

考虑集卡到港时间延迟的堆场起重机动态调度研究

王耀宗, 胡志华*, 田曦丹, 陈婉婷

(上海海事大学 物流研究中心, 上海 201306)

摘要: 在集装箱码头的集港作业中,集卡延迟到港可能导致已制订堆场起重机作业计划变得次优甚至不可行.因此,及时为延迟到港任务更新作业计划具有必要性.考虑集卡到港时间延迟的堆场起重机动态调度问题,基于任务批次分配策略,将调度期划分为多个时段,在每个时段内生成作业计划.以完成时间最短为目标,建立起重机调度混合整数规划优化模型,并提出下界推导模型.提出迭代重优化框架,每当任务延迟到港,就会重优化相应批次的作业计划.在框架内,设计遗传算法求解各批次原有作业计划;设计贪婪插入算法重优化带延迟到港的任务,更新原有计划.实验结果表明,模型、下界推导模型具有正确性和有效性,算法具备良好的求解性能;缩短调度时段长度有利于快速生成起重机作业计划;延迟到港箱量占比越大,作业时间越长,且随着调度时段长度减小,这一影响逐渐弱化.

关键词: 堆场起重机;集卡到港时间延迟;动态调度;迭代重优化;贪婪策略

中图分类号: U691

文献标识码: A

doi: 10.7511/dllgxb202303009

0 引言

在自动化集装箱码头中,堆场面积占码头总面积的40%至60%,是码头重要的作业区域^[1].自动化堆场连接岸边船舶的装卸与外集卡的运输,用于集装箱的堆存和中转,其作业效率制约码头的整体性能.为提升堆场的作业能力,码头采用穿越式双起重机的作业工艺,协同作业箱区内任意位置的装卸和堆垛任务^[1].然而,在同一箱区作业的两台起重机可能发生干涉^[2],即如果小起重机穿越正在某一贝位作业的大起重机或者两台起重机同时同一贝位作业,将发生干涉.因此,起重机的调度计划应为一个无干涉的作业序列.

在集港作业中,集卡可能受交通拥堵等不确定因素的影响而延迟到港.通常,在集装箱船靠泊前的1~3 d,码头为准备装船的集装箱制订集港计划,集卡按照预约的时段依次到达堆场.然而,不确定因素可能导致集卡未能按照预约的时段到达堆场,实际交箱时间滞后于计划的交箱时间,对原有作业计划产生扰动.因此,码头需要及时调整

作业计划以降低扰动前后规划结果的偏差.集卡延迟到港使得码头的调度作业具有动态性特征.码头通常以静态方式制订作业计划^[3],即调度期初做好所有任务的起重机作业计划,但该作业模式的时效性较差,如果出现集卡到港时间延迟而产生即时请求,原有作业序列将不再满足最优性甚至不可行,影响堆场的作业效率.

在码头多级作业场景下,作业工序相互耦合与装卸决策相互依赖是设备调度的复杂性所在,而双起重机非交叉约束进一步增加了建模的难度^[4].Vis等^[5]把双起重机问题转化为单起重机问题进行求解;Gharehgozli等^[6]把起重机的调度问题抽象为具有优先约束的多重旅行商问题;郑红星等^[7]研究考虑倒箱的混堆装船箱区内起重机调度优化问题,提出了带滚动时域的启发式算法求解多起重机作业序列,并设计了带有解空间切割的遗传算法求解单个时域对应的子调度问题;Hu等^[8]把起重机之间的安全距离转化为时间间隔以处理干涉,提出了混合整数规划模型和非线性模型刻画批次任务之间的顺序约束,并设计了

收稿日期: 2022-03-14; 修回日期: 2023-03-25.

基金项目: 国家自然科学基金资助项目(71871136);上海市自然科学基金资助项目(23ZR1426500).

作者简介: 王耀宗(1993—),男,博士生, E-mail: yzwang_hit@163.com; 胡志华*(1977—),男,博士,教授,博士生导师, E-mail: zhhu@shmtu.edu.cn.

精确算法和遗传算法求解起重机的作业计划; Han 等^[9]以提箱完成时间最短为目标建立起重机调度模型,优化装船过程中起重机的行驶路径; 秦磊磊等^[10]以最小化任务完成时间为目标建立考虑交接区缓冲贝位决策的双起重机调度混合整数规划模型,求解最优缓冲贝位的位置与起重机作业序列. 以上研究为堆场起重机的调度优化提供了理论及方法上的参考.

一些研究考虑了不确定因素的码头作业场景. 为降低集卡到港时间不确定对出口箱堆存和装船效率的影响,范厚明等^[11]构建了双层混合整数规划模型优化出口箱位分配与起重机作业序列,通过不断平衡层与层之间的最优解优化堆场堆存和装船作业效率;针对码头装卸作业时间不确定性,孙玉姣等^[12]以最小化作业完成时间、岸边无集卡与堆场无集卡的时间和为目标,建立了混合整数规划模型,研究了考虑装卸时间受作业顺序与位置等影响下码头生产调度问题,为码头提供动态场景下优化装卸作业的参考;范厚明等^[13]关注交箱序列不确定的起重机调度问题,根据出口箱预约信息提出了新的堆存策略,建立了堆场箱位分配及多起重机调度集成优化模型,同时对于预约时段的长度及准确性对调度结果的影响做了鲁棒性分析,补充了不确定情景下码头的运作优化研究. 此外, Dorndorf 等^[14]关注穿越式 Triple-ASC 调度,当一个任务完成或一个新任务到达时,动态调整调度计划以优化任务的分配和起重机的作业序列.

在动态调度问题的研究中, Pillac 等^[15]综述了动态车辆路径优化问题,着重研究了确定性、随机性两个特征组合下的车辆调度问题. 针对客户路线变更的路径优化问题, Ozbaygin 等^[16]建立了混合整数规划模型,在迭代重优化框架内,基于初始计划动态调整路线变更后的计划;为了减小重调度对系统的扰动,宁涛等^[17]从干扰管理角度研究了车辆行驶时间延迟问题,以最小化用户时间窗偏离度和最小化配送成本为目标建立了数学规划模型;针对自动导引车 (AGV) 动态调度问题,丁一等^[18]采用周期和事件混合驱动的方式——根据 AGV 的状态定义若干触发规则响应突发事件,并结合周期型调度策略生成调度计划;杨珍花等^[19]研究了甩挂车辆的动态调度问题,建立了挂车装卸时间不确定混合整数规划模型,并提出了

多阶段动态优化算法生成牵引车的调度方案.

综上所述,关于动态车辆路径优化和柔性车间动态调度问题的研究已取得丰硕的成果. 多数文献采用周期驱动或事件驱动的重调度方式处理延迟或紧急任务,即在特定时刻或状态下调整原有计划. 然而,该重调度方式下,如何在复杂场景下界定事件的存在形式与如何选择调度时段长度是研究的难点,尤其在集装箱码头这一多级物流系统中,难以穷尽作业场景下所有状态及其对应事件. 为此,本文在原有研究基础上,考虑在集卡到港时间延迟情形下的出口箱作业,以作业完成时间最短为目标,建立双起重机动态调度混合整数规划模型,提出一种迭代重优化框架处理延迟到港任务. 由于存在非交叉约束,堆场起重机调度的难度显著增加,原有研究中, Ng^[20]证明了其为 NP-Hard 问题. 为了求解大规模作业计划,把遗传算法和贪婪插入算法纳入迭代重优化框架内,用遗传算法求解初始作业计划,用贪婪插入算法重优化延迟到港任务. 贪婪插入算法的局部搜索技术具有快速生成最优解的优势,能够及时更新作业计划,响应动态性的要求.

