

# 摆放约束的二维装箱研究

周雨菁, 陈国卫

(海军工程大学 管理工程与装备经济系, 武汉 430000)

**摘要:**如何实现装载率更高的装载方式、提高容器装载空间的利用率,从而提高服务水平、降低运营成本,对于企业提高竞争力具有十分重要的意义。二维装箱有一种“最左最低对齐”的经典解法,其基本原理是将物品按宽度进行降序排列,再把物品装载至目前能容纳其体积的尽量低、尽量左的位置,但实际操作中,为便于查看货物信息,摆放货物时需要货物至少一面朝外。考虑上述摆放约束,在“最左最低对齐”算法的基础上,提出一种二维装箱问题优化解法,并结合实际算例进行分析。

**关键词:**二维装箱问题;摆放约束;启发式算法

中图分类号:TB497

文献标识码:A

文章编号:1672-545X(2020)07-0137-04

## 0 引言

在货物流通过程中,普遍存在将货物装载至某种容器的过程,而采用何种方式装载、如何提高装载率,被称作装箱问题,早在 1970 年装箱问题已被证明是 NP 完全问题,无法在有效时间内求得最优解。在面对规模较小的装箱问题时,可以通过如线性规划法等精确算法求得最优解。吉尔莫和格莫瑞通过线性规划方法求得了一个实际问题最优解,克里斯托菲德斯和惠特洛克使用树搜索方法解决了二维下料装箱问题并达到最优,比斯利使用树搜索法解决了二维非下料装箱问题也同样达到最优<sup>[1-3]</sup>。但面对数据规模较大的装箱问题时,精确算法就难以在有效时间内得到最优解。随着学术成果的创新,启发式算法应用于数据规模较大的装箱问题上已经能在有效时间内得到较优解。贝克和科夫曼等人提出了一种十分经典的装箱算法,即最低最左对齐算法,首先根据待装载货物的尺寸由宽至窄排序,再将待装载货物放置在能容纳其体积的尽量低、尽量左的位置。此种算法时间复杂度为  $O(n^2)$ <sup>[4]</sup>。查泽尔提出了最低最左左填充算法,此算法为最低最左算法的修改版,在最低最左算法的基础上,再将待装载货物放入与其最为匹配的位置,此种算法时间复杂度为  $O(n^3)$ <sup>[5]</sup>。李昆鹏等人提出了更切合实际的装箱问题,即相同供应商或客户的货物需要装载至同一容器,否则会造成损失<sup>[6]</sup>。本文考虑实际案例 A 公司的货物

摆放约束,提出了一种新的装箱问题,即所有零件必须有一面朝外以及只允许同类周转箱叠放,在此约束下进行研究。

## 1 问题模型构建与分析

本文以 A 汽车制造公司为例,研究考虑货物摆放约束的二维装箱问题。A 公司是行业内将准时生产制应用得较好的汽车制造企业,依据物料需求计划进行零部件的订购,其长期合作的供应商遍布全国,甚至部分核心零部件,如发动机、变速箱等更是由海外供应商供应。A 公司在某省会城市有两个整车厂并建有中心仓库,在全国范围内建有八个分库,每个分库负责其一定半径内供应商货物的运输、暂存和周转。半径范围内的零部件供应商将零部件配送至分库后,再由分库统一配送至中心仓库,而半径范围外的零部件供应商则直接将零部件运输至中心仓库或生产流水线边的暂存仓库。A 公司的物流配送网络见图 1。

