

文章编号:1006-2467(2020)07-0718-10

DOI: 10.16183/j.cnki.jsjtu.2018.339

# 考虑平板车合作运输的船舶分段堆场间调度

李柏鹤<sup>1,2</sup>, 蒋祖华<sup>1,2</sup>, 陶宁蓉<sup>3</sup>, 孟令通<sup>1,2</sup>, 郑虹<sup>1</sup>

(1. 上海交通大学 机械与动力工程学院, 上海 200240; 2. 上海交通大学 高新船舶与深海开发装备协同创新中心, 上海 200240; 3. 上海海洋大学 工程学院, 上海 201306)

**摘要:** 针对船厂运输任务分段的平板车缺乏有效调度的现状,以多种类型平板车和任务分段的承重约束、任务时间窗约束作为约束条件,并考虑实际运输时单个大型分段质量超出现有单个平板车最大承重能力的情况,采用多个平板车合作运输的策略解决该问题.建立以平板车完成任务的空载行驶时间以及平板车的等待时间权重和作为优化目标的模型,设计禁忌搜索算法对模型进行求解.最后以某船厂实际数据为例,对模型的应用进行实例验证.研究表明,所提方法可以求解较优的调度方案,并解决大型分段的运输,以实现资源的高效利用.

**关键词:** 堆场间调度; 多类型平板车; 多车合作运输; 时间窗约束; 禁忌搜索算法

**中图分类号:** TP 391; U 672

**文献标志码:** A

## Ship Block Transportation Scheduling Considering Cooperative Transportation of Flatcars

LI Baihe<sup>1,2</sup>, JIANG Zuhua<sup>1,2</sup>, TAO Ningrong<sup>3</sup>, MENG Lingtong<sup>1,2</sup>, ZHENG Hong<sup>1</sup>

(1. School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China; 2. Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration, Shanghai Jiao Tong University, Shanghai 200240, China; 3. College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China)

**Abstract:** This paper aims at solving the problem of ineffective dispatching status for flatcars in shipyard transportation tasks using the task time window and the loading constraints of various types of flatcars and blocks as constraints. Considering the fact that the weight of a single large block exceeds the maximum loading capacity of a flatcar during actual transportation, this paper proposes a method of cooperative transportation using multiple flatcars. Aimed at establishing an optimization model, the optimal objects set the no-load travel time and waiting time for the task as the weight of the flatcars. Then, a tabu search algorithm is designed to solve the model proposed. Taking the actual data of a shipyard as an example, the application of the model is verified by specific examples. The results show that the method proposed provides a better scheduling scheme which can solve the transportation problem of large blocks and better realize the efficient utilization of resources.

**Key words:** dispatch between storage yards; multi-type flatcars; multi-flatcar cooperative transportation; time window; tabu search algorithm

收稿日期:2018-10-30

基金项目:国家自然科学基金(71501125),工信部高技术船舶项目联装函[2018]35号(18GC04246)资助项目

作者简介:李柏鹤(1993-),男,黑龙江省齐齐哈尔市人,硕士生,研究方向为船舶分段堆场间调度.

通信作者:蒋祖华,男,教授,博士生导师,电话(Tel.):021-34206819;E-mail:zhjiang@sjtu.edu.cn.

符号说明

$c_l^m$ —第  $l$  辆平板车的承重能力  
 $h$ —工厂或堆场序号,  $h = 1, 2, \dots, k$   
 $i$ —运输任务编号,  $i = 1, 2, \dots, n$   
 $j$ —运输任务编号,  $j = 1, 2, \dots, n$   
 $l$ —平板车编号,  $l = 1, 2, \dots, p$   
 $m_i$ —第  $i$  个任务分段的质量  
 $Q$ —无穷大的正数  
 $t^E$ —车场开放时间,即平板车最早可出发时间  
 $t^L$ —车场关闭时间,即平板车最晚回归时间  
 $t_i^b$ —执行第  $i$  个任务所需要的时间  
 $t_i^e$ —第  $i$  个任务的时间窗起始点,即任务可以开始执行的时间点  
 $t_i^s$ —第  $i$  个任务的实际开始执行时间

船舶制造是典型的拉式生产方式,船体建造工艺流程以中间产品“分段”作为组织生产的基本作业单元<sup>[1]</sup>. 分段在船坞之外的作业场或堆场进行组立装配、预舾装、涂装和舾装等建造工艺,最后到船坞进行总组搭载. 船体分段质量较大,一般为 200~300 t,个别可达到 500 t. 因此,分段在作业场地和堆场之间的运输主要依靠重型平板车.

一般来说,船厂中的平板车资源都是有限的. 目前,某大型船厂平板车的最大承重能力为 425 t,将质量超出现有平板车最大承重能力的单个分段,称之为大型分段. 大型分段需要采用多个平板车进行合作运输. 实际中,某大型船厂每天要完成约 100 个分段的运输. 研究如何对船厂有限的运输资源进行充分利用、如何运输大型分段、如何提高平板车的利用率以及如何在保证任务准时进行的情况下,降低无效作业的等待时间等问题,具有十分重要的现实应用价值.

国内外船舶分段运输问题的研究相对较少. Joo 等<sup>[2]</sup>以拖期时间和延迟时间的权重和作为目标函数,设计了遗传和启发式进化两种算法对问题进行求解. Wang 等<sup>[3]</sup>以船舶分段运输过程中平板车的空驶时间、拖期时间和延迟时间的总和作为目标函数,提出一种贪心启发式算法对问题进行求解. Salehi 等<sup>[4]</sup>认为运输过程中排放的气体在温室气体排放中占较大的比例,因此提出一个双目标混合整数非线性规划,目标函数为总运输成本和总碳排放量,并提出一种构造式启发式算法用于求解问题. Hu 等<sup>[5]</sup>考虑了多个单类型平板车运输一个大型分段的问题,建立了以最小化完工时间为优化目标的数学模型,为了提高求解方法的计算性能,用贪婪插

