

三维装载与CVRP联合多目标优化问题的模型及算法

王超^{1,2}, 金淳¹, 韩庆平³

(1. 大连理工大学 系统工程研究所, 辽宁 大连 116024; 2. 大连交通大学 软件学院, 辽宁 大连 116052; 3. 美国佛罗里达大西洋大学 信息技术及运作管理系, 波卡拉顿 FL 33431)

摘要: 提出三维装载与CVRP联合多目标优化问题(3LCVRPMO)模型, 该模型在三维装载约束下的CVRP问题(3LCVRP)的基础上, 考虑了配送车辆数目及路径总距离两个目标函数. 在权衡装箱和路径优化两个优化过程的基础上, 构建了多阶段/两层混合算法架构(MSOTLH)及其算法, 并对路径优化偏好的3LCVRPMO问题进行求解. 基于3LCVRP问题相关算例的数据实验结果表明, 所提出的3LCVRPMO模型及MSOTLH算法是有效的.

关键词: 车辆路径问题; 三维装箱问题; 联合优化问题; 多目标优化; 启发式算法

中图分类号: O224

文献标志码: A

Model and algorithm for multi-objective joint optimization of three-dimensional loading and CVRP

WANG Chao^{1,2}, JIN Chun¹, HAN Jim C³

(1. Institute of Systems Engineering, Dalian University of Technology, Dalian 116024, China; 2. School of Software, Dalian Jiaotong University, Dalian 116052, China; 3. Department of Information Technology and Operations Management, Florida Atlantic University, Boca Raton FL 33431, USA. Correspondent: JIN Chun, E-mail: jinchun@dlut.edu.cn)

Abstract: A multi-objective joint optimization problem combining the three-dimensional loading problem with the capacitated vehicle routing problem(3LCVRPMO) is proposed. In 3LCVRPMO, two targets including the minimum number of vehicles and the shortest distance of path are considered. An architecture of multi-stage or two layers hybrid algorithm(MSOTLH) is established to deal with path optimization preference of 3LCVRPMO based on trading off packing and path optimization. In numerical experiments, the relevant instances of the 3LCVRP are referenced, and the results show the effectiveness of the proposed 3LCVRPMO model and MSOTLH algorithm.

Keywords: capacitated vehicle routing problem; three-dimensional container loading problem; joint optimization problem; multi-objective optimization; heuristic algorithm

0 引言

将装箱问题与CVRP问题相结合的考虑三维装载约束下的CVRP问题(3LCVRP)^[1]是近年来研究的新热点问题. 3LCVRP的难点在于它涉及CVRP及三维装箱两个NP难问题. 关于CVRP问题, 往往采用启发式算法求解, 如节约算法、插入法、扫描算法、禁忌搜索算法、遗传算法、模拟退火法、蚁群算法等. 与3LCVRP相关的三维装箱问题包括容器装载问题(3D-CLP)^[1-2]、箱柜装载问题(3D-BPP)^[2-3]、条带装载问题(3D-SPP)^[4-6]等. 其中: 3D-CLP是将一系列长方体物品装载到长方体容器中, 使空间利用率最大; 3D-

SPP是依据容器的垂直截面将系列长方体物品装入, 使得装入容器的装载长度最小. 关于三维装箱问题的求解, 人们提出了许多方法, 其中包括文献[7-8]提出的分枝定界算法, George等^[9]基于“层”的启发式算法, Pisinger等^[4]通过拓展分层的条带进行求解的启发式算法, Lim等^[10]的切割算法, 何琨等^[11]的按最大穴度装载算法, Bortfeldt等^[12]的禁忌搜索算法, Gehring等^[13]的遗传算法和张德富等^[14]的混合模拟退火的元启发算法等. 文献[3, 5, 15]对3LCVRP联合优化问题展开了相关研究和探索, 但总体而言, 关于三维装载与CVRP联合优化问题的研究尚处于起步阶段.

收稿日期: 2015-02-03; 修回日期: 2015-10-27.

基金项目: 国家自然科学基金项目(71271041).

作者简介: 王超(1974-), 女, 博士生, 从事物流管理及算法分析与设计的研究; 金淳(1963-), 男, 教授, 博士生导师, 从事物流与供应链管理、系统仿真等研究.

目前, 3LCVRP问题大多以路径优化为单一目标, 即仅考虑了路径距离与运输成本的正相关性, 而实际上该问题与两个优化过程密切相关: 装箱优化与路径优化, 其对应的目标分别为车辆数目最小(与装车的空间效率相关)和车辆路径最短(与CVRP问题相关), 此外, 还应当考虑装箱优化结果对路径优化结果的影响. 在实际物流作业中, 配送车辆数目也对运输成本有一定的影响. 为此, 本文提出三维装载与CVRP联合问题的多目标优化模型(3LCVRPMO), 并构建其多阶段/两层混合算法, 试图为更有效地解决此类联合优化问题提出新思路.