1 动态调度问题描述

为了快速响应延迟到港任务需求,以降低对起重机调度结果的影响,把集港作业的调度期 T 划分为 m 个时段 $T_i, i \in \{1, 2, \dots, m\}$, 按照调度时段 T_i 的长度,把任务划分为多个批次的子任务 φ_i , 在调度期内分批次生成作业计划. 一旦出现延迟到港任务的即时请求,根据其延迟到港时间,把该任务临时分配到相应的调度时段内,与该调度时段的原有计划一起执行重调度作业. 基于此,提出迭代重优化框架,动态优化计划期内的任务,其关键是把延迟到港任务分配到某一调度时段内执行重调度,同时刷新该调度时段之后的若干批次作业计划. 值得注意的是,把调度时段的划分和存在延迟任务的重调度纳入该框架内,降低了作业计划调整的难度与成本. 基于迭代重优化框架的动态优化流程如下:

步骤 1 划分调度时段集合 Ω 及其对应的任务集合 φ . $\Omega = \{T_1, T_2, \dots, T_m\}$, 设 T_i^+, T_i^- 分别为调度时段 T_i 的上、下界;按照交箱时段把属于同一调度时段 T_i 内的任务归类到同一批,记为

φ_i , 即 $\varphi = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$.

步骤2 在零时刻下, 向后生成3批作业计划 P_i , 即生成包括调度时段 T_1, T_2, T_3 在内的3批作业计划 P_1, P_2, P_3 . 置 $i=1$.

步骤3 在当前时刻 t 下 ($t \in [T_i^-, T_i^+]$), 向后生成1批作业计划, 即生成在调度时段 T_{i+3} 对应的作业计划 P_{i+3} .

步骤4 如果存在延迟到港任务 r , 设该任务 r 对应的交箱时刻为 I_r . 若 $I_r \in [T_j^-, T_j^+]$, $j = \{i+3, i+4, \dots, m\}$, 则向后刷新 $T_{i+3}, T_{i+4}, \dots, T_m$ 与 $\varphi_{i+3}, \varphi_{i+4}, \dots, \varphi_m$, 调整调度时段集合 Ω 及其对应的任务集合 φ , 并返回步骤3. 如果不存在延迟到港任务, 则执行步骤5.

步骤5 生成起重机作业计划 P_i , 记录完成集合 P_i 中所有任务的时刻 t^* .

步骤6 更新当前时刻: $t \leftarrow t^*$, 若 $t \leq T_m^+$, 则更新任务执行进度, $i \leftarrow i+1$, 并返回步骤3; 否则, 结束优化, 输出作业计划 $P_i, i = \{1, 2, \dots, m\}$.

动态优化流程的核心是边作业边分配. 即同一时刻起重机执行已分配的作业计划, 同时码头为剩余的集港作业分配作业计划; 此外, 不断检测是否存在延迟任务, 为延迟到港的集装箱临机分配作业计划, 并重优化该调度时段内的作业计划. 以上基于迭代更新的动态优化框架是调度优化模型及其优化算法设计的前提. 在该动态优化框架下, 每一时刻作业系统内仅存在少量的作业计划, 不仅降低了计算成本, 也保证了在小范围内为延迟到港集装箱任务分配作业计划的便利性, 避免延迟到港的不确定性给系统带来较大的扰动.

2 调度优化模型

2.1 问题特征分析

本文研究的目的是优化集港作业内任务的分配与起重机的作业序列, 最小化任务完成时间; 如果存在延迟到港任务, 则对原调度计划进行重优化, 以保证调度计划的最优性, 同时减少作业计划调整所产生的延误时间. 因此, 关键优化问题: (1) 决策每个任务的执行者(起重机); (2) 决策每台起重机执行任务的顺序; (3) 如果调度计划存在干涉, 执行规避干涉; (4) 如果存在延迟到港任务, 执行重优化.

2.2 变量定义

模型的基本符号及其变量定义如下:

J : 子任务集合, 同一调度时段内的任务集合, $i, j \in J; J^- = J \cup \{O\}, J^+ = J \cup \{\bar{O}\}$. 其中 $O, \bar{O}$ 分别为虚拟起始和终止任务.

J^d : 延迟到港的任务集合, $i, j \in J^d$.

K : 起重机 k 集合, $k \in K, K = \{1, 2\}$.

B : 贝位 b 集合, $b \in B, B = \{0, 1, \dots, 41\}$.

O_i, D_i : 任务 i 的起始贝位和目标贝位, $O_i = 0, D_i \in B \setminus \{0, 41\}$.

T_{ij} : 起重机作业任意两个任务 i, j 之间的空载时间, $i, j \in J$.

T_v : 起重机移动一个贝位的时间, $T_v = u$.

T_{ii} : 起重机作业任务 i , 从起始贝位 O_i 到目标贝位 D_i 的移动时间, $\forall i \in J$.

T_s : 起重机提升或释放一个集装箱的服务时间, $T_s = 7.5u$.

x_{kij} : 0-1 变量. 若起重机 k 依次作业任务 i, j 取1; 否则取0.

y_{ki} : 0-1 变量. 若任务 i 由起重机 k 作业取1; 否则取0.

s_{ki}, e_{ki} : 任务 i 由起重机 k 作业的开始、完成时间.

s_i, e_i : 任务 i 由起重机作业的开始、完成时间.

a_{ij} : 0-1 变量. 若任务 i 的作业开始时间在任务 j 之前取1; 否则取0. 其中 $O_i = O_j, \forall i, j \in J$.

b_{ij} : 0-1 变量. 若任务 i 的作业完成时间在任务 j 之前取1; 否则取0. 其中 $D_i = D_j, \forall i, j \in J$.

2.3 模型建立

以任务完成时间最短为优化目标, 建立考虑集卡到港时间延迟的起重机调度混合整数规划模型, 确定每一个任务(包括延迟到港任务)的分配结果与每台起重机的作业序列. 因此, 优化模型的目标函数 f^1 为最小化所有任务的最大完成时间 e_i , 即 $f^1 = \min_{i \in J \cup J^d} e_i$.

