



(12) 发明专利

(10) 授权公告号 CN 115081674 B

(45) 授权公告日 2023.04.18

(21) 申请号 202210498853.7

G06Q 10/083 (2023.01)

(22) 申请日 2022.05.09

G06Q 10/0631 (2023.01)

(65) 同一申请的已公布的文献号

G06Q 10/067 (2023.01)

申请公布号 CN 115081674 A

(56) 对比文件

CN 112686458 A, 2021.04.20

(43) 申请公布日 2022.09.20

US 2022035374 A1, 2022.02.03

(73) 专利权人 香港理工大学深圳研究院

US 7725339 B1, 2010.05.25

地址 518057 广东省深圳市南山区粤海街
道高新技术产业园南区粤兴一道18号
香港理工大学产学研大楼205室

审查员 戴诚

(72) 发明人 许敏

(74) 专利代理机构 深圳市君胜知识产权代理事
务所(普通合伙) 44268

专利代理人 谢松

(51) Int.Cl.

G06Q 10/047 (2023.01)

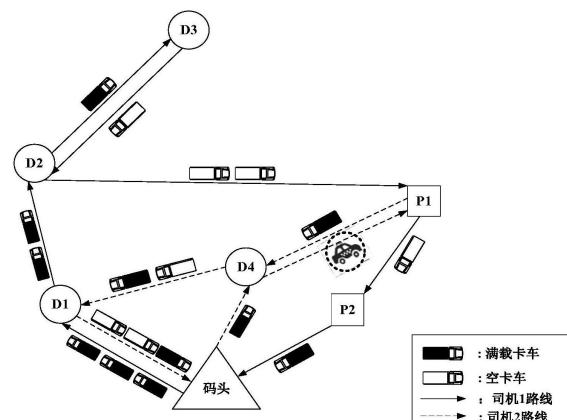
权利要求书4页 说明书17页 附图2页

(54) 发明名称

新型卡车排队行驶模式下的本地集装箱运
输排版优化方法

(57) 摘要

本发明公开了一种新型卡车排队行驶模式
下的本地集装箱运输排版优化方法,包括步骤:
获取网络图;根据网络图,确定总运营成本的
目标函数;其中,总运营成本包括:司机雇佣成
本、卡车部署成本、考虑到列队节油的卡车总油
耗成本以及替代运输模式产生的旅行成本;确定
网络图对应的初始解;在约束条件下,对初始解进
行更新以最小化目标函数,得到网络图对应的更新
解。卡车可以共享,可以根据需要调整无人驾驶
卡车的领头卡车,本发明在较短时间内为大规模
集装箱拖运任务制定司机和卡车的最优雇佣数
量、服务路线和调度计划,从而提高卡车的利
用率,减少卡车的数量,降低总运营成本。



1. 一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,包括步骤:

获取网络图;其中,所述网络图包括节点集合和弧的集合,所述节点集合中的节点包括送货客户节点、取货客户节点以及码头,所述弧的集合中的弧为所述节点集合中任意两个节点之间的连接弧线;所述送货客户节点包括:卸货前的任务节点和卸货后的任务节点,所述取货客户节点包括:装货前的任务节点和装货后的任务节点;

根据所述网络图,确定总运营成本的目标函数;其中,所述总运营成本包括:司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本;

确定所述网络图对应的初始解;

在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解;

所述目标函数为:

$$\min_{x,y,\alpha,\beta,s} \left\{ \lambda_1 \sum_{v \in V} \sum_{j \in C} x_{0j}^v + \lambda_2 \sum_{k \in K} \sum_{j \in C} y_{0j}^k + c_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} [\beta_{ij} + (1-\eta)(\sum_{k \in K} y_{ij}^k - \beta_{ij})] + c'_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} (\sum_{v \in V} x_{ij}^v - \beta_{ij}) \right\}$$

其中,min表示最小化函数,x表示司机的二元路线决策变量,y表示卡车的二元路线决策变量,a表示卡车穿过弧的二元决策变量,b表示在弧上有卡车穿过的二元决策变量,s表示时间变量, λ_1 表示司机的每日固定成本,v表示司机,V表示司机的集合,j表示任务节点,C表示除码头外的任务节点的集合, x_{0j}^v 表示司机v在弧(0,j)上的二元路线决策变量,0表示码头, λ_2 表示卡车的每日固定成本分,k表示卡车,K表示卡车的集合, y_{0j}^k 表示卡车k在弧(0,j)上的二元路线决策变量, c_{ij} 表示在弧(i,j)上卡车的燃油消耗成本,i表示任务节点,N表示包含码头的节点的集合, β_{ij} 表示在弧(i,j)上有卡车穿过的二元决策变量, η 表示卡车队列中卡车的省油率, y_{ij}^k 表示卡车k在弧(i,j)上的二元路线决策变量, c'_{ij} 表示在弧(i,j)上替代运输模式的旅行成本, x_{ij}^v 表示司机v在弧(i,j)上的二元路线决策变量, Σ 表示求和, \in 表示属于。

2. 根据权利要求1所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,所述约束条件为:

$$\sum_{i \in N} \alpha_{id}^{0d} = 1, \forall d \in D_1$$

$$\sum_{j \in N \setminus d} (\sum_{p \in P_1} \alpha_{dj}^{dp} + \alpha_{dj}^{d0}) = 1, \forall d \in D_2$$

$$\sum_{i \in N \setminus p} (\sum_{d \in D_2} \alpha_{ip}^{dp} + \alpha_{ip}^{0p}) = 1, \forall p \in P_1$$

$$\sum_{j \in N} \alpha_{pj}^{p0} = 1, \forall p \in P_2$$

$$\alpha_{0i}^{od} = 0, \forall i \in C, o \in C, d \in N, (o,d) \in R$$

$$\alpha_{i0}^{od} = 0, \forall i \in C, o \in N, d \in C, (o,d) \in R$$

$$\sum_{i \in \mathbf{N} \setminus o} \alpha_{io}^{od} = 0, \forall (o, d) \in \mathbf{R}$$

$$\sum_{i \in \mathbf{N} \setminus d} \alpha_{di}^{od} = 0, \forall (o, d) \in \mathbf{R}$$

$$\sum_{v \in \mathbf{V}} x_{0i}^v \leq 1, \forall v \in \mathbf{V}$$

$$\sum_{v \in \mathbf{V}} \sum_{i \in \mathbf{N} \setminus j} x_{ij}^v = 1, \forall j \in \mathbf{C}$$

$$\sum_{i \in \mathbf{N} \setminus j} x_{ij}^v = \sum_{i \in \mathbf{N} \setminus j} x_{ji}^v, \forall j \in \mathbf{C}, v \in \mathbf{V}$$

$$\sum_{i \in \mathbf{C}} y_{0i}^k \leq 1, \forall k \in \mathbf{K}$$

$$\sum_{k \in \mathbf{K}} y_{ij}^k \geq 1, \forall i \in \mathbf{C}_1, j = i'$$

$$\sum_{i \in \mathbf{N} \setminus j} y_{ij}^k = \sum_{i \in \mathbf{N} \setminus j} y_{ji}^k, \forall j \in \mathbf{C}, k \in \mathbf{K}$$

$$1 \leq \sum_{k \in \mathbf{K}} \sum_{i \in \mathbf{N} \setminus \{j, j'\}} y_{ij}^k \leq L, \forall j \in \mathbf{C}$$

$$y_{ij}^k \leq \sum_{v \in \mathbf{V}} x_{ij}^v, \forall i, j \in \mathbf{N}, j \neq i, i' \neq i, \forall k \in \mathbf{K}$$

$$\sum_{k \in \mathbf{K}} y_{ij}^k / L \leq \beta_{ij} \leq \sum_{k \in \mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{N}, j \neq i$$

$$\sum_{v \in \mathbf{V}} x_{ij}^v \geq \beta_{ij}, \forall i, j \in \mathbf{N}, j \neq i$$

$$\sum_{i \in \mathbf{C} \setminus \{g, g'\}} \alpha_{ig}^{od} = \sum_{j \in \mathbf{C} \setminus \{g, g'\}} \alpha_{gj}^{od}, \forall (o, d) \in \mathbf{R}, g \in \mathbf{C}, g \neq o \neq d$$

$$\sum_{i \in \mathbf{N}} \alpha_{oi}^{od} = \sum_{j \in \mathbf{N}} \alpha_{jd}^{od}, \forall (o, d) \in \mathbf{R}$$

$$\sum_{(o, d) \in \mathbf{R}} \sum_{i \in \mathbf{N} \setminus \{j, j'\}} \alpha_{ij}^{od} = \sum_{i \in \mathbf{N} \setminus \{j, j'\}} \sum_{k \in \mathbf{K}} y_{ij}^k, \forall j \in \mathbf{C}$$

$$\sum_{(o, d) \in \mathbf{R}} \alpha_{ij}^{od} \leq \sum_{k \in \mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{C}, i \neq j$$

$$s_i + p_i \leq s_{i'}, \forall i \in \mathbf{C}_1$$

$$s_i \geq t_{0i} \cdot \beta_{0i} + (1 - \beta_{0i}) \cdot t'_{0i}, \forall i \in \mathbf{C}$$

$$s_i + t_{i0} \leq T, \forall i \in \mathbf{C}_2$$

$$s_j \geq s_i + t_{ij} \cdot \beta_{ij} + (1 - \beta_{ij}) \cdot t'_{ij} - M \cdot (\sum_{v \in \mathbf{V}} x_{ij}^v - 1), \forall i, j \in \mathbf{C}, i \neq j$$

$$x_{ij}^v, y_{ij}^k, \alpha_{ij}^{od}, \beta_{ij} \in \{0, 1\}, \forall i, j \in \mathbf{N}, i \neq j, v \in \mathbf{V}, k \in \mathbf{K}, (o, d) \in \mathbf{R}$$

