

## 冷链物流越库调度的拉格朗日松弛算法

周炳海<sup>†</sup>, 宗 师

(同济大学 机械与能源工程学院, 上海 201804)

**摘要:** 为了提高冷链物流的运输效率, 解决越库在冷链物流中的应用问题, 提出了基于拉格朗日松弛算法的冷链物流的越库调度方法. 首先进行了问题域的描述并做出了具体假设, 基于问题域以最小化卡车等待时间和越库内部运输成本为目标, 建立越库调度的整数规划数学模型. 然后, 提出了针对越库调度模型的拉格朗日松弛算法, 松弛复杂约束后根据决策变量将松弛问题分解为若干子问题, 采用次梯度算法求解松弛模型. 最后, 对各种不同规模的越库模型进行仿真实验, 并与传统的贪婪算法进行对比, 结果表明, 所提出的调度算法适用于问题的求解, 并可以在较短时间内获得良好的近优解.

**关键词:** 越库; 冷链物流; 拉格朗日松弛; 次梯度算法; 调度

**引用格式:** 周炳海, 宗师. 冷链物流越库调度的拉格朗日松弛算法. 控制理论与应用, 2020, 37(3): 505 – 512

DOI: 10.7641/CTA.2019.90176

## Lagrangian relaxation algorithm for cross-dock scheduling & problems in cold-chain logistics

ZHOU Bing-hai<sup>†</sup>, ZONG Shi

(School of Mechanical Engineering, Tongji University, Shanghai 201804, China)

**Abstract:** In order to effectively improve the efficiency of cold-chain logistics and solve the scheduling problems in cold-chain logistics, a scheduling method based on a Lagrangian relaxation algorithm is proposed. First of all, a cross-dock scheduling problem in cold-chain logistics is presented and a few assumptions of the problem are depicted in detail in the paper. Then an integer mathematical programming model is constructed with objectives of minimizing total truck waiting time and inner cross-dock transportation cost. On that basis, a Lagrangian relaxation algorithm aiming at cross-docking scheduling is developed, which decomposes the relaxed problem into several sub-problems, and a Subgradient algorithm is introduced to solve these sub-problems. Finally, simulation experiments of different problem scales are carried out to analyze the proposed algorithm and compared with a Greedy algorithm. Results indicate that the two proposed algorithms are fit to solving the problem and could obtain satisfactory near-optimal solutions within the reasonable time.

**Key words:** corss-dock; cold-chain logistics; Lagrangian relaxation; subgradient algorithm; scheduling

**Citation:** ZHOU Binghai, ZONG Shi. Lagrangian relaxation algorithm for cross-dock scheduling problems in cold-chain logistics. *Control Theory & Applications*, 2020, 37(3): 505 – 512

### 1 引言

越库(cross-docking)是一种物流策略,旨在使货物在供应链中快速的运输和分拣.越库也是一种仓储策略,目的是减少或消除仓储,从而降低供应链的成本.越库这一策略现已经广泛地应用于零售、制造等各个行业.

多年来,越库调度问题引起了国内外学者的关注,并针对不同的应用场景进行了深入的研究.Nassief等<sup>[1]</sup>研究了带有临时存储区域的越库门分配问题(cro-

ss-dock assignment problems, CDAPs),给出了两个整数规划模型,并对模型参数进行了灵敏度分析.Yin等<sup>[2]</sup>考虑了绿色环保的越库调度问题,在越库模型中增添了二氧化碳浓度的约束,并提出一种新的记忆人工蜂群算法进行求解.Karimi等<sup>[3]</sup>研究了越库网络在伊朗邮政系统内的应用,通过设置邮件的预设到达时间以控制和提高整个邮政系统的送件速度.Serrano等<sup>[4]</sup>考虑在雷诺汽车供应链的越库内部加入包装车间,实现了运输和加工同时进行的需求.Nassief等<sup>[5]</sup>

收稿日期: 2019-03-24; 录用日期: 2019-07-18.

<sup>†</sup>通信作者. E-mail: bhzhou@tongji.edu.cn; Tel.: +86 13564164374.

本文责任编辑: 薛安克.

国家自然科学基金项目(71471135)资助.

Supported by the National Natural Science Foundation of China (71471135).

采用了拉格朗日松弛算法研究越库调度问题,但是仅研究了卡车到门的分配,未考虑卡车的排序问题.上述文献研究了越库在各类不同供应链或物流中的应用,其研究成果为接来越库调度问题的研究提供了很好的参考和借鉴.

冷链物流作为一种特殊的物流方式,其最大的技术难点在于严格的时间和低温控制.冷链物流强调货物运输过程中的全局控制,如果有环节出现“断链”,即货物温度高于其最高贮藏温度,就会影响到产品质量与安全,使得整个供应链的可靠性大大降低.此外,高昂的冷库存储成本也阻碍着冷链物流的应用和发展.许多学者和企业也一直在寻找一种针对冷链物流的及时且高效的运作方法.越库调度策略在物流系统中具有及时性、高效性、低成本的特点,越库作为供应链上的中转站和连接点大大提升了物流效率,降低了仓储成本.因此,将越库调度策略运用到冷链物流中,一方面,减少了冷库存储需求,大大降低了成本;另一方面,越库使得货物具有了快速被分拣并装载的能力,货物暴露在常温下的时间大大降低,减少了“断链”的风险.