$t_{ij}^v$ —执行完第  $i$  个任务后,空驶到第  $j$  个任务的起点所花费的时间,  $i = 0$  或者  $j = 0$  表示平板车要去车场  
 $t_i^v$ —第  $i$  个任务的时间窗终点,即任务必须开始执行的时间截点  
 $x_i$ —第  $i$  个任务是否是超重任务,即一个平板车无法运输,若为 1 表示是;若为 0 表示否  
 $y_{il}$ —第  $l$  辆平板车是否执行第  $i$  个任务的 0-1 决策变量  
 $z_{ijl}$ —第  $l$  辆平板车上第  $i, j$  个任务执行顺序的 0-1 决策变量,即如果第  $l$  辆平板车执行完第  $i$  个任务后,接着执行第  $j$  个任务,则  $z_{ijl} = 1$ ;否则  $z_{ijl} = 0$   
 $z_{i0l}$ —第  $l$  辆平板车执行其最后 1 个任务的 0-1 决策变量  
 $z_{0il}$ —第  $l$  辆平板车执行其第 1 个任务的 0-1 决策变量  
 $\alpha$ —平板车执行所有任务的空载行驶时间权重  
 $\beta$ —平板车执行所有任务的等待时间权重

入算法和遗传算法生成初始解,设计顺序插入法求解模型. 张志英等<sup>[6]</sup>构建了考虑堆场信息、分段进出场次序等因素的最短场路模型并对其进行优化,以最小化临时分段移动量和平板车在堆场中的行驶距离为优化目标,确定分段在堆场中的最优停放位置和进出场路. 王冲等<sup>[7]</sup>为解决船厂平板运输车搬运船舶分段的日程计划问题,建立了利用最少数量的平板车完成分段搬运作业,以及所有分段搬运作业完成时间最小化的两阶段优化模型,提出了基于遗传算法的两种编码方法以实现模型求解.

上述研究均没有考虑当单个分段的质量超出平板车最大承重能力时,应该如何调度多种类型的平板车. 而本文的研究目的是解决该运输难点,采用多个平板车合作运输大型分段的策略,且设定有多种承重能力的平板车可供选择. 同时,为了保证生产的准时性,引入任务时间窗紧约束作为约束条件,开展具体研究.

1 问题描述

研究的问题可描述为有  $n$  个待执行的运输任务和  $p$  个运输平板车,每个运输任务包括:任务编号、分段编号、分段质量、运输起点、运输终点、运输时间窗(任务最早可执行时间以及任务最晚的执行时间);每个运输平板车包括:平板车编号、平板车承重能力. 每个任务必须由满足其分段承重需求的平板车在时间窗范围内开始从运输起点运送至运输终点. 其中对于大型分段,一个平板车由于承重能力限制无法运输,此时就需要多个平板车合作运输. 某船厂的道路路口情况、堆场位置以及平板车停放位置如图 1 所示. 其中:虚线为平板车空载行驶的路径;

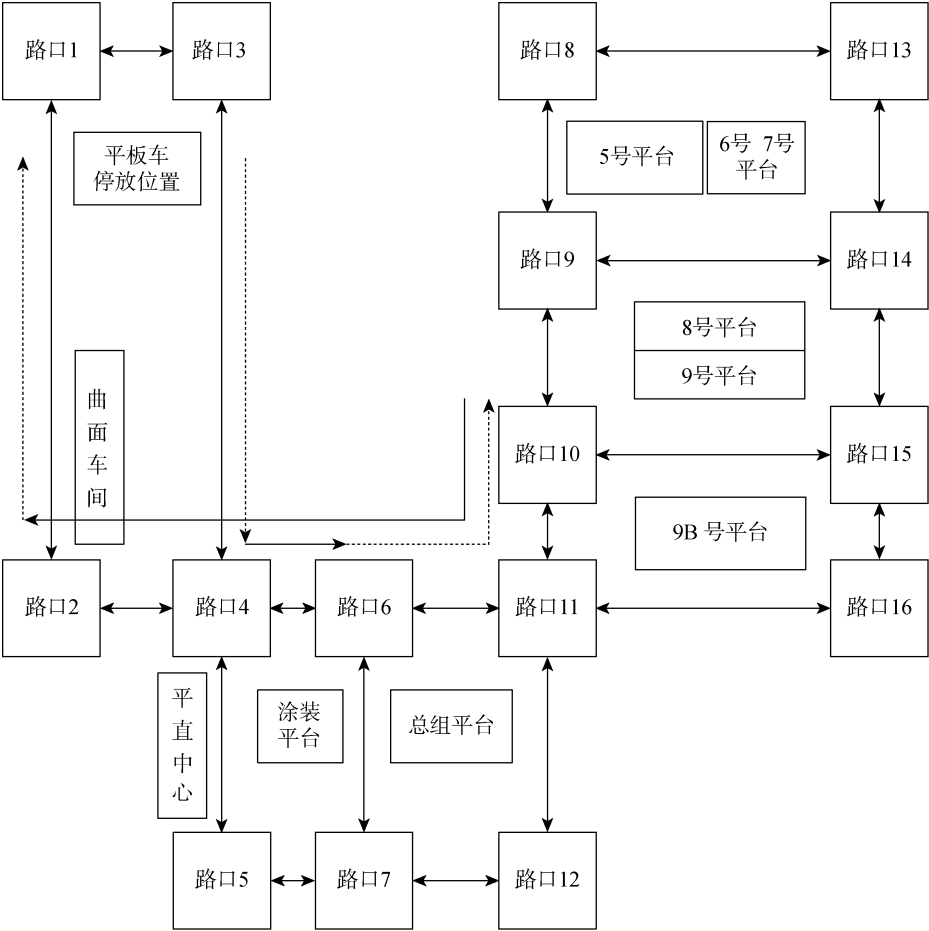


图 1 道路及堆场布局图  
Fig. 1 Distribution of roads and yards

实线为平板车负载行驶的路径. 假设平板车  $L_1$  的任务序列为任务  $O_i$  和  $O_j$ , 任务  $O_i$  为将目标分段从平直中心运输到涂装平台; 任务  $O_j$  为将目标分段从 9 号平台运输至曲面车间. 则平板车的实际执行流程为从平板车的停放位置出发, 行驶至平直中心并提取分段, 负载行驶至涂装平台后放下分段; 然后, 空驶运行至 9 号平台并提取分段, 负载行驶至曲面车间后空驶至平板车停放位置, 结束运输.

