

计算机集成制造系统
Computer Integrated Manufacturing Systems
ISSN 1006-5911, CN 11-5946/TP

《计算机集成制造系统》网络首发论文

题目：基于列生成和分支定界算法的两阶段二次装箱问题
作者：杨玉冰，杨松坡
收稿日期：2022-05-26
网络首发日期：2022-10-11
引用格式：杨玉冰，杨松坡. 基于列生成和分支定界算法的两阶段二次装箱问题[J/OL]. 计算机集成制造系统.
<https://kns.cnki.net/kcms/detail/11.5946.TP.20221011.1404.002.html>



网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

基于列生成和分支定界算法的两阶段二次装箱问题

杨玉冰, 杨松坡⁺

(北京工业大学 北京市交通工程重点实验室, 北京 100124)

摘要: 为将切割和装箱问题作为一条完整的生产物流链进行研究, 设计考虑将两者契合在一起, 构建了两阶段二次装箱模型, 第一阶段为二维切割问题, 第二阶段为二维装箱问题。为求解上述问题, 提出了两阶段基于列生成和分支定界算法的求解方式。利用列生成算法对问题的线性松弛主问题进行求解后, 使用分支定界进行后续问题的求解, 最终原整数规划问题通过该方法得到了最优解。通过二次装箱策略将装箱问题分解为一次装箱和二次装箱, 进而简化其求解复杂度。一次装箱类似于快递包装, 在箱子上做上标记, 记录箱子内装载货物的规格和数量位置, 二次装箱类似于包裹的转载运输。二次装箱策略可以对货物进行分类包装, 能够在一定程度上减少货物在后续搬运过程中辨识工作进而减少物流企业在各需求地的搬运成本, 帮助物流企业在后续配送过程中提高工作效率。通过对实例的测试, 结果表明列生成和分支定界算法适用于所提出模型的求解, 研究成果为二维切割和装箱组合问题的求解提供了有效途径。

关键词: 物流工程; 二维装箱问题; 二次装箱策略; 列生成算法; 分支定界; 两阶段求解

中图分类号: TP18

文献标识码: A

Two-stage quadratic packing problem based on column generation and branch and bound algorithm

YANG Yubing, YANG Songpo⁺

(Beijing Key Laboratory of Traffic Engineering, Beijing University of Technology, Beijing 100124, China)

Abstract: In order to study the problem of cutting and packing as a complete production logistics chain, the design considerations combine the two together and construct a two-stage quadratic packing model. The first stage is a two-dimensional cutting problem, and the second stage is a packing problem. To solve the above problem, a two-stage method based on column generation and branch and bound algorithm is proposed. After solving the main problem of linear relaxation by using the column generation algorithm, the subsequent problems are solved by using the branch and bound method. Finally, the optimal solution of the original integer programming problem is obtained by this way. The problem is decomposed into primary and secondary packing by the quadratic packing strategy, by this way the complexity of solving the problem is simplified. The primary packing is similar to the express packing, and marks are made on the boxes to record the specifications and quantity positions of the goods inside the boxes. The secondary

收稿日期: 2022-05-26; 修订日期: 2022-09-08。Received 26 May 2022; accepted 08 Sep. 2022.

基金项目: 国家自然科学基金资助项目(72001008, 52002008); 北京市自然科学基金资助项目(9214022)。Foundation item: Project supported by the National Natural Science Foundation, China (No. 72001008, 52002008), and the Beijing Natural Science Foundation, China (No. 9214022).

packing is similar to the reloading and transportation of packages. The quadratic packing strategy can classify the goods, which can reduce the identification work of the goods in the subsequent handling process to a certain extent, and then reduce the handling cost of logistics enterprises in each demand place, it can also help logistics enterprises improve work efficiency in the follow-up delivery process. Through the test of the example, the results show that that the column generation and branch and bound algorithms are suitable for solving the proposed model, and the research results provide an effective way for solving the two-dimensional cutting and packing combination problems.

Keywords: logistics engineering; two-dimensional packing problem; quadratic packing strategy; column generation algorithm; branch-and-bound; two-stage solution

0 引言

近年来, 切割和装箱问题广泛存在于物流运输的车厢装载、集装箱装载、托盘装载, 以及工厂企业的材料切割、成品包装、设施布局规划等场合中^[1-2]。针对二维切割问题, 需要将一组规格相同的货物切割成更小的单元满足运输需求; 针对二维包装问题, 需要将一类或多类半成品包装到一个或多个有限容器内, 达到节省成本的目的。以上两类问题密切相关, 经济目标一致, 如合理设置中间环节, 可将两者组合进行统一考虑。本文为了保持生产物流链条的连续性、完整性, 以及便于原材料加工场和物流公司之间的联系和统一调度, 将切割和装箱问题进行组合, 引入两阶段法进行求解。耦合后的二维切割和装箱问题, 是一个复杂的组合优化问题, 在模型和求解方法方面均具有重要的研究价值。