针对冷链物流的特点,提出了越库内的货物时间窗口,要求每种货物在规定的时间内通过越库<sup>[6]</sup>.此外为了贴合实际应用,考虑设置越库门前临时存储区域,内设冷库,并分别给定不同的存储容量.基于越库的基本运作原理和冷链物流的特殊要求,提出了冷链物流越库调度的数学模型,并采用拉格朗日松弛算法,根据决策变量松弛复杂约束,提高模型求解速度和稳定性<sup>[7]</sup>.

## 2 数学建模

### 2.1 问题描述

研究的越库配送调度主要包含3个部分,示意图见图1.

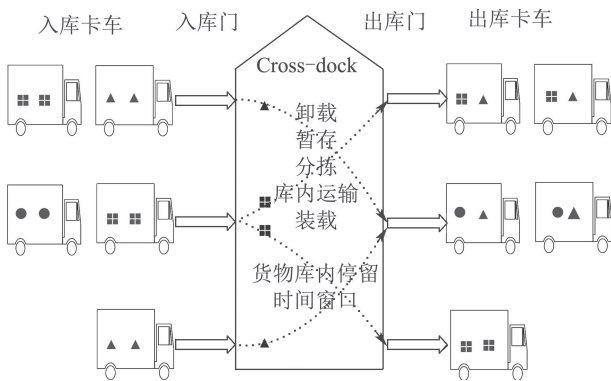


图1 冷链物流中的越库调度问题

Fig. 1 Cross-dock scheduling problem in cold-chain logistics

图1中: 1) 入库卡车到越库门的分配和入库卡车的排序: 将装载着货物的入库卡车分别分配至一入库门, 并对分配到同一入库门的卡车进行排序, 按照排

序进入越库. 2) 货物的库内运输: 入库卡车进入越库后, 将货物卸载于入库门前的临时存储区域, 然后立即将货物运输至为其分配的出库门, 暂存在出库门的临时存储区域内等待装载. 此阶段考虑冷链物流中, 每个货物在库内的时间窗口约束, 根据不同的货物种类, 要求货物在规定的时间内完成库内运输. 3) 出库卡车到越库门的分配和出库卡车的排序: 将出库卡车分别分配至一出库门, 并对分配到同一出库门的卡车进行排序, 按照排序进入越库, 装货后离开越库<sup>[8]</sup>.

### 2.2 数学模型

基于上述的问题描述, 建立如下数学模型:

#### 1) 模型假设.

① 每个越库门在同一时间只能服务于一辆运输卡车; ② 货物在越库内部运输时, 越库门距离越远, 运输时间越长; ③ 将货物分为 $k$ 个种类, 每个种类的货物对应于一个时间窗口约束 $t_k$ , 每种货物从入库到出库必须要在该时间窗口约束内完成; ④ 临时存储区域设有冷库, 可保证货物在低温下的存储. 同时, 每个临时存储区域设有容量限制, 区域内货物之和不得超出此容量限制.

#### 2) 参数.

$k \in P$ 为货物的种类;  $i \in I$ 为入库门编号;  $j \in J$ 为出库门编号;  $m \in M$ 为入库卡车编号;  $n \in N$ 为出库卡车编号;  $ULT$ 为一辆入库卡车的卸货时间;  $LT$ 为一辆出库卡车的装货时间;  $w_k$ 为货物 $k$ 的数量;  $s_m$ 为入库卡车 $m$ 装载的货物数量;  $r_n$ 为出库卡车 $n$ 装载的货物数量;  $S_i$ 为入库门 $i$ 前的临时存储区域容量;  $R_j$ 为出库门 $j$ 前的临时存储区域容量;  $d_{ij}$ 为入库门 $i$ 到出库门 $j$ 的运输时间成本;  $t_k$ 为货物 $k$ 在越库内部运输的时间窗口.

#### 3) 决策变量.

$z_{kij}$ : 0~1变量, 表示货物 $k$ 是否由入库门 $i$ 卸载并发送出库门 $j$ 装载;  $x_{mi}$ : 0~1变量, 表示入库卡车 $m$ 是否被分配到入库门 $i$ 进行卸货;  $y_{nj}$ : 0~1变量, 表示出库卡车 $n$ 是否被分配到出库门 $j$ 进行卸货;  $p_{mn}, q_{mn}$ : 0~1变量, 表示入(出)库卡车 $m$ 在卡车入(出)库排序中是否排在入(出)库卡车 $n$ 的前面;  $P_m, Q_n$ : 整数变量, 表示卡车在卡车入(出)库排序中的位次;  $C_m$ : 整数变量, 表示入库卡车 $m$ 进入越库进行卸货的时间;  $L_n$ : 整数变量, 表示出库卡车 $n$ 装货完成后离开越库的时间.

根据上述问题描述、模型假设、参数以及决策变量, 对所研究的越库模型建模如下:

$$\min \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J w_k d_{ij} z_{kij}, \quad (1)$$

$$\min \left\{ \sum_{m=1}^M (P_m + I) \cdot ULT + \sum_{n=1}^N L_n + J \cdot LT \right\}, \quad (2)$$

$$\text{s.t. } \sum_{i=1}^I \sum_{j=1}^J z_{kij} = 1, k = 1, \dots, P, \quad (3)$$

$$\begin{cases} \sum_{j=1}^J z_{kij} = x_{mi}, k = 1, \dots, P, \\ i = 1, \dots, I, m = 1, \dots, M, \end{cases} \quad (4)$$

$$\begin{cases} \sum_{i=1}^I z_{kij} = y_{nj}, k = 1, \dots, P, \\ i = 1, \dots, I, m = 1, \dots, M, \end{cases} \quad (5)$$