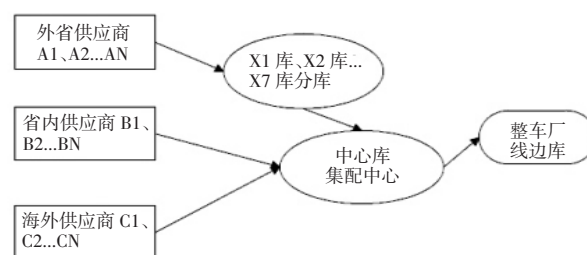


图 1 A 公司配送网络图

收稿日期:2020-04-07

作者简介:周雨菁(1993-),女,湖北武汉人,硕士研究生,助理实验师,研究方向:装备经济管理。

目前 A 公司的物流配送网络中,从分库运送至中心库的过程,装载率在 75%左右;从供应商运送至分库或中心库的过程,装载率在 65%左右;从具有集散功能的中心库运送至整车装配厂的过程,装载率在 50%左右。在物流配送网络中,集散点能起到提高物流效率的作用,而 A 公司的物流配送网络在经过集散点后装载率却明显降低了。经实地调查研究,这个现象的原因在于:一是零部件数量繁多、种类繁多,装载过程需考虑其体积、重量、包装要求等,十分复杂;二是准时生产制需要多批次小批量的供货,A 公司中心库向整车厂的供货间隔为 2 h,批量为 60;三是准时生产制要求将零部件送至必要的位置,所以零部件还需按照工位分类,一个托盘上的零部件,仅对应一个工位。由于复杂程度高的零部件和固定且较小的批量,再加之零部件周转箱必须有一面朝外的装载约束,导致 A 公司中心库至整车厂环节的装载效率低下。本文以此实际案例为背景,深入研究考虑货物摆放约束的二维装箱问题。

假定某一零部件批次,待装载货物集合  $A$ , 含有  $n$  个零部件周转箱,  $a_i = (l_i, w_i)$ ,  $i = 1, 2, \dots, n$ , 其中  $l_i$  和  $w_i$  为零部件周转箱的长度和宽度。所用托盘的长度和宽度分别为  $L$  和  $W$ 。且对于  $\forall a_i \in A$ ,  $l_i \leq W$  且  $w_i \leq L$ 。  $x_i$  表示零部件  $a_i$  最左侧与托盘左边缘之间的距离,  $y_i$  表示零部件  $a_i$  最上侧与托盘上边缘之间的距离。  $b_{ij}$  表示零部件  $a_i$  是否装载在托盘  $j$  上, 是则为 1, 否则为 0。  $z_j$  表示托盘是否被使用, 是则为 1, 否则为 0。  $r_i$  表示零部件是否被旋转放置, 是则为 1, 否则为 0。  $m_{ij}$  表示相邻的两个零部件周转箱的左右位置关系, 若周转箱  $i$  在周转箱  $j$  的左侧, 则为 1, 否则为 0;  $n_{ij}$  表示相邻的两个零部件周转箱的前后位置关系, 若周转箱  $i$  在周转箱  $j$  前侧, 则为 1, 否则为 0。  $S_{ij}$  表示周转箱  $i$  和周转箱  $j$  是否放置于同一托盘上, 是则为 1, 否则为 0。

模型目标函数:

$$\min \sum_{i=1}^n Z_i \quad (1)$$

约束条件:

$$x_i + (1 - r_i)l_i + r_i w_i \leq L \quad \forall i, j \quad (2)$$

$$y_i + (1 - r_i)w_i + r_i l_i \leq W \quad \forall i, j \quad (3)$$

$$x_i + (1 - r_i)l_i + r_i w_i \leq x_j + (1 - m_{ij})M \quad \forall i, j, ij \quad (4)$$

$$y_i + (1 - r_i)w_i + r_i l_i \leq y_j + (1 - n_{ij})M \quad \forall i, j, ij \quad (5)$$

$$m_{ij} + m_{ji} + n_{ij} + n_{ji} \geq S_{ij} \quad \forall i, j \text{ 都有 } i < j \quad (6)$$

$$x_j[y_i + (1 - r_i)l_i + r_i w_i - L][y_i + (1 - r_i)w_i + r_i l_i - W] = 0 \quad \forall i \quad (7)$$

$$\sum_{j=1}^n b_{ij} = 1 \quad \forall i \quad (8)$$

$$\sum_{j=1}^n b_{ij} \leq z_j M \quad \forall j \quad (9)$$

$$s_{ij} \geq b_{ik} + b_{jk} - 1 \quad \forall i, j, k \text{ 都有 } i < j \quad (10)$$