1 3LCVRPMO的描述及数学模型

1.1 问题的描述

本文3LCVRPMO问题可以描述为: 给定一个图 $G(V, E)$. 其中: $V = \{v_i | i = 0, 1, \dots, n\}$ 是一个配送中心及 n 个客户的顶点集合, 客户的顶点集合 $V_D = V/\{v_0\}$; $E = \{\overrightarrow{v_i v_j} | i, j = 0, 1, \dots, n \text{ 且 } i \neq j\}$ 是各顶点间所有边的集合, 顶点 v_i 与 v_j 之间的距离用 c_{ij} 表示. 需求集合定义为四元组 $D = \{d_i | i = 0, 1, \dots, n\}$, $d_i = \{(l_{ij}, w_{ij}, h_{ij}, r_{ij}) | j = 1, 2, \dots, b_i\}$ 是第 i 个客户的需求集合, 包括货物的长、宽、高及重量信息, b_i 是第 i 个客户的货箱数量, 其总体积、总重量表示为 $\left\{ \sum_{j=1}^{b_i} l_{ij} w_{ij} h_{ij}, \sum_{j=1}^{b_i} r_{ij} \right\} = \{A_i, r_i\}$, 配送中心的需求 $d_0 = 0$. 设配送车辆型号统一, 车厢内尺寸的长、宽、高分别对应 L_0, W_0, H_0 , 最大载重量为 R , 配送中心派出车辆的动态集合为 $C = \{1, 2, \dots, K\}$. 对于每一辆车, 其装载的货物重量必须满足货车的载重量约束, 货物摆放须满足空间约束. 车辆从配送中心出发, 完成对所载货物的服务后, 返回配送中心.

关于装载规则, 若某个需求 d_i 能被一个车厢容纳, 则不拆分需求; 否则, 按整车专运装载直至 d_i 的非整车剩余部分, 更新 d_i , 将其转化为不拆分需求情况.

决策变量 x_{ij}^k, p_{ik}^r 和 p_{ik}^A 定义如下:

$$x_{ij}^k = \begin{cases} 1, & \text{车辆 } k \text{ 从点 } i \text{ 行驶至 } j; \\ 0, & \text{其他.} \end{cases}$$

$$p_{ik}^r = \begin{cases} p, & \text{车辆 } k \text{ 装载 } d_i \text{ 的重量与 } r_i \text{ 之比;} \\ 0, & \text{车辆 } k \text{ 未装载点 } i \text{ 的货物.} \end{cases}$$

$$p_{ik}^A = \begin{cases} p', & \text{车辆 } k \text{ 装载 } d_i \text{ 的体积与 } A_i \text{ 之比;} \\ 0, & \text{车辆 } k \text{ 未装载点 } i \text{ 的货物.} \end{cases}$$

其中: $0 < p, p' \leq 1$. 以车厢的左、后、下方顶点为原点建立三维直角坐标系, 位置变量 $xx_{ij}^k, yy_{ij}^k, zz_{ij}^k$ 分别表示第 i 个客户的第 j 个货箱的左、后、下方顶点在第 k 辆车车厢中 X, Y, Z 方向的坐标值, 并设与该顶点相连的体对角线的另一个顶点坐标为 $(txx_{ij}^k,$

$tyy_{ij}^k, tzz_{ij}^k)$. 货箱在车厢中按其长、宽、高 $\{l_{ij}, w_{ij}, h_{ij}\}$ 三个元素排列, 有6种正交摆放方式, 用方向数 $\text{ort} = 1, 2, \dots, 6$ 表示. 平行于 X 轴的边称为相对宽, 用 $\psi_X(l_{ij}, w_{ij}, h_{ij}, \text{ort})$ 表示; 相应地, 平行于 Y, Z 轴的边分别称为相对高 $\psi_Y(l_{ij}, w_{ij}, h_{ij}, \text{ort})$ 和相对长 $\psi_Z(l_{ij}, w_{ij}, h_{ij}, \text{ort})$. 车厢的三维直角坐标系如图1所示, 设长方体 A 此时摆放方式 $\text{ort} = 1$, 相对宽 $\psi_X(l_{ij}, w_{ij}, h_{ij}, \text{ort}) = \psi_X(l_{ij}, w_{ij}, h_{ij}, 1) = l_{ij}$.

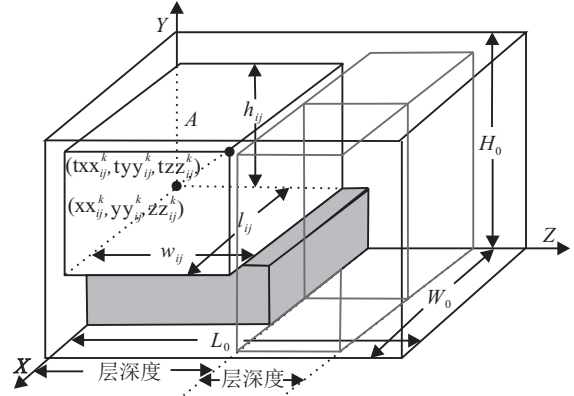


图1 车厢的三维直角坐标系

3LCVRPMO问题的两个目标是: 最小化路径距离, 即所有路线的总距离; 最小化配送中心派出的车辆数目.