双起重机调度问题与车辆路径优化问题的建模机理一致, 均有分配约束和序约束, 即双起重机调度问题的分配约束确定起重机作业任务的集合, 序约束确定每台起重机作业任务的顺序. 不同的是, 在堆场起重机调度问题上, 需求是同质的, 起重机也无容量约束. 因此, 将起重机作业任务的开始和完成时间作为求解该问题的主要决策, 而该决策过程也是序约束的实现方式; 另外, 要求每

个任务都被某一起重机作业,即对每一个任务而言,出度和入度相等且均为1.在不考虑起重机干涉的情况下,双起重机调度问题约束为

$$\sum_{j \in J \cup J^d} x_{kj} = \sum_{i \in J \cup J^d} x_{ki} = 1, \forall k \in K \quad (1)$$

$$\sum_{k \in K} y_{ki} = 1, \forall i \in J \cup J^d \quad (2)$$

$$\sum_{j \in J^+ \cup J^d} x_{kj} = y_{ki}; \forall k \in K, \forall i \in J^- \cup J^d \quad (3)$$

$$\sum_{i \in J^- \cup J^d} x_{ki} = \sum_{i \in J^+ \cup J^d} x_{ki}; \forall k \in K, \forall j \in J \cup J^d \quad (4)$$

$$e_{ki} \geq s_{ki} + T_{ii} + 2T_s + (y_{ki} - 1)M_1; \forall i \in J \cup J^d, \forall k \in K \quad (5)$$

$$s_{kj} \geq e_{ki} + T_{ij} + (x_{kj} - 1)M_1; \forall i \in J^- \cup J^d, j \in J^+ \cup J^d, \forall k \in K \quad (6)$$

$$s_{ki} \leq y_{ki}M_1; \forall i \in J^- \cup J^d, \forall k \in K \quad (7)$$

$$e_{ki} \leq y_{ki}M_1; \forall i \in J^- \cup J^d, \forall k \in K \quad (8)$$

$$M_1 = 2NT_s + \sum_{i,j \in J \cup J^d} (T_{ii} + T_{ij}) \quad (9)$$

式(1)确保有且仅有一个任务从虚拟起始任务处之后允许起重机作业,有且仅有一个任务在虚拟终止任务处结束作业;式(2)保证每一个任务只能由同一台起重机作业;式(3)确保起重机 k 作业任务 i 之后,紧接着作业任务 j ;式(4)保证每一个任务的出度和入度相等,即每一个任务被起重机作业;式(5)为任务作业的开始和完成时间的关系;式(6)约束起重机作业任务的先后关系;式(7)、(8)为作业时间上限约束;式(9)为 M_1 的最小值.然而,由于干涉的存在,两台起重机的装卸决策相互依赖,约束(1)~(9)的优化结果可能出现无效解,即所得解包含干涉存在的形式.因此,为了生成无干涉的调度计划,基于以上所界定的干涉形式及其规避规则,约束两台起重机在同一贝位作业时的依赖关系,即:

$$s_i = \sum_{k \in K} s_{ki}, \forall i \in J^- \cup J^d \quad (10)$$

$$e_i = \sum_{k \in K} e_{ki}, \forall i \in J^- \cup J^d \quad (11)$$

$$a_{ij} + a_{ji} = 1; \forall i, j \in J \cup J^d, O_i = O_j, i \neq j \quad (12)$$

$$b_{ij} + b_{ji} = 1; \forall i, j \in J \cup J^d, D_i = D_j, i \neq j \quad (13)$$

$$s_j \geq s_i + T_s + (a_{ij} - 1)M_1; \forall i, j \in J \cup J^d, O_i = O_j, i \neq j \quad (14)$$

$$e_j \geq e_i + T_s + (b_{ij} - 1)M_1; \forall i, j \in J \cup J^d, D_i = D_j, i \neq j \quad (15)$$

决策变量的取值如下:

$$s_{ki}, e_{ki}, s_i, e_i \geq 0; \forall i \in J \cup J^d, \forall k \in K$$

$$x_{kj}, y_{ki}, a_{ij}, b_{ij} \in \{0, 1\}; \forall i, j \in J \cup J^d$$

式(10)、(11)为任务作业的开始和完成时间;式(12)、(13)界定出现干涉情况,限定任务 i 或任务 j 其中一个完成时间较晚;干涉规避的同步约束如式(14)、(15)所示,限定位于同一贝位的两个任务作业开始和完成时间之间的依赖关系.

2.4 下界推导

双起重机调度问题求解的关键是计算单台起重机的作业序列,Vis等^[21]提出了一种基于动态规划的最优下界推导方法,其目标是找到一条包含所有任务的起重机运行最短路径.基于Vis等^[21]的下界推导方法,文献[5]证明了双起重机作业完成时间近似等于单台起重机作业所有任务时间(travel time)的一半.本文提出一种考虑两台起重机作业负载均衡的下界推导方法,使得所求的任务完成时间更接近所有任务作业时间的一半,缩小下界值与最优值之间的差距.增加整数决策变量 p_{ki} ,表示起重机 k 作业任务 i 的次序,即为序约束的关键决策,建立松弛模型计算下界 f_{lb} .目标函数(16)是最小化两台起重机作业均衡条件下的完成时间,即下界 f_{lb} .

$$f_{lb} = \left(2NT_s + \sum_{i,j \in J, k \in K} x_{ki} T_{ij} + \sum_{i \in J} T_{ii} \right) / 2 \quad (16)$$

$$\sum_{k \in K} y_{ki} = |K| \quad (17)$$

$$\sum_{i \in J} y_{ki} \leq N/2, \forall k \in K \quad (18)$$

$$p_{ki} - p_{kj} + M_2 x_{kj} \leq M_2 - 1;$$

$$\forall i \in J^-, \forall j \in J, \forall k \in K \quad (19)$$

$$1 \leq p_{ki} \leq M_2; \forall i \in J^-, \forall k \in K \quad (20)$$

$$M_2 = N \quad (21)$$

式(17)表示起重机在虚拟开始任务处作业,且任务必须分配给两台起重机;式(18)表示起重机作业任务量约束,为了均衡每台起重机的作业负载;式(19)、(20)表示每台起重机作业任务的序约束,用于消除子回路;式(21)表示 M_2 的一个取值.

3 迭代重优化框架及其算法

3.1 迭代重优化框架

为了求解考虑集卡延迟到港的堆场起重机动态调度问题,在迭代更新的动态优化流程基础上,提出一种动态处理延迟到港任务的方法——迭代

重优化框架及其算法。首先,在所划分的时段 T_i 内,依次生成对应批次任务的起重机作业计划;如果起重机在作业之前无延迟到港任务,则执行原有作业计划。一旦存在延迟到港任务,原有作业计划变得次优甚至不可行。因此,根据起重机作业进度和任务延迟到港时间,对特定批次任务进行重优化。重优化的前提是,定位延迟到港任务的到港时间落在哪个调度时段内,即延迟任务属于哪一批次的作业计划。然后,把延迟到港任务和该批次原有任务一起执行重优化生成更新的作业计划。简而言之,只要存在延迟到港任务,就会生成一批更新的作业计划。

基于此,设计两个优化算法支撑迭代重优化框架:遗传算法用于生成各时段所对应批次任务的作业计划;贪婪插入算法用于重优化带延迟到港批次任务的作业计划,生成更新的作业计划,应对动态性。贪婪插入算法具有快速搜索的特征,能够在遗传算法求解结果的基础上,把延迟到港任务插入原有作业计划序列中最合适的位置。迭代重优化框架及其算法构成如图1所示。以上算法的构成逻辑及其形成的闭环优化策略,其益处体现在3个方面:首先,遗传算法仅仅作用于特定批次的任务集合而不是全部任务集合,因此,搜索空间限定在较小的范围内,增加了搜索到全局最优解的可能性^[22];其次,由于原有作业计划仅限定