对于切割问题而言, 国内外许多学者都对一维切割问题进行了深入研究, 提出了许多求解程序, 并将其扩展到二维切割的情况。二维切割问题属于组合优化的范畴, 其应用领域很广泛, 如: 钢铁、木材、玻璃、皮革、矿产、石料等制造业。这类问题通常是大规模问题, 即切割模式的数量是极其庞大的, 其已被证明是 NP-难问题, 即非多项式时间可解问题。Gilmore 和 Gomory^[3]提出了采用列生成算法来解决一维下料问题的线性松弛问题, 该算法最大限度地减少了毛坯卷的切割次数, 以满足成品卷的需求, 在此基础上, 他们(Gilmore 和 Gomory^[5])还对二维下料问题进行了开创性的研究, 提出了 K 阶段切割模式以及 Guillotine 切割条件约束。Alvarez Valdes^[7]等人开发并比较了几种基于列生成的启发式方法, 以解决一般二维切割问题, 为了解决列生成的子问题, 他们在动态规划的基础上还开发了一些启发式算法。Cintra^[8]等人利用列生成对二维 Guillotine 切割条件下的切割问题及其允许正交旋转的变体进行了研究, 他们提出了矩形背包问题的动态规划算法, 然后在基于列生成的切割问题中使用这些算法生成不同的切割模式。Belov^[10]等对分支定价算法进行了改进, 将通用切割平面与分支-定价算法结合为分支-定价-切割算法, 并研究了其对一维和二维切割问题效用。为求解子问题中带有切割平面约束的线性规划 (Linear Problem, 简称 LP) 松弛问题, 其将每个分支定

价节点上的 LP 松弛问题通过 CHVATAL-GOMORY 和 GOMORY 混合整数切割进行加强。测试结果表明,运用该方法,可以很容易找到一维切割问题和二维切割问题最优解。Melega 等^[11]进行了集成批量和切割库存问题的文献综述。

对于装箱问题而言, Caprara 等^[12]以及 Martello 等^[13]都利用分支定界算法对装箱问题及其变体进行了研究。Vanderbeck^[14]设计了基于列生成的有效精确算法即分支定价算法来解决切割和装箱问题。Elhedhli 等^[15]提出了利用分支定价算法来解决存在冲突的装箱问题。Pisinger 和 Sigurd^[17]利用开发的分支-定价-切割算法,对二维装箱问题进行了研究,在定价子问题当中,其提出了一种约束规划和分支-切割算法相结合的方法来求解由此产生的二维背包问题,该方法得到了比前人更严格的下界,为使用列生成来解决大规模的二维装箱实例问题做出了先例。装箱问题在其他领域也有着很广泛的应用,如 Iori 等^[18]考虑到了二维装箱在带装载约束路径问题上的应用,采用分支-切割算法对路径组合方案搜索,并利用分支定界算法检测路径装箱可行性,但对于路径和装箱约束而言,其将两者分割开来独立建模,未能形成可直接求解的完整模型。Lodi 等人^[19]对几种类型的二维装箱问题进行了综述。上述文章大多数考虑的都为规格相同的料板/箱子的切割/装箱问题,并且仅是对切割和装箱其中一个问题进行考虑,没有将两者进行组合,忽略了生产物流链的连续性和完整性。

本文构建了基于列生成和分支定界算法求解的两阶段二次装箱模型。以储存场切割原材料得到各种类型客户所需求的半成品后,再将其移交物流公司进行装箱操作这一完整的生产运输链条为研究对象,分别建立了切割和装箱模型。在得到储存场切割的半成品后,采用两次装箱策略,首先将根据订单切割出来的半成品装入适宜规格的箱子中进行打包处理(一次装箱),并在箱子上做上标记,记录箱子内装载货物的规格和数量位置;接下来,将运输至中转站的打包半成品装到统一规格的集装箱中(二次装箱)。两次装箱策略能够有效的对货物进行保护,并有效减少货物后续在需求地搬运过程中的费用,既能防止半成品发生损坏变形、防止其在运输过程中发生受潮、发霉或被有害生物啃食等情况,也方便管理员对货物的辨识以及便于各种装卸搬运人员或机械的工作,有利于提高装卸搬运机械的工作效率。在模型求解方面,采用基于列生成和分支定界算法的求解方式对上述两阶段分别进行求解。

1 切割-装箱数学模型

二维切割和装箱问题是由组合问题和调度问题紧密结合的复杂问题。本文考虑一类可切割原材料,与一个容量不限的原材料储存场,该场地根据客户的订单需求进行原材料的切割,在得到满足

客户订单需求的半成品后，为了满足货运运输，对切割后的半成品进行打包处理（一次装箱），并在箱子上做上标记，记录箱子内装载货物的规格和数量位置，接下来将打包半成品运输至中转站后，将其装至集装箱中（二次装箱），至此完成切割-装箱流程。

1.1 问题假设

为了简化实际切割-装箱流程存在多种不确定因素，建立数学模型时考虑如下假设：

- 假设(1)，客户的订单需求，箱子（小母板）的规格以及大集装箱的规格是已知的；
- 假设(2)，原材料、箱子（小母板）以及集装箱具有足够的数量可以满足客户的订单需求；
- 假设(3)，半成品在切割和装箱时，其中一边必须与所使用物体的边正交或者平行；
- 假设(4)，所有货物的宽和高不可超过所用物体的边界；
- 假设(5)，货物之间不可发生重叠。