1.2 优化模型的建立

3LCVRPMO的数学模型具体表示如下:

$$\text{obj. } \min P = \sum_{k \in C} \sum_{j \in V} \sum_{i \in V} c_{ij} x_{ij}^k, \quad i \neq j; \quad (1)$$

$$\min K = \sum_{k \in C} \sum_{j \in V_D} x_{0j}^k. \quad (2)$$

$$\text{s.t. } \sum_{k \in C} \sum_{i \in V_D} x_{0i}^k = \sum_{k \in C} \sum_{j \in V_D} x_{j0}^k = K; \quad (3)$$

$$\sum_{i \in V} x_{ih}^k - \sum_{j \in V} x_{hj}^k = 0, \quad h \in V_D, k \in C; \quad (4)$$

$$\sum_{i \in B} \sum_{j \in B} x_{ij}^k \leq |B| - 1, \quad B \subseteq V_D, |B| \geq 2, \quad (5)$$

$$\sum_{i \in V_D} d_i p_{ik}^r \leq R, \quad k \in C; \quad (6)$$

$$\sum_{k \in C} p_{ik}^r = 1, \quad i \in V_D; \quad (7)$$

$$\sum_{k \in C} p_{ik}^A = 1, \quad i \in V_D; \quad (8)$$

$$txx_{ij}^k = xx_{ij}^k + \psi_X(l_{ij}, w_{ij}, h_{ij}, \text{ort}),$$

$$tyy_{ij}^k = yy_{ij}^k + \psi_Y(l_{ij}, w_{ij}, h_{ij}, \text{ort}),$$

$$tzz_{ij}^k = zz_{ij}^k + \psi_Z(l_{ij}, w_{ij}, h_{ij}, \text{ort}),$$

$$i \in V_D, j \in \{1, 2, \dots, b_i\}, k \in C; \quad (9)$$

$$txx_{ij}^k < W_0, tyy_{ij}^k < H_0, tzz_{ij}^k < L_0, \quad i \in V_D,$$

$$j \in \{1, 2, \dots, b_i\}, k \in C; \quad (10)$$

$$xx_{ij}^k \geq txx_{lt}^k \vee xx_{ij}^k \geq txx_{lt}^k \vee yy_{ij}^k \geq tyy_{lt}^k \vee yy_{ij}^k \geq tyy_{lt}^k \geq$$

$$tyy_{ij}^k \vee zz_{ij}^k \geq tzz_{lt}^k \vee zz_{ij}^k \geq tzz_{lt}^k, k \in C, i, l \in V_D,$$

$$j \in \{1, 2, \dots, b_i\}, t \in \{1, 2, \dots, b_l\}; \quad (11)$$

$$\min\{zz_{ij}^k\}x_{il}^k \geq \max\{tzz_{lt}^k\}x_{il}^k, k \in C, i, l \in V_D,$$

$$j \in \{1, 2, \dots, b_i\}, t \in \{1, 2, \dots, b_l\}. \quad (12)$$

其中: 式(1)为目标函数1, 即最小化各车辆路线的总距离 P ; 式(2)为目标函数2, 即最小化派出的配送车辆数 $K = |C|$. 当目标函数有字典序(1) < (2)时, 优先进行路径优化, 同时求解相对最小配送车辆数目, 此时称之为路径优化偏好; 同理, 当有字典序(1) > (2)时, 优先最小化配送车辆数 K , 然后进行路径优化, 称之为车辆数优化偏好. 在实际应用中, 可视资源及需求等具体情况选择不同偏好. 式(3)~(12)是约束条件: 式(3)表示配送中心派出 K 辆车, 并最终返回出发点; 式(4)是连通性约束, 对于每一个客户顶点, 到访服务的车辆必须驶离; 式(5)是排除子回路约束; 式(6)是车辆承重约束, 即每辆车所装载各客户货物的重量之和不能超过车辆的最大载重量; 式(7)和(8)分别表示各客户需求的总重量及总体积都被装载; 式(9)是正交放置约束; 式(10)约束货物都放置在车厢内部, 不能超出车厢的范围; 式(11)表示车厢中的任意两个货箱不能重叠放置; 式(12)是车辆路径问题(VRP)装车时要遵从的后进先出(LIFO)原则.