$$\sum_{i=1}^I x_{mi} = 1, m = 1, \dots, M, \quad (6)$$

$$\sum_{j=1}^J y_{nj} = 1, n = 1, \dots, N, \quad (7)$$

$$\sum_{m=1}^M s_m x_{mi} \leq S_i, i = 1, \dots, I, \quad (8)$$

$$\sum_{n=1}^N r_n y_{nj} \leq R_j, j = 1, \dots, J, \quad (9)$$

$$\begin{cases} \sum_{k=1}^P s_m z_{kij} \leq S_i y_{nj}, j = 1, \dots, J, \\ i = 1, \dots, I, m = 1, \dots, M, n = 1, \dots, N, \end{cases} \quad (10)$$

$$\begin{cases} \sum_{k=1}^P r_n z_{kij} \leq R_j x_{mi}, j = 1, \dots, J, \\ i = 1, \dots, I, m = 1, \dots, M, n = 1, \dots, N, \end{cases} \quad (11)$$

$$\sum_{i=1}^I \sum_{j=1}^J z_{kij} d_{ij} \leq t_k, k = 1, \dots, P, \quad (12)$$

$$\begin{cases} x_{mi} + x_{m'i} \leq p_{mm'} + p_{m'm} + 1, \\ m, m' = 1, \dots, M, m \neq m', \end{cases} \quad (13)$$

$$\begin{cases} p_{mm'} + p_{m'm} \leq 1, \\ m, m' = 1, \dots, M, m \neq m', \end{cases} \quad (14)$$

$$\begin{cases} y_{nj} + y_{n'j} \leq q_{nn'} + q_{n'n} + 1, \\ n, n' = 1, \dots, M, n \neq n', \end{cases} \quad (15)$$

$$q_{nn'} + q_{n'n} \leq 1, n, n' = 1, \dots, N, n \neq n', \quad (16)$$

$$P_m = \sum_{m'=1}^M p_{m'm} + 1, m, m' = 1, \dots, M, \quad (17)$$

$$Q_n = \sum_{n'=1}^N q_{n'n} + 1, n, n' = 1, \dots, N, \quad (18)$$

$$\begin{cases} C_m = z_{kij} P_m ULT, \\ m = 1, \dots, M, i = 1, \dots, I, \end{cases} \quad (19)$$

$$\begin{cases} C_m + d_{ij} - M(1 - x_{mi} z_{kij} y_{nj}) \leq L_n, \\ m = 1, \dots, M, n = 1, \dots, N, \\ i = 1, \dots, I, j = 1, \dots, J, k = 1, \dots, P, \end{cases} \quad (20)$$

$$L_{n'} + LT - M(1 - q_{nn'}) \leq L_n, n = 1, \dots, N, \quad (21)$$

$$z_{kij}, x_{mi}, y_{nj}, p_{mn}, q_{mn} \in \{0, 1\}. \quad (22)$$

上述模型中, 目标函数(1)是库内运输成本之和; 目标函数(2)是卡车等待时间之和; 约束(3)表明每种货物只能且必须被分配给一个入库门和一个出库门; 约束(4)–(7)表明每辆卡车只能且必须分配给一个越库门; 约束(8)–(11)给定了每个越库门前临时存储区域的容量限制; 约束(12)表明每一种货物 $k$ 在库内运输时与其对应的时间窗口; 约束(13)–(16)表明卡车在越库门前等待队列中相互的先后顺序; 约束(17)–(18)表明了每辆卡车在越库门前等待队列中的位次; 约束(19)为入库卡车的入库卸货时间; 约束(20)表明了入库卡车的卸货时间与出库卡车的装货时间的关系; 约束(21)为出库卡车装货时间的先后关系; 约束(22)为0~1变量约束。

### 3 拉格朗日松弛算法

#### 3.1 松弛约束

将原模型中的约束(4)–(5)和约束(10)–(11)松弛到目标函数(1)中, 引入拉格朗日乘子 $\mu, \nu, \lambda$ 和 $\gamma$ , 形成如下的松弛问题:

$$\begin{aligned} L(\mu, \nu, \lambda, \gamma) = \min \{ & \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J w_k d_{ij} z_{kij} + \\ & \sum_{k=1}^P \sum_{i=1}^I \mu_{ki} \left( \sum_{m=1}^M z_{kij} - x_{mi} \right) + \\ & \sum_{k=1}^P \sum_{j=1}^J \nu_{kj} \left( \sum_{n=1}^N z_{kij} - y_{nj} \right) + \\ & \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J \lambda_{kij} \left( \sum_{n=1}^N r_n z_{kij} - R_j x_{mi} \right) + \\ & \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J \gamma_{kij} \left( \sum_{m=1}^M s_m z_{kij} - S_i y_{nj} \right) \}. \end{aligned}$$

将原模型中的约束(19)松弛到目标函数(2)中, 引入拉格朗日乘子 $\eta$ , 形成如下的松弛问题:

$$\begin{aligned} L(\eta) = \min \{ & (1 - \eta) ULT \sum_{m=1}^M P_m + \\ & (1 - \eta) \sum_{n=1}^N L_n + I \cdot ULT + J \cdot LT \}, \end{aligned}$$

s.t. 式(13)(18)(20)(22).