$$s_i \geq 0, \forall i \in \mathbf{N}$$

其中, α_{id}^{0d} 表示连接OD对(o, d)的行程中是否穿过弧(i, d)的二元决策变量,d表示送货

任务的任务节点, D_1 表示卸货前的任务节点的集合, α_{dj}^{dp} 表示连接OD对 (d, p) 的行程中是否穿过弧 (d, j) 的二元决策变量, p 表示取货任务的任务节点, α_{dj}^{d0} 表示连接OD对 $(d, 0)$ 的行程中是否穿过弧 (d, j) 的二元决策变量, $N \setminus d$ 表示 N 除去 d 的集合, D_2 表示卸货后的任务节点的集合, α_{ip}^{dp} 表示连接OD对 (d, p) 的行程中是否穿过弧 (i, p) 的二元决策变量, α_{ip}^{0p} 表示连接OD对 $(0, p)$ 的行程中是否穿过弧 (i, p) 的二元决策变量, $N \setminus p$ 表示 N 除去 p 的集合, P_1 表示装货前的任务节点的集合, α_{pj}^{p0} 表示连接OD对 $(p, 0)$ 的行程中是否穿过弧 (p, j) 的二元决策变量, P_2 表示装货后的任务节点的集合, α_{oi}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 $(0, i)$ 的二元决策变量, o 表示任务起始的节点, R 表示OD对的集合, α_{i0}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 $(i, 0)$ 的二元决策变量, α_{io}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, o) 的二元决策变量, $N \setminus o$ 表示 N 除去 o 的集合, α_{di}^{od} 表示连接OD对 $(0, d)$ 的行程中是否穿过弧 (d, i) 的二元决策变量, $N \setminus d$ 表示 N 除去 d 的集合, \forall 表示存在, x_{0i}^v 表示司机 v 在弧 $(0, i)$ 上的二元路线决策变量, $N \setminus j$ 表示 N 除去 j 的集合, x_{ji}^v 表示司机 v 在弧 (j, i) 上的二元路线决策变量, y_{0i}^k 表示卡车 k 在弧 $(0, i)$ 上的二元路线决策变量, C_1 表示 D_1 和 P_1 的并集, i' 表示同一客户节点中与任务节点 i 对应的任务节点, y_{ji}^k 表示卡车 k 在弧 (j, i) 上的二元路线决策变量, $N \setminus \{j, j'\}$ 表示 N 除去 j 和 j' 的集合, j' 表示同一客户节点中与任务节点 j 对应的任务节点, L 表示卡车队列中最大卡车数量, α_{ig}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, g) 的二元决策变量, α_{gj}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (g, j) 的二元决策变量, g 表示 C 中除了 o 和 d 之外的任务节点, g' 表示同一客户节点中与任务节点 g 对应的任务节点, $C \setminus \{g, g'\}$ 表示 C 除去 g 和 g' 的集合, α_{oi}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (o, i) 的二元决策变量, α_{jd}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (j, d) 的二元决策变量, α_{ij}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, j) 的二元决策变量, s_i 表示任务节点 i 的服务开始时间, p_i 表示任务节点 i 的卸货/装货时间, $s_{i'}$ 表示任务节点 i' 的服务开始时间, t_{0i} 表示司机通过卡车从码头 0 转移到任务节点 i 的旅行时间, β_{0i} 表示在弧 $(0, i)$ 上有卡车穿过的二元决策变量, t'_{0i} 表示司机通过替代运输模式在弧 $(0, i)$ 上的旅行时间, t_{i0} 表示司机通过卡车在弧 $(i, 0)$ 上的旅行时间, T 表示每日最长工作时间, C_2 表示 D_2 和 P_2 的并集, s_j 表示任务节点 j 的服务开始时间, t_{ij} 表示司机通过卡车在弧 (i, j) 上的旅行时间, t'_{ij} 表示司机通过替代运输模式在弧 (i, j) 上的旅行时间, M 表示常数。

3. 根据权利要求1-2任意一项所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法, 其特征在于, 所述确定所述网络图对应的初始解包括:

为司机构造一个字符串, 从码头开始依次插入尚未服务的任务节点; 其中, 在插入尚未服务的任务节点时, 选择与上次插入的任务节点的距离最近的任务节点插入;

当该司机的字符串中插入的任务节点达到卡车队列中最大卡车数量约束时, 结束该司机的字符串;

为下一司机构造一个字符串, 从码头依次插入尚未服务的任务节点, 直到所有任务节

点都已插入到字符串中,得到所述网络图对应的初始解。

4. 根据权利要求1-2任意一项所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,所述在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解,包括:

当所述初始解不满足任务节点时间安排的约束条件时,重新确定所述网络图对应的初始解;

当所述初始解满足任务节点时间安排的约束条件时,基于所述目标函数,对所述初始解进行评估,得到评估结果;

当所述评估结果满足预设要求时,将所述初始解作为所述网络图对应的更新解。

5. 根据权利要求4所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,所述在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解,还包括:

当所述评估结果不满足预设要求时,采用邻域搜索算子对所述初始解进行搜索得到邻域解;其中,所述邻域搜索算子包括:交换算子、反转算子以及插入算子;

当所述邻域解对应的目标函数值小于所述初始解对应的目标函数值时,将所述邻域解作为所述网络图对应的初始解,并降低所述初始解对应的温度;

当所述初始解符合终止条件时,将所述初始解作为所述网络图对应的更新解。

6. 根据权利要求5所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,所述在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解,还包括:

当所述邻域解对应的目标函数值大于或等于所述初始解对应的目标函数值时,根据Metropolis准则,确定所述网络图对应的初始解,并降低所述初始解对应的温度;

当所述初始解不符合终止条件时,继续判断所述初始解是否满足任务节点时间安排的约束条件,直至所述初始解满足评估结果满足预设要求或所述初始解符合终止条件时,将所述初始解作为所述网络图对应的更新解。

7. 根据权利要求6所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其特征在于,所述终止条件包括:所述初始解对应的温度小于预设的温度阈值或所述初始解的迭代次数达到最大连续无提高迭代次数;

降低所述初始解对应的温度时,根据退火速率降低温度。

8. 一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至7中任一项所述方法的步骤。

9. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现权利要求1至7中任一项所述的方法的步骤。

新型卡车排队行驶模式下的本地集装箱运输排版优化方法

技术领域

[0001] 本发明涉及非轨道行驶的陆地车辆的操作和运输技术领域,尤其涉及的是一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法。

背景技术

[0002] 全球贸易在很大程度上依赖于集装箱运输,其中多式联运集装箱通过轮船、铁路和卡车在海港、码头和当地客户之间移动(DHL,2021年)。在整个运输链中,通过卡车在码头和当地客户之间进行的短途集装箱运输,承运人分别为收货人和托运人提供送货和取货服务,这被称为当地集装箱运输问题(LCDP)。尽管与远洋运输和陆基长途运输相比运输距离非常短,但LCDP的运营成本在整个集装箱运输链总成本中约占比高达80% (Macharis和Bontekning,2004)。因此,本地集装箱运输服务的成本效率对多式联运集装箱运输的盈利能力尤为重要。在传统的集装箱拖运模式下,每个收货人/托运人需要一名司机带着一辆满载/空载卡车离开码头,并在客户现场等待卸货/装货工作;卸货/装货工作完成后,司机应将空/载卡车运回码头。近几十年来,为了降低总运营成本,集运模式在许多方面都得到了改进,例如空卡车的共享,其中从收货人处卸下的空卡车可以被托运人重复使用,而无需返回码头(Imai et al.,2007)。随着联网和自动驾驶汽车(CAV) 技术的发展,新兴的卡车排队行驶模式有助于减少LCDP中的劳动力。在这种模式中,一组自动驾驶汽车以较小的间隔距离一起行驶,只有领头的卡车需要司机的驾驶。此外,许多现场试验发现,卡车排队行驶还可以通过降低空气动力阻力来降低油耗,尤其是对于跟随卡车,这种模式显著节省约10~20%的燃油(Davila等人,2013年;Lammert等人,2014年;Lu 和Shladover,2014年)。如何利用新兴技术获得最大的成本效益是集装箱运输服务提供商面临的主要问题之一。

[0003] 在过去几十年中,考虑到各种实际约束和特征,如资源约束、灵活订单和旅行速度的不确定性等,众多学者对LCDP开展了相关研究。例如, Zhang考虑码头可用的卡车和空集装箱数量有限。他们在有向图上建立了一个混合整数规划(MIP) 模型,并提出了一种基于反应式禁忌搜索的算法来解决这个问题。Escudero研究了考虑意外事件导致的行程时间不确定性的每日行程问题。他们制定了一个实时优化模型,并提出了一种用于车队动态规划和管理的遗传算法。Zhan为具有灵活集装箱订单的动态集装箱运输服务建立了一个混合整数非线性规划模型,并开发了四种不同的求解方法来解决该问题。考虑到集装箱的可用性, Benantar解决了现实中的集装箱运输问题。在资源利用率高、成本效益高的新集运模式下,也有许多关于LCDP 的研究。例如, Imai提出了一个考虑空车共享的LCDP,并开发了一个基于拉格朗日松弛的启发式算法,通过解决几个子问题获得接近最优的解决方案。之后,为了进一步提高运输效率,Xue提出了领头卡车-跟随卡车分离模式,其中领头卡车可以与其配套的跟随卡车分离,使司机能够继续为下一个客户服务,而无需在当前客户现场等待卸货/装货工作。他们将这个问题描述为一个带有时间约束的车辆路径问题并提出了一种禁忌搜索算法。Song在Xue的基础上考虑了仓库必要的集装箱维护。他们将该问题视为非对称车辆路径问题,并将其表述为MILP模型。针对这一问题,提出了一种分支定价切割算法。Song

和Xue都假设一辆拖拉机一次只能搭载一辆拖车。后来,一些研究放宽了这一假设,允许拖拉机一次牵引两个或多个拖车(集装箱)。具体而言,Zhang等人探索了可折叠集装箱运输服务,在该服务中,卡车可以一次运输一个装载的集装箱或多个折叠集装箱。针对这一问题,他们提出了一种基于里程的卡车状态转移方法和一种改进的反应式禁忌搜索算法。Wang和Zhang提出了LCDP的双拖车分离模式,其中一辆牵引车可以承载两辆拖车。Zhang等人将其进一步扩展到多拖车下拉模式,其中开发了一个混合整数非线性规划模型和回溯自适应阈值接受算法来解决该问题。