2 多阶段/两层混合算法架构 (MSOTLH)

目前国内外既有研究中对于 3LCVRP 的求解大致有两种思路: 一是在进行路径优化的同时检验三维装载约束是否满足^[3]; 二是将求解过程分为清晰的几个阶段, 如先进行三维装载求解, 再进行路径优化两个独立的阶段^[5]. 称思路一为两层优化, 思路二为多阶段优化. 由于 3LCVRPMO 问题比 3LCVRP 更复杂, 本文仅针对 3LCVRPMO 问题中的路径优化偏好目标进行求解, 权衡装箱与路径优化两个优化过程, 融合这两种求解思路的长处, 构建多阶段/两层混合算法架构 (MSOTLH) 及相关算法, 具体介绍如下.

2.1 MSOTLH 的基本框架

MSOTLH 的基本框架如图 2 所示, 由各客户独立装箱的条带装箱问题 (SPP) 计算、综合评估判断、多阶段算法及两层算法 4 部分组成.

2.2 MSOTLH 的基本流程

第 1 部分: 各客户独立 SPP 计算. 分别将每一个 d_i 按重量及三维约束独立进行 3D-SPP 计算, 每一个客户都会生成一个装载长度 $\Pi(i)$, 记录相应的条带装载率 $\varepsilon(i)$, 计算如下:

$$\varepsilon(i) = \sum_{j \in d_i} w_{ij} h_{ij} l_{ij} / W_0 H_0 \Pi(i). \quad (13)$$

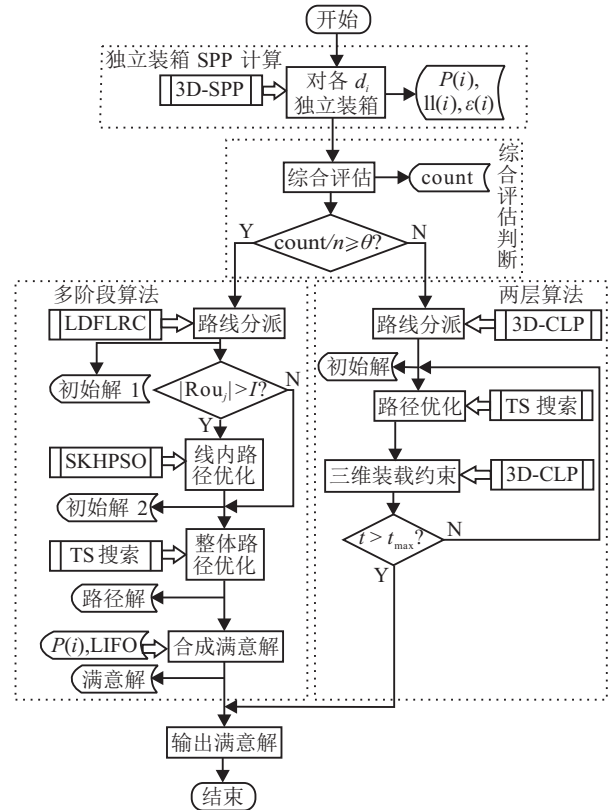


图 2 MSOTLH 算法架构

第 2 部分: 综合评估判断. 判断每一客户的 $\varepsilon(i)$ 是否不小于条带装载率阈值 ε_0 , 并统计满足该条件的客户数量 $count$, 当 $count$ 与客户总数 n 的比值不小于给定常数 θ 时, 算法进入第 3 部分, 即多阶段算法; 否则, 调用第 4 部分, 即两层算法.

第 3 部分: 多阶段算法 (MA). 对于所有装载长度 $\Pi(i)$, 在重量及长度约束下, 用装箱问题 (BPP) 算法进行路线分派, 得到最小配送车辆数及初始解, 再进行路线内及路线间的路径优化, 得出路径优化方案, 并结合 LIFO 原则得到最终的三维装载方案及满意解.

第 4 部分: 两层算法 (TLA). 对于所有 d_i , 以车厢为容器进行依次三维装箱计算, 得到初始解, 松弛 LIFO 原则, 用禁忌搜索在重量和三维装载约束下进行路径优化, 得到最终的路径优化与装载方案.

3 MSOTLH 算法的实现

3.1 各客户独立 SPP 计算

在该阶段中, 每一个 d_i 都被定义成一个相应的 3D-SPP 实例单独进行计算. 本节装箱过程仅考虑两种情况: 同一类型和不同类型.