拉格朗日对偶问题可表示为

$$\begin{aligned} \max \{ & L(\mu, \nu, \lambda, \gamma) \} = \\ \max \{ & \min \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J (w_k d_{ij} + \mu_{ki} + \nu_{kj} + \\ & r_n \lambda_{kij} + s_m \gamma_{kij}) z_{kij} \} + \\ \max \{ & \min \sum_{m=1}^M \sum_{i=1}^I \left( \sum_{k=1}^P \mu_{ki} + \sum_{j=1}^J R_j \lambda_{kij} \right) x_{mi} \} + \\ \max \{ & \min \sum_{n=1}^N \sum_{j=1}^J \left( \sum_{k=1}^P \nu_{kj} + \sum_{i=1}^I S_i \gamma_{kij} \right) y_{nj} \}, \end{aligned}$$

$$\begin{aligned} \max\{L(\eta)\} = \\ \max\{\min(1-\eta)ULT \sum_{m=1}^M P_m + I \cdot ULT\} + \\ \max\{\min(1-\eta) \sum_{n=1}^N L_n + J \cdot LT\}. \end{aligned}$$

### 3.2 求解子问题

根据拉格朗日分解思想, 可将 $L(\mu, \nu, \lambda, \gamma)$ 和 $L(\eta)$ 根据不同的决策变量分解成多个子问题: 1) 关于决策变量 $z$ 的子问题 $L_z(\mu, \nu, \lambda, \gamma)$ ; 2) 关于决策变量 $x$ 的子问题 $L_x(\nu, \lambda)$ ; 3) 关于决策变量 $y$ 的子问题 $L_y(\nu, \gamma)$ ; 4) 关于决策变量 $P_m$ 的子问题 $L_p(\eta)$ ; 5) 关于决策变量 $L_n$ 的子问题 $L_q(\eta)$ <sup>[9]</sup>.

1) 求解子问题 $L_z(\mu, \nu, \lambda, \gamma)$ .

$L_z(\mu, \nu, \lambda, \gamma)$ 为关于决策变量 $z$ 的子问题, 是货物在越库内部运输路线的分配问题, 可以被分解为基于货物种类 $k$ 的独立子问题

$$L_z(\mu, \nu, \lambda, \gamma) = \min \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J (w_k d_{ij} + \mu_{ki} + \nu_{kj} + r_n \lambda_{kij} + s_m \gamma_{kij}) z_{kij},$$

$$\text{s.t. } \sum_{i=1}^I \sum_{j=1}^J z_{kij} = 1, \quad k = 1, \dots, P, \quad (23)$$

$$\sum_{i=1}^I \sum_{j=1}^J z_{kij} d_{ij} \leq t_k, \quad k = 1, \dots, P, \quad (24)$$

$$z_{kij} \in \{0, 1\}. \quad (25)$$

基于决策变量约束(23)和货物的时间窗口约束(24), 求解出子问题 $L_z$ , 从而得到货物 $k$ 在越库中的由入库门到出库门的移动路线.

2) 求解子问题 $L_x(\mu, \lambda)$ 和 $L_y(\nu, \gamma)$ .

$L_x$ 为关于决策变量 $x$ 的子问题, 是入库卡车到入库门的分配问题, 可以被分解为基于入库门 $i$ 的独立子问题

$$L_x(\mu, \lambda) = \min \left\{ - \sum_{m=1}^M \sum_{i=1}^I \left( \sum_{k=1}^P \mu_{ki} + \sum_{j=1}^J R_j \lambda_{kij} \right) x_{mi} \right\},$$

$$\text{s.t. } \sum_{i=1}^I x_{mi} = 1, \quad m = 1, \dots, M, \quad (26)$$

$$\sum_{m=1}^M s_m x_{mi} \leq S_i, \quad i = 1, \dots, I, \quad (27)$$

$$x_{mi} \in \{0, 1\}. \quad (28)$$

同理,  $L_y$ 为关于决策变量 $y$ 的子问题, 是出库卡车到出库门的分配问题, 可以被分解为基于出库门 $j$ 的独立子问题

$$L_y(\nu, \gamma) = \min \left\{ - \sum_{n=1}^N \sum_{j=1}^J \left( \sum_{k=1}^P \nu_{kj} + \sum_{i=1}^I S_i \gamma_{kij} \right) y_{nj} \right\},$$

$$\text{s.t. } \sum_{j=1}^J y_{nj} = 1, \quad n = 1, \dots, N, \quad (29)$$

$$\sum_{n=1}^N r_n y_{nj} \leq R_i, \quad j = 1, \dots, J, \quad (30)$$

$$z_{kij}, x_{mi}, y_{nj}, p_{mn}, q_{mn} \in \{0, 1\}. \quad (31)$$

求解出子问题 $L_x$ 和 $L_y$ 得到卡车到越库门的分配情况.

3) 求解子问题 $L_p(\eta)$ 和 $L_q(\eta)$ .

将求解子问题 $L_x(\mu, \lambda)$ 和 $L_y(\nu, \gamma)$ 得到的 $x_{mi}$ 和 $y_{nj}$ 代入式(32)–(33), 即

$$x_{mi} + x_{m'i} \leq p_{mm'} + p_{m'm} + 1, \quad (32)$$

$$y_{nj} + y_{n'j} \leq q_{nn'} + q_{n'n} + 1, \quad (33)$$

其中:  $m, m' = 1, \dots, M, m \neq m', n, n' = 1, \dots, M, n \neq n'$ . 然后进行车辆排序问题的求解, 即求解子问题 $L_p(\eta)$ 和 $L_q(\eta)$ .