对于大型分段, 两个平板车之间的合作运输工作原理如图 2 所示. 其中:  $L_i$  为运输平板车序号;  $B_i$  为船舶分段;  $S_h$  为工作场或堆场序号. 以平板车  $L_1$  和  $L_2$  为例, 平板车  $L_1$  从车场出发, 空驶至  $S_1$  堆场, 装载分段  $B_1$  后负载行驶至  $S_3$  堆场, 卸下分段; 空驶至  $S_5$ , 由于分段  $B_3$  是大型分段, 需要  $L_1$  和  $L_2$  合作运输, 所以等  $L_1$  和  $L_2$  均到达  $S_5$  后, 开始运输分段  $B_3$  至  $S_6$ . 对于合作运输大型分段的平板车而言, 若一辆平板车早到, 一辆平板车晚到, 将会导致早到的平板车只能原地等待, 这在生产过程中是极度浪费资源的, 且等待时间也会造成生产进程的滞后. 因

此, 从平板车运输任务分段的全过程出发, 以平板车执行任务时的空载行驶时间和平板车由于早于时间窗到达产生的等待时间, 或者平板车合作运输时产生的等待时间的权重和作为优化目标进行研究, 以期达到合作运输效率优化的效果.

考虑到堆场的实际情况以及便于堆场调度问题的研究, 作出如下假设: ① 平板车运输分段过程一旦开始执行, 就不可被其他任务打断; ② 平板车在同一时刻最多只能运输一个分段; ③ 不考虑道路因素 (道路上是否有平板车、道路宽度、道路上可以通行多少平板车) 对平板车通行的影响; ④ 为了方便说明, 假定大型分段的质量不超过两个平板车的承重能力之和; ⑤ 合作运输的平板车可以保持一样的速度.

本研究要解决的问题有: ① 运输任务的最优序列; ② 运输任务对应哪些平板车执行, 大型任务由哪些平板车运输; ③ 各个任务的开始执行时间和完成时间. 时间变量单位均为分钟, 以开始工作时间作为 0 点, 其余时间均在此基础上累加.

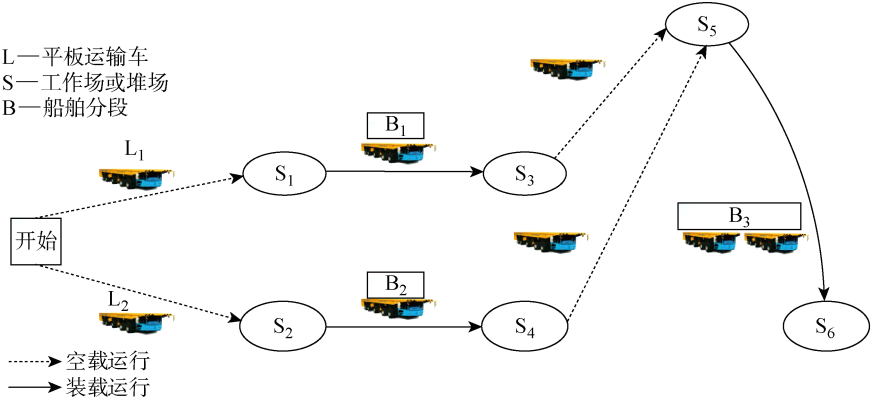


图 2 平板车合作运输示意图

Fig. 2 Diagram of multi-flatcars cooperative transportation

2 堆场间运输调度模型

2.1 目标函数

建立平板车在堆场间的运输调度模型,优化目标在调度过程中的时间因素. 首先,平板车执行任务所消耗的时间是确定的,但平板车执行任务之间的空载行驶时间是不确定的,其会随着平板车上任务序列的改变而改变;其次,平板车早到堆场或工作场时,由于时间窗紧约束的原因,会使平板车产生等待时间;再次,由于考虑的是平板车合作运输的大型分段,合作运输的平板车如果到达堆场或者工作场的时间不同,也会使早到的平板车产生等待时间. 而等待时间会产生附加成本,即资源的浪费,进而影响生产进程. 因此,模型以平板车执行任务的空载行驶时间与等待时间的权重和作为目标函数,优化运输成本. 现有研究文献中,以最小化任务完工时间作为目标函数的也较多,本文模型中由于考虑了时间窗紧约束,严格控制每个任务的执行时间,所以不需要优化任务完工时间.

(1) 平板车完成所有任务的任务间空驶时间:

$$f_1 = \sum_{l=1}^p \sum_{j=1, j \neq i}^n \sum_{i=1}^n z_{ijl} t_{ij}^u$$

(2) 平板车从车场出发空驶到其第 1 个任务起点的时间,以及平板车执行完任务空驶返回车场的时间:

$$f_2 = \sum_{l=1}^p \sum_{i=1}^n z_{0il} t_{0i}^u + \sum_{l=1}^p \sum_{i=1}^n z_{i0l} t_{i0}^u$$

(3) 平板车早于时间窗到达或者由于协同运输而产生的等待时间:

$$f_3 = \sum_{l=1}^p \sum_{j=1, j \neq i}^n \sum_{i=1}^n z_{ijl} (t_j^s - t_i^s - t_i^b - t_{ij}^u)$$

目标函数以上述三者的权重和作为优化目标,

使其最小化.

$$\min f = \alpha(f_1 + f_2) + \beta f_3$$

2.2 约束条件

(1) 非大型分段由 1 个且只由 1 个平板车进行运输,大型分段由 2 个平板车进行运输:

$$\sum_{l=1}^p y_{il} = 1 + x_i, \quad \forall i$$

(2) 每个平板车上的第 1 个任务只有 1 个:

$$\sum_{i=1}^n z_{0il} = 1, \quad \forall l$$

(3) 每个平板车上的最后 1 个任务只有 1 个:

$$\sum_{i=1}^n z_{i0l} = 1, \quad \forall l$$

(4) 平板车上执行的相邻 2 个任务之间的时间存在约束:

$$t_i^s + t_i^b + z_{ijl} t_{ij}^u - t_j^s \leq (1 - z_{ijl})Q, \quad \forall i, j, l$$

(5) 任务时间窗约束:

$$t_i^e \leq t_i^s \leq t_i^v, \quad \forall i$$

(6) 车场时间窗约束:

$$t_i^s + t_i^b + t_{i0}^u \leq t^L, \quad \forall i$$

$$t^E \leq t_i^s - t_{0i}^u, \quad \forall i$$

(7) 平板车承重能力与任务分段质量约束:

$$m_i \leq \sum_{l=1}^p y_{il} c_l^m, \quad \forall i$$

(8) 对于大型分段任务来说,执行任务的平板车必须都到任务始点地,任务才能开始执行:

$$t_i^s + t_i^b + z_{ijl} t_{ij}^u - t_j^s \leq (1 - x_j)Q, \quad \forall i, j, l$$