1) 同一类型: 当某 d_i 是同一规格的货箱时, 分别将货箱的长、宽、高三个维度尺寸设置成层的深度, 去寻找能装载 d_i 的最小装载长度 $\Pi(i)$, 即

$$\Gamma = \left\lfloor \frac{W_0}{\psi_X(l, w, h, \text{ort})} \right\rfloor \times \left\lfloor \frac{H_0}{\psi_Y(l, w, h, \text{ort})} \right\rfloor, \quad (14)$$

$$\Pi(i) = \min \left\{ \left\lfloor \frac{b_i}{\max\{\Gamma\}} \right\rfloor \times \psi_Z(l, w, h, \text{ort}) \right\}. \quad (15)$$

2) 不同类型: 如果 d_i 的货箱类型多于一种, 则对于相应 3D-SPP 实例调用一系列容器长度 L 递减的 3D-CLP 求解过程, 当 L 对应的装载空间容纳不下 d_i 时, L 不再被减小^[5]. 存储装箱计划 $IP(i)$ 、对应总重量 $r(i)$ 、最小装载长度 $\ell(i)$ 及相应的 $\varepsilon(i)$, 具体算法描述如下:

Step 1: 给定容器宽、高和长度分别为 W_0, H_0, L_0 , 对当前的 d_i 进行 3D-CLP 问题求解;

Step 2: 按给定步长参数 ΔL 减小装载空间的长度, $L = L - \Delta L$;

Step 3: 对当前 d_i 以及宽、高、长为 (W_0, H_0, L) 的装载空间进行 3D-CLP 求解, 如果 L 对应的装载空间可以容纳 d_i , 则转向 Step 2, 否则进行 Step 4;

Step 4: 存储最后一次完全装载 d_i 的装箱方案 $IP(i)$ 、最小装载长度 $\ell(i)$ 、重量 $r(i)$, 计算相应的 $\varepsilon(i)$.

关于 3D-CLP 问题的求解, 本文在 Pisinger^[4] 的相关算法基础上, 加入了重力约束及层深度修正线, 在每装入一个箱子时记录其 tzz_{ij} , 该层中 $\max\{tzz_{ij}\} - \min\{zz_{ij}\}$ 为实际层深度, 可参见图 1, 该方法使得三维装载过程更加紧凑.

3.2 综合评估判断

对独立装箱 SPP 的结果进行分析统计. 该阶段算法描述如下:

```
count = 0.
for (i = 1; i ≤ n; i++)
  if(ε(i) ≥ ε0) count++
  if(count/n ≥ θ)
    call MA;
  else
    call TLA.
```

在混合算法架构下, 当 $\varepsilon(i) \geq \varepsilon_0$ 时, 认为 $\varepsilon(i)$ 较好, θ 是 $\varepsilon(i)$ 较好的客户数比例. $\varepsilon(i)$ 较好说明 d_i 独立占用长度为 $\ell(i)$ 的车厢空间的装载效率较好, 在全部 $\varepsilon(i)$ 都较好的情况下, 这种装载就近似于重量约束下对 $\ell(i)$ 进行装载. 但这种情况过于理想, 在通常情况下, 阈值参数 ε_0 与 θ 的取值范围为 $[0.6, 0.8]$. 当 $\varepsilon(i) < \varepsilon_0$ 的客户数达到一定比例时, 每一客户需要整块地占用某段车厢的这种装载方式效率较差, 会导致配送车辆数增加, 也极有可能使路径距离变差.

3.3 多阶段算法(MA)

MA 是 MSOTLH 的第 3 部分, 即当大部分客户的独立条带装载率较高时, 每个客户的货箱可以独立装入车厢中的某一段, 这时可以将 3LCVRPMO 问题转化为 LCVRP 问题^[16]. MA 求解 LCVRP 的步骤如下:

Step 1: 路线分派. 对集合 $\{\ell(i) | i \in V_D, \ell(i) <$

$L_0\}$ 进行 BPP 问题的求解.

Step 2: 路线内路径优化. 对于客户数目大于给定门槛值 I 的路线, 求解旅行商问题 (TSP).

Step 3: 整体路径优化. 采用禁忌搜索算法 (TS) 优化以上步骤得出的初始解, 并存储路径优化结果 X , 其中包括各路线的向量 $X(j), j = 1, 2, \dots, K$.

Step 4: 合成最终满意解. 对于每一路线 j , 结合 $X(j)$ 及其各客户的 $IP(i)$, 以 LIFO 原则进行装车, 并存储该车辆的装车方案 $IP'(j)$.

上述 4 个步骤的具体求解过程如下文所述.

3.3.1 路线分派

经过独立装箱阶段, d_i 获得了相应装载最小长度 $\ell(i)$ 及总重量 $r(i)$. 路线分派就是对集合 $\{\{\ell(i), r(i) | i \in V_D\}$ 进行 BPP 求解. 本文采用的方法是在最大需求适宜最大装载空间 (LDFLRC) 算法^[16] 的基础上加入重量约束, 算法调整如下.

Step 1: 初始化配送车辆数目

$$K = \max \left\{ \sum_{i \in V_D} r(i)/R, \sum_{i \in V_D} \ell(i)/L_0 \right\}. \quad (16)$$

每辆车可装载重量 $RC(i) = R$; 装载长度 $q_i = L_0$; 各路线的客户集合 $Rou_i = \emptyset, i = 1, 2, \dots, K$.