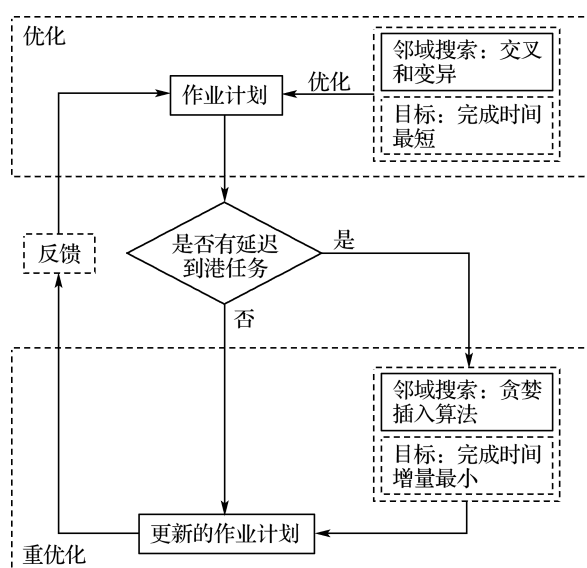


图1 迭代重优化框架及其算法

Fig. 1 Iterative re-optimization framework and its algorithm

在某一批次任务,针对延迟到港任务,贪婪插入算法能够在其有限邻域内快速生成最优解,完成对应批次任务的重优化;最后,考虑延迟到港任务的重优化是基于某个批次的作业计划,重优化后生成的作业计划与原有作业计划形成较为平稳的过渡,避免由于作业计划大范围调整对起重机作业产生较大的扰动。

3.2 算法设计

(1) 编码方式和解码策略

采用随机排列编码向量表示双起重机作业序列中的任务,即染色体上每一位元素(等位基因)都是互异的,代表一个作业任务,如图2所示。在解码过程中,根据染色体上等位基因的排列顺序依次确定起重机的作业序列。

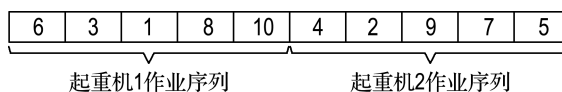


图2 双起重机作业序列的染色体编码结构

Fig. 2 Chromosome coding structure of dual crane sequences

遗传算法的解码策略是将编码向量转化为两台起重机的作业序列,以便利用适应度函数计算编码向量对应的评价值。根据双起重机作业的顺序约束和干涉规避约束,将编码向量转换为可行的双起重机作业序列,并使遗传算法的交叉算子和变异算子在全局范围内寻优,最终确定最优解。以某一批次的作业任务为例,说明解码策略及其对应的算法,如算法1所示。

算法1 基于随机排列编码的解码策略

输入: (1) 参数: $[J, K, O_i, D_i]$;

(2) 随机排列序列: S_0 。

输出: (1) S_k : 最优双起重机作业计划, $\forall k \in K$;

(2) $[s_{ki}, e_{ki}]$: 作业计划中,任务作业开始、完成时间, $\forall i \in J, \forall k \in K$;

(3) f : 完成时间(makespan)。

步骤1 初始化: 令 f 为足够大的实数; s_{ki}, e_{ki} 置零。

步骤2 for x in X . 其中, $X = |S_0|$ 。

步骤2.1 拆分作业序列 S_0 , 前后段依次分配给两台起重机。其中, 起重机1的作业序列 $S_1^* \leftarrow S_0[:x]$; 起重机2的作业序列 $S_2^* \leftarrow S_0[x:]$ 。

步骤2.2 计算作业序列 S_k^* 对应的 $s_{ki}^*, e_{ki}^*, k \in K$ 。

步骤2.3 检测干涉的存在与执行规避。

For i, j in S_1^*, S_2^*

若 $O_i = O_j$ 且 $\Delta = |s_{1,i}^* - s_{2,j}^*| \leq T_s$ 或若 $D_i = D_j$ 且 $\Delta = |e_{1,i}^* - e_{2,j}^*| \leq T_s$, 则令作业序列 S_1^* 中 $s_{1,i}^* \leftarrow s_{1,i}^* + (T_s - \Delta)$, $e_{1,i}^* \leftarrow e_{1,i}^* + (T_s - \Delta)$.

步骤 2.4 分别计算两台起重机作业序列对应的完成时间 f^{1*}, f^{2*} .

步骤 2.5 令 $f^* = \max(f^{1*}, f^{2*})$.

步骤 2.6 如果 $f^* < f$, 则 $f \leftarrow f^*$, $S_k \leftarrow S_k^*$, $s_{ki} \leftarrow s_{ki}^*$, $e_{ki} \leftarrow e_{ki}^*$.

步骤 3 输出 $f, S_k, s_{ki}, e_{ki}, i \in J, k \in K$.

(2) 贪婪插入算法

一旦存在延迟到港任务(即时请求), 随机分配该任务到相近的调度时段内, 用贪婪插入算法重优化该时段内原有作业计划. 贪婪插入示意如图 3 所示, 把新增任务依据完成时间增量最小的原则插入原有序列的相应位置. 若存在多个新任务, 则按照移动时间 T_{ii} 从小到大进行递增排序, 依次执行贪婪插入操作, 如算法 2 所示.

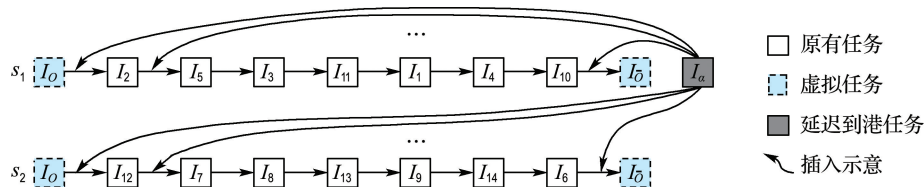


图 3 延迟到港任务重优化示意图

Fig. 3 Schematic diagram of re-optimization with delayed arrival tasks

算法 2 迭代重优化框架下贪婪插入算法

输入: (1) 参数: $[J, K, O_i, D_i]$;

(2) 原有作业序列: $S_{kp}, \forall k \in K, p \in P_i = \{p_i, p_{i+1}, p_{i+2}\}$;

(3) 当前时刻: τ .

输出: (1) P_p : 更新后的第 p 批次作业计划, $\forall p \in P_i$;

(2) $[s_{ki}, e_{ki}]$: 作业计划中, 批次 p 作业的开始和完成时间, $\forall i \in J, \forall k \in K$.

变量: B_p, E_p : 批次 p 作业计划的开始、完成时间;

δ : 任务开始执行前, 允许变更计划的最大持续时间.

步骤 1 零时刻下, 产生 3 批作业序列, 即初始作业计划 P_p , 完成时间为 $E_p, p \in \{1, 2, 3\}$. 令 $m = 1$.

步骤 2 起重机执行作业计划 P_m .

步骤 3 在 $\tau \in (B_m, E_m)$ 下, 向后生成 1 批作业计划 P_{m+3} .

步骤 4 若延迟到港集装箱的到港时间落在第 p 批次作业计划中, 即 $\tau < B_p - \delta, p = \{m+1, m+2\}$, 则重优化原有作业序列; 否则, 转到步骤 7.

步骤 4.1 把延迟到港任务集合 $N(j \in N)$ 按照移动时间 T_{jj} 递增排序, 依次执行插入操作. 取 $l = \arg \min e_{ki}$, 把任务 l 插入序列 S_{kp} 中, 使完成时间的增量最小.

步骤 4.2 更新任务序列, $S_{kp} \leftarrow S_{kp} \cup \{l\}$.