1.2 切割模型

切割问题中各符号和含义如下表 1 所示：

切割问题主要决定每个切割模式需要使用多少次（原材料的数量）才能满足需求，并使所使用的原材料总数最少，首先将切割问题表述为整数线性规划问题(Integer linear programming, 简称 ILP)。

表 1 切割问题参数及变量符号说明

集合：			
T	切割模式集合， $T = \{t t = 1, 2, \cdots, n\}$ 种		
I	半成品类别集合， $I = \{i i = 1, 2, \cdots, m\}$ 类		
参数：			
n	切割问题中所有可能半成品切割模式的数量	H	高度
m	半成品的种类	W	宽度
$d_i^{(B)}$	客户对第 i 类半成品的需求量	a_{it}	第 t 种切割模式中切割原材料得到的第 i 类半成品的数量
决策变量：			
x_t	第 t 种切割模式所使用原材料的数量，此值非负		
l_i	半成品 i 最左侧与原材料左侧边缘之间的距离		
u_i	半成品 i 最上侧与原材板上边缘之间的距离		

切割问题整数线性规划模型如下：

$$\min Q = \sum_{t \in T} x_t \quad (1)$$

$$\text{subject to } \sum_{t \in T} a_{it} x_t \geq d_i^{(B)}, \forall i \in I \quad (2)$$

$$l_i + W_i^{(B)} \leq W^{(Y)}, \forall i \in I \quad (3)$$

$$u_i + H_i^{(B)} \leq H^{(Y)}, \forall i \in I \quad (4)$$

$$l_i, u_i \geq 0, \forall i \in I \quad (5)$$

$$x_t \in \mathbf{Z}^+, \forall t \in T \quad (6)$$

其中：目标函数（1）为最小化切割原材料的数量（ Q ）；约束（2）为订单需求约束，即第 t 种切割模式切割第 i 类半成品的数量（ a_{it} ）与各模式所使用原材料个数（ x_t ）的乘积之和要满足客户对第 i 类半成品的需求量（ $d_i^{(B)}$ ）。约束（3）和（4）为尺寸约束，即半成品 i 在被切割时不能超出原材料的范围，其中 $W_i^{(B)}$ 、 $H_i^{(B)}$ 为半成品 i 的宽度和高度， $W^{(Y)}$ 、 $H^{(Y)}$ 为原材料的宽度和高度；约束（5）、（6）为变量的类型和取值范围。

1.3 装箱模型

二维装箱问题根据箱子的规格类型可以分为两类，一类是定宽无限长的箱子，其要将给定的小矩形零件以尽可能密的方式装入到箱子中，这种装箱问题称为条形装箱问题，其目标是使箱子的高度尽可能的小，例如布匹的剪裁就属于条形装箱问题；另一类是箱子的长和宽有固定的数值，给定一系列的小矩形零件，需要将所有的小矩形零件装入到箱子中，求使得所用箱子数量最小的方案，这种类型的装箱问题称为箱式矩形装箱问题，例如玻璃切割，木材切割就属于这类的装箱问题^[20]。其中箱式矩形装箱问题根据箱子规格的不同又分为：规格相同的箱式矩形装箱问题和规格不同的箱式矩形装箱问题^[20]。在建立二维装箱模型时，本文对规格相同的箱式矩形装箱和规格不同的箱式矩形装箱问题进行了综合考虑，并且在对两种问题建模时，考虑到了规格相同箱子和规格不同箱子的成本因素。

装箱问题中各符号和含义如下表 2 所示：

表 2 装箱问题参数及变量符号说明

集合：	
U	箱子(小母板)集合, $U = \{u u = 1, 2, \dots, b\}$ 种
K	第一次装箱问题的装箱模式集合, $K = \{k k = 1, 2, \dots, N\}$
J	第二次装箱问题的装箱模式集合, $J = \{j j = 1, 2, \dots, M\}$
G	第二次装箱问题中的箱子类型集合, $G = \{g g = 1, 2, \dots, R\}$
参数：	
$S_i^{(FB)}$	第一次装箱中第 i 类半成品的面积
$S_j^{(SJ)}$	第二次装箱中第 j 种装箱模式对应集装箱的面积
$C_u^{(X)}$	箱子(小母板)的成本
$C_{min}^{(X)}$ 、 $C_{max}^{(X)}$	可选的箱子（小母板）中最小、最大成本
$C_j^{(SJ)}$	第二次装箱中第 j 种装箱模式对应集装箱的成本