[0004] 现有技术中,通过CAV技术,车辆/卡车列队在运输过程中具有进一步降低成本和提高效率的巨大潜力,已经有一些研究将新兴的卡车排队行驶技术纳入LCDP。在这些研究中,You等人率先提出了一个通用LCDP,考虑到在卡车列队运行模式下共享空的卡车,在这种模式下,只有领头卡车需要司机的驾驶,并由一组无人驾驶卡车自动跟踪。他们还假设,所有司机和卡车必须在规定的司机每日最长工作时间内完成客户的所有送货/取货请求后返回码头,并提出了一种基于蚁群算法的启发式求解方法。Xue等人后来将卡车排排的节油效益纳入了一个特殊的基于卡车排队行驶的 LCDP模型中,在该模型中,不同排的卡车不能在整个服务过程中队列在一起,并且不允许共享空卡车。最近,You等人考虑了LCDP中与负载相关的燃油成本和司机的多行程以及卡车排队行驶,其中司机可以在一个工作日内多次进入码头,但不允许在客户之间共享空卡车。他们建立了一个MIP 模型,并提出了一个定制的分支和价格切割算法来解决这个问题。然而, Xue等人、You等人和You等人的研究都假设,在重新获得司机或领头卡车的指令之前,在客户现场空置的无人驾驶卡车将无法用于其他服务任务,这可能会导致卡车利用率低下,总运营成本较高。

[0005] 因此,现有技术还有待于改进和发展。

发明内容

[0006] 本发明要解决的技术问题在于,针对现有技术的上述缺陷,提供一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法,旨在解决现有技术中卡车利用率低下导致总运营成本较高的问题。

[0007] 本发明解决技术问题所采用的技术方案如下:

[0008] 一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其中,包括步骤:

[0009] 获取网络图;其中,所述网络图包括节点集合和弧的集合,所述节点集合中的节点包括送货客户节点、取货客户节点以及码头,所述弧的集合中的弧为两个所述节点之间的连接弧线;所述送货客户节点包括:卸货前的任务节点和卸货后的任务节点,所述取货客户节点包括:装货前的任务节点和装货后的任务节点;

[0010] 根据所述网络图,确定总运营成本的目标函数;其中,所述总运营成本包括:司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本;

[0011] 确定所述网络图对应的初始解;

[0012] 在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解。

[0013] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其中,所述目

标函数为：

$$[0014] \quad \min_{x,y,\alpha,\beta,s} \left\{ \lambda_1 \sum_{v \in V} \sum_{j \in C} x_{0j}^v + \lambda_2 \sum_{k \in K} \sum_{j \in C} y_{0j}^k + c_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} [\beta_{ij} + (1-\eta)(\sum_{k \in K} y_{ij}^k - \beta_{ij})] + c'_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} (\sum_{v \in V} x_{ij}^v - \beta_{ij}) \right\}$$

[0015] 其中, min表示最小化函数,x表示司机的二元路线决策变量,y表示卡车的二元路线决策变量,a表示卡车穿过弧的二元决策变量,b表示在弧上有卡车穿过的二元决策变量,s表示时间变量,λ₁表示司机的每日固定成本,v表示司机,V表示司机的集合,j表示任务节点,C表示除码头外的任务节点的集合,x_{0j}^v表示司机v在弧(0,j)上的二元路线决策变量,0表示码头,λ₂表示卡车的每日固定成本分,k表示卡车,K表示卡车的集合,y_{0j}^k表示卡车k在弧(0,j)上的二元路线决策变量,c_{ij}表示在弧(i,j)上卡车的燃油消耗成本,i表示任务节点,N表示包含码头的节点的集合,β_{ij}表示在弧(i,j)上有卡车穿过的二元决策变量,η表示卡车队列中卡车的省油率,y_{ij}^k表示卡车k在弧(i,j)上的二元路线决策变量,c'_{ij}表示在弧(i,j)上替代运输模式的旅行成本,x_{ij}^v表示司机v在弧(i,j)上的二元路线决策变量,Σ表示求和,∈表示属于。

[0016] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其中,所述约束条件为:

$$[0017] \quad \sum_{i \in N} \alpha_{id}^{0d} = 1, \forall d \in D_1$$

$$[0018] \quad \sum_{j \in N \setminus d} (\sum_{p \in P_1} \alpha_{dj}^{dp} + \alpha_{dj}^{d0}) = 1, \forall d \in D_2$$

$$[0019] \quad \sum_{i \in N \setminus p} (\sum_{d \in D_2} \alpha_{ip}^{dp} + \alpha_{ip}^{0p}) = 1, \forall p \in P_1$$

$$[0020] \quad \sum_{j \in N} \alpha_{pj}^{p0} = 1, \forall p \in P_2$$

$$[0021] \quad \alpha_{0i}^{od} = 0, \forall i \in C, o \in C, d \in N, (o, d) \in R$$

$$[0022] \quad \alpha_{i0}^{od} = 0, \forall i \in C, o \in N, d \in C, (o, d) \in R$$

$$[0023] \quad \sum_{i \in N \setminus o} \alpha_{io}^{od} = 0, \forall (o, d) \in R$$

$$[0024] \quad \sum_{i \in N \setminus d} \alpha_{di}^{od} = 0, \forall (o, d) \in R$$

$$[0025] \quad \sum_{i \in C} x_{0i}^v \leq 1, \forall v \in V$$

$$[0026] \quad \sum_{v \in V} \sum_{i \in N \setminus j} x_{ij}^v = 1, \forall j \in C$$

$$[0027] \quad \sum_{i \in N \setminus j} x_{ij}^v = \sum_{i \in N \setminus j} x_{ji}^v, \forall j \in C, v \in V$$

$$[0028] \quad \sum_{i \in C} y_{0i}^k \leq 1, \forall k \in K$$

$$[0029] \quad \sum_{k \in K} y_{ij}^k \geq 1, \forall i \in C, j = i'$$

- [0030] $\sum_{i \in \mathbf{N} \setminus j} y_{ij}^k = \sum_{i \in \mathbf{N} \setminus j} y_{ji}^k, \forall j \in \mathbf{C}, k \in \mathbf{K}$
- [0031] $1 \leq \sum_{k \in \mathbf{K}} \sum_{i \in \mathbf{N} \setminus \{j, j'\}} y_{ij}^k \leq L, \forall j \in \mathbf{C}$
- [0032] $y_{ij}^k \leq \sum_{v \in \mathbf{V}} x_{ij}^v, \forall i, j \in \mathbf{N}, j \neq i, i', \forall k \in \mathbf{K}$
- [0033] $\sum_{k \in \mathbf{K}} y_{ij}^k / L \leq \beta_{ij} \leq \sum_{k \in \mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{N}, j \neq i$
- [0034] $\sum_{v \in \mathbf{V}} x_{ij}^v \geq \beta_{ij}, \forall i, j \in \mathbf{N}, j \neq i$
- [0035] $\sum_{i \in \mathbf{C} \setminus \{g, g'\}} \alpha_{ig}^{od} = \sum_{j \in \mathbf{C} \setminus \{g, g'\}} \alpha_{gj}^{od}, \forall (o, d) \in \mathbf{R}, g \in \mathbf{C}, g \neq o \neq d$
- [0036] $\sum_{i \in \mathbf{N}} \alpha_{oi}^{od} = \sum_{j \in \mathbf{N}} \alpha_{jd}^{od}, \forall (o, d) \in \mathbf{R}$
- [0037] $\sum_{(o, d) \in \mathbf{R}} \sum_{i \in \mathbf{N} \setminus \{j, j'\}} \alpha_{ij}^{od} = \sum_{i \in \mathbf{N} \setminus \{j, j'\}} \sum_{k \in \mathbf{K}} y_{ij}^k, \forall j \in \mathbf{C}$
- [0038] $\sum_{(o, d) \in \mathbf{R}} \alpha_{ij}^{od} \leq \sum_{k \in \mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{C}, i \neq j$
- [0039] $s_i + p_i \leq s_{i'}, \forall i \in \mathbf{C}_1$
- [0040] $s_i \geq t_{0i} \cdot \beta_{0i} + (1 - \beta_{0i}) \cdot t'_{0i}, \forall i \in \mathbf{C}$
- [0041] $s_i + t_{i0} \leq T, \forall i \in \mathbf{C}_2$
- [0042] $s_j \geq s_i + t_{ij} \cdot \beta_{ij} + (1 - \beta_{ij}) \cdot t'_{ij} - M \cdot (\sum_{v \in \mathbf{V}} x_{ij}^v - 1), \forall i, j \in \mathbf{C}, i \neq j$
- [0043] $x_{ij}^v, y_{ij}^k, \alpha_{ij}^{od}, \beta_{ij} \in \{0, 1\}, \forall i, j \in \mathbf{N}, i \neq j, v \in \mathbf{V}, k \in \mathbf{K}, (o, d) \in \mathbf{R}$
- [0044] $s_i \geq 0, \forall i \in \mathbf{N}$
- [0045] 其中, α_{id}^{0d} 表示连接OD对 (0, d) 的行程中是否穿过弧 (i, d) 的二元决策变量, d 表示送货任务的任务节点, \mathbf{D}_1 表示卸货前的任务节点的集合, α_{dj}^{dp} 表示连接OD对 (d, p) 的行程中是否穿过弧 (d, j) 的二元决策变量, p 表示取货任务的任务节点, α_{dj}^{d0} 表示连接OD对 (d, 0) 的行程中是否穿过弧 (d, j) 的二元决策变量, N \ d 表示 N 去除 d 的集合, \mathbf{D}_2 表示卸货后的任务节点的集合, α_{ip}^{dp} 表示连接OD对 (d, p) 的行程中是否穿过弧 (i, p) 的二元决策变量, α_{ip}^{0p} 表示连接OD对 (0, p) 的行程中是否穿过弧 (i, p) 的二元决策变量, N \ p 表示 N 去除 p 的集合, \mathbf{P}_1 表示装货前的任务节点的集合, α_{pj}^{p0} 表示连接OD对 (p, 0) 的行程中是否穿过弧 (p, j) 的二元决策变量, α_{pj}^{p0} 表示连接OD对 (p, 0) 的行程中是否穿过弧 (p, j) 的二元决策变量, N \ p 表示 N 去除 p 的集合, \mathbf{P}_2 表示装货后的任务节点的集合, α_{0i}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (0, i) 的二元决策变量, o 表示任务起始的节点, R 表示 OD 对的集合, α_{i0}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, 0) 的二元决策变量, α_{io}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, o) 的二元决策变量, N \ o 表示 N 去除 o 的集合, α_{di}^{od} 表示连接OD对 (0, d) 的行程中是否穿过弧 (d, i) 的二元决策变量, N \ d 表示 N 去除 d 的集合, \forall 表示存在, x_{0i}^v 表示司机 v 在弧 (0, i) 上的二元决策变量。