$L_p(\eta)$ 为关于决策变量 $P_m$ 的子问题是入库卡车在每一个入库门的排序问题:

$$L_p(\eta) = \min \left\{ (1-\eta)ULT \sum_{m=1}^M P_m + I \cdot ULT \right\},$$

$$\text{s.t. } x_{mi} + x_{m'i} \leq p_{mm'} + p_{m'm} + 1, \quad (34)$$

$$p_{mm'} + p_{m'm} \leq 1, \quad (35)$$

$$p_{mn} \in \{0, 1\}, \quad (36)$$

其中 $m, m' = 1, \dots, M, m \neq m'$ .

$L_q(\eta)$ 为关于决策变量 $L_n$ 的子问题是出库卡车在每一个出库门的排序问题:

$$L_q(\eta) = \min \left\{ (1-\eta) \sum_{n=1}^N L_n + J \cdot LT \right\},$$

$$\text{s.t. } \begin{cases} y_{nj} + y_{n'j} \leq q_{nn'} + q_{n'n} + 1, \\ n, n' = 1, \dots, M, n \neq n', \end{cases} \quad (37)$$

$$q_{nn'} + q_{n'n} \leq 1, \quad n, n' = 1, \dots, N, n \neq n', \quad (38)$$

$$P_m = \sum_{m'=1}^M p_{m'm} + 1, \quad m, m' = 1, \dots, M, \quad (39)$$

$$Q_n = \sum_{n'=1}^N q_{n'n} + 1, \quad n, n' = 1, \dots, N, \quad (40)$$

$$\begin{cases} C_m + d_{ij} - M(1 - x_{mi} z_{kij} y_{nj}) \leq L_n, \\ m = 1, \dots, M, n = 1, \dots, N, i = 1, \dots, I, \\ j = 1, \dots, J, k = 1, \dots, P, \end{cases} \quad (41)$$

$$L_{n'} + LT - M(1 - q_{nn'}) \leq L_n, \quad n = 1, \dots, N, \quad (42)$$

$$q_{mn} \in \{0, 1\}. \quad (43)$$

求解出子问题 $L_p(\eta)$ 和 $L_q(\eta)$ 得到卡车在越库门的排序.

### 3.3 次梯度算法

引入次梯度算法来求解拉格朗日对偶问题, 每次

迭代过程中根据如下公式进行拉格朗日乘子更新:

$$(\mu, \nu, \lambda, \gamma, \eta)^{t+1} = (\mu, \nu, \lambda, \gamma, \eta)^t + \beta \frac{z_{UP}^t - L(\mu, \nu, \lambda, \gamma, \eta)}{\|s(\mu, \nu, \lambda, \gamma, \eta)\|^2},$$

其中:

$$\begin{aligned} s(\mu, \nu, \lambda, \gamma, \eta) = & \{(\sum_{j=1}^J z_{kij}(\mu, \nu, \lambda, \gamma) - x_{mi}(\mu, \lambda)), \\ & (\sum_{i=1}^I z_{kij}(\mu, \nu, \lambda, \gamma) - y_{nj}(\nu, \gamma)), \\ & (\sum_{k=1}^P r_j z_{kij}(\mu, \nu, \lambda, \gamma) - R_j x_{mi}(\mu, \lambda)), \\ & (\sum_{k=1}^P s_i z_{kij}(\mu, \nu, \lambda, \gamma) - S_i y_{nj}(\nu, \gamma)), \\ & ((\sum_{m=1}^M z_{kij} P_m ULT - \sum_{n=1}^N L_n) \eta + I \cdot ULT + J \cdot LT)\}, \end{aligned}$$

$z_{UP}^t$ 为每一步目标函数的值. 目标函数(1)

$$\min \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J w_k d_{ij} z_{kij}$$

是货物在越库内部的运输成本, 而目标函数(2)

$$\min \{ \sum_{m=1}^M (P_m + I) \cdot ULT + \sum_{n=1}^N L_n + J \cdot LT \}$$

是卡车在越库门的等待时间. 对两个目标函数进行无量纲化处理并加权相加, 得到一个统一的目标函数<sup>[10]</sup>

$$\begin{aligned} (1-b) \min & \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J w_k d_{ij} z_{kij} + \\ b \min & \{ \sum_{m=1}^M (P_m + I) \cdot ULT + \sum_{n=1}^N L_n + J \cdot LT \}. \end{aligned}$$

在次梯度法的应用过程中,  $z_{UP}^t$ 可表示为

$$z_{UP}^t = (1-b) \cdot L(\mu, \nu, \lambda, \gamma) + b \cdot L(\eta),$$

则 $z_{UP}^t$ 为运用次梯度法求解松弛模型的过程中, 每一步迭代求解的目标函数.

### 3.4 构造可行解

上述拉格朗日松弛的过程在大多数情况下得到的是目标函数的下界及其对应的不可行解. 现采用拉格朗日启发式算法, 对上述传统的拉格朗日算法进行改进, 获得可行解.

拉格朗日启发式算法共包含两个过程: 第1步, 如上述过程采用次梯度法进行优化迭代计算; 第2步, 采用系数修正法对第一步得到的解进行可行化. 系数修正法是拉格朗日启发式算法的一种形式, 用于检验用次梯度法求解的当前解是否满足各条被松弛的约束. 若存在未被满足的松弛约束, 则选取出其对应的目标函数系数中的最小值 $d_i$ , 并更新其对应的拉格朗日 $\lambda^{(t+1)} = \lambda^t + d_i$ , 直到此条约束被满足. 直到当所有

的约束(无论是否被松弛)都被满足时, 则当前解为可行解.