步骤 5 执行干涉规避措施, 如算法 1 步骤 2.3. 更新批次任务的作业开始时间 s_{ki}^* 和作业完成时间 e_{ki}^* , 即 $[s_{ki}, e_{ki}] \leftarrow [s_{ki}^*, e_{ki}^*]$.

步骤 6 生成重优化后的作业计划 $P_p, p = \{m+1, m+2\}$.

步骤 7 当 $\tau \geq B_{m+1}$ 时, 起重机执行作业计划 P_{m+1} ;

令 $m \leftarrow m+1$, 若 $m \leq 3$, 返回步骤 3; 否则, 结束重优化.

4 数值实验

设计多组算例验证堆场起重机调度模型和松弛模型的有效性、算法 1 的计算性能以及算法 2 的重优化性能. 计算在 Windows 10 操作系统上用 Python 语言编程实现, 计算机配置: Intel(R) Core(TM) i7-10510 CPU @ 1.8 GHz, 16 GB RAM. 用 CPLEX 作为模型求解的分支切割(branch-and-cut)求解器.

数据来源于自动化码头作业数据生成平台(<https://www.instances.de/dfg/>), 分别生成多组小规模算例($N \leq 12$)和大规模算例($N \leq 50$). 箱区规模为 40 贝位 $\times$ 6 栈 $\times$ 5 层. 堆场起重机移动速度为 3 m/s, 即起重机在 4 s(u)内通过一个贝位. 另外, 设起重机抓取或释放一个集装箱需要 30 s($7.5u$). 对于起重机而言, 其调度主要关注进口箱任务的起始贝位(O_i)和目标贝位(D_i). 由实例生成器导出的小规模算例($N=5$)见表 1.

4.1 实验设计

实验设计的目的及其相应设置见表 2.

4.2 实验结果与分析

4.2.1 模型与算法有效性验证 模型和算法 1 的规划结果见表 3. 其中, f_{lb} 表示松弛模型所计算的松弛解. 结果显示, 松弛解 f_{lb} 与最优解 f_1 的最

表1 算例示例($N=5$)
Tab.1 Instance example ($N=5$)

序号	O_i	D_i	贝位
1	41	12	12
2	41	8	8
3	41	19	19
4	41	2	2
5	41	7	7

注：数据来源于 <https://www.instances.de/dfg/>.

大误差为3.91%，说明松弛模型为该数学规划问题提供了一个良好下界，可作为模型和算法优化性能的评判基准. 算法1与CPLEX所求得最优解之间的平均误差仅为0.49%；当任务规模增加到12时，CPLEX无法在可接受的时间内求得最优解，但算法1能够在较短时间内求得近似最优解. 由此可见，算法1在求解双起重机调度问题上具备良好性能. 此外，双起重机作业路径如图4所示，两台起重机在同一时间 f 同一贝位 b 下的(重载)路径无交点，表明模型的干涉规避约束有效.

表2 实验设置
Tab.2 Experiment settings

实验序号	实验目的	实验设置	结果
1	验证调度模型有效性、下界模型逼近最优解的效果和算法1的优化性能	①任务数量： $N=\{6,8,10,12\}$ ，各3组；②对于每组算例，分别用CPLEX和算法1求解；③显示 $N=10$ 时双起重机作业路径	表3和图4
2	寻找遗传算法的交叉概率和变异概率最佳组合	①任务数量： $N=\{30\}$ ；②交叉概率 P_c 和变异概率 $P_m:[0,1]$ ，步长0.1；③组合 P_c 和 P_m ，令遗传算法迭代100次，求解最优值和平均值，寻找最优参数组合的分布	图5
3	验证算法的求解性能	①任务数量： $N=\{30,50\}$ ，各20组；②组合 P_c 和 P_m ，分别计算每组任务的最小值、最大值和平均值	表4、表5和图6
4	研究集卡到港时间延迟下不同调度时段长度对作业计划的影响	①调度时段长度： $T=\{300u,600u,900u,M\}$ ；②任务数量： $N=\{30,40,50,75,100,125,150,175\}$.	表6和图7
5	研究延迟到港任务的临机分配及其对规划结果的影响	①任务数量： $N=\{30,50,100\}$ ；②延迟到港箱量与原始箱量的比值 $\beta=\{0.1,0.2,0.3\}$ ；③调度时段长度： $T=\{300u,600u,900u\}$	图8

注：实验3中 P_c 和 P_m 的取值范围基于实验2参数调节确定.

表3 模型和算法1的规划结果对比
Tab.3 Programming result comparison of the model and algorithm 1

序号	N	f_{lb}/u	模型(CPLEX)		$G_1/\%$	算法1				$G_2/\%$
			f_1/u	T_{CPU}/s		f_{min}/u	f_{max}/u	f_{ave}/u	T_{CPU}/s	
1	6	115.00	117.00	0.19	1.71	117.00	117.00	117.00	0.84	0
2	6	135.00	140.50	0.20	3.91	140.50	151.00	140.70	0.79	0.14
3	6	173.50	177.50	0.19	2.25	177.50	184.50	177.70	0.81	0.11
4	8	243.50	248.00	6.22	1.81	248.00	277.50	248.40	1.52	0.16
5	8	250.50	254.40	9.03	1.53	254.50	270.00	255.00	1.48	0.24
6	8	240.00	244.00	5.55	1.64	246.00	264.50	246.35	1.55	0.96
7	10	290.50	292.50	1 134.16	0.68	292.50	321.00	294.60	2.45	0.72
8	10	313.00	316.00	1 652.09	0.95	316.00	327.00	318.24	3.12	0.71
9	10	287.50	294.50	1 145.92	2.38	294.50	325.00	295.86	2.18	0.46
10	12	344.50	351.50	3 600.00	1.99	351.50	380.00	354.45	3.71	0.84
11	12	392.00	398.00	3 600.00	1.51	398.50	427.00	402.28	3.26	1.08
12	12	368.00	373.00	3 600.00	1.34	373.50	394.50	374.90	3.29	0.51

注： f_{min} 、 f_{max} 和 f_{ave} 分别表示10次求解结果的最小值、最大值和平均值； T_{CPU} 为对应的计算时间； $G_1=(f_1-f_{lb})/f_1\times 100\%$ ； $G_2=(f_{ave}-f_1)/f_1\times 100\%$.

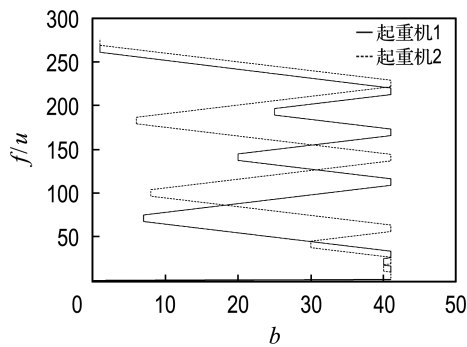


图 4 双起重机作业路径 ($N=10$)
Fig. 4 The operation paths of dual crane ($N=10$)

4.2.2 算法参数调节与性能验证 为研究算法的参数取值对优化结果的影响,在 $[0,1]$ 以步长 0.1 分别取交叉概率 P_c 和变异概率 P_m ,组合后获得非连续分布解的变化规律,如图 5 的等值线所示.由于算例结构的一致性,通过分析算法 1 在 P_c 和 P_m 组合下解的演变规律,限定参数的取值范围,以此作为其他实验中参数选择的依据,避免大范围参数调节造成的冗余计算影响求解效率.