$W_u^{(X)}、H_u^{(X)}$	箱子(小母板)的宽度、高度
$W^{(J)}、H^{(J)}$	集装箱的宽度、高度
中间变量:	
$p_{ik}^{(F)}$	第一次装箱中第 k 种装箱模式装载第 i 类半成品的数量
$q_{gj}^{(S)}$	第二次装箱中第 j 种装箱模式装载第 g 类箱子(小母板)的数量
y_g	第二次装箱订单需求量
$W_k^{(FX)}、H_k^{(FX)}$	第一次装箱中第 k 种装箱模式所选取的箱子(小母板)的宽度、高度
$W_g^{(SX)}、H_g^{(SX)}$	第二次装箱中, 第 g 类箱子(小母板)的宽度、高度
$S_g^{(SX)}$	第二次装箱中第 g 类箱子的面积, $S_g^{(SX)} = W_g^{(SX)} \times H_g^{(SX)}$
$S_k^{(FX)}$	第一次装箱中第 k 种装箱模式所选用箱子的面积, $S_k^{(FX)} = W_k^{(FX)} \times H_k^{(FX)}$
$C_k^{(FX)}$	第一次装箱中第 k 种装箱模式所选取的箱子(小母板)的成本
决策变量:	
y_k	第一次装箱中第 k 种装箱模式对应箱子(小母板)的使用数量
x_j	第二次装箱中第 j 种装箱模式对应集装箱的使用数量
$l_{ik}^{(F)}$	第一次装箱模式 k 中, 第 i 类半成品最左侧与箱子左侧边缘之间的距离
$o_{ik}^{(F)}$	第一次装箱模式 k 中, 第 i 类半成品最上侧与箱子上边缘之间的距离
$l_{gj}^{(S)}$	第二次装箱模式 j 中, 第 g 类箱子最左侧与集装箱左侧边缘之间的距离
$o_{gj}^{(S)}$	第二次装箱模式 j 中, 第 g 类箱子最上侧与集装箱的上边缘的距离

则二维装箱整数规划模型如下:

$$\min Z = \sum_{j \in J} C_j^{(S1)} x_j \quad (7)$$

$$\text{subject to } \sum_{k \in K} p_{ik}^{(F)} y_k \geq d_i^{(B)}, \quad \forall i \in I \quad (8)$$

$$\sum_{j \in J} q_{gj}^{(S)} x_j \geq y_g, \quad \forall g \in G \quad (9)$$

$$\sum_{i \in I} \sum_{k \in K} p_{ik}^{(F)} S_i^{(FB)} \leq \sum_{k \in K} S_k^{(FX)} \leq \sum_{k \in K} S_k^{(FX)} y_k \quad (10)$$

$$\sum_{g \in G} \sum_{j \in J} q_{gj}^{(S)} S_g^{(X)} \leq \sum_{j \in J} S_j^{(SJ)} \leq \sum_{j \in J} S_j^{(SJ)} x_j \quad (11)$$

$$\sum_{i \in I} \sum_{k \in K} p_{ik}^{(F)} \leq \sum_{i \in I} d_i^{(B)} \quad (12)$$

$$\sum_{g \in G} \sum_{j \in J} q_{gj}^{(S)} \leq \sum_{g \in G} y_g \quad (13)$$

$$\sum_{i \in I} S_i^{(FB)} d_i^{(B)} \leq \sum_{k \in K} S_k^{(FX)} y_k \quad (14)$$

$$\sum_{g \in G} S_g^{(SX)} y_g \leq \sum_{j \in J} S_j^{(SJ)} x_j \quad (15)$$

$$\sum_{k \in K} C_{\min}^{(X)} y_k \leq \sum_{k \in K} C_k^{(FX)} y_k \leq \sum_{k \in K} C_{\max}^{(X)} y_k \quad (16)$$

$$l_{ik}^{(F)} + W_i^{(B)} \leq W_k^{(FX)}, \forall i \in I, k \in K \quad (17)$$

$$o_{ik}^{(F)} + H_i^{(B)} \leq H_k^{(FX)}, \forall i \in I, k \in K \quad (18)$$

$$l_{gj}^{(S)} + W_g^{(SX)} \leq W_j^{(J)}, \forall g \in G, j \in J \quad (19)$$

$$o_{gj}^{(S)} + H_g^{(SX)} \leq H_j^{(J)}, \forall g \in G, j \in J \quad (20)$$

$$l_{ik}^{(F)}, o_{ik}^{(F)}, l_{gj}^{(S)}, o_{gj}^{(S)} \geq 0, \forall i \in I, g \in G, k \in K, j \in J \quad (21)$$