Step 2: 对集合 $\{\{\ell(i), r(i) | i \in V_D\}$ 按 $\ell(i) \prec r(i)$ 的字典序, 按 $\ell(i)$ 的非升序及 $r(i)$ 的非降序排列, 结果记为

$$D' = \{d'_{1M_1}, d'_{2M_2}, \dots, d'_{iM_i}, \dots, d'_{nM_n} | d'_{iM_i} = (\ell(M_i), r(M_i)), M_i \in V_D, i = 1, 2, \dots, n\}. \quad (17)$$

Step 3: 从 D' 中依次取 $\ell(M_i), D' = D' - \{d'_{iM_i}\}$, 并找到当前最大装载长度 $q_\theta = \max\{q_i\}$ 的车辆 θ . 如果 $q_\theta > \ell(M_i)$ 并且 $RC(\theta) \geq r(M_i)$, 则 $Rou_\theta = Rou_\theta + \{M_i\}, q_\theta = q_\theta - \ell(M_i), RC(\theta) = RC(\theta) - r(M_i)$; 否则, 添一辆空车将其装入, $K = K + 1, Rou_K = M_i, q_K = L_0 - \ell(M_i), RC(K) = R - r(M_i)$.

Step 4: 重复执行 Step 3 直至 $D' = \emptyset$.

Step 5: 输出各路线的客户集合 $Rou_j, j = 1, 2, \dots, K$.

3.3.2 路线内路径优化

当某一路线的客户数 $|Rou_j| > I$ (设定的门槛值) 时, 对其进行路线内路径优化, 即路线内各客户配送顺序的排序问题. 算法描述如下:

```
for(j = 1; j ≤ K; j++)
```

```
  if(|Rouj| > I)
```

```
    将 Rouj 与 v0 涵盖的子图 G' 作为输入,
    调用基于类 Kruskal 的混合粒子群算法
    (SKHPSO)[17] 求解 TSP.
```

3.3.3 整体路径优化

MA 中禁忌搜索(TS)算法进一步优化的基本步骤如下。

Step 1: 初始化禁忌表 $U\{\} = \emptyset, t = 0$;

Step 2: 将 MA 前两阶段生成的初始解作为 TS 的初始解 X ;

Step 3: 当 $t > t_{\max}$ 时, 算法结束, 输出 X , 包括各路线的优化序列 $X(j) (j = 1, 2, \dots, K)$, 否则进行 Step 4;

Step 4: 对当前解 X 用置换、插入、2-Opt 等算子生成 X 的有效邻域移动集 $S(X)$;

Step 5: 选择一个最好的邻域移动 $s (s \in S \wedge s \notin U)$, 从而得到一个新解 X' , 且 $X \leftarrow X'$;

Step 6: s 入栈 $U\{\}$, $U\{0\}$ 出栈, $t \leftarrow t + 1$, 转向 Step 3.

3.3.4 合成最终满意解

将路径优化结果 X 与各客户的装箱方案 $IP(i)$ 按照 LIFO 规则结合起来, 便得到 MA 的最终满意解, 如第 j 条路线, 按其路线优化序列 $X(j)$ 的逆序逐块装载, 直至最后将其第 1 个访问客户的需求集整块装入。

3.4 两层算法 (TLA)

TLA 是 MSOTLH 的第 4 部分, 当 $\varepsilon(i) < \varepsilon_0$ 的客户数目多于 $n(1 - \theta)$ 时, 松弛 LIFO 约束, 提高满载率, 从而提高优化效率. TLA 求解步骤如下。

Step 1: 用 3D-CLP 求解算法对所有 d_i 进行依次装载计算, 当某 d_i 不能被完全载入当前车辆时, 加一辆空车, 完整装入此 d_i , 接着对未装载的 d_i 依次装载, 直至所有的 d_i 装载完毕, 生成初始解;

Step 2: 对初始解进行 TS 搜索, 进一步优化计算, 基本与 3.3.3 节的 Step 3~Step 6 相同, 但在 TLA 中, 当 TS 生成邻域移动集时, 不仅要检验载重量约束, 还要调用 3D-CLP 算法检验三维装载约束, 当两个约束条件都满足时, 生成的邻域解才是可行解。

4 数据实验及结果分析

由于目前没有针对 3LCVRPMO 的标准测试算例, 本文参考了与 3LCVRP 问题相关的 GI 系列及 3LCVRP 两组标准算例, 并基于该算例对路径优化偏好的 3LCVRPMO 问题及 MSOTLH 架构和具体算法进行测试. 机器型号为 Lenovo 系列 ThinkPad, CPU 主频 2.5 GHz, 内存 4 G. 在 GI 系列算例中, 每个客户的货箱数较多, 但货箱尺寸类型较少^[5]; 而在 3LCVRP 算例中, 每个客户的货箱数较少, 但其尺寸类型各不相同^[3]. 考虑到整体车辆的满载率, 本文取 $\varepsilon_0 = 0.8$, $\theta = 0.7$, 主要测试结果如下。