路线决策变量, $N \setminus j$ 表示 N 除去 j 的集合, x_{ji}^v 表示司机 v 在弧 (j, i) 上的二元路线决策变量, y_{0i}^k 表示卡车 k 在弧 $(0, i)$ 上的二元路线决策变量, C_1 表示 D_1 和 P_1 的并集, i' 表示同一客户节点中与任务节点 i 对应的任务节点, y_{ji}^k 表示卡车 k 在弧 (j, i) 上的二元路线决策变量, $N \setminus \{j, j'\}$ 表示 N 除去 j 和 j' 的集合, j' 表示同一客户节点中与任务节点 j 对应的任务节点, L 表示车队列中最大卡车数量, α_{ig}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, g) 的二元决策变量, α_{gj}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (g, j) 的二元决策变量, g 表示 C 中除了 o 和 d 之外的任务节点, g' 表示同一客户节点中与任务节点 g 对应的任务节点, $C \setminus \{g, g'\}$ 表示 C 除去 g 和 g' 的集合, α_{oi}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (o, i) 的二元决策变量, α_{jd}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (j, d) 的二元决策变量, α_{ij}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, j) 的二元决策变量, s_i 表示任务节点 i 的服务开始时间, p_i 表示任务节点 i 的卸货/装货时间, $s_{i'}$ 表示任务节点 i' 的服务开始时间, t_{0i} 表示司机通过卡车从码头 0 转移到任务节点 i 的旅行时间, β_{0i} 表示在弧 $(0, i)$ 上有卡车穿过的二元决策变量, t'_{0i} 表示司机通过替代运输模式在弧 $(0, i)$ 上的旅行时间, t_{i0} 表示司机通过卡车在弧 $(i, 0)$ 上的旅行时间, T 表示每日最长工作时间, C_2 表示 D_2 和 P_2 的并集, s_j 表示任务节点 j 的服务开始时间, t_{ij} 表示司机通过卡车在弧 (i, j) 上的旅行时间, t'_{ij} 表示司机通过替代运输模式在弧 (i, j) 上的旅行时间, M 表示常数。

[0046] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法, 其中, 所述确定所述网络图对应的初始解包括:

[0047] 为司机构造一个字符串, 从码头开始依次插入尚未服务的任务节点; 其中, 在插入尚未服务的任务节点时, 选择与上次插入的任务节点的距离最近的任务节点插入;

[0048] 当该司机的字符串中插入的任务节点达到最大卡车排大小约束时, 结束该司机的字符串;

[0049] 为下一司机构造一个字符串, 从码头依次插入尚未服务的任务节点, 直到所有任务节点都已插入到字符串中, 得到所述网络图对应的初始解。

[0050] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法, 其中, 所述在约束条件下, 对所述初始解进行更新以最小化所述目标函数, 得到所述网络图对应的更新解, 包括:

[0051] 当所述初始解不满足任务节点时间安排的约束条件时, 重新确定所述网络图对应的初始解;

[0052] 当所述初始解满足任务节点时间安排的约束条件时, 基于所述目标函数, 对所述初始解进行评估, 得到评估结果;

[0053] 当所述评估结果满足预设要求时, 将所述初始解作为所述网络图对应的更新解。

[0054] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法, 其中, 所述在约束条件下, 对所述初始解进行更新以最小化所述目标函数, 得到所述网络图对应的更新解, 还包括:

[0055] 当所述评估结果不满足预设要求时, 采用邻域搜索算子对所述初始解进行搜索得到邻域解; 其中, 所述邻域搜索算子包括: 交换算子、反转算子以及插入算子;

- [0056] 当所述领域解对应的目标函数值小于所述初始解对应的目标函数值时,将所述领域解作为所述网络图对应的初始解,并降低所述初始解对应的温度;
- [0057] 当所述初始解符合终止条件时,将所述初始解作为所述网络图对应的更新解。
- [0058] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其中,所述在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解,还包括:
- [0059] 当所述领域解对应的目标函数值大于或等于所述初始解对应的目标函数值时,根据Metropolis准则,确定所述网络图对应的初始解,并降低所述初始解对应的温度;
- [0060] 当所述初始解不符合终止条件时,继续判断所述初始解是否满足任务节点时间安排的约束条件,直至所述初始解满足评估结果满足预设要求或所述初始解符合终止条件时,将所述初始解作为所述网络图对应的更新解。
- [0061] 所述的新型卡车排队行驶模式下的本地集装箱运输排版优化方法,其中,所述终止条件包括:所述初始解对应的温度小于预设的温度阈值或所述初始解的迭代次数达到最大连续无提高迭代次数;
- [0062] 降低所述初始解对应的温度时,根据退火速率降低温度。
- [0063] 一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,其中,所述处理器执行所述计算机程序时实现如上任一项所述方法的步骤。
- [0064] 一种计算机可读存储介质,其上存储有计算机程序,其中,所述计算机程序被处理器执行时实现如上任一项所述的方法的步骤。
- [0065] 有益效果:卡车可以共享,可以根据需要调整无人驾驶卡车的领头卡车,本发明在较短时间内为大规模集装箱拖运任务制定司机和卡车的最优雇佣数量、服务路线和调度计划,从而提高卡车的利用率,减少卡车的数量,降低总运营成本。

附图说明

- [0066] 图1是本发明中卡车排队行驶模式在LCDP中的应用示例的示意图。
- [0067] 图2是本发明中启发式构造算法HCSA框架。
- [0068] 图3是CPU运行时间随客户规模的增加的变化。

具体实施方式

- [0069] 为使本发明的目的、技术方案及优点更加清楚、明确,以下参照附图并举实施例对本发明进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。
- [0070] 请同时参阅图1-图3,本发明提供了一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法的一些实施例。
- [0071] 如图1-图3所示,本发明的一种新型卡车排队行驶模式下的本地集装箱运输排版优化方法,包括以下步骤:
- [0072] 步骤S100、获取网络图;其中,所述网络图包括节点集合和弧的集合,所述节点集合中的节点包括送货客户节点、取货客户节点以及码头,所述弧的集合中的弧为两个所述节点之间的连接弧线;所述送货客户节点包括:卸货前的任务节点和卸货后的任务节点,所

述取货客户节点包括:装货前的任务节点和装货后的任务节点。

[0073] 步骤S200、根据所述网络图,确定总运营成本的目标函数;其中,所述总运营成本包括:司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本。

[0074] 步骤S300、确定所述网络图对应的初始解。

[0075] 步骤S400、在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解。

[0076] 具体地,本申请中的卡车采用卡车列队,卡车列队中有至少一个卡车,只有领头卡车需要司机的驾驶,并由一组无人驾驶卡车自动跟踪领头卡车。卡车列队中的卡车数量小于或等于最大卡车数量。卡车可以共享,也就是说,可以根据需要调整无人驾驶卡车的领头卡车,从而提高卡车的利用率,减少卡车的数量,降低总运营成本。

[0077] 现有的研究主要是针对传统卡车排队行驶模式下的本地集装箱运输排版优化方法。本发明提出的优化方法针对新型本地集装箱运输模式,相比现有模式更加灵活和高效,不仅更加节省人力、物力和财力,提高服务效率,降低运营总成本,而且卡车近距离地排队行驶能降低卡车所受到的空气阻力,有效减小卡车油耗和污染物排放。另外,本专利提供了新型运输模式下的本地集装箱问题的优化方法,实现了在较短时间内为大规模集装箱拖运任务制定司机和卡车的最优雇佣数量、服务路线和调度计划,为管理者或计划者提供了有力的技术支持与保障。

[0078] 为了尽可能降低总运营成本,进行数学建模,形成网络图 $G = (N, A)$ 中 N 表示点的集合, $A = \{(i, j) | i, j \in N, i \neq j\}$ 为弧的集合。注意 $N = D \cup P \cup \{0\}$,其中D 和P为送货和取货客户节点集,而0则表示码头,卡车为单集装箱容量的同类自动卡车车队。在新型卡车排队行驶操作模式(IPOM)中,司机可以通过卡车或其他交通方式从一个客户节点转移到另一个客户节点,采用其它交通方式(替代运输模式可以是除卡车以外的其它交通工具,如摩托车、出租车、共享车辆、叫车服务和公交车)转移时,司机可以驾驶另一个客户节点的无人驾驶卡车,实现卡车车队中无人驾驶卡车的重排。因此会有两个与弧 $(i, j) \in A$ 相关的旅行时间 t_{ij} 和 t'_{ij} ,分别表示采用卡车和其他替代交通工具穿过弧 (i, j) 所需的时间。卡车的燃油消耗成本和替代运输模式的旅行成本分别用 c_{ij} 和 c'_{ij} 表示。在运输过程中,卡车以队列的方式行驶以节省燃油。并且出于安全考虑,在每个弧上一起行驶的卡车数量不能超过最大卡车数量L。关于节油效果我们假设队列中卡车可达到的省油率为 η ,而领头的卡车不能节油。例如,由m辆卡车成的队列穿过弧 $(i, j) \in A$ 所产生的总燃油成本将通过以下约束计算: $c_{ij} [1 + (1 - \eta)(m - 1)]$ 。此外,司机和卡车的每日固定成本分别用 λ_1 和 λ_2 表示。

[0079] 在IPOM模式下,除卡车外,司机还可以使用附近可用的其他交通方式,前往其他客户地点执行服务任务,并将之前空置在客户点的卡车驶回相应地点。使用图1中所示的示例来说明IPOM的操作特征。在此示例中,将为四个送货客户(包括D1、D2、D3和D4)和两个取货客户(即P1和 P2)提供服务。图1中显示,司机1带着一个由三辆满载卡车(司机1位于领头卡车,后面跟着2辆无人驾驶卡车)组成的队列离开码头。司机1 将卡车列队分别停在D1、D2和D3处,在D1和D2处时分别分离1辆无人驾驶卡车进行卸货,领头卡车在D3处进行卸货,司机1 在D3处等待,直到卸货工作完成。然后,司机1再次前往D2,领头卡车与卸完货的1辆无人驾驶卡车(即空卡车),形成一个由两辆空卡车组成的队列,而D1处的卸完货的1辆无人驾驶卡