其中(1)为模型目标函数,即所用托盘的最小数量;(2)、(3)表示尺寸约束,即周转箱的长或宽不可以超过托盘的长和宽;(4)至(6)表示周转箱之间的位置约束,若两个周转箱放置于同一托盘上,则要么左右相邻,要么前后相邻,并且这两种情况不可能同时发现;(7)表示周转箱至少有一面朝外;(8)表示周转箱被放置在唯一的托盘上;(9)表示将装载周转箱的托盘标记为已使用托盘;(10)表示两个周转箱是否放置于同一个托盘上。

## 2 优化方案算法说明

案例中的装箱问题是三维装箱问题,但在建模过程将其简化为了二维装箱问题。因为 A 公司允许相同的零部件周转箱纵向堆叠,但不可超过一定的高度,因此先将零部件周转箱进行垂直方位的纵向堆叠,然后再进行码放,可以大大简化问题。在待装载零部件周转箱一定的时候,所用托盘数和装载率显然是成反比的,使用的托盘越少,装载率越高。本文使用启发式算法求解在 A 公司的装载约束下,完成某批次零部件装载使用托盘的最少数量。

算法求解步骤如下:

步骤 1: 准时生产制的一项重要理念就是“将必要的零件,以必要的数量,在必要的时间,送到必要的地方”,所以应先将数据按时点进行分,再将同时点的批次代入求解。

步骤 2: A 公司有若干个用于装卸货的到货平台,零部件周转箱会放置于到货平台,且仅相同到货平台的零部件周转箱才可放置在同一托盘上。检验到货平台上的零部件周转箱是否装载完毕,若装载完毕,执行步骤 5,否则,执行步骤 3。

步骤 3: 根据实际装载规则,先将尺寸完全相同的周转箱进行纵向堆叠。考虑装载的稳固和货车尺寸的标准,托盘装载高度不得高于 1.2 m。

步骤 4: 将周转箱进行纵向堆叠后,从二维的角度考虑堆叠物的装载方式。依据堆叠物底部的尺寸选择托盘,在规则下进行装载,直至托盘上无法继续放入周转箱。若依然有未装箱完毕的零部件周转箱,则继续执行步骤 4,否则,执行步骤 2。

步骤 5:将已装载周转箱的托盘标记为已使用状态,待所有周转箱装载完毕后,记录托盘的最终使用数量,输出结果。

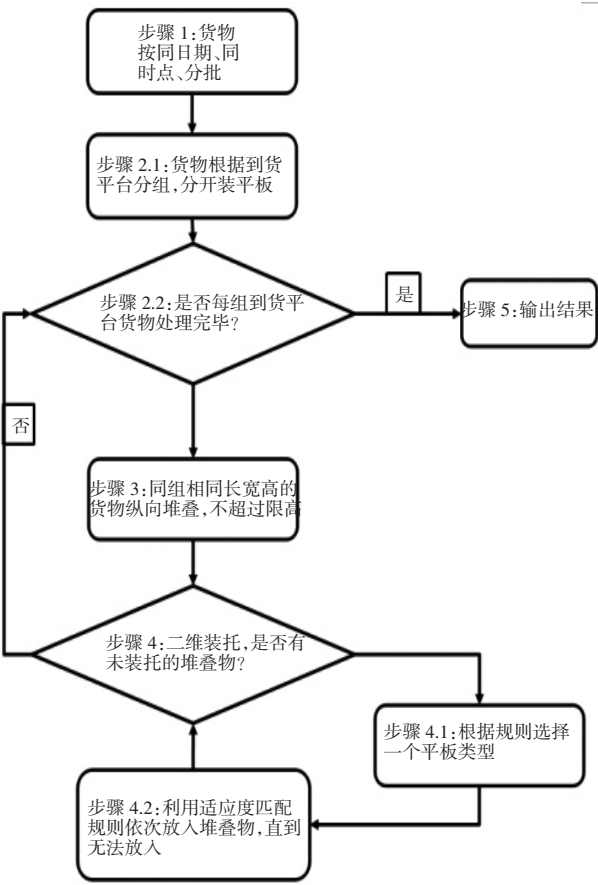


图 2 启发式算法流程图

3 A 公司入厂物流装载率算例分析

3.1 第一阶段装载率优化方案实施结果

A 公司生产模式成熟,一条装配线年产量为 26 万辆,每日入厂物流数据量庞大,目前使用的托盘有两种,尺寸分别为 2.02 m \* 1.11 m 的大托盘和 1.21 m\*1.11 m 的小托盘,使用的零部件周转箱主要有 18 种。