(9) 保证平板车上执行完一个任务后接下来的任务要满足在解中的出现次数约束:

$$\sum_{j=1, j \neq i}^n z_{jil} + z_{0il} = y_{il}, \quad \forall i, l$$

(10) 保证平板车上执行的任务要满足在解中的出现次数约束:

$$\sum_{j=1, j \neq i}^n z_{ijl} + z_{i0l} = y_{il}, \quad \forall i, l$$

(11) 决策变量的取值范围约束:

$$\sum_{j=1, j \neq i}^n z_{ijl} \leq y_{il}, \quad \forall i, l$$

3 基于禁忌搜索的模型求解

船舶分段堆场间的运输调度模型为混合整数线性规划问题. 对于小规模问题, 可以采用商业求解器, 如 CPLEX 求出最优解. 但在实际应用中, 需要搬运的分段数量较大, 现有求解器很难在有限时间内求出高效解. 禁忌搜索算法在组合优化问题中具有较强的效果, 广泛应用于旅行商问题以及车辆路径问题. 禁忌搜索算法依靠禁忌表的存在, 可以在搜索中跳出局部最优, 不同的构造邻域方法也增加了邻域空间的多样性, 具有较好的求解效果. 因此, 设计禁忌搜索算法在分钟级别的时间内高效求解模型, 具有较高的实际应用价值.

3.1 解的编码方式

模型解的编码与解码方式如图 3 所示. 编码方式为 3 个一维数组串行, 表示平板车运输的分段及其运输顺序. 如图 3(a)中, 第 1 行表示任务序列编号, 第 2 行和第 3 行表示任务被执行的平板车编号. 若第 2 行和第 3 行的平板车编号相同, 则表示该任务是由同一个平板车执行的; 若第 2 行和第 3 行平板车编号不相同, 则表示该任务是由两个平板车合

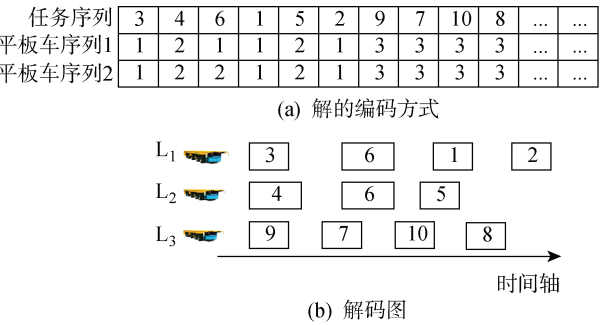


图 3 编码与解码图  
Fig. 3 Coding and decoding

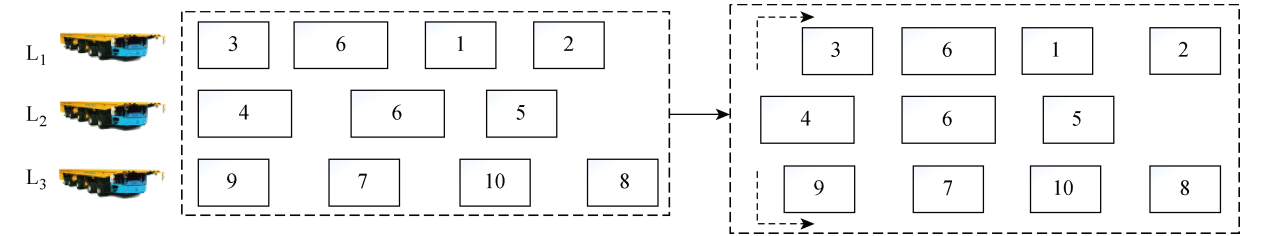


图 4 大型分段的时间更新示意图  
Fig. 4 Diagram of time update of large-blocks

作运输的. 解码后的运输方案如图 3(b)所示, 即平板车 1 执行第 3、6、1、2 个任务; 平板车 2 执行第 4、6、5 个任务; 平板车 3 执行第 9、7、10、8 个任务. 其中任务 6 为大型任务, 由平板车 1 和平板车 2 合作运输. 同理, 如果考虑 3 个平板车合作运输, 则用 4 个一维数组串行表示平板车运输的分段及运输顺序.

3.2 构造初始种群

采用随机构造初始种群的方式, 先随机生成任务序列, 再根据承重约束为每个任务分配平板车, 对于大型任务则随机产生两个符合其承重约束的平板车. 由于模型中存在时间窗约束, 很可能构造的个体是不可行的, 但对于不可行解, 在计算适应度时会给予惩罚, 所以在迭代中, 不可行解会被逐渐取代.

3.3 个体适应度评估

个体代表模型的解, 个体适应度函数就是模型需要优化的目标函数, 对于所提模型的适应度越低, 代表个体越好. 个体解码后, 可以获得每个车的运输方案、计算平板车从车场出发至执行第 1 个任务的空载行驶时间, 以及执行后续任务的空载行驶时间与执行完所有任务后回到车场的空载行驶时间. 在计算平板车等待时间时, 首先需要注意大型任务要等到执行大型任务的所有平板车均到达其起始点才可以开始执行. 大型分段的时间更新示意图如图 4 所示. 由图 4 可知, 第 1 辆平板车上任务 6 的执行时间要后移, 因为第 2 辆平板车还未到达. 其次, 需要调整每个平板车上第 1 个任务的开始执行时间, 以最小化等待时间. 由于任务 6 是大型任务, 任务 6 的开始时间被延后, 导致  $L_1$  在执行完任务 3 后存在一段等待时间, 此时需要对任务 6 之前的任务开始执行时间进行调整. 将任务 3 的开始时间延后, 此时等待时间就消除了(见图 4 右框). 同理, 第 3 辆平板车也需要对任务 9 的开始时间进行调整, 这样可以减少平板车的等待时间. 在算法求解过程中可以出现不可行解, 但一定要对其适应度加上一定的惩罚, 且根据不可行解的出现情况, 动态改变惩罚系数.

3.4 邻域结构

如何从当前解出发,构造高效的邻域解是禁忌搜索算法的核心所在.设计 5 种邻域结构,具体操作过程如下.