$$x_j, y_k, y_g \in \mathbf{Z}^+, \forall j \in J, k \in K, g \in G \quad (22)$$

其中，目标函数(7)表示最小化二维装箱所用成本(Z)；约束(8)为第一次装箱订单需求约束，即各装箱模式装载第 i 类半成品的数量($p_{ik}^{(F)}$)与各模式使用次数(y_k)乘积之和需满足客户对第 i 类半成品的需求($d_i^{(B)}$)；约束(9)为第二次装箱订单需求约束，即各模式装载第 g 类箱子(小母板)的数量($q_{gj}^{(S)}$)与各模式的使用次数(x_j)乘积之和需满足在第一次装箱中得出的箱子(小母板)个数(y_g)；约束(10)为第一次装箱面积约束，其中所有装箱模式所装载的半成品总面积($\sum_{i \in I} \sum_{k \in K} p_{ik}^{(F)} S_i^{(FB)}$)不超过所有装箱模式所选择箱子(小母板)的总面积($\sum_{k \in K} S_k^{(FX)}$)，还不能超过装箱各模式所选择箱子(小母板)的面积($S_k^{(FX)}$)与各模式所使用的次数(y_k)的乘积之和($\sum_{k \in K} S_k^{(FX)} y_k$)；约束(11)为第二次装箱面积约束，其中所有装箱模式所装载的箱子(小母板)总面积($\sum_{g \in G} \sum_{j \in J} q_{gj}^{(S)} S_g^{(X)}$)不超过所有装箱模式所选择集装箱的总面积($\sum_{j \in J} S_j^{(SJ)}$)，还不能超过装箱各模式所选择集装箱的面积($S_j^{(SJ)}$)与各模式所使用的次数(x_j)的乘积之和($\sum_{j \in J} S_j^{(SJ)} x_j$)；约束(12)为第一次装箱中，所有装箱模式所装载的半成品数量和($\sum_{i \in I} \sum_{k \in K} p_{ik}^{(F)}$)不超过客户订单需求的半成品总数($\sum_{i \in I} d_i^{(B)}$)；约束(13)为第二次装箱中，所有装箱模式所装载的小母板的数量($\sum_{g \in G} \sum_{j \in J} q_{gj}^{(S)}$)不超过第一次装箱问题得到的小母板的总数($\sum_{g \in G} y_g$)；约束(14)为第一次装箱各模式所选择的小母板的面积($S_k^{(FX)}$)与各模式所使用的次数(y_k)的乘积之和不能小于订单需求中所有半成品总面积($\sum_{i \in I} S_i^{(FB)} d_i^{(B)}$)；约束(15)第二次装箱各模式所选择的集装箱的面积($S_j^{(SJ)}$)与各模式所使用的次数(x_j)的乘积之和不能小于第一次装箱所得

出的订单需求中所有小母板总面积 ($\sum_{g \in G} S_g^{(SX)} y_g$)；约束 (16) 为第一次装箱所用总成本 Z 应在最大

成本 ($\sum_{k \in K} C_{\max}^{(X)} y_k$) 与最小成本 ($\sum_{k \in K} C_{\min}^{(X)} y_k$) 之间；约束 (17) (18) 为第一次装箱尺寸约束，半成品 i 在被装箱时不能超出箱子（小母板）的范围；约束 (19) (20) 为第二次装箱尺寸约束，即箱子（小母板） g 在被装箱时不能超出集装箱的范围。约束 (21) (22) 为变量的类型和取值范围。

1.3.1 模型降维

装箱整数线性规划模型中含有大量变量和约束，难以直接求解较大规模算例。本文提出一种降维方法将上述装箱模型转化为两次装箱模型。具体如下：首先，在储存场切割原材料得到各种类型客户所需求的半成品后，给出有限的箱子（小母板）种类。然后在此基础上进行第一次装箱求解得到箱子（小母板）所用的最优规格以及数量。最后，再依据上述所得进行第二次装箱求解得到集装箱的所用数量。依据该思路对装箱整数规划模型进行重构，分别得到一次装箱模型和二次装箱模型。其中，在第一次装箱中，考虑规格不同的箱式矩形装箱；在二次装箱中，考虑规格相同的箱式矩形装箱，下面给出两次装箱具体模型。

① 不同规格矩形装箱

储存场切割原材料得到各种类型客户所需半成品后，将得到的半成品装到规格不同的矩形箱子（小母板），即对半成品进行打包处理，并在箱子上做上标记，记录箱子内装载货物的规格和数量位置，称之为第一次装箱问题。详细地，首先，将规格不同矩形装箱问题表述为 ILP(整数线性规划) 问题。这里我们将二维装箱模型中的可选箱子（小母板）的种类数 b 固定，然后得到简化后的一次装箱模型，即规格不同矩形的装箱问题模型。该装箱问题包括决定使用何种类型的箱子（小母板），使用哪些装箱模式以及每种模式必须使用多少次才能满足订单需求，并最小化所使用的矩形箱子（小母板）的总成本。

我们有如下的 ILP 公式：

$$\min Z_1 = \sum_{k \in K} C_k^{(FX)} y_k \quad (23)$$

$$\text{subject to } l_{ik}^{(F)}, o_{ik}^{(F)} \geq 0, \forall i \in I, k \in K \quad (24)$$

$$y_k \in \mathbf{Z}^+, \forall k \in K \quad (25)$$

约束 (8)、(10)、(12)、(14)、(17) (18)

其中：目标函数 (23) 表示最小化第一次装箱成本 (Z_1)；该模型应同时满足约束 (8)、(10)、(12)、(14)、(17)、(18)。

② 规格相同矩形装箱问题

在这里，本文以储存场切割原材料完并进行一次装箱到规格不同的小母板后，最后将规格不同

的小母板装到规格相同的集装箱中这类实际问题为例进行构建第二次装箱模型。

首先将规格相同矩形的装箱问题表述为 ILP(整数线性规划)问题。该装箱问题包括决定每种模式需要使用多少次（集装箱数量）才能满足需求，并使所使用的集装箱总成本最小。

我们有如下的 ILP 公式：

$$\min Z_2 = \sum_{j \in J} C_j^{(S)} x_j \quad (26)$$

$$\text{subject to } l_{gj}^{(S)}, o_{gj}^{(S)} \geq 0, \forall g \in G, j \in J \quad (27)$$

$$x_j \in \mathbf{Z}^+, \forall j \in J \quad (28)$$

约束 (9)、(11)、(13)、(15)、(19)、(20)