4.1 GI 系列算例测试

对于 GI 系列所有算例, 点的坐标、货箱尺寸及数

量是相同的. 对其 25 个客户的需求进行独立 SPP 计算, 其中第 23 个客户的装载结果为: $ll(23) = 876$, 共分 12 层, 层布局如图 3 所示。

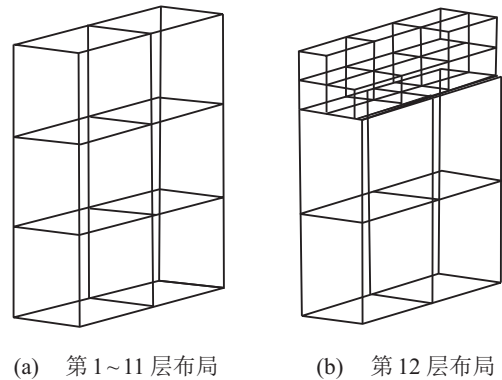


图 3 IP(23) 的装载层布局

在图 3 中: 第 1~11 层的布局相同. 在综合评估阶段, 由于 72% 客户的 $\varepsilon(i) \geq \varepsilon_0$, MSOTLH 调用了 MA. 路径优化的结果为 $P = 442.271$, $K = 5$, 具体路径如图 4 所示。

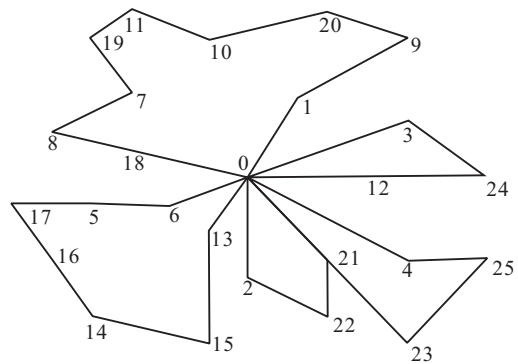


图 4 GI 系列算例的路径优化

在 MA 的求解结果中, 每一客户的所有货箱被整块地装载在某个车厢的某一段中, 装车严格遵循了 LIFO 原则。

4.2 3LCVRP 算例测试

对 3LCVRP 的 27 个算例进行测试, MSOTLH 调用了 TLA, 主要结果如表 1 所示。

在表 1 中: K 表示车辆数, P 表示满意解的距离, $3D_P$ 表示仅三维装载约束下的满意解距离, dv 表示 TLA 算法得出的 P 与 $3D_P$ 列结果的偏差. 由表 1 可知: 1) 目标函数 1: TLA 算法解得的 P 在与文献 [1] 和文献 [3] 中相比较, 尤其与同条件的 $3D_P$ 比较时, 得到了比较满意的结果. 本文在客户数小于 40 的 9 个算例中找到了更短的满意解距离; 而在大于 40 个客户的算例中, 除了第 25 个算例之外, 其余的 dv 均小于 10%. 2) 目标函数 2: 设 $3D_P$ 列对应的 K 值与算例中给出的值一致, 将其与 TLA 算法的结果进行比较发现, 在多数算例中的 K 值有差别, 并且在部分算例, 如算例 12 和算例 27 中, 配送车辆数和路径距离两个目标的冲突已显现。

表1 3LCVRP算例的测试结果

算例名与 客户数	K	文献[1]		VRLH1 ^[3]		MAOTLH调用 TLA		
		P	P	3D_P	P	K	dv /%	
1(15)	4	316.32	302.02	297.65	291.2	4		
2(15)	5	350.58	334.96	334.96	326.69	5		
3(20)	4	447.73	392.63	381.36	376.74	4		
4(20)	6	448.48	437.19	430.89	430.89	5	0	
5(21)	6	464.24	443.61	397.16	418.72	5	5.43	
6(21)	6	504.46	498.16	498.07	500.49	6	0.49	
7(22)	6	831.66	769.68	741.80	735.28	4		
8(22)	6	871.77	810.89	735.14	768.93	5	4.6	
9(25)	8	660.10	630.13	631.82	628.67	9		
10(29)	8	911.16	820.35	739.94	751.14	6	1.59	
11(29)	8	819.36	803.61	723.44	758.09	6	4.79	
12(30)	9	651.58	614.59	623.10	605.63	10		
13(32)	8	2928.34	2645.95	2348.48	2390.96	5	1.81	
14(32)	9	1559.64	1368.42	1234.54	1199.01	6		
15(32)	9	1452.34	1341.14	1202.34	1230.84	6	2.37	
16(35)	11	707.85	698.61	704.47	707.84	12	0.48	
17(40)	14	920.87	866.40	928.93	887.12	15		
18(44)	11	1400.52	1207.72	1108.37	1137.36	8	2.62	
19(50)	12	871.29	741.74	678.59	732.08	9	7.88	
20(71)	18	732.12	587.95	520.55	552.77	13	6.19	
21(75)	17	1275.2	1090.22	964.66	1057.66	13	9.64	
22(75)	18	1277.94	1147.80	1041.92	1143.69	14	9.77	
23(75)	17	1258.16	1130.54	995.22	1076.05	14	8.12	
24(75)	16	1307.09	1116.13	1053.41	1129	15	7.18	
25(100)	22	1570.72	1407.36	1238.83	1405.31	17	13.4	
26(100)	26	1847.95	1600.35	1444.58	1506.83	20	4.31	
27(100)	23	1747.52	1529.86	1342.23	1450.41	19	8.06	