(1) M1. 让 1 个任务从 1 辆平板车迁移到另外 1 辆平板车.

**步骤 1** 在解的任务序列上随机地选择 1 个任务.

**步骤 2** 若该任务是大型任务,则在满足大型任务的承重约束下生成新的平板车赋值给解的第 3 行序列;若该任务不是大型任务,则在满足任务承重约束下生成新的平板车,并赋值给解的第 2 行和第 3 行序列.

M1 的具体领域结构图如图 5 所示.

(2) M2. 让 1 辆平板车上的 2 个任务的执行顺序发生改变.

**步骤 1** 在解的平板车序列中随机选择 1 个平板车.

**步骤 2** 在该平板车上选择 2 个任务,交换其位置及平板车号.

M2 的具体领域结构图如图 6 所示.

(3) M3. 让两个车之间交换任务片段,这样更有利于保护时间窗的约束<sup>[8]</sup>.

**步骤 1** 在解的平板车序列中选择 2 个平板车.

**步骤 2** 选择平板车上除了开始和末尾的 1 个任务序列片段,即 2 个连续的任务,与另外 1 个平板车上的任务序列片段进行交换.

**步骤 3** 检查新的解是否满足大型任务的承重约束,若不满足,则返回步骤 1;若满足则结束.

M3 的具体领域结构图如图 7 所示.

(4) M4. 让 1 辆车的上 1 个任务迁移到该车上的另 1 个任务的前面,以增加任务序列的多样性.

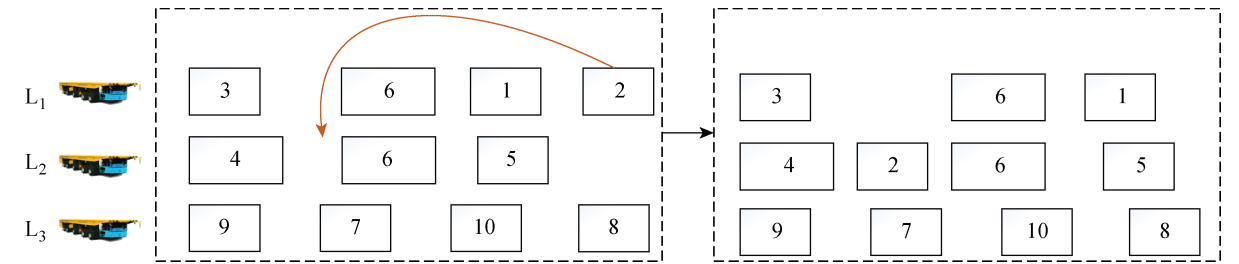


图 5 M1 的邻域结构  
Fig. 5 Neighborhood structure of M1

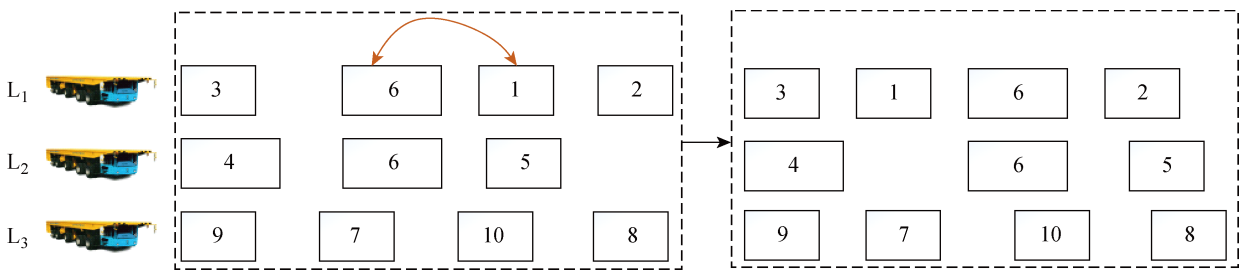


图 6 M2 的邻域结构  
Fig. 6 Neighborhood structure of M2

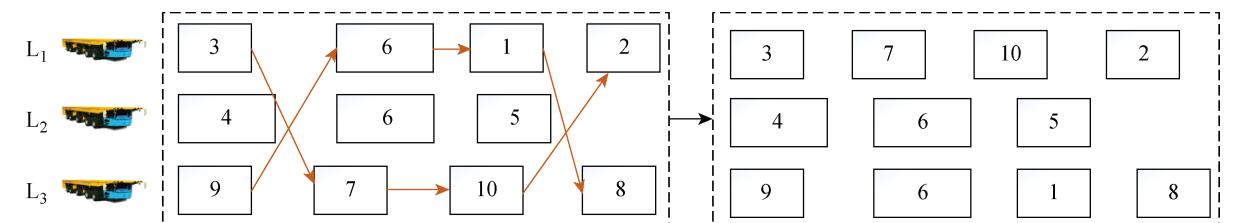


图 7 M3 的邻域结构  
Fig. 7 Neighborhood structure of M3



**步骤1** 在解的平板车序列中选择1辆平板车.

**步骤2** 选择平板车上的1个任务,再选择另外1个任务.

**步骤3** 将选择的第1个任务迁移到选择的第2个任务位置前.

M4的具体邻域结构图如图8所示.

(5) M5. 让1辆车的上1个任务迁移到该车上

的另1个任务之后,以增加任务序列的多样性.

**步骤1** 在解的平板车序列中选择1辆平板车.

**步骤2** 选择平板车上的1个任务,再选择另1个任务.

**步骤3** 将选择的第1个任务迁移到选择的第2个任务位置后.

M5的具体邻域结构图如图9所示.