其中：目标函数 (26) 表示最小化第二次装箱所用成本 (Z_2)；该模型应同时满足约束 (9)、(11)、(13)、(15)、(19)、(20)。

2 算法分析

列生成算法先使用一小部分初始解求解受限主问题(restrict master problem, 简称 RMP)。然后, 初始基本解决方案由检验数为正的新生成的列迭代更新, 这些列由适当的检验数子问题(Reduced costsubproblem)求解生成。当检验数子问题的解不再提供大于 0 (针对最小化问题) 的检验数时, 线性规划问题 (Linear programming problem 简称 LP) 被解为最优。

分支定界算法在得到原问题去掉整数约束后的线性规划问题的初始基本解决方案后, 若解不满足整数要求, 将原问题划分为若干子问题, 每个对应的子问题又可继续分解, 直到子问题不能再继续分解或不能产生最优解。把原问题分解为子问题的过程称之为分支, 在分支过程中, 为每一子问题确定其对应的目标值的界限称之为定界。

考虑到我们求解的底层算法为列生成以及分支定界算法, 则求解算法的时间复杂度的为两种算法时间复杂度之和, 经查阅 Cintra 等人^[8]的文献可知, 列生成算法平均生成的列数为 $O(m^2)$, 其中 m 代表订单需求的类型数, 即小矩形的类型数。查阅其他关于分支定界的资料可知, 分支定界的时间复杂度为 $O(m^2 \times 2^m)$, 其中 m 同为订单需求的类型数。则算法总时间复杂度为 $O(m^2 \times (2^m + 1))$ 。

2.1 切割问题

2.1.1 切割 RMP 问题

对于切割问题而言, 由于切割模型包含的变量数量庞大, 对大规模问题直接枚举所有列的方式不可行。为此, 先构造 m 个可行切割模式, 由这 m 个切割模式构成一个初始可行装箱模式矩阵, 即先枚举部分列构造一个规模较小的主问题模型。此时, 系数矩阵为满秩状态, 能够保证一开始能够求解, 然后将 x_i 线性松弛, 仅使 $x_i \geq 0$, 这样其就变为一个易于求解的 RMP 问题, 从而构成切割问题的限制线性主问题模型。

有以下切割 RMP 问题：

$$\min Q_1 = \sum_{i \in I} x_i \quad (29)$$

$$\text{subject to } \sum_{i \in I} a_{it} x_i \geq d_i^{(B)}, \forall i \in I \quad (30)$$

$$x_i \geq 0, \forall i \in I \quad (31)$$

2.1.2 切割检验数子问题模型

从现实意义的角度来看主问题与检验数子问题的关系时，主问题中的每一“列”代表了不同的原材料的切割模式，在求解检验数的迭代过程中，若当前存在的“列”不能有效满足客户订单需求时，则会“加列”，即增加另外一种新的切割模式，随着加入的“列”增多，原材料的切割模式也不断增加，当不能加入新的列时则表明没有更优类型的切割模式，即主问题取得最优解。切割问题中子问题的目的是寻找新的较优可行切割方案。当将该可行方案加入到目标函数中时，目标函数值可以得到下降。对于这里的切割问题来讲，因为只存在一种原材料规格，所以只对应一个子问题。定义符号和含义如下表 3 所示：

表 3 切割子问题参数及变量符号说明

集合：	
$C_i^T = [C_1^T, \dots, C_m^T]$	当前线性规划目标函数的系数，此处为 1
参数：	
A	当前线性规划的基矩阵
$\pi_i^T = C_i^T A^{-1}$	切割 RMP 问题中第 i 类半成品需求约束所对应的对偶变量
c	RMP 问题中目标函数的系数，在此模型中此值为 1；
S	面积
决策变量：	
a_i	通过子问题新生成的切割模式其切割第 i 类半成品的数量

根据影子价格系数构造辅助二维背包子问题模型如下：

$$\max Q_2 = \sum_{i \in I} \pi_i^T a_i - c \quad (32)$$

$$\text{subject to } \sum_{i \in I} S_i^{(B)} a_i \leq S^{(Y)} \quad (33)$$

$$a_i \leq d_i^{(B)}, \forall i \in I \quad (34)$$

$$a_i \in \mathbf{Z}^+, \forall i \in I \quad (35)$$

约束 (3)、(4)、(5)。

其中：目标函数 (32) 为最大化切割 RMP 问题检验数 (Q_2)。约束 (33) 为切割方案切割出的第 i 类半成品的数量 a_i 与第 i 类半成品面积 $S_i^{(B)}$ 的乘积之和不超过原材料的面积 $S^{(Y)}$ ；约束 (34) 为切割方案切割出的第 i 类半成品的数量不能超过订单需求中半成品的数量 $d_i^{(B)}$ ；约束 (35) 为正整数约束。该模型应同时满足约束 (3)、(4) 和 (5)。