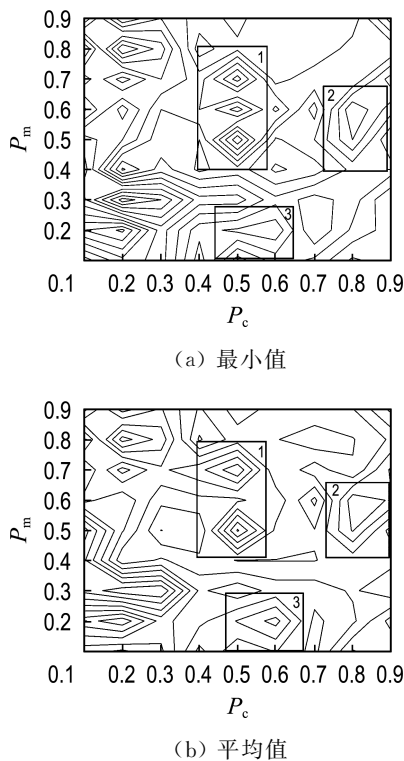


图 5 算法 1 优化结果的等值线分布
Fig. 5 The contour distribution of optimization results by algorithm 1

图 5 中方框内的若干组等值线描述了最优组合参数周围 ($d \leq 0.05$) 计算结果的变化趋势.如

图 5(a) 的方框 1 内存在 3 组等值线,对于第一组等值线 $P_c=0.5$,当 $P_m=0.5$ 时(令 $P_m^*=0.5$),取得最优值;当 $P_m > (或 <) P_m^*$ 时($d \leq 0.05$),计算结果偏离最优值;同理,可得方框 1 中其他两组数据的变化趋势.需要注意的是,当 $P_c=0.5$ 时,在解平面内对应 3 个最优的 P_m ,分别为 0.5、0.6 和 0.7,说明最优参数组合呈现多样性特征.此外,通过最小值与平均值的对比发现,平均值的等值线稀疏于最小值,且最优解的组数(即方框内等值线集聚组数)出现退化,表明计算结果取平均值更具一般性.

为进一步验证算法 1 的计算性能,分别选取任务规模为 30 和 50 进行测试.不失一般性地,取不同的交叉和变异概率组合,减小固定交叉和变异概率对求解结果的局限性,计算结果见表 4、5.计算结果表明,算法求解结果与松弛模型下界之间的平均误差均不超过 2%;另外,图 6 为算法求

表 4 大规模算例下算法 1 的规划结果 ($N=30$)
Tab. 4 The programming results of algorithm 1 for large-scale instance ($N=30$)

序号	f_{lb}/u	f_{min}/u	f_{max}/u	f_{ave}/u	T_{CPU}/s	$G_3/\%$
1	784.00	797.00	827.00	797.92	37.83	1.78
2	703.00	712.00	755.00	716.40	34.13	1.91
3	673.00	689.00	718.00	692.26	33.05	2.86
4	867.50	888.00	907.00	890.16	33.70	2.61
5	865.50	878.00	911.00	881.56	39.07	1.86
6	918.00	926.00	964.00	929.24	33.86	1.22
7	790.00	805.00	825.00	807.27	37.51	2.19
8	793.50	807.00	827.00	808.59	35.29	1.90
9	805.00	821.00	839.00	822.63	37.61	2.19
10	719.00	736.00	762.00	736.94	33.60	2.50
11	705.00	719.00	735.00	719.42	38.77	2.05
12	804.00	816.00	842.00	816.71	37.33	1.58
13	805.00	819.00	858.00	821.46	37.65	2.04
14	732.50	742.00	771.00	747.37	34.08	2.03
15	811.50	831.00	850.00	831.91	37.69	2.52
16	804.50	812.00	848.00	813.25	33.79	1.09
17	810.50	823.00	852.00	826.36	34.02	1.96
18	846.50	859.00	865.00	859.17	33.79	1.50
19	785.50	795.00	820.00	797.85	38.67	1.57
20	850.00	864.00	885.00	865.61	38.77	1.84

注: $G_3 = (f_{ave} - f_{lb}) / f_{lb} \times 100\%$.

表 5 大规模算例下算法 1 的规划结果($N=50$)

Tab. 5 The programming results of algorithm 1 for large-scale instance ($N=50$)

序号	f_{lb}/u	f_{min}/u	f_{max}/u	f_{ave}/u	T_{CPU}/s	$G_4/\%$
1	1 253.00	1 268.00	1 275.00	1 269.50	166.18	1.54
2	1 443.50	1 457.50	1 461.00	1 460.00	170.18	1.85
3	1 484.00	1 496.50	1 513.00	1 506.00	152.72	1.54
4	1 460.50	1 493.00	1 485.00	1 475.50	167.67	1.25
5	1 350.50	1 371.00	1 385.00	1 371.00	156.93	2.11
6	1 405.50	1 424.00	1 430.50	1 425.50	151.97	2.38
7	1 370.50	1 384.00	1 390.00	1 388.50	154.41	1.24
8	1 292.00	1 312.00	1 320.00	1 319.00	169.57	2.14
9	1 251.50	1 272.00	1 272.00	1 272.00	170.13	1.74
10	1 491.50	1 504.00	1 520.00	1 519.00	166.79	1.03
11	1 362.00	1 374.50	1 379.00	1 378.50	157.55	2.12
12	1 312.00	1 323.00	1 340.00	1 333.00	152.29	1.80
13	1 310.00	1 323.00	1 330.50	1 327.00	170.14	1.72
14	1 399.00	1 413.50	1 422.50	1 421.00	168.11	1.37
15	1 177.50	1 199.50	1 211.00	1 202.00	156.43	1.99
16	1 346.50	1 360.50	1 374.00	1 367.00	169.74	1.57
17	1 258.50	1 274.50	1 287.00	1 276.00	155.41	2.38
18	1 457.50	1 471.50	1 475.50	1 474.00	159.29	1.25
19	1 424.50	1 436.00	1 440.00	1 444.00	156.72	1.28
20	1 494.00	1 509.00	1 518.00	1 513.50	151.36	1.39

注： $G_4=(f_{ave}-f_{lb})/f_{lb}\times 100\%$ 。

表 6 调度时段长度对规划结果的影响

Tab. 6 The effect of the duration of scheduling cycle on the programming results

序号	N	$T=M(\infty)$	$T=900u$		$T=600u$		$T=300u$	
		f_{ave}/u	f_{ave1}/u	$H_1/\%$	f_{ave2}/u	$H_2/\%$	f_{ave3}/u	$H_3/\%$
1	30	717.52	717.52	0	775.34	8.06	794.23	10.69
2	40	961.00	1 019.71	6.11	1 024.66	6.62	1 056.50	9.94
3	50	1 270.36	1 326.99	4.46	1 347.53	6.07	1 379.38	8.58
4	75	1 999.46	2 062.80	3.17	2 088.07	4.43	2 154.04	7.73
5	100	2 774.00	2 844.11	2.53	2 864.31	3.26	2 961.24	6.75
6	125	3 558.28	3 639.20	2.27	3 677.37	3.35	3 801.00	6.82
7	150	4 317.54	4 389.06	1.66	4 441.86	2.88	4 587.39	6.25
8	175	5 139.60	5 239.16	1.94	5 289.69	2.92	5 324.18	3.59

注： $H_a=(f_{avea}-f_{ave})/f_{ave}\times 100\%$ ，其中， $a\in\{1,2,3\}$ 。

解任务规模为 50 的收敛曲线,在 150 代内收敛到最优目标值. 因此,在优化结果和计算时间两方面验证了算法具备的良好优化性能.