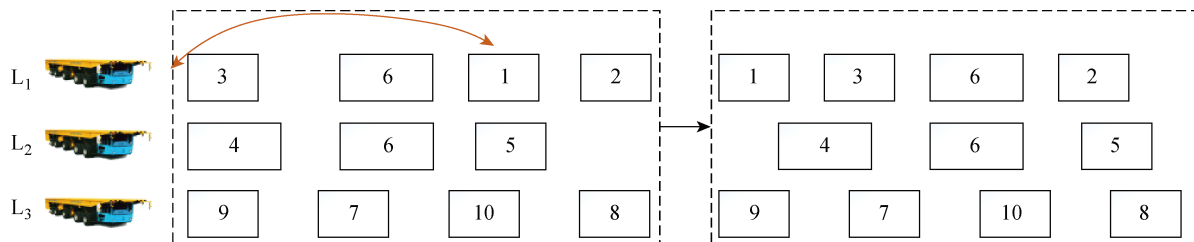


图8 M4的邻域结构

Fig. 8 Neighborhood structure of M4

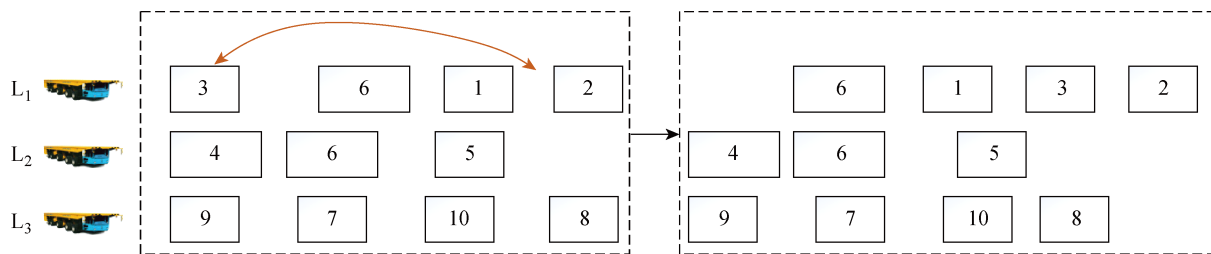


图9 M5的邻域结构

Fig. 9 Neighborhood structure of M5

### 3.5 惩罚系数的更新

由于模型的约束较多,很难一直在可行解空间搜索模型的解,且对于上述5种邻域构造方法均不保证时间窗约束的成立,所以对于不满足时间窗约束的个体,在计算个体适应度时需加上额外的惩罚.文献[9]通过研究证明,动态系数的方式效果更好,即如果本次迭代中最好的解不是可行解,则惩罚系数应变大,以减少不可行解的存在;如果本次迭代中最好的解是可行解,则惩罚系数应变小,以增大不可行解的存在.

### 3.6 禁忌表及特赦规则

由于上述设定的5种邻域结构是不同的,所以针对每个邻域结构分别建立禁忌表,以防止每种策略产生重复的邻域解.特赦规则为如果由禁忌表中的操作产生的解优于当前最优解,则触发特赦规则.对于第1种邻域结构,用 $(i, l_1, l_2)$ 表示任务 $i$ 由 $L_{l_1}$ 车迁移到 $L_{l_2}$ 车,则用 $(i, l_2, l_1)$ 作为禁忌元素加入到禁忌表中以防止最优解的操作重复出现;同理,第2种邻域结构用 $(i, j, l)$ 表示车 $L_l$ 上任务 $i$ 和任务 $j$ 交换,则禁忌元素为 $(j, i, l)$ 和 $(i, j, l)$ ;第3种邻域

结构用 $(i_1, j_1, l_1; i_2, j_2, l_2)$ 表示 $L_{l_1}$ 车上 $(i_1, j_1)$ 任务片段与 $L_{l_2}$ 车上 $(i_2, j_2)$ 任务片段交换,则禁忌元素为 $(i_1, j_1, l_2; i_2, j_2, l_1)$ ;第4种邻域结构用 $(i, j, l)$ 表示 $L_l$ 车上任务 $i$ 迁移到任务 $j$ 前面;第5种邻域结构用 $(i, j, l)$ 表示 $L_l$ 车上任务 $i$ 迁移到任务 $j$ 后面.禁忌长度与邻域空间有关,一般为 $10 \lg n$ .

## 4 数值试验

### 4.1 实例验证

为了验证所提模型及算法的可行性与正确性,以上海某船厂的实际调度任务为例,利用Java编程求解问题,程序运行电脑配置为2.7 GHz Intel Core i5 处理器、8 GB 1867 MHz DDR3 内存、MacOS High Sierra 系统.系统输入为:①任务分段信息;②平板车信息;③时间窗约束信息;④算法参数——禁忌搜索的邻域空间大小为20 000,禁忌长度为57,迭代次数为300,无改善解代数为100.运输任务信息和时间窗约束信息如表1所示,平板车信息如表2所示.其中: $\Delta t = [t_i^s, t_i^e]$ 为时间窗.

获得的调度结果如表3所示.由表3可知,平板

表 1 运输任务和时间窗信息表

Tab. 1 List of transportation tasks and time windows

O <sub>i</sub>	始点堆位	终点堆位	m <sub>i</sub> /t	Δt/min	O <sub>i</sub>	始点堆位	终点堆位	m <sub>i</sub> /t	Δt/min
O <sub>1</sub>	P6207	T1101	238	[0,200]	O <sub>21</sub>	P6207	T1101	238	[150,450]
O <sub>2</sub>	T1805	T1301	244	[0,200]	O <sub>22</sub>	T1805	T1301	244	[150,450]
O <sub>3</sub>	P6207	涂装车间	264	[0,200]	O <sub>23</sub>	P6207	涂装车间	264	[150,450]
O <sub>4</sub>	P7101	涂装车间	276	[0,200]	O <sub>24</sub>	P7101	涂装车间	276	[150,450]
O <sub>5</sub>	P7107	总组平台	280	[0,200]	O <sub>25</sub>	P7107	总组平台	280	[150,450]
O <sub>6</sub>	T1805	小组立车间	283	[0,200]	O <sub>26</sub>	T1805	综合堆场	283	[150,450]
O <sub>7</sub>	P6207	小组立车间	283	[0,200]	O <sub>27</sub>	P6207	集配堆场	283	[150,450]
O <sub>8</sub>	P6207	涂装车间	288	[0,200]	O <sub>28</sub>	P6207	涂装车间	450	[250,520]
O <sub>9</sub>	P6207	总组平台	450	[0,200]	O <sub>29</sub>	P6207	总组平台	200	[250,520]
O <sub>10</sub>	P6207	总组平台	220	[0,200]	O <sub>30</sub>	P6207	总组平台	220	[250,520]
O <sub>11</sub>	P6207	T1125	220	[50,250]	O <sub>31</sub>	P6207	T1125	220	[250,520]
O <sub>12</sub>	曲面车间	总组平台	222	[50,250]	O <sub>32</sub>	曲面车间	总组平台	222	[250,520]
O <sub>13</sub>	P6207	涂装车间	226	[50,250]	O <sub>33</sub>	P6207	涂装车间	226	[250,520]
O <sub>14</sub>	曲面车间	小组立车间	226	[50,250]	O <sub>34</sub>	曲面车间	加工堆场	226	[250,520]
O <sub>15</sub>	P7107	总组平台	242	[50,250]	O <sub>35</sub>	P7107	总组平台	242	[250,520]
O <sub>16</sub>	P7314	T1125	258	[100,300]	O <sub>36</sub>	P7314	T1125	258	[350,520]
O <sub>17</sub>	P7511	曲面车间	276	[100,300]	O <sub>37</sub>	P7511	曲面车间	276	[350,520]
O <sub>18</sub>	P7314	集配堆场	450	[100,300]	O <sub>38</sub>	P7314	平直广场	450	[350,520]
O <sub>19</sub>	P6207	T1101	319	[100,300]	O <sub>39</sub>	P6207	T1101	319	[350,520]
O <sub>20</sub>	T1805	P7314	326	[100,300]	O <sub>40</sub>	T1805	P7314	326	[350,520]