本文选取 A 公司某年 11 月 02 日 19 时的数据进行测算,测算结果如表 1 所示。从测算结果可知,目前使用的托盘尺寸不能满足 A 公司的装载要求,因此需对托盘尺寸作出优化。并且在对 A 公司某年 11 月 02 日全天的数据进行测算后,发现导致托盘装载率低的原因主要有两个:一是因为同一到货平台某批次待装载的零部件极少或仅有单个;二是因为托盘尺寸与周转箱尺寸未能实现和数、倍数关系。到货批次受生产计划限制,难以优化,因此对托盘尺寸的优化也是至关重要的。

表 1 第一阶段测算结果示例

| 车号      | 大托盘数 | 小托盘数 | 装载率 1        | 装载率 2 | 装载率 3 | 装载率 4 | 装载率 5 |
|---------|------|------|--------------|-------|-------|-------|-------|
| EA9D797 | 4    | 1    | 0.908        | 0.908 | 0.908 | 0.787 | 0.458 |
| EAHD068 | 1    | 1    | 0.851        | 0.221 |       |       |       |
| EAHD139 | 0    | 1    | 0.785        |       |       |       |       |
| EAHD208 | 1    | 1    | 0.718        | 0.391 |       |       |       |
| EAHD218 | 1    | 1    | 0.783        | 0.448 |       |       |       |
| EAKD657 | 0    | 1    | 0.211        |       |       |       |       |
| EALD010 | 0    | 1    | 0.105        |       |       |       |       |
| EARD296 |      |      | 存在宽度超过托盘的零部件 |       |       |       |       |

3.2 第二阶段装载率优化方案实施结果

根据第一阶段测算结果,拟对托盘的尺寸作出修改,综合考虑零部件尺寸和货车、叉车等 A 公司目前使用的物流设施,本文设计了三种新的托盘,尺寸最大的 A 类托盘为 236 cm \* 142 cm,尺寸中等的 B 类托盘为 199 cm \* 120 cm 以及尺寸最小的 C 类托盘为 111 cm \* 111 cm;并制定了相应的选择策略:

(1)待装载周转箱的长度大于 1.4 m,或待装载周转箱宽度大于 0.84 m,或待装载周转箱数量多于 15 且底面积大于 0.36 m<sup>2</sup>,则选择使用 A 类托盘。

(2)待装载周转箱的长度大于 1.11 m、待装载周转箱宽度小于 0.84 m,或待装载周转箱数量大于 6 且小于 15 个,或待装载周转箱平均底面积大于 0.308 平方米且小于 0.36 m<sup>2</sup> 时,选择使用 B 类托盘。

(3)待装载周转箱长度小于 1.11 m、待装载周转箱宽度小于 0.84 m 的待装载周转箱,且待装载周转箱总数小于 6 且平均底面积小于 0.308 m<sup>2</sup> 时,选择使用 C 类托盘。

选取 A 公司某年 11 月 2 日 2 时的数据进行测算,测算结果如表 2 所示。所有零部件均能按规则装载至托盘,且托盘使用数量降低,装载率有所提高。

表 2 第二阶段测算结果示例

| 车号      | A 类托盘 | B 类托盘 | C 类托盘 | 装载率 1 | 装载率 2 | 装载率 3 | 装载率 4 |
|---------|-------|-------|-------|-------|-------|-------|-------|
| EA9D797 | 1     | 3     | 0     | 0.847 | 0.842 | 0.817 | 0.663 |
| EAHD068 | 0     | 1     | 1     | 0.780 | 0.230 |       |       |
| EAHD139 | 0     | 0     | 2     | 0.674 | 0.193 |       |       |
| EAHD208 | 0     | 1     | 1     | 0.645 | 0.426 |       |       |
| EAHD218 | 0     | 1     | 1     | 0.735 | 0.489 |       |       |
| EAKD657 | 0     | 0     | 1     | 0.230 |       |       |       |
| EALD010 | 0     | 0     | 1     | 0.115 |       |       |       |
| EARD296 | 1     | 0     | 0     | 0.901 |       |       |       |