车(即空卡车),则归入另一名司机(本例中为司机2)的卡车队列,并司机2由带回码头。由于空卡车可以在客户之间共享,司机1将分别将送货客户D2和D3释放的两辆空卡车给取货客户P1 和P2,以供再利用。司机1的卡车列队到达P1处后,释放1辆无人驾驶的空卡车,并继续驾驶领头卡车到达P2处,在P2处的装载工作完成后,司机1将带着满载的领头卡车返回码头,而P1处无人驾驶的空卡车装载工作完成后,由司机2运送回码头,此时,该卡车作为领头卡车。至于司机2 从码头出发时,带着一辆满载的卡车,并停在客户D4进行卸货工作。由于IPOM允许司机在停下带领的所有卡车后能够采用其他交通工具(比如出租车)移动到P1处,并将满载的卡车开回去,而无需在D4等待卸货任务的完成,然后重新访问D4处,形成两辆卡车的队列。由于司机2在再次到达 D4后可能在最长工作时间内会有额外的可利用时间,司机2以继续前往 D1,形成一个总共三辆卡车的队列,最后返回码头。但是,如果没有IPOM,司机2必须在D4等待,直到卸货工作完成,这样司机2可能没有足够的时间在P1完成提货而不违反最长工作时间,从而更不可能服务D1。在这种情况下,需要另外一名或两名司机驾驶卡车将P1和D1处的卡车运回码头。由以上示例所示,IPOM在通过节省劳动力和卡车来提高运输效率方面具有巨大潜力。

[0080] 对于本发明所研究的LCDP问题,其中空卡车可能来自码头或是完成卸货工作的送货客户处。此外,多辆卡车以列队的形式一起行驶的列队模式将迫使卡车在从各自的起点到各自的目标客户的途中穿过许多临时客户点。客户之间共享空卡车和卡车列队行驶模式使得为卡车具体移动轨迹的建模和计算完成全部客户服务所需的卡车数量方面带来了挑战。事实上,每个客户在装货/卸货工作前后的两个阶段中都与两个不同的任务相关。出于建模目的,可以将每个客户节点拆分为两个任务节点,以表示客户任务的两个阶段。具体地说,用 C_1 表示第一阶段任务节点的集合,包括需要满载卡车的送货任务节点集合(即送货任务中卸货前的任务节点的集合 D_1) 和需要空卡车的取货任务节点集合(即取货任务中装货前的任务节点的集合 P_1)。类似地,用 C_2 表示第二阶段任务节点的集合,包括清空卡车的送货任务节点集合(即送货任务中卸货后的任务节点的集合 D_2) 和装载后的取货任务节点集合(即取货任务中装货后的任务节点的集合 P_2)。用 $C=C_1 \cup C_2$ 表示除码头外所有任务节点的集合,其中有 $C_1=D_1 \cup P_1$ 以及 $C_2=D_2 \cup P_2$ 。然后可以将客户节点集N转换为 $N=D_1 \cup D_2 \cup P_1 \cup P_2 \cup \{0\}=C \cup \{0\}$ 。请注意,客户的第一阶段任务节点 $i \in C_1$ 对应的第二阶段任务节点表示为 $i' \in C_2$,反之亦然。根据客户节点两阶段任务节点的定义,OD对可定义如下:由一个任务起始的节点o和一个任务的终点节点d组成的OD对用于指示从起始点出发的卡车专用于服务于终点任务节点。对于特定的OD对,卡车在从起点到目的地的路线上顺序穿过的一组任务节点可以视为转运节点。码头不能作为连接OD对的路径中的传输节点。请注意,每个任务节点只能作为一个起点或终点一次,因此在研究的问题中OD对的数量是有限的。结合以上阐述的卡车移动特点和客户节点拆分策略,采用如下约束条件(1) - (8) 来跟踪卡车移动轨迹。

$$[0081] \quad \sum_{i \in N} \alpha_{id}^{0d} = 1, \forall d \in D_1 \quad (1)$$

$$[0082] \quad \sum_{j \in N \setminus d} \left(\sum_{p \in P_1} \alpha_{dj}^{dp} + \alpha_{dj}^{d0} \right) = 1, \forall d \in D_2 \quad (2)$$

$$[0083] \quad \sum_{i \in N \setminus p} \left(\sum_{d \in D_2} \alpha_{ip}^{dp} + \alpha_{ip}^{0p} \right) = 1, \forall p \in P_1 \quad (3)$$

$$[0084] \quad \sum_{j \in N} \alpha_{pj}^{p0} = 1, \forall p \in P_2 \quad (4)$$

$$[0085] \quad \alpha_{0i}^{od} = 0, \forall i \in C, o \in C, d \in N, (o, d) \in R \quad (5)$$

$$[0086] \quad \alpha_{i0}^{od} = 0, \forall i \in C, o \in N, d \in C, (o, d) \in R \quad (6)$$

$$[0087] \quad \sum_{i \in N \setminus o} \alpha_{io}^{od} = 0, \forall (o, d) \in R \quad (7)$$

$$[0088] \quad \sum_{i \in N \setminus d} \alpha_{di}^{od} = 0, \forall (o, d) \in R \quad (8)$$

[0089] 其中, α_{id}^{0d} 表示连接OD对(0, d)的行程中是否穿过弧(i, d)的二元决策变量, d表示送货任务的任务节点, D_1 表示卸货前的任务节点的集合, α_{dj}^{dp} 表示连接OD对(d, p)的行程中是否穿过弧(d, j)的二元决策变量, p表示取货任务的任务节点, α_{dj}^{d0} 表示连接OD对(d, 0)的行程中是否穿过弧(d, j)的二元决策变量, $N \setminus d$ 表示N除去d的集合, D_2 表示卸货后的任务节点的集合, α_{ip}^{dp} 表示连接OD对(d, p)的行程中是否穿过弧(i, p)的二元决策变量, α_{ip}^{0p} 表示连接OD对(0, p)的行程中是否穿过弧(i, p)的二元决策变量, $N \setminus p$ 表示N除去p的集合, P_1 表示装货前的任务节点的集合, α_{pj}^{p0} 表示连接OD对(p, 0)的行程中是否穿过弧(p, j)的二元决策变量, P_2 表示装货后的任务节点的集合, α_{0i}^{od} 表示连接OD对(o, d)的行程中是否穿过弧(0, i)的二元决策变量, o表示任务起始的节点, C表示除码头外的任务节点的集合, R表示OD对的集合, α_{i0}^{od} 表示连接OD对(o, d)的行程中是否穿过弧(i, 0)的二元决策变量, α_{io}^{od} 表示连接OD对(o, d)的行程中是否穿过弧(i, o)的二元决策变量, $N \setminus o$ 表示N除去o的集合, α_{di}^{od} 表示连接OD对(0, d)的行程中是否穿过弧(d, i)的二元决策变量, $N \setminus d$ 表示N除去d的集合, \forall 表示存在。

[0090] 约束条件(1) - (4) 表示卡车移动要求。更具体地说, 约束条件(1) 规定, 对于第一阶段送货任务节点, 必须将满载的卡车从码头运输到该节点。约束(2) 表示对于第二阶段送货任务节点, 其释放的空卡车可以运送到码头或第一阶段取货任务节点以供再使用; 类似地, 约束(3) 表示对于第一阶段取货任务节点对空卡车的需求可以由码头或第二阶段送货任务节点来满足。约束(4) 确保对于第二阶段取货任务节点, 完成装货任务后装满载的卡车必须返回码头。约束(5) - (8) 是关于OD对特征的约束。具体而言, 约束条件(5) 和(6) 确保码头不能用于连接OD对。约束条件(7) 和(8) 强制要求OD对的起始节点和终点不能充当连接该OD对的转运节点。

[0091] 为优化司机和卡车的数量、行驶路线和时间表, 除了司机的二元路线决策变量 $x_{ij}^v, \forall (i, j) \in A, v \in V$ 和卡车的二元路线决策变量 $y_{ij}^k, \forall (i, j) \in A, k \in K$, 分别表示司机 $v \in V$ 在其行程中是否沿着弧(i, j)行驶, 以及卡车 $k \in K$ 是否经由 $i \in N$ 节点到另一个节点 $j \in N$, 我们还需要定义二元决策变量 $\beta_{ij}, \forall (i, j) \in A$ 和连续时间变量 $s_i, \forall i \in N$, 以分别表示是否至少有一辆卡车穿过弧(i, j)和任务节点 $i \in C$ 的服务开始时间。此外, 用 $p_i, i \in C_1$ 表示每位客户所需的集装箱卸货/装货时间。为了方便读者阅读, 本研究中使用的符号说明见附录。本发

明对IPOM下的LCDP建模如下目标函数：

$$\begin{aligned} [0092] \quad & \min_{x,y,\alpha,\beta,s} \quad \lambda_1 \sum_{v \in V} \sum_{j \in C} x_{0j}^v + \lambda_2 \sum_{k \in K} \sum_{j \in C} y_{0j}^k + c_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} [\beta_{ij} + (1-\eta)(\sum_{k \in K} y_{ij}^k - \beta_{ij})] + \\ & c'_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} (\sum_{v \in V} x_{ij}^v - \beta_{ij}) \end{aligned} \quad (9)$$

[0093] 其中, \min 表示最小化函数, x 表示司机的二元路线决策变量, y 表示卡车的二元路线决策变量, α 表示卡车穿过弧的二元决策变量, β 表示在弧上有卡车穿过的二元决策变量, s 表示时间变量, λ_1 表示司机的每日固定成本, v 表示司机, V 表示司机的集合, j 表示任务节点, C 表示除码头外的任务节点的集合, x_{0j}^v 表示司机 v 在弧 $(0, j)$ 上的二元路线决策变量, 0 表示码头, λ_2 表示卡车的每日固定成本分, k 表示卡车, K 表示卡车的集合, y_{0j}^k 表示卡车 k 在弧 $(0, j)$ 上的二元路线决策变量, c_{ij} 表示在弧 (i, j) 上卡车的燃油消耗成本, i 表示任务节点, N 表示包含码头的节点的集合, β_{ij} 表示在弧 (i, j) 上有卡车穿过的二元决策变量, η 表示卡车队列中卡车的省油率, y_{ij}^k 表示卡车 k 在弧 (i, j) 上的二元路线决策变量, c'_{ij} 表示在弧 (i, j) 上替代运输模式的旅行成本, x_{ij}^v 表示司机 v 在弧 (i, j) 上的二元路线决策变量, Σ 表示求和, \in 表示属于。