2.2 装箱问题

2.2.1 装箱 RMP 问题

由于装箱模型包含的变量数量庞大, 对大规模问题直接枚举所有列的方式不可行。

① 规格不同矩形装箱 RMP 问题

对于规格不同矩形装箱 RMP 问题, 针对变量数量过大, 本文先构造 m 个可行装箱模式, 由这 m 个装箱模式构成一个初始可行装箱模式矩阵, 即先枚举部分列构造一个规模较小的主问题模型, 此时, 系数矩阵为满秩状态, 能够保证一开始能够求解。然后将 y_k 线性松弛, 仅使 $y_k \geq 0$, 这样其就变为一个易于求解的 RMP 问题, 从而构成第一次装箱的限制线性主问题模型。

$$\min Z_3 = \sum_{k \in I} C_k^{(FX)} y_k \quad (36)$$

$$\text{subject to } \sum_{k \in I} p_{ik}^{(F)} y_k \geq d_i^{(B)}, \quad \forall i \in I, \quad (37)$$

$$y_k \geq 0, \quad \forall k \in I \quad (38)$$

约束 (10)、(12)、(14)。

② 规格相同矩形装箱 RMP 问题

对于规格相同矩形装箱 RMP 问题, 构造方法相似, 先构造 R 个可行装箱模式, 由这 R 个装箱模式构成一个初始可行装箱模式矩阵, 此时, 系数矩阵为满秩状态, 能够保证一开始能够求解。然后再将 x_j 线性松弛, 仅使 $x_j \geq 0$, 这样其也变为一个易于求解的 RMP 问题, 从而构成第二次装箱的限制线性主问题模型。

$$\min Z_4 = \sum_{j \in G} C_j^{(SJ)} x_j \quad (39)$$

$$\text{subject to } \sum_{j \in G} q_{gj}^{(S)} x_j \geq y_g, \quad \forall g \in G \quad (40)$$

$$x_j \geq 0, \quad \forall j \in G \quad (41)$$

约束 (11)、(13)、(15)。

2.2.2 装箱检验数子问题模型

装箱子问题的目的是寻找新的较优可行装箱方案。当将该可行方案加入到主问题目标函数中, 目标函数值可以得到下降。对于这里的装箱问题来讲, 第一次装箱过程中存在 b 种规格箱子 (小母板) 可供选择, 所以第一次装箱检验数子问题对应有 b 个, 但为了提高模型的优化速度, 本文考虑只加入检验数最大的一列。对于第二次装箱过程而言, 因为只存在一种规格集装箱可供选择, 所以这里的第二次装箱检验数子问题对应有 1 个。

① 规格不同矩形装箱检验数子问题

定义符号和含义如下表 4 所示:

表 4 第一次装箱子问题参数及变量符号说明

集合:	
$F_i^T = [F_1^T, \dots, F_m^T]$	当前线性规划目标函数的系数
参数:	
P	当前线性规划的基矩阵
$\sigma_i^T = F_i^T P^{-1}$	第一次装箱 RMP 问题中第 i 类订单需求约束所对应的对偶变量
$S_u^{(X)}$	第 u 种箱子（小母板）的面积
$C_u^{(X)}$	第 u 种箱子（小母板）的成本
决策变量:	
$p_i^{(F)}$	通过子问题新生成的装箱模式中装载第 i 类订单需求的数量

根据影子价格系数构造辅助检验数子问题模型如下:

$$\max Z_5 = \sum_{i \in I} \sigma_i^T p_i^{(F)} - C_u^{(X)} \quad (42)$$

$$\text{subject to } \sum_{i \in I} S_i^{(B)} p_i^{(F)} \leq S_u^{(X)}, \forall u \in U \quad (43)$$

$$p_i^{(F)} \leq d_i^{(B)}, \forall i \in I \quad (44)$$

$$p_i^{(F)} \in Z^+, \forall i \in I \quad (45)$$

约束 (17)、(18)。

其中: 目标函数 (42) 为最大化第一次装箱 RMP 问题检验数 (Z_5)。约束 (43) 为装箱方案装载第 i 类半成品的数量 $p_i^{(F)}$ 与第 i 类半成品面积 $S_i^{(B)}$ 的乘积之和不超过小母板的面积 $S_u^{(X)}$; 约束 (44) 为装箱方案装载第 i 类半成品的数量 $p_i^{(F)}$ 不能超过订单需求中半成品的数量 $d_i^{(B)}$; 约束 (45) 为正整数约束。该模型应同时满足约束 (17) 和 (18), 其为第一次装箱尺寸约束, 半成品 i 在被装箱时不能超出箱子 (小母板) 的范围。

②规格相同矩形装箱检验数子问题

定义符号和含义如下表 5 所示:

表 5 第二次装箱子问题参数及变量符号说明

集合:	
$L_g^T = [L_1^T, \dots, L_N^T]$	当前线性规划目标函数的系数, 此处为 1
参数:	
Q	当前线性规划的基矩阵
$\omega_g^T = L_g^T Q^{-1}$	第二次装箱 RMP 问题中第 g 类订单需求约束所对应的对偶变量
变量:	
$q_g^{(S)}$	通过子问题新生成的装箱模式装载第 g 类订单需求的数量