3.3 大规模装载率优化方案实施结果

将 A 公司某年 11 月 1 日全天的数据代入程序进行测算,结果如表 3 所示。在第一阶段,不改变托盘尺寸的前提下,平均装载率能提高 17.81%左右;在第二阶段,对托盘尺寸进行优化后,平均装载率能够



提升 2.48%,且使用托盘的数量减少了 20.47%。将因为零部件配送计划导致的超低装载率情况,例如:某时点某到货平板仅有一个待装载周转箱,或是某时点某货车的待装载周转箱在不同的到货平板。这部分数据剔除后,托盘的平均装载率进一步提高,优化效果更为明显。

表 3 优化方案结果对比

|        | 优化前   | 第一阶段   | 第二阶段   | 剔除异常后  |
|--------|-------|--------|--------|--------|
| 平均装载率  | 50%左右 | 67.81% | 69.49% | 78.43% |
| 使用平板数量 | 未记录数据 | 298    | 237    | 172    |

## 4 结论

本文深入研究汽车制造公司入厂物流环节实际装载现状,对其零部件装载环节进行研究,并结合 A 汽车制造公司实际操作中的具体案例进行分析,综合考虑零部件装载时需有一面朝外、纵向堆叠限制,设计了一套基于算法的装载率优化方案,并运用到实际情境中,优化结果表明,通过提出的相应流程和算法能比实际操作获得更好的解,减少入厂物流环节的作业次数,提高入厂物流环节的装载率。优化方

案的不足之处主要有,未能考虑入厂物流环节货车的装载率,从整体环节提升装载率的处理有待提升。在未来的研究中,可以在本文的基础上,建立托盘装载至货车的模型,使生成的解在整体上更优化。

### 参考文献:

- [1] Gilmore, P.C.. The cutting stock problem [C]. IBM Proc. Combinatorial Problems, 1966(13):211-224.
- [2] Christofides, N., C. Whitlock. An algorithm for two-dimensional cutting problems[J]. Oper. Res. 1977, 25(1):30-44.
- [3] Beasley, J.E.. An exact two-dimensional non-guillotine cutting tree search procedure [J]. Oper. Res. Soc. 1985, 33(1):49-64.
- [4] Baker B S, Coffman Jr E G, Rivest R L. Orthogonal packing in two dimensions[J]. SIAM Journal on Computing, 1980(9):846-855.
- [5] Chazelle, B. The bottom-left bin packing heuristic: An efficient implementation[J]. IEEE Trans. Comput., 1983, 32(8):697-707.
- [6] Kunpeng Li, Hailan Liu, Yong Wu, Xianhao Xu. A two-dimensional bin-packing problem with conflict[J]. International Journal of Production Research, 2014.

## Two Dimensional Packing with Placement Constraints

ZHOU Yu-jing, CHEN Guo-wei

(Department of Management Engineering and Equipment Economics, Naval Engineering University, Wuhan 430000, China)

**Abstract:** How to achieve a higher loading rate, improve the utilization rate of container loading space, so as to improve the service level and reduce the operating cost, is of great significance to improve the competitiveness of enterprises. There is a classical solution of “leftmost and lowest alignment” in two-dimensional packing. Its basic principle is to arrange the items in descending order according to the width, and then load the items to the position as low as possible and as far left as possible to accommodate their volume. However, in actual operation, in order to check the information of goods, the goods should be placed with at least one side outward. Considering the above placement constraints, based on the “leftmost and lowest alignment” algorithm, an optimal solution of two-dimensional bin packing problem is proposed and analyzed with an actual example.

**Key words:** 2D bin-packing problem; placement restrictions; heuristic algorithm