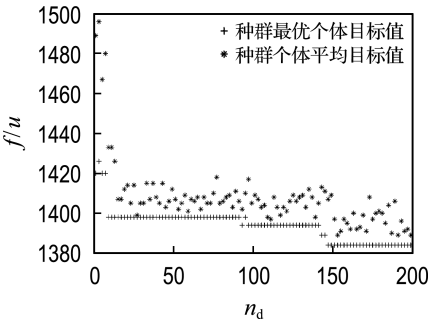


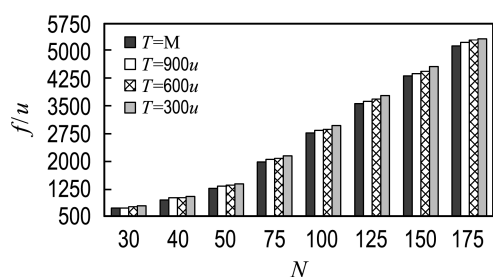
图 6 算法 1 的收敛图

Fig. 6 Convergence diagram of algorithm 1

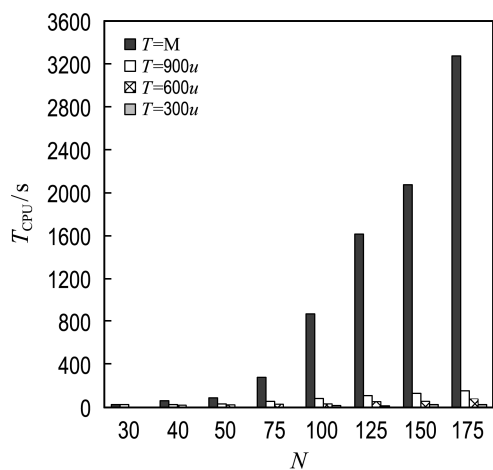
4.2.3 迭代重优化框架有效性验证与分析 在迭代重优化框架下,把某一时间段内的集港作业划分为若干调度时段,使出口箱任务分别在相应的调度时段内生成作业计划,一旦出现延迟到港集装箱的即时请求,可随机分配至与到港时间最近的调度时段内. 为验证该框架的有效性,依次取不同长度的调度时段和不同延迟到港的箱量,求解相应的规划结果.

(1)调度时段长度对规划结果的影响

调度时段长度 T 分别取 M 、 $900u$ 、 $600u$ 和 $300u$,其中 M 表示不对集港作业进行拆分. 规划结果如表 6、图 7 所示,可见:调度时段长度越小,作业完成时间越长,但计算时间明显缩短. 因此,均衡解的质量与计算时间,选取合适的调度时段长度有利于提高码头制订作业计划的效率.



(a) 作业完成时间



(b) 计算时间

图 7 不同调度时段长度下作业完成时间及其计算时间

Fig. 7 The makespan and computing time in various scheduling cycles

(2) 延迟到港的箱量分析

定义延迟程度 β 表示某一交箱时段内延迟到港箱量 N_d 与原始箱量 N_o 的比值, 即 $\beta = N_d / N_o$. 以调度时段 T 为基准, 用算法 1 生成基础作业计划, 调用算法 2 对延迟到港任务执行重调度, 计算结果如图 8 所示.

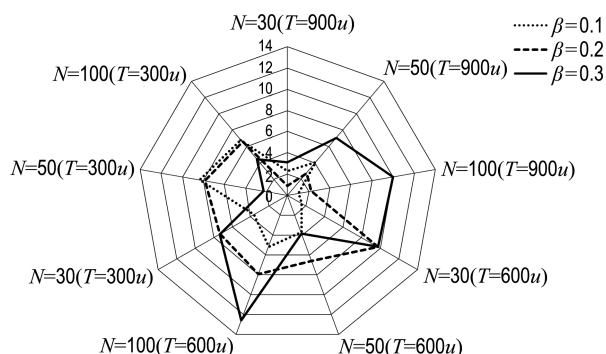


图 8 延迟到港箱量对规划结果的影响

Fig. 8 The influence of delayed arrival volume on the programming result

结果显示, 延迟到港箱量占比 β 越大, 作业时间的增量 Δ 越大; 而随着调度时段 T 的减小, 这一影响逐渐弱化. 其原因在于, 随着延迟任务的数量增加, 重调度对原作业计划的扰动增大, 可行解数量减少(或增加)的变化幅度增大; 而缩短调度时段, 有利于为延迟到港任务分配更适合的调度时段, 从而增加重调度结果需求到最优解的可能性.

5 结 语

自动化集装箱码头的堆场作业受诸多外部不确定因素的干扰, 制约作业效率的提升, 如交通拥堵、雨雪天气影响集卡的到达时间, 对原有作业计划产生扰动. 针对集卡到港时间延迟的堆场起重机动调度问题, 本文提出了迭代重优化框架, 把集港作业的调度期划分为多个时段, 出口箱任务分别在相应的时段内被安排作业, 一旦出现延迟到港集装箱的即时请求, 可临时分配至与到港时间邻近的时段内执行重调度. 借助自动化码头作业数据生成平台, 构造小规模和大规模算例, 验证本文提出的模型和算法的有效性. 通过结果分析验证了模型和算法的有效性, 并提出两方面有益于码头运作优化的管理启示.

一方面, 在计划制订之前, 若待作业的任务量较少, 调度人员可增大调度时段长度, 以保证解的质量, 缩短作业的持续时长; 若任务量较多, 可适当减小调度时段长度, 从而缩短计算时间以提高计算效率. 另一方面, 码头运营者需及时与客户沟通, 获取集卡的行驶状况、预计到达堆场的时间等信息, 为可能延迟到港的集卡提前部署应对策略: 当延迟到港箱量占比较小时, 可适当增大调度时段长度, 保证整体作业水平; 当延迟到港箱量占比较大时, 可减小调度时段长度从而降低重调度对作业系统的扰动.

本文研究还存在一些不足, 即便是分段调度提供了应对集卡作业不确定性的方式, 但难以快速消除集卡到港动态性对系统的扰动. 未来的研究可结合强化学习方法, 在历史作业数据中学习相关场景下的调度经验, 从而及时、准确地调用相应策略生成或调整作业计划.