表 2 平板车信息  
Tab. 2 Flatcar informations

L <sub>l</sub>	c <sub>l</sub> <sup>m</sup> /t
L <sub>7</sub>	250
L <sub>8</sub>	270
L <sub>9</sub>	325
L <sub>13</sub>	380
L <sub>14</sub>	380
L <sub>15</sub>	425

车与执行任务之间满足承重约束;任务开始执行时间满足时间窗约束;第 9、18、28、38 号任务为大型任务,由两辆平板车协同运输.实例结果验证了所提算法的有效性.然而,调度结果中出现了承重能力为 425、380 t 的平板车运输质量为 200 t 分段的情况,这主要是由于调度结果是从整个调度周期的总目标函数考虑而导致的.如果质量为 200 t 的分段都由承重能力为 250 t 或 270 t 的平板车运输,很有可能造成 250、270 t 的平板车之后的运输任务不满足时间窗约束或者运输成本较高,所以算法在求解时舍弃了成本较高的解.

表 3 调度结果  
Tab. 3 Scheduling results

O <sub>i</sub>	L <sub>l</sub>	c <sub>l</sub> <sup>m</sup> /t	t <sub>i</sub> <sup>s</sup> /min	O <sub>i</sub>	L <sub>l</sub>	c <sub>l</sub> <sup>m</sup> /t	t <sub>i</sub> <sup>s</sup> /min
O <sub>1</sub>	L <sub>14</sub>	380	23. 13	O <sub>21</sub>	L <sub>13</sub>	380	295. 43
O <sub>2</sub>	L <sub>14</sub>	380	10. 43	O <sub>22</sub>	L <sub>9</sub>	325	150. 00
O <sub>3</sub>	L <sub>9</sub>	325	162. 78	O <sub>23</sub>	L <sub>13</sub>	380	182. 31
O <sub>4</sub>	L <sub>13</sub>	380	18. 80	O <sub>24</sub>	L <sub>9</sub>	325	340. 97
O <sub>5</sub>	L <sub>13</sub>	380	136. 49	O <sub>25</sub>	L <sub>14</sub>	380	340. 58
O <sub>6</sub>	L <sub>13</sub>	380	10. 43	O <sub>26</sub>	L <sub>13</sub>	380	284. 52



(续表)

$O_i$	$L_l$	$c_l^m/\text{t}$	$t_i^s/\text{min}$	$O_i$	$L_l$	$c_l^m/\text{t}$	$t_i^s/\text{min}$
$O_7$	$L_{13}$	380	76.89	$O_{27}$	$L_{15}$	425	327.92
$O_8$	$L_{14}$	380	81.93	$O_{28}$	$L_8, L_7$	250+270	300.69
$O_9$	$L_{15}, L_7$	425+250	192.31	$O_{29}$	$L_{13}$	380	407.39
$O_{10}$	$L_{14}$	380	192.31	$O_{30}$	$L_7$	250	250.00
$O_{11}$	$L_8$	270	245.84	$O_{31}$	$L_9$	325	280.09
$O_{12}$	$L_8$	270	222.36	$O_{32}$	$L_9$	325	384.22
$O_{13}$	$L_{15}$	425	136.58	$O_{33}$	$L_{14}$	380	284.86
$O_{14}$	$L_{15}$	425	131.54	$O_{34}$	$L_8$	270	292.60
$O_{15}$	$L_{14}$	380	239.03	$O_{35}$	$L_{14}$	380	385.81
$O_{16}$	$L_9$	325	220.91	$O_{36}$	$L_{13}$	380	353.26
$O_{17}$	$L_{15}$	425	238.43	$O_{37}$	$L_{14}$	380	430.43
$O_{18}$	$L_7, L_{14}$	380+250	140.05	$O_{38}$	$L_8, L_7$	250+270	358.81
$O_{19}$	$L_{13}$	380	241.32	$O_{39}$	$L_{13}$	380	454.72
$O_{20}$	$L_{15}$	425	284.11	$O_{40}$	$L_{13}$	380	497.92

4.2 数值实验分析

4.2.1 实验条件 为了验证所提禁忌搜索算法求解模型的性能,以多组船厂的实际数据进行实验.任务数量分别为 6、8、10、12、20、30、40、50;大型任务数分别为 1、1、1、1、2、3、4、5;平板车数量为 2、3、4、5、6、7、8.以任务执行时间、各任务之间的行驶时间作为输入量.时间窗根据实际背景进行调整.选用 IBM ILOG CPLEX 优化求解器求解模型,禁忌搜索基于 Java 语言编写.考虑到实际应用及便于对比等因素,设定 CPLEX 求解器的求解时间为 3 600 s.