### 3.5 拉格朗日松弛算法流程

**步骤1** 初始化, 令当前迭代次数 $t = 0$ , 令所有拉格朗日乘子为0;

**步骤2** 求解每个子问题, 目标函数值加权相加, 得到下界;

**步骤3** 对松弛问题的解通过系数修正法进行可行化;

**步骤4** 判断是否符合次梯度法的停止准则, 若符合则算法停止, 否则继续;

**步骤5** 进行拉格朗日乘子更新, 返回步骤2<sup>[11]</sup>.

## 4 实验与分析

本实验结果通过以下数据进行衡量: 1) 每一组数据运算的CPU时间; 2) 对偶间隙(%gap);

$$\%gap = 100 \frac{(OPT - LP)}{LP},$$

其中LP是每组实验通过拉格朗日松弛算法得到的目标函数值下界, OPT是根据系数修正法得到的可行解;

3) 对比算法的偏移(%dev):

$$\%dev = 100 \frac{(UB - OPT)}{OPT},$$

其中UB是每一组实验数据通过贪婪算法得到的目标函数值上界. 通过可行解和下界、上界比较得到了对偶间隙(%gap)和对比算法的偏移(%dev), 这两个参数是拉格朗日松弛算法重要的评价指标, 反映了可行解的优良程度和当前算法的可靠性. 设定最大迭代次数为500, 次梯度优化算法中取参数 $\beta = 2^{[12]}$ .

本实验测试了所提出的拉格朗日松弛算法求解各种不同规模越库调度问题的性能. 问题实例数据取值如下: 卡车数量 $M, N = 5, 6, 7, 8, 9, 10, 11, 12, 15$ ; 货物数量 $P = 3, 4, 5, 6$ ; 每种货物的数量 $w_k$ 在 $[200, 500]$ 间随机生成; 临时存储区域的总量

$$\sum_i S_i = \sum_j R_j = \sum_k w_k \times (1 + \alpha),$$

其中:  $\alpha$ 分别取0.1, 0.15, 0.2, 0.3; 库内运输成本

$$d_{ij} = 100 + |i - j| \times 10,$$

每种货物 $k$ 的库内运输时间窗口在库内最长运输时间的60%~100%间离散均匀分布产生. 对于每种车-门数量组合, 根据临时存储区域总量放宽 $\alpha$ 产生3~4种实验实例, 一共产生49组实验.

原模型内双目标分别为库内运输成本之和以及卡车等待时间之和, 两个目标数值差异较大, 对最终结果的影响程度也相差较大. 本实验用无量纲化加权的方法, 将原模型内的双目标转换为单目标. 经过多次

实验, 取 $b = 0.95$ , 此时统一的目标函数为

$$0.05 \min \sum_{k=1}^P \sum_{i=1}^I \sum_{j=1}^J w_k d_{ij} z_{kij} + \\ 0.95 \min \left\{ \sum_{m=1}^M (P_m + I) \cdot ULT + \sum_{n=1}^N L_n + J \cdot LT \right\}.$$

此时两个目标的数值接近, 对模型的影响程度相近. 以问题1为例, 经多次迭代, 库内运输成本之和为11520, 卡车等待时间之和为741, 采用 $b = 0.95$ 的加权方法后两目标函数分别为576和704, 符合预期.

采用贪婪算法作为对比算法: 一方面通过贪婪算法得到每组实验目标函数的初始上界; 另一方面与通

过拉格朗日松弛算法得到的近优解进行对比, 判断解的质量. 对比贪婪算法思路如下: 将每辆卡车的货物由多至少进行排列, 按照此顺序依次进入越库, 每辆卡车在入库时选取当前库门剩余临时存储区域最大的入库门进入; 入库后, 依然按照上述排列顺序依次出库, 每辆出库卡车依次选取当前库门剩余临时存储区域最大的出库门.

算法采用以下计算机配置进行实现: CPU: 1.6 GHz Intel Core i5; 内存: 4 GB; 电脑系统: Windows 7; 编程语言: MATLAB.

实验结果如表1所示.

表1 不同规模问题的拉格朗日松弛算法与贪婪算法的结果对比

Table 1 Comparison of Lagrangian relaxation algorithm and greedy algorithm of different scale problems