5 结论

本文提出了3LCVRPMO问题,在权衡装箱和路径优化两个优化过程的基础上,构建了MSOTLH,并由此设计了针对路径优化偏好的3LCVRPMO问题的启发式算法.数据实验表明:1)当多数客户的条带装载率较高时,多阶段算法可以得到较满意解;在其余情况下,用两层优化算法在一定程度下松弛LIFO约束可以得到较满意解.2)3LCVRP算例的测试结果呈现了配送车辆数的差别及配送车辆数与路径总距离的矛盾.

本文研究也为有效求解3LCVRP联合优化问题提供了新思路.

参考文献(References)

- [1] Gendreau M, Iori M, Laporte G, et al. A tabu search algorithm for a routing and container loading problem[J]. *Transportation Science*, 2006, 40(3): 342-350.
- [2] 张德富,魏丽军,陈青山,等.三装箱问题的组合启发式[J]. *软件学报*, 2007, 18(9): 2083-2089.
(Zhang D F, Wei L J, Chen Q S, et al. A combinational heuristic algorithm for the three-dimensional packing problem[J]. *J of Software*, 2007, 18(9): 2083-2089.)
- [3] Bortfeldt A. A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints[J]. *Computers & Operations Research*, 2012, 39(9): 2248-2257.

- [4] Pisinger D. Heuristics for the container loading problem[J]. *European J of Operational Research*, 2002, 141(2): 383-392.
- [5] Bortfeldt A, Homberger J. Packing fist, routing second-a heuristic for the vehicle routing and loading problem[J]. *Computers & Operations Research*, 2013, 40(3): 873-885.
- [6] Bortfeldt A, Mack D. A heuristic for the three-dimensional strip packing problem[J]. *European J of Operational Research*, 2007, 183(3): 1267-1279.
- [7] Martello S, Pisinger D, Vigo D. The three-dimensional bin packing problem[J]. *Operations Research*, 2000, 48(2): 256-267.
- [8] Fekete S P, Schepers J. An exact algorithm for higher dimensional orthogonal packing[J]. *Operations Research*, 2007, 55(3): 569-587.
- [9] George J A, Robinson D F. A heuristic for packing boxes into a container[J]. *Computers and Operations Research*, 1980, 7(3): 147-156.
- [10] Lim A, Rodrigues B, Yang Y. 3-D container packing heuristics[J]. *Applied Intelligence*, 2005, 22(2): 125-134.
- [11] 何琨,黄文奇.求解长方体Packing问题的捆绑穴度算法[J]. *软件学报*, 2011, 22(5): 843-851.
(He K, Huang W Q. Cuboid arrangement approach based on caving degree for solving the cuboid packing problem[J]. *J of Software*, 2011, 22(5): 843-851.)
- [12] Bortfeldt A, Gehring H. Applying tabu search to container loading problems[C]. *Operations Research Proc. Berlin: Springer*, 1998: 533-538.
- [13] Gehring H, Bortfeldt A. A parallel genetic algorithm for solving the container loading problem[J]. *Int Trans in Operational Research*, 2002, 9(4): 497-511.
- [14] 张德富,彭煜,朱文兴,等.求解三维装箱问题的混合模拟退火算法[J]. *计算机学报*, 2009, 32(11): 2147-2156.
(Zhang D F, Peng Y, Zhu W X, et al. A hybrid simulated annealing algorithm for the three-dimensional packing problem[J]. *Chinese J of Computers*, 2009, 32(11): 2147-2156.)
- [15] Fuellerer G, Doerner K F, Hartl R F, et al. Metaheuristics for vehicle routing problems with three-dimensional loading constraints[J]. *European J of Operational Research*, 2010, 201(3): 751-759.
- [16] Wang C, Jin C, Han J. A multistage algorithm for multi-objective joint optimization of loading problem and capacity vehicle routing problem[J]. *ICIC Express Letters Part B: Applications*, 2014, 5(5): 1453-1459.
- [17] 王超,金淳,韩庆平.求解旅行商问题的基于类Kruskal的混合粒子群算法[J]. *运筹与管理*, 2014, 23(3): 30-37.
(Wang C, Jin C, Han J C. Similar kruskal-based hybrid particle swarm optimization algorithm for traveling salesman problem[J]. *Operations Research and Management Science*, 2014, 23(3): 30-37.)

(责任编辑: 闫 妍)