[0094] 具体地, 目标函数中第一项 $\lambda_1 \sum_{v \in V} \sum_{j \in C} x_{0j}^v$ 为司机雇佣成本, 第二项 $\lambda_2 \sum_{k \in K} \sum_{j \in C} y_{0j}^k$ 为卡车部署成本, 第三项 $c_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} [\beta_{ij} + (1-\eta)(\sum_{k \in K} y_{ij}^k - \beta_{ij})]$ 为考虑到列队节油的卡车总油耗成本, 第四项 $c'_{ij} \sum_{i \in N, i \neq j} \sum_{j \in N} (\sum_{v \in V} x_{ij}^v - \beta_{ij})$ 为替代运输模式产生的旅行成本。

[0095] 目标函数服从约束条件(1) - (8) 和 (10) - (29), 约束条件(10) - (29) 如下:

$$[0096] \quad \sum_{i \in C} x_{0i}^v \leq 1, \forall v \in V \quad (10)$$

$$[0097] \quad \sum_{v \in V} \sum_{i \in N \setminus j} x_{ij}^v = 1, \forall j \in C \quad (11)$$

$$[0098] \quad \sum_{i \in N \setminus j} x_{ij}^v = \sum_{i \in N \setminus j} x_{ji}^v, \forall j \in C, v \in V \quad (12)$$

$$[0099] \quad \sum_{i \in C} y_{0i}^k \leq 1, \forall k \in K \quad (13)$$

$$[0100] \quad \sum_{k \in K} y_{ij}^k \geq 1, \forall i \in C, j = i' \quad (14)$$

$$[0101] \quad \sum_{i \in N \setminus j} y_{ij}^k = \sum_{i \in N \setminus j} y_{ji}^k, \forall j \in C, k \in K \quad (15)$$

$$[0102] \quad 1 \leq \sum_{k \in K} \sum_{i \in N \setminus \{j, j'\}} y_{ij}^k \leq L, \forall j \in C \quad (16)$$

$$[0103] \quad y_{ij}^k \leq \sum_{v \in V} x_{ij}^v, \forall i, j \in N, j \neq i, i', \forall k \in K \quad (17)$$

$$[0104] \quad \sum_{k \in K} y_{ij}^k / L \leq \beta_{ij} \leq \sum_{k \in K} y_{ij}^k, \forall i, j \in N, j \neq i \quad (18)$$

$$[0105] \quad \sum_{v \in V} x_{ij}^v \geq \beta_{ij}, \forall i, j \in N, j \neq i \quad (19)$$

$$[0106] \quad \sum_{i \in C \setminus \{g, g'\}} \alpha_{ig}^{od} = \sum_{j \in C \setminus \{g, g'\}} \alpha_{gj}^{od}, \forall (o, d) \in R, g \in C, g \neq o \neq d \quad (20)$$

$$[0107] \quad \sum_{i \in N} \alpha_{oi}^{od} = \sum_{j \in N} \alpha_{jd}^{od}, \forall (o, d) \in R \quad (21)$$

$$[0108] \quad \sum_{(o, d) \in R} \sum_{i \in N \setminus \{j, j'\}} \alpha_{ij}^{od} = \sum_{i \in N \setminus \{j, j'\}} \sum_{k \in K} y_{ij}^k, \forall j \in C \quad (22)$$

$$[0109] \quad \sum_{(o, d) \in R} \alpha_{ij}^{od} \leq \sum_{k \in K} y_{ij}^k, \forall i, j \in C, i \neq j \quad (23)$$

$$[0110] \quad s_i + p_i \leq s_{i'}, \forall i \in C_1 \quad (24)$$

$$[0111] \quad s_i \geq t_{0i} \cdot \beta_{0i} + (1 - \beta_{0i}) \cdot t'_{0i}, \forall i \in C \quad (25)$$

$$[0112] \quad s_i + t_{i0} \leq T, \forall i \in C_2 \quad (26)$$

$$[0113] \quad s_j \geq s_i + t_{ij} \cdot \beta_{ij} + (1 - \beta_{ij}) \cdot t'_{ij} - M \cdot (\sum_{v \in V} x_{ij}^v - 1), \forall i, j \in C, i \neq j \quad (27)$$

$$[0114] \quad x_{ij}^v, y_{ij}^k, \alpha_{ij}^{od}, \beta_{ij} \in \{0, 1\}, \forall i, j \in N, i \neq j, v \in V, k \in K, (o, d) \in R \quad (28)$$

$$[0115] \quad s_i \geq 0, \forall i \in N \quad (29)$$

[0116] 其中, x_{0i}^v 表示司机 v 在弧 $(0, i)$ 上的二元路线决策变量, $N \setminus j$ 表示 N 除去 j 的集合, x_{ji}^v 表示司机 v 在弧 (j, i) 上的二元路线决策变量, y_{0i}^k 表示卡车 k 在弧 $(0, i)$ 上的二元路线决策变量, C_1 表示 D_1 和 P_1 的并集, i' 表示同一客户节点中与任务节点 i 对应的任务节点, y_{ji}^k 表示卡车 k 在弧 (j, i) 上的二元路线决策变量, $N \setminus \{j, j'\}$ 表示 N 除去 j 和 j' 的集合, j' 表示同一客户节点中与任务节点 j 对应的任务节点, L 表示卡车队列中最大卡车数量, α_{ig}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, g) 的二元决策变量, α_{gj}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (g, j) 的二元决策变量, g 表示 C 中除了 o 和 d 之外的任务节点, g' 表示同一客户节点中与任务节点 g 对应的任务节点, $C \setminus \{g, g'\}$ 表示 C 除去 g 和 g' 的集合, α_{oi}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (o, i) 的二元决策变量, α_{jd}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (j, d) 的二元决策变量, α_{ij}^{od} 表示连接OD对 (o, d) 的行程中是否穿过弧 (i, j) 的二元决策变量, s_i 表示任务节点 i 的服务开始时间, p_i 表示任务节点 i 的卸货/装货时间, $s_{i'}$ 表示任务节点 i' 的服务开始时间, t_{0i} 表示司机通过卡车从码头 0 转移到任务节点 i 的旅行时间, β_{0i} 表示在弧 $(0, i)$ 上有卡车穿过的二元决策变量, t'_{0i} 表示司机通过替代运输模式在弧 $(0, i)$ 上的旅行时间, t_{i0} 表示司机通过卡车在弧 $(i, 0)$ 上的旅行时间, T 表示每日最长工作时间, C_2 表示 D_2 和 P_2 的并集, s_j 表示任务节点 j 的服务开始时间, t_{ij} 表示司机通过卡车在弧 (i, j) 上的旅行时间, t'_{ij} 表示司机通过替代运输模式在弧 (i, j) 上的旅行时间, M 表示常数。

[0117] 其中 M 为一个足够大的数字。约束 (9) 所示的目标函数是总运营成本的总和, 包括司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本。约束条件 (10) - (12) 构成司机行驶路线。具体来说, 约束 (10) 强制每个司机最

多只能离开码头一次。约束(11)规定每个任务节点由一个司机访问一次。约束(12)确保每名司机流量守恒。约束条件(13)-(15)构成卡车行驶路线。约束(13)限制每辆卡车不能重复被重复部署。约束(14)确保至少有一辆卡车停留在客户所在地,直到装/卸货工作完成。约束(15)确保每辆卡车流量守恒。约束(16)限制最大货车列队行驶规模。约束(17)-(19)指定任务节点之间司机路线和卡车移动路线之间的关系。具体来说,约束(17)规定卡车在没有司机引导的情况下不能移动。约束(18)确定二元决策变量 β_{ij} 的值,以表示是否至少存在一辆卡车穿过弧(i,j)。约束(19)意味着司机可以在没有卡车的情况下穿过弧。约束(20)-(23)跟踪卡车流量。约束(20)是每个OD对的流量守恒。约束(21)确保每个OD对的流量平衡。约束(22)保证针对每个任务节点,用于通过该任务节点连接OD对的弧数等于卡车流量。约束(23)表示用于连接OD对的弧被至少一辆卡车穿过。约束(24)-(27)是任务节点时间安排的约束。约束(24)确保必须满足所需的装/卸货时间。约束条件(25)和(26)规定每个任务必须在司机每日最长工作时间内完成。约束(27)指定同一司机连续访问的两个客户之间的服务开始时间关系。约束(28)-(29)定义了决策变量的值域。

[0118] 步骤S300具体包括:

[0119] S310、为司机构造一个字符串,从码头开始依次插入尚未服务的任务节点;其中,在插入尚未服务的任务节点时,选择与上次插入的任务节点的距离最近的任务节点插入。

[0120] S320、当该司机的字符串中插入的任务节点达到最大卡车排大小约束时,结束该司机的字符串。

[0121] S330、为下一司机构造一个字符串,从码头依次插入尚未服务的任务节点,直到所有任务节点都已插入到字符串中,得到所述网络图对应的初始解。

[0122] 具体地,本发明考虑的新型集装箱运输操作模式比现有操作模式更加复杂。具体来说,客户之间的空卡车共享、卡车排队行驶模式和允许司机使用除卡车外的其他可用运输模式将使得对所需司机和卡车的实际数量、卡车在客户之间的移动方式、司机在客户之间行驶的时间,以及每条弧上(一起)行驶的卡车数量的计算变得更加困难。因此,我们根据问题特点设计了一种特殊的启发式构造算法(HCSA)来求解提出的问题。HCSA的框架如图2所示。利用贪心算法来构造初始解。具体来说,为每个司机构造一个字符串,将根据从码头开始到最后一个插入节点的最近距离选择一个尚未服务的任务节点。如果在不违反最大卡车排大小约束的情况下无法插入任何一个尚未服务的任务节点,结束当前字符串。此过程将从码头再次开始,并继续构造下一个字符串,直到所有任务节点都已插入到字符串中。在初始解的生成过程中,维持卡车排大小约束的可行性是一个挑战,因为每个字符串中所占用的卡车的实际数量(即每个字符串的车队大小)将根据试图插入字符串中的下一个任务节点的类型而动态变化:

