husseini S M. An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times[J]. *Applied Mathematics and Computation*, 2006, 180(1): 111-127.