| 问题 | 卡车数 $\times$ 越库门数 | 临时存储区域宽放比 | 拉格朗日松弛算法 |      |         | 对比贪婪算法   |       |
|----|-------------------|-----------|----------|------|---------|----------|-------|
|    |                   |           | 目标函数可行解值 | 对偶间隙 | CPU时间/s | 目标函数上界   | 偏移    |
| 1  | 5 $\times$ 3      | 15        | 1280     | 9.27 | 16.886  | 1408     | 14.52 |
| 2  | 5 $\times$ 3      | 20        | 1329.55  | 7.03 | 15.91   | 1612.5   | 12.76 |
| 3  | 5 $\times$ 3      | 30        | 1301     | 7.02 | 15.936  | 1576     | 12.76 |
| 4  | 6 $\times$ 3      | 15        | 1539.15  | 6.33 | 26.811  | 2081.25  | 26.62 |
| 5  | 6 $\times$ 3      | 20        | 1602.3   | 4.32 | 26.514  | 2128.4   | 24.27 |
| 6  | 6 $\times$ 3      | 30        | 1539.85  | 4.29 | 25.453  | 2007.95  | 26061 |
| 7  | 7 $\times$ 3      | 15        | 1994.05  | 4.81 | 47.043  | 2359.875 | 13.37 |
| 8  | 7 $\times$ 3      | 20        | 1948.5   | 4.96 | 47.69   | 2258.8   | 15.11 |
| 9  | 7 $\times$ 3      | 30        | 1865.95  | 3.55 | 53.188  | 2312.675 | 21.98 |
| 10 | 7 $\times$ 4      | 15        | 1817.75  | 6.04 | 118.322 | 2525.25  | 30.53 |
| 11 | 7 $\times$ 4      | 20        | 1814     | 6.23 | 123.468 | 2524.325 | 30.53 |
| 12 | 7 $\times$ 4      | 30        | 1820.65  | 5.38 | 140.137 | 2506.85  | 31.24 |
| 13 | 8 $\times$ 3      | 15        | 2079.3   | 5.17 | 106.756 | 2736     | 24.08 |
| 14 | 8 $\times$ 3      | 20        | 2113.95  | 5.03 | 111.306 | 2718.9   | 22.91 |
| 15 | 8 $\times$ 3      | 30        | 2092.2   | 5.27 | 167.472 | 2606.95  | 23.49 |
| 16 | 8 $\times$ 4      | 15        | 2050     | 4.96 | 140.528 | 2986.25  | 36.3  |
| 17 | 8 $\times$ 4      | 20        | 2090     | 4.76 | 152.81  | 2928     | 33.42 |
| 18 | 8 $\times$ 4      | 30        | 2090     | 4.76 | 226.699 | 2928     | 33.42 |
| 19 | 9 $\times$ 3      | 15        | 2370.75  | 5.17 | 408.316 | 2946.2   | 19.3  |
| 20 | 9 $\times$ 3      | 20        | 2517.25  | 4.53 | 363.15  | 3001.25  | 13.12 |
| 21 | 9 $\times$ 3      | 30        | 2380.15  | 4.8  | 340.158 | 2982.45  | 19.3  |
| 22 | 9 $\times$ 4      | 10        | 2427.8   | 5.08 | 494.003 | 3191.1   | 24.5  |
| 23 | 9 $\times$ 4      | 15        | 2406.5   | 4.97 | 453.737 | 3191.1   | 24.5  |
| 24 | 9 $\times$ 4      | 20        | 2382.2   | 4.32 | 453.868 | 3086.345 | 29.83 |
| 25 | 9 $\times$ 4      | 30        | 2323.35  | 5.08 | 455.136 | 3054     | 30.38 |
| 26 | 10 $\times$ 3     | 10        | 2763.4   | 5.25 | 847.115 | 3437.75  | 14.43 |
| 27 | 10 $\times$ 3     | 15        | 2728.15  | 5.09 | 678.582 | 3337.125 | 16.1  |
| 28 | 10 $\times$ 3     | 20        | 2733     | 4.92 | 580.693 | 3337.13  | 16.1  |
| 29 | 10 $\times$ 3     | 30        | 2971.45  | 4.96 | 590.377 | 3172     | 6.74  |
| 30 | 10 $\times$ 4     | 10        | 2797.5   | 5.86 | 389.199 | 3506.75  | 21.56 |
| 31 | 10 $\times$ 4     | 15        | 2597.5   | 4.76 | 532.835 | 3570.75  | 30.92 |
| 32 | 10 $\times$ 4     | 20        | 2543.5   | 5.56 | 541.308 | 3570.75  | 32.45 |
| 33 | 10 $\times$ 4     | 30        | 2687.5   | 4.76 | 701.738 | 3541     | 26.54 |

(转下页)

(接上页)

| 问题 | 卡车数 $\times$ 越库门数 | 临时存储区域宽放比 | 拉格朗日松弛算法 |      |           | 对比贪婪算法   |       |
|----|-------------------|-----------|----------|------|-----------|----------|-------|
|    |                   |           | 目标函数可行解值 | 对偶间隙 | CPU时间/s   | 目标函数上界   | 偏移    |
| 34 | 10 $\times$ 5     | 10        | 2527.5   | 6.82 | 742.803   | 3804.375 | 43.35 |
| 35 | 10 $\times$ 5     | 15        | 2677.5   | 5.36 | 716.115   | 3840.625 | 35.32 |
| 36 | 10 $\times$ 5     | 20        | 2537.5   | 4.76 | 808.187   | 3590.45  | 42.79 |
| 37 | 10 $\times$ 5     | 30        | 2657.5   | 4.76 | 795.166   | 3804.375 | 36.34 |
| 38 | 11 $\times$ 5     | 10        | 3008.5   | 6.86 | 1294.109  | 4309.125 | 31.41 |
| 39 | 11 $\times$ 5     | 15        | 3008.5   | 5.54 | 1382.954  | 4437.65  | 33.86 |
| 40 | 11 $\times$ 5     | 20        | 3018.5   | 7.41 | 1253.493  | 4064.775 | 32.18 |
| 41 | 11 $\times$ 5     | 30        | 3028.5   | 4.76 | 1257.295  | 4468.75  | 35.51 |
| 42 | 11 $\times$ 6     | 10        | 2998.5   | 9.09 | 1609.718  | 4817.425 | 38.69 |
| 43 | 11 $\times$ 6     | 15        | 2800     | 7.45 | 1524.341  | 4634.375 | 41.4  |
| 44 | 11 $\times$ 6     | 20        | 2888.5   | 7.41 | 1552.369  | 4823.85  | 46.64 |
| 45 | 11 $\times$ 6     | 30        | 2908.5   | 6.54 | 1548.82   | 4574.4   | 46.99 |
| 46 | 15 $\times$ 6     | 10        | 4111.5   | 9.09 | 5880.221  | 6243.6   | 37.39 |
| 47 | 15 $\times$ 6     | 15        | 4201.5   | 7.26 | 6384.5145 | 6235.1   | 35.68 |
| 48 | 15 $\times$ 6     | 20        | 4006     | 7.41 | 5886.209  | 6200.4   | 36.94 |
| 49 | 15 $\times$ 6     | 30        | 4110.5   | 6.54 | 6514.831  | 6187.2   | 40.67 |