[0123] (1)对于每个任务节点*i* ∈ D₁,其所需的卡车只能由码头满足。因此,车队规模增加了1。

[0124] (2)对于每个任务节点*i* ∈ D₂,如果*i'* ∈ D₁已经包含在字符串中,则车队规模大小保持不变;否则,车队规模将增加1。

[0125] (3)对于每个任务节点*i* ∈ P₁,其需求可以由码头或之前从第二阶段送货任务节点释放的空卡车来满足。因此,如果当前字符串至少包含一辆可用的空卡车,车队大小保持不变;否则,车队规模将增加1。

[0126] (4) 对于每个任务节点 $i \in P_2$, 如果 $i' \in P_1$ 已经包含在字符串中, 则车队大小保持不变; 否则, 车队规模将增加1。

[0127] 步骤S400具体包括:

[0128] S410、当所述初始解不满足任务节点时间安排的约束条件时, 重新确定所述网络图对应的初始解。

[0129] S420、当所述初始解满足任务节点时间安排的约束条件时, 基于所述目标函数, 对所述初始解进行评估, 得到评估结果。

[0130] S430、当所述评估结果满足预设要求时, 将所述初始解作为所述网络图对应的更新解。

[0131] 具体地, 在确定初始解时判断了是否符合卡车排大小的约束条件, 由于, 无人驾驶的卡车在领头卡车的引领下行驶, 但是卡车队列中的卡车的数量有约束限制, 需要满足小于或等于卡车队列中最大卡车数量L的要求。满足该要求时, 则生成初始解。然后进行实践可行性检验, 约束条件(24)-(27)是任务节点时间安排的约束, 若初始解不满足任务节点时间安排的约束条件时, 返回步骤S300, 重新确定初始解。若初始解满足任务节点时间安排的约束条件时, 基于所述目标函数, 对所述初始解进行评估, 得到评估结果, 例如, 可以评估司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本。

[0132] 举例说明, 以总运营成本进行评估, 预设要求可以是总运营成本小于预设成本, 评估结果满足预设要求时, 表明初始解的质量较高, 则可以将该初始解作为更新解。

[0133] 以上的启发式求解方法虽然可以获得一个可行解, 但可能并非一个高质量的解, 因此, 我们将模拟退火机制融入到启发式求解算法中, 以使解的结构多样化, 并以有效的迭代方式指导搜索到更好的解。

[0134] 步骤S400还包括:

[0135] S440、当所述评估结果不满足预设要求时, 采用邻域搜索算子对所述初始解进行搜索得到邻域解; 其中, 所述邻域搜索算子包括: 交换算子、反转算子以及插入算子。

[0136] S450、当所述领域解对应的目标函数值小于所述初始解对应的目标函数值时, 将所述邻域解作为所述网络图对应的初始解, 并降低所述初始解对应的温度。

[0137] S460、当所述初始解符合终止条件时, 将所述初始解作为所述网络图对应的更新解。

[0138] 具体地, 评估结果不满足预设要求时, 需要对初始解进行更新。采用三种不同的邻域搜索算子, 即交换、反转和插入。具体来说, 交换算子是交换解中的字符串中两个随机选择的节点的位置。反转是随机选择字符串中的一对节点, 然后反转它们之间的子字符串顺序来执行的。插入算子将随机选择的节点被移除并重新插入同一字符串中的另一个位置。模拟退火在每次迭代中将从上述三个邻域搜索算子中随机选择一个, 以获得更好的领域解。

[0139] 步骤S400还包括:

[0140] S470、当所述领域解对应的目标函数值大于或等于所述初始解对应的目标函数值时, 根据Metropolis准则, 确定所述网络图对应的初始解, 并降低所述初始解对应的温度。

[0141] S480、当所述初始解不符合终止条件时, 继续判断所述初始解是否满足任务节点

时间安排的约束条件,直至所述初始解满足评估结果满足预设要求或所述初始解符合终止条件时,将所述初始解作为所述网络图对应的更新解。

[0142] 具体地,所述终止条件包括:所述初始解对应的温度小于预设的温度阈值或所述初始解的迭代次数达到最大连续无提高迭代次数。降低所述初始解对应的温度时,根据退火速率降低温度。模拟退火算法(以下简称为 SA)从一个启发式构造的初始解开始 S^0 。然后,通过预定义的邻域结构,将当前解 S (从初始解 S^0 开始)迭代转换为其邻域解 S' 。如果 S' 的目标函数值小于 S 的目标函数值,我们将用领域解 S' 替换当前解 S ;否则,我们将以Metropolis概率接受新的解,例如, $e^{\frac{-\Delta H}{T}}$,其中 ΔH 代表解 S' 和解 S 的目标函数值的差(例如, $\text{obj}(S') - \text{obj}(S)$), T 是当前温度。由以上可以看出,SA算法允许以较小的概率接受不好的解,这有效地降低了陷入局部极小值的可能性。重复该搜索过程,直到满足停止条件,即预先指定的温度阈值 T^{end} 或最大连续无提高迭代次数 B_{\max} 。最终,返回最优解 S^* 。SA算法的本质是将温度从给定的初始温度 T^0 逐渐降低到指定的终止温度 T^{end} 。经过多次连续尝试后,当前温度 T 将根据冷却规则降低 $T \leftarrow T \cdot \epsilon$,为了获得更好的解决方案,其中 ϵ 是退火速率,即0和1之间的常数,以控制冷却速度。

[0143] 具体实施例一

[0144] 我们进行了一个计算实验来评估所提出的模型和算法的有效性。实验采用了随机生成的算例来测试模型和算法。客户在长度为200km的正方形区域内随机选择,而码头位于区域中心,坐标为(100,100)。每个客户只有一个送货或取货请求,其中所需的卸货或装货时间,以小时为单位,并从集合{2,3,...,5}中随机生成。本章指定摩托车作为可供司机在任何客户地点使用的替代交通方式。考虑到卡车的行驶速度为60km/h,摩托车的行驶速度为35km/h,可以根据两个客户之间的欧几里得距离获得两个客户之间的行驶时间。根据考虑卡车列队行使的LCDP的现有参数设置,本文将模型参数设置如下:形成排的节油系数 η 、最大卡车排队行驶数量 L 和司机日最长工作时间 T 分别设置为0.1、6和16。至于成本相关的参数,雇佣司机和卡车的每日固定成本 λ_1 和 λ_2 分别设置为100和50。卡车的单位燃油消耗成本和摩托车的单位旅行成本 c_{ij} 和 c'_{ij} 分别设置为1和0.5。基于一些初步实验和SA准则,我们设置了以下算法参数:初始的温度 T^0 ,温度阈值 T^{end} ,退火速率 ϵ 和最大连续无提高迭代次数 B_{\max} ,在实验中分别设置为20、0.0001、0.9999和3000。此外,使用 $T^{\text{end}}=0.0001$ 和 $B_{\max}=3000$ 作为HCSA的停止条件。基于这些参数设置,我们对多组客户规模从4到400不等的随机算例进行了数值实验。每组算例均以“a+b”命名,其中a和b分别表示送货和取货客户的数量。对于每个客户规模,将生成五个随机算例,并展示平均求解结果。

[0145] 为了评估使用HCSA求解LCDP-IPOM的求解性能,我们将比较使用 HCSA和直接使用 CPLEX商业求解器求解所提出的模型得到的结果。CPLEX求解每个算例的最大运行时间设置为1小时。请注意,HCSA算法求解时每个算例将运行3次,并展示平均计算结果。表2给出了 HCSA算法和CPLEX的求解结果,每个指标的最小值以粗体突出显示。对于这两种方法,我们都展示了在时间限制内获得的目标函数值(Obj)和经过的CPU 运行时间(time)。我们还将展示HCSA和CPLEX之间的平均目标值差距(Gap)。如表2所示,我们发现CPLEX无法在1小时内为超过30个客户的算例找到可行解,只有规模最小的实例可以由CPLEX直接求解。相比之下,所提出的HCSA平均可以在8分钟内为所有算例找到高质量的解。更具体地说,对于CPLEX

可以在时间限制内解决的大多数情况,HCSA可以在比CPLEX短得多的时间内获得相同或更好的解决方案,即平均69s v.s.3004s。对于CPLEX难以求解的算例,HCSA平均只在0.3小时内找到了高质量的解。特别地,HCSA可以在43分钟内为规模达到400(200+200)的算例获得良好的解。上述发现一方面证明了所提出问题的相当大的计算复杂性,以及证明了所提出的算法在解决该问题方面的有效性和效率。为了进一步说明HCSA的计算性能,我们在图3中可视化了HCSA的CPU运行时间相对于客户规模的变化。一个值得注意的观察结果是,随着客户规模的增加,CPU时间呈线性增长趋势。这与我们的预期一致,即提出的基于模拟退火的启发式方法利用了元启发式的优势,可以在合理时间内找到相当好的解,这表明HCSA在LCDP的实际应用中具有巨大的潜力。

[0146] 表2.HCSA和CPLEX求解结果比较

| a+b | HCSA | | CPLEX | | Gap | |
|--------|------------------|----------|------------------|----------|-------|---------|
| | Obj ¹ | Time (s) | Obj ² | Time (s) | | |
| [0147] | 2+2 | 254.41 | 84 | 254.41 | 45 | 0.00% |
| | 3+3 | 257.14 | 74 | 256.23 | 3,600 | 0.36% |
| | 2+4 | 208.01 | 64 | 208.00 | 3,600 | 0.00% |
| | 4+2 | 310.96 | 69 | 310.80 | 3,600 | 0.05% |
| | 4+4 | 310.42 | 81 | 311.93 | 3,600 | -0.48% |
| | 2+6 | 387.14 | 61 | 410.73 | 3,600 | -5.74% |
| | 6+2 | 501.71 | 72 | 515.53 | 3,600 | -2.68% |
| | 5+5 | 460.42 | 79 | 460.73 | 3,600 | -0.07% |
| | 2+8 | 422.45 | 78 | 514.39 | 3,600 | -17.87% |
| | 8+2 | 601.97 | 81 | 712.58 | 3,600 | -15.52% |
| | 4+6 | 378.68 | 83 | 410.99 | 3,600 | -7.86% |
| | 6+4 | 472.14 | 91 | 511.06 | 3,600 | -7.62% |
| | 10+10 | 1,049.43 | 195 | 2,968.36 | 3,600 | -64.65% |
| | 15+15 | 1,515.36 | 224 | 3,515.69 | 3,600 | -56.90% |

| | | | | | | |
|--------|---------|-----------|-------|---|-------|---|
| | 20+20 | 2,130.23 | 358 | - | 3,600 | - |
| | 30+30 | 2,970.02 | 494 | - | 3,600 | - |
| | 40+40 | 4,583.23 | 603 | - | 3,600 | - |
| | 50+50 | 5,180.36 | 724 | - | 3,600 | - |
| [0148] | 100+100 | 12,269.27 | 1,321 | - | 3,600 | - |
| | 150+150 | 19,114.89 | 1,885 | - | 3,600 | - |
| | 200+200 | 25,834.83 | 2,603 | - | 3,600 | - |
| | 平均值 | - | 467 | - | - | - |
| | | | | | | |