4.2.2 试验结果与分析 算法的性能评价指标为  $GAP=\frac{f^{ts}-f^c}{f^{ts}}$ ,其中: $f^{ts}$ 为禁忌搜索算法的求解结果; $f^c$ 为 CPLEX 的求解结果.小规模任务数和平板车数组合情况下 2 种方法的求解结果如表 4 所示,

表 4 小规模任务的 CPLEX 与禁忌搜索求解对比  
Tab. 4 Comparison of CPLEX and tabu search on small-scale tasks

$n$	$L_l$	$f^c$	$t^c/\text{s}$	$f^{ts}$	$t^{ts}/\text{s}$	GAP
6	$L_2$	46.2	0.818	46.2	0.596	0
6	$L_3$	59.0	1.095	59.0	0.425	0
8	$L_2$	75.3	0.787	75.3	0.565	0
8	$L_3$	73.3	29.172	73.3	8.590	0
10	$L_3$	121.2	23.276	121.2	8.238	0
10	$L_4$	108.0	214.117	108.2	15.785	0
12	$L_3$	153.7	26.418	155.5	15.775	0.011 58
12	$L_4$	148.7	243.750	151.5	16.026	0.018 48

取 3 次求解的平均值作为禁忌搜索算法的最终结果.其中: $t^c$ 为 CPLEX 的求解时间; $t^{ts}$ 为禁忌搜索的求解时间.中等规模任务数和平板车数组情况下 2 种方法的求解结果如表 5 所示.

由表 4 可知,随着任务数和平板车数量的增加,CPLEX 的求解时间急剧增加.禁忌搜索算法在任务数为 6、8、10 时能够求出最优解,在未求出最优解的测试用例上求出近最优解,GAP 小于 0.02,同时求解时间并没有急剧增加,证明禁忌搜索算法是有效且可行的.由表 5 可知,当任务数为 20、平板车数为 4 时,CPLEX 已经在 3 600 s 内无法求出最优解.针

表 5 中等规模任务的 CPLEX 与禁忌搜索求解对比  
Tab. 5 Comparison of CPLEX and tabu search on medium-scale tasks

$n$	$L_l$	$f^c$	$t^c/\text{s}$	$f^{ts}$	$t^{ts}/\text{s}$	GAP
20	$L_4$	248.75	3 600	257.9	28.798	0.035 480
20	$L_5$	244.63	3 600	254.8	31.270	0.039 913
20	$L_6$	245.65	3 600	247.9	45.756	0.009 076
30	$L_4$	360.40	3 600	353.5	40.470	−0.019 520
30	$L_5$	330.25	3 600	328.4	63.954	−0.005 630
30	$L_6$	343.45	3 600	330.9	62.206	−0.037 950
40	$L_6$	468.43	3 600	472.9	62.890	0.009 452
40	$L_7$	450.01	3 600	453.2	61.153	0.007 040
40	$L_8$	479.88	3 600	441.1	61.574	−0.087 920
50	$L_6$	702.46	3 600	611.3	78.528	−1.491 200
50	$L_7$	698.89	3 600	593.4	75.504	−0.177 780
50	$L_8$	788.26	3 600	602.1	77.359	−0.309 180

对较大规模的任务数和平板车数时,禁忌搜索算法均可以在 80s 内求得结果,且在任务数为 30、40、50 时求得的结果明显优于 CPLEX. 其他任务数和平板车数上 GAP 均小于 0.04,验证了禁忌搜索算法在实际应用中具有较好的效果.

由于模型目标函数考虑了等待时间,所以需要 考虑所有任务的开始执行时间,但是禁忌搜索算法却不能遍历所有任务的开始时间,只能根据时间窗和任务序列的时间连贯性进行微调整,这就导致了大规模任务时禁忌搜索求解结果与最优解在等待时间上会有一定的差距.

5 结语

考虑平板车合作运输的船舶分段堆场间的运输调度模型,通过平板车合作运输解决大型分段的运输难点,提出通过调整任务的开始时间以减小平板车的等待时间. 从数值结果可以看出,所提算法在规模较大的情况下依然可以快速求得质量较高的解,验证了所设计的禁忌搜索算法具有较好的效果,可用于实际生产. 但如何提高大规模问题解的精度、如何使算法求解稳定、如何设计策略优化平板车的等待时间等问题仍需进一步的研究.

参考文献:

[ 1 ] 钟宏才, 蒋如宏, 谭家华, 等. 造船专业化生产单元模型及其作业安排优化[J]. 中国造船, 2004, 45 (2): 7-13.  
ZHONG Hongcai, JIANG Ruhong, TAN Jiahua, *et al.*. Research on cellular manufacturing model and production plan optimum in shipbuilding[J]. **Ship-building of China**, 2004, 45(2): 7-13.  
[ 2 ] JOO C M, KIM B S. Block transportation scheduling under delivery restriction in shipyard using meta-heuristic algorithms [ J ]. **Expert Systems with**

**Applications**, 2014, 41(6): 2851-2858.  
[ 3 ] WANG C, MAO Y S, HU B Q, *et al.*. Ship block transportation scheduling problem based on greedy algorithm[J]. **Journal of Engineering Science & Technology Review**, 2016, 9(2): 93-98.  
[ 4 ] SALEHI M, JALALIAN M, VALI SIAR M M. Green transportation scheduling with speed control; Trade-off between total transportation cost and carbon emission[J]. **Computers & Industrial Engineering**, 2017, 113: 392-404.  
[ 5 ] HU Z H, WEI C. Synchronizing vehicles for multi-vehicle and one-cargo transportation[J]. **Computers & Industrial Engineering**, 2018, 119: 36-49.  
[ 6 ] 张志英, 徐建祥, 计峰. 基于遗传算法的船舶分段堆场调度研究[J]. 上海交通大学学报, 2013, 47(7): 1036-1042.  
ZHANG Zhiying, XU Jianxiang, JI Feng. Shipbuilding yard scheduling approach based on genetic algorithm[J]. **Journal of Shanghai Jiao Tong University**, 2013, 47(7): 1036-1042.  
[ 7 ] 王冲, 茅云生, 辛锺桂. 基于遗传算法的船舶分段运输调度方法[J]. 上海交通大学学报, 2017, 51(3): 338-343.  
WANG Chong, MAO Yunsheng, SHIN Jonggye. Ship block transportation scheduling approach based on genetic algorithm[J]. **Journal of Shanghai Jiao Tong University**, 2017, 51(3): 338-343.  
[ 8 ] TAILLARD É, BADEAU P, GENDREAU M, *et al.*. A tabu search heuristic for the vehicle routing problem with soft time windows[J]. **Transportation Science**, 1997, 31(2): 170-186.  
[ 9 ] CORDEAU J F, LAPORTE G, MERCIER A. A unified tabu search heuristic for vehicle routing problems with time windows[J]. **Journal of the Operational Research Society**, 2001, 52(8): 928-936.

(本文编辑:石易文)