从表1可以得出如下结论:

1) 大多数情况下,对于每一对车-门数量组合,随着临时存储区域宽放的增加,对偶间隙(%gap)会降低.因为临时存储区域总量增加后,对卡车入库的货物容量约束会降低,货物入库的条件得以放松,因此解的质量会有所提高;

2) 大多数情况下,当卡车数量不变,库门数量增加时,对偶间隙(%gap)会降低.因为此时,分配到每个门的卡车数量相对减少,卡车可停靠的库门资源相对增加,卡车的排序问题相对简化,因此解的质量也会有所提高;

3) 随着实验实例的规模增加,目标函数上界值(即对比贪婪算法的目标函数值)相较于目标函数近优解的偏移(%dev)越来越大.因此随着实验规模的增加,可行解数量的增多,拉格朗日松弛算法的优势逐渐显现,其解的质量相对于贪婪算法愈发优良;

4) 本实验所有实例的平均对偶间隙和平均CPU时间分别为5.74%和990.67 s.因此,拉格朗日松弛算法能够在合理的时间内得到较为满意的目标函数下界和近优可行解.

## 5 结论

针对以冷链物流为背景的越库调度问题,考虑到实际运输过程中货物的冷藏条件,提出了带有货物时间窗口约束的越库调度问题,并对该问题进行描述,以最小化库内运输成本和卡车排序等待时间为目标,建立数学模型.继而采用了拉格朗日松弛

算法,根据重要的决策变量将原问题分解为五个子问题,并通过次梯度算法进行模型的求解.实验分析了不同规模的越库调度实例,并得到相应结论.结果表明,拉格朗日松弛算法能够在合理的时间内得到较为满意的目标函数下界和近优可行解.后续研究将考虑如何进一步的提升解的质量,并提高算法运行速度.

## 参考文献:

- [1] NASSIEF W, CONTRERAS I, JAUMARD B. A comparison of formulations and relaxations for cross-dock door assignment problem. *Computers & Operations Research*, 2018, 94: 76 – 88.
- [2] YIN P Y, CHUANG Y L. Adaptive memory artificial bee colony algorithm for green vehicle routing with cross-docking. *Applied Mathematical Modelling*, 2016, 40(21/22): 9302 – 9315.
- [3] KARIMI H, SETAK M. A bi-objective incomplete hub location-routing problem with flow shipment scheduling. *Applied Mathematical Modelling*, 2018, 57: 406 – 431.
- [4] SERRANO C, DELORME X, DOLGUI A. Distribution and operation planning at a cross-dock platform: A case of study at Renault. *International Conference on Advanced Logistics & Transport*. [s.l.]: IEEE, 2015, 4: 193 – 198.
- [5] NASSIEF W, CONTRERAS I, AS'AD R. A mixed-integer programming formulation and Lagrangean relaxation for the cross-dock door assignment problem. *International Journal of Production Research*, 2015, 54(2): 1 – 15.
- [6] QIJUN Q, ZHANG Z, SONG X. Application research of cross docking logistics in food cold-chain logistics. *International Conference on Information Management*. Xi'an, China: IEEE, 2009: 236 – 240.
- [7] ZHOU Binghai, ZHONG Zhenyi. Lagrangian relaxation algorithm for scheduling problems of reentrant hybrid flow shops. *Control Theory & Applications*, 2015, 32(7): 881 – 886.  
(周炳海, 钟臻怡. 可重入混合流水车间调度的拉格朗日松弛算法. 控制理论与应用, 2015, 32(7): 881 – 886.)

- [8] WISITTIPANICH W, HENGMEETCHAI P. Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization. *Computers & Industrial Engineering*, 2017, 113: 793 – 802.
- [9] ASSADI M T, BAGHERI M. Differential evolution and population-based simulated annealing for truck scheduling problem in multiple door cross-docking systems. *Computers & Industrial Engineering*, 2016, 96(20): 149 – 161.
- [10] ZHANG Z H, LI B F, QIAN X. An integrated supply chain network design problem for bidirectional flows. *Expert Systems with Applications*, 2014, 41(9): 4298 – 4308.
- [11] SERRANO C, DELORME X, DOLGUI A. Scheduling of truck arrivals, truck departures and shop-floor operation in a cross-dock platform, based on trucks loading plans. *International Journal of Production Economics*, 2017, 194: 102 – 112.
- [12] EKSIOĞLU S D, EKSIOĞLU B, ROMEIJN H E. A Lagrangean heuristic for integrated production and transportation planning problems in a dynamic, multi-item, two-layer supply chain. *IIE Transactions*, 2007, 39(2): 191 – 201.

#### 作者简介:

**周炳海** 博士, 博士生导师, 主要从事离散系统建模、调度与仿真等方向的研究, E-mail: bhzhou@tongji.edu.cn;

**宗 师** 硕士研究生, 主要从事智能优化和生产系统建模、调度等方向的研究, E-mail: 18361228325@163.com.