[0149] 注: $\text{Gap} = (\text{Obj}^1 - \text{Obj}^2) / \text{Obj}^2 \times 100\%$

[0150] 基于上述任意一实施例所述新型卡车排队行驶模式下的本地集装箱运输排版优化方法,本发明还提供了一种计算机设备的实施例。

[0151] 本发明的计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,所述处理器执行所述计算机程序时实现如下步骤:

[0152] 获取网络图;其中,所述网络图包括节点集合和弧的集合,所述节点集合中的节点包括送货客户节点、取货客户节点以及码头,所述弧的集合中的弧为两个所述节点之间的连接弧线;所述送货客户节点包括:卸货前的任务节点和卸货后的任务节点,所述取货客户节点包括:装货前的任务节点和装货后的任务节点;

[0153] 根据所述网络图,确定总运营成本的目标函数;其中,所述总运营成本包括:司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本;

[0154] 确定所述网络图对应的初始解;

[0155] 在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图对应的更新解。

[0156] 基于上述任意一实施例所述新型卡车排队行驶模式下的本地集装箱运输排版优化方法,本发明还提供了一种计算机可读存储介质的实施例。

[0157] 本发明的计算机可读存储介质,其上存储有计算机程序,所述计算机程序被处理器执行时实现如下步骤:

[0158] 获取网络图;其中,所述网络图包括节点集合和弧的集合,所述节点集合中的节点包括送货客户节点、取货客户节点以及码头,所述弧的集合中的弧为两个所述节点之间的连接弧线;所述送货客户节点包括:卸货前的任务节点和卸货后的任务节点,所述取货客户节点包括:装货前的任务节点和装货后的任务节点;

[0159] 根据所述网络图,确定总运营成本的目标函数;其中,所述总运营成本包括:司机雇佣成本、卡车部署成本、考虑到列队节油的卡车总油耗成本以及替代运输模式产生的旅行成本;

[0160] 确定所述网络图对应的初始解;

[0161] 在约束条件下,对所述初始解进行更新以最小化所述目标函数,得到所述网络图

对应的更新解。

[0162] 应当理解的是，本发明的应用不限于上述的举例，对本领域普通技术人员来说，可以根据上述说明加以改进或变换，所有这些改进和变换都应属于本发明所附权利要求的保护范围。

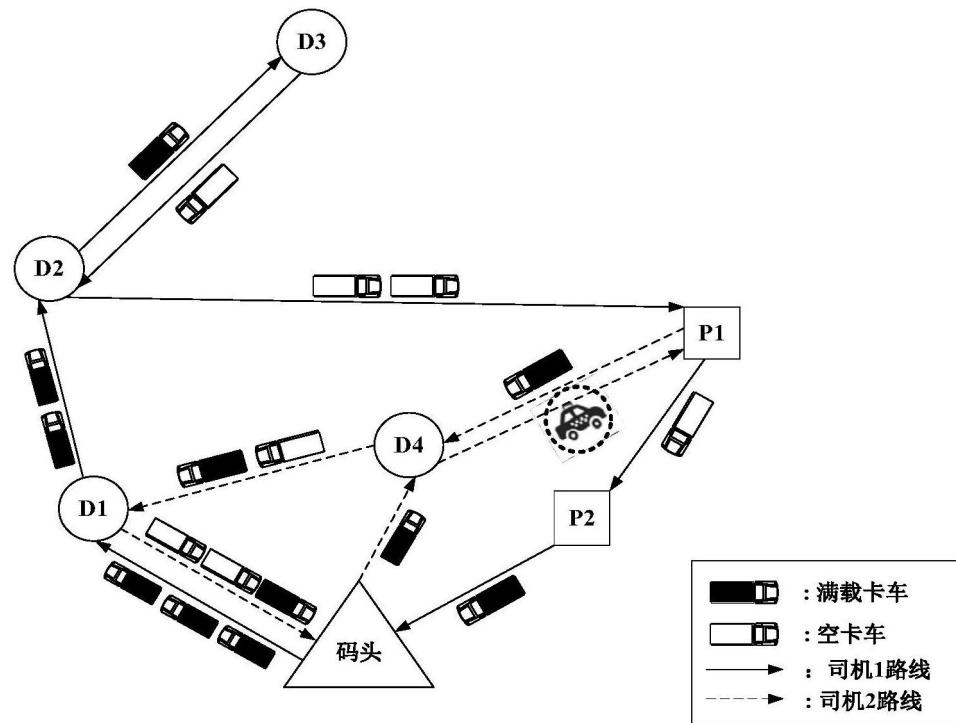


图1

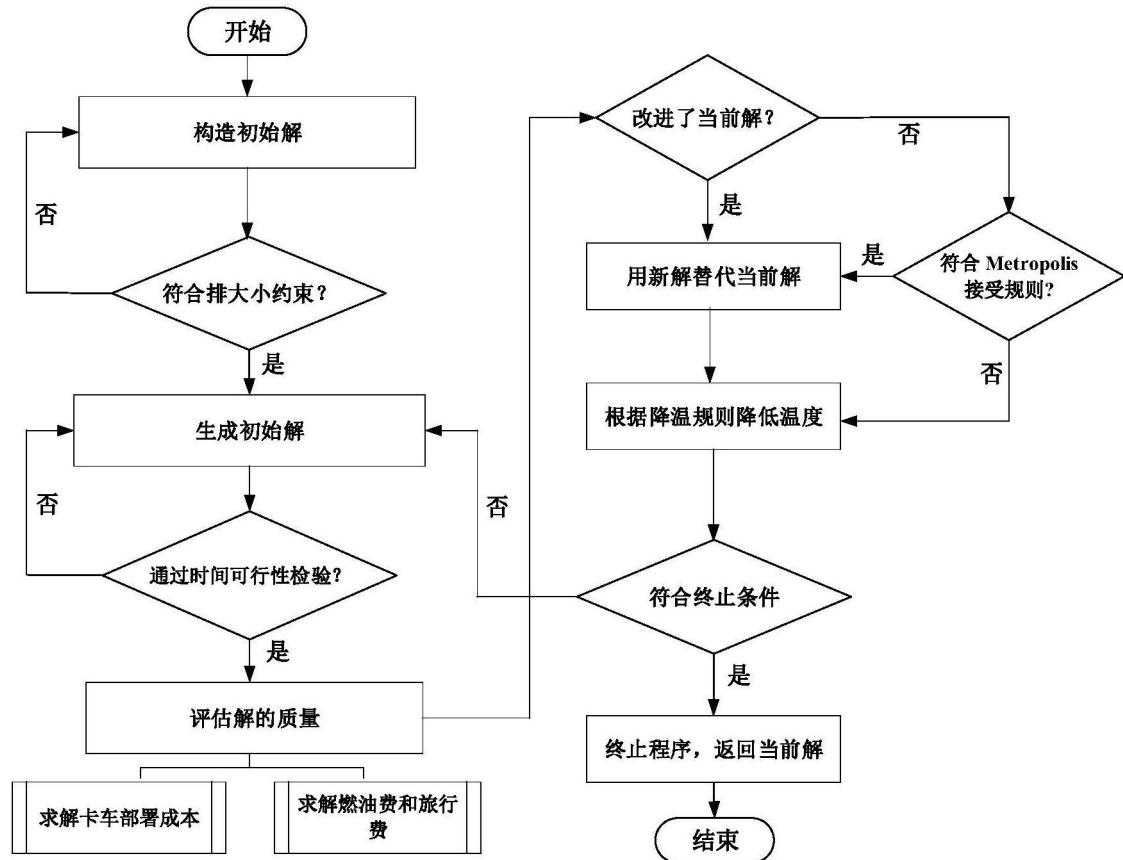


图2

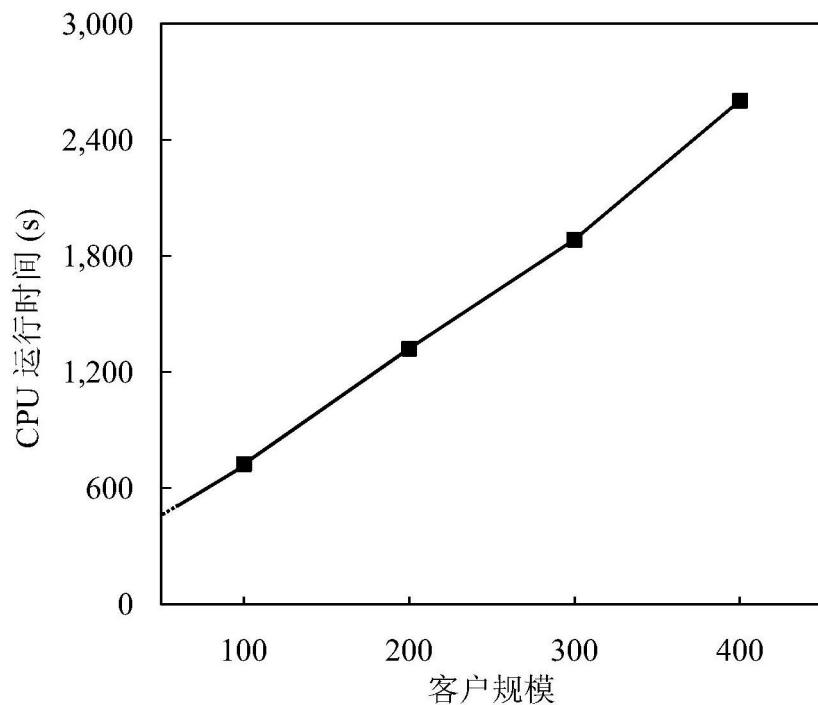


图3