参考文献:

- [1] CARLO H J, VIS I F A, ROODBERGEN K J. Transport operations in container terminals: Literature overview, trends, research directions and classification scheme [J]. **European Journal of Operational Research**, 2014, **236**(1): 1-13.
- [2] NOSSACK J, BRISKORN D, PESCH E. Container dispatching and conflict-free yard crane routing in an automated container terminal [J]. **Transportation Science**, 2018, **52**(5): 1059-1076.
- [3] ZHEN Lu, YU Shucheng, WANG Shuaian, *et al.* Scheduling quay cranes and yard trucks for unloading operations in container ports [J]. **Annals of Operations Research**, 2019, **273**(1): 455-478.
- [4] BOYSEN N, BRISKORN D, MEISEL F. A generalized classification scheme for crane scheduling with interference [J]. **European Journal of Operational Research**, 2017, **258**(1): 343-357.
- [5] VIS I F A, CARLO H J. Sequencing two cooperating automated stacking cranes in a container terminal [J]. **Transportation Science**, 2010, **44**(2): 169-182.
- [6] GHAREHGOZLI A H, LAPORTE G, YU Yugang, *et al.* Scheduling twin yard cranes in a container block [J]. **Transportation Science**, 2015, **49**(3): 686-705.
- [7] 郑红星, 董译文, 匡海波, 等. 考虑倒箱的混堆装船箱区内场桥调度优化 [J]. **系统工程理论与实践**, 2016, **36**(9): 2362-2373.
ZHENG Hongxing, DONG Yiwen, KUANG Haibo, *et al.* Yard cranes scheduling with relocation at a mixed storage block for ship loading [J]. **Systems Engineering - Theory and Practice**, 2016, **36**(9): 2362-2373. (in Chinese)
- [8] HU Zhihua, SHEU J B, LUO J X. Sequencing twin automated stacking cranes in a block at automated container terminal [J]. **Transportation Research Part C: Emerging Technologies**, 2016, **69**: 208-227.
- [9] HAN Xiaolong, WANG Qianqian, HUANG Jiwei. Scheduling cooperative twin automated stacking cranes in automated container terminals [J]. **Computers and Industrial Engineering**, 2019, **128**: 553-558.
- [10] 秦磊磊, 胡志华, 郑玉馨, 等. 自动化集装箱码头同轨双堆垛起重机交接缓冲贝位优化 [J]. **大连理工大学学报**, 2021, **61**(5): 537-545.
- [11] QIN Leilei, HU Zhihua, ZHENG Yuxin, *et al.* Handover buffer bay optimization for same-track double stacking cranes in automated container terminal [J]. **Journal of Dalian University of Technology**, 2021, **61**(5): 537-545. (in Chinese)
- [11] 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] FAN Houming, KONG Liang, YUE Lijun. Yard crane scheduling and storage space allocation considering container storage, retrieval, and pre-marshalling [J]. **Operations Research and Management Science**, 2021, **30**(6): 26-34. (in Chinese)
- [12] 孙玉姣, 胡祥培, 曾庆成. 集装箱码头装卸作业顺序多目标优化调度模型 [J]. **系统工程理论与实践**, 2020, **40**(1): 195-209.
- [12] SUN Yujiao, HU Xiangpei, ZENG Qingcheng. A multi-objective optimization model for operation scheduling in container terminals [J]. **Systems Engineering - Theory and Practice**, 2020, **40**(1): 195-209. (in Chinese)
- [13] 范厚明, 孔 靓, 马梦知, 等. 交箱序列不确定下的堆场箱位分配及多场桥调度优化 [J]. **系统工程理论与实践**, 2021, **41**(5): 1294-1306.
- [13] FAN Houming, KONG Liang, MA Mengzhi, *et al.* Storage space allocation and multi-yard cranes scheduling in terminal yard with container delivery time uncertain [J]. **Systems Engineering - Theory and Practice**, 2021, **41**(5): 1294-1306. (in Chinese)
- [14] DORNDORF U, SCHNEIDER F. Scheduling automated triple cross-over stacking cranes in a container yard [J]. **OR Spectrum**, 2010, **32**(3): 617-632.
- [15] PILLAC V, GENDREAU M, GUÉRET C, *et al.* A review of dynamic vehicle routing problems [J]. **European Journal of Operational Research**, 2013, **225**(1): 1-11.
- [16] OZBAYGIN G, SAVELSBERGH M. An iterative re-optimization framework for the dynamic vehicle routing problem with roaming delivery locations [J]. **Transportation Research Part B: Methodological**, 2019, **128**: 207-235.
- [17] 宁 涛, 王旭坪, 胡祥培. 行驶受扰延迟下配送车辆调度的干扰管理决策模型 [J]. **系统工程理论与实践**

- 实践, 2019, **39**(5): 1236-1245.
- NING Tao, WANG Xuping, HU Xiangpei. Disruption management decision model for vehicle scheduling under traffic disruption delay [J]. **Systems Engineering - Theory and Practice**, 2019, **39**(5): 1236-1245. (in Chinese)
- [18] 丁一, 陈婷. 基于滚动时域优化策略的多载AGV充电调度[J]. 中国航海, 2020, **43**(2): 80-85.
- DING Yi, CHEN Ting. Charging scheduling of multi-loading AGV based on rolling time domain optimization strategy [J]. **Navigation of China**, 2020, **43**(2): 80-85. (in Chinese)
- [19] 杨珍花, 王滋承, 魏照坤, 等. 挂车装卸时间不确定的甩挂车辆动态调度优化[J]. 系统工程理论与实践, 2021, **41**(5): 1081-1095.
- YANG Zhenhua, WANG Zicheng, WEI Zhaokun, *et al.* Dynamic tractor and semi-trailer routing problem with the non-deterministic operation time of trailers [J]. **Systems Engineering - Theory and Practice**, 2021, **41**(5): 1081-1095. (in Chinese)
- [20] NG W C. Crane scheduling in container yards with inter-crane interference [J]. **European Journal of Operational Research**, 2005, **164**(1): 64-78.
- [21] VIS I F A, ROODBERGEN K J. Scheduling of container storage and retrieval [J]. **Operations Research**, 2009, **57**(2): 456-467.
- [22] 秦进, 倪玲霖, 王承娜, 等. 集装箱码头岸桥调度优化模型及算法[J]. 西南交通大学学报, 2013, **48**(1): 184-192.
- QIN Jin, NI Linglin, WANG Chengna, *et al.* Optimization model and algorithm for quay crane scheduling in container terminals [J]. **Journal of Southwest Jiaotong University**, 2013, **48**(1): 184-192. (in Chinese)

Research on dynamic scheduling for yard crane with truck arrival time delay

WANG Yaozong, HU Zhihua*, TIAN Xidan, CHEN Wanting

(Logistics Research Center, Shanghai Maritime University, Shanghai 201306, China)

Abstract: For the container collection operations at terminals, delayed arrival of the truck may cause the planned scheme of yard crane to become sub-optimal or even infeasible. Therefore, it is necessary to update the scheme in time for the delayed arrival tasks. Considering the dynamic scheduling problem with truck arrival time delay, based on the batch-based task assigned strategy, the scheduling period is divided into several scheduling cycles, and the schemes are generated within each cycle. A mixed integer programming model is established to sequence the handling tasks under the minimization of the makespan. Moreover, a lower bound derivation model is proposed. An iterative solution framework is proposed, in which the scheme is re-optimized whenever the truck arrival time delay is revealed. In the framework, a genetic algorithm is designed to solve the planned scheme of the corresponding batch tasks in each cycle, and a greedy insertion algorithm is proposed to re-optimize batch tasks with delayed arrival and update the scheme. The experimental results show that the model and the lower bound derivation model have correctness and validity, and the algorithm has good performance. Shortening the scheduling cycles helps to generate the scheme quickly for crane operation. The larger the proportion of delayed arrival containers, the bigger the makespan. And this effect becomes weakened by the reduction in the scheduling cycle.

Key words: yard crane; truck arrival time delay; dynamic scheduling; iterative re-optimization; greedy strategy