根据影子价格系数构造辅助检验数子问题模型如下:

$$\max Z_6 = \sum_{g \in G} \omega_g^T q_g^{(S)} - C_j^{(J)} \quad (46)$$

$$\text{subject to } \sum_{g \in G} S_g^{(SX)} q_g^{(S)} \leq S_j^{(SJ)} \quad (47)$$

$$\sum_{g \in G} q_g^{(S)} \leq \sum_{g \in G} y_g \quad (48)$$

$$q_g^{(S)} \in \mathbf{Z}^+, \forall g \in G \quad (49)$$

约束 (19)、(20)

其中：目标函数 (46) 为最大化第二次装箱 RMP 问题检验数 (Z_6)。约束 (47) 为装箱方案装载第 g 类箱子的数量 $q_g^{(S)}$ 与第 g 类箱子面积 $S_g^{(SX)}$ 的乘积之和不超过集装箱的面积 $S_j^{(SJ)}$ ；约束 (48) 为新生成的第二次装箱方案装载第 g 类箱子的总数量 $\sum_{g=1}^G q_g^{(S)}$ 不能超过需求中箱子的总需求 $\sum_{g=1}^G y_g$ ；约束 (49) 为正整数约束。该模型应同时满足约束 (19) 和 (20)，其为第一次装箱尺寸约束，即箱子（小母板） g 在被装箱时不能超出集装箱的范围。

2.3 分支定界算法

在得到线性松弛问题的解后，结果中可能会出现小数的情况，这种情况在实际之中是不可接受的，为了得到整数线性规划的解，本文在得到线性松弛问题的解后，采用分支定界算法进而得到整数规划的解。

2.4 算法流程图

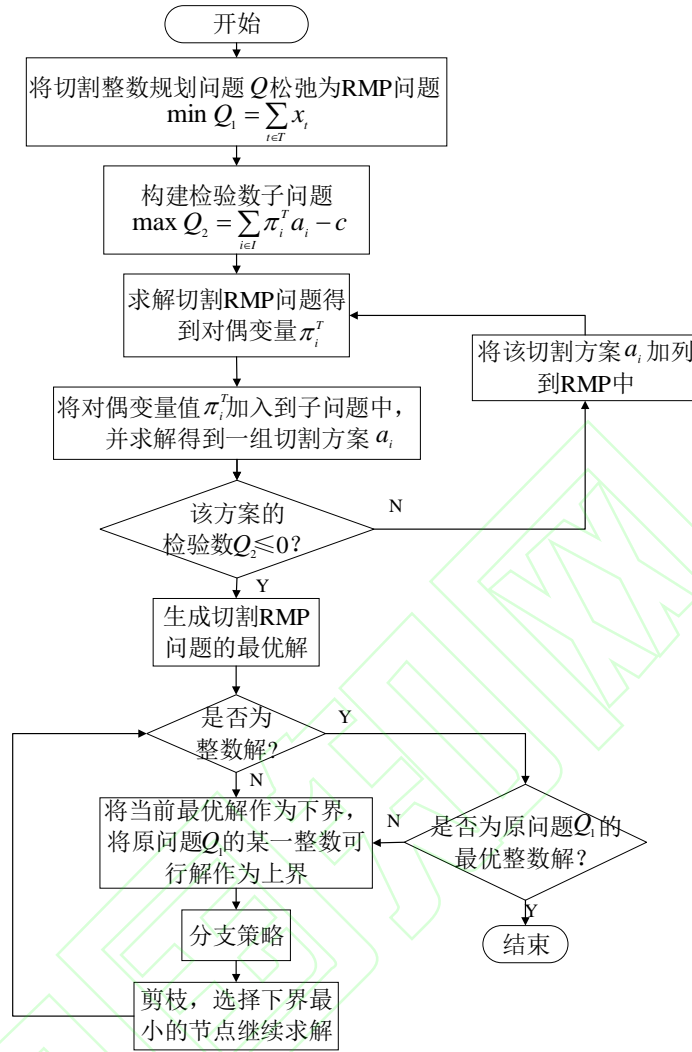


图 1 求解切割模型算法流程图

本文通过对切割得到的半成品进行打包操作这一中间环节，将切割和装箱问题契合在一起，构建了二维切割和二维装箱问题相组合的两阶段各自的规划模型，通过设置降维方法，即在半成品切割完后，在对半成品进行打包操作之前，给出有限的打包箱子（小母板）的种类，依据该方法对装箱整数规划模型进行降维进而得到一次装箱模型和二次装箱模型。

2.4.1 切割模型算法流程图

切割模型算法流程图如图 1。

其中，分支策略是指：在求得的线形松弛问题的最优解中任选一个不符合整数条件的变量，假设该变量为 x_j ，其数值对应为 b_j ，以 $[b_j]$ 表示不超过 b_j 的最大整数，构造两个相互排斥的不等式约束条件 $x_j \leq [b_j]$ 和 $x_j \geq [b_j] + 1$ ，将这两个约束条件分别加入线形松弛问题中去，得到两个子问题，即两个分支。用单纯形等方法求出两个子问题的最优解。再从求得的结果中找出最优值的最小者为新

的下界 \underline{z} ，从已符合整数条件的各分支中，找出目标函数值的最小者作为新的上界 \bar{z} 。

2.4.2 装箱模型算法流程图

装箱模型算法流程图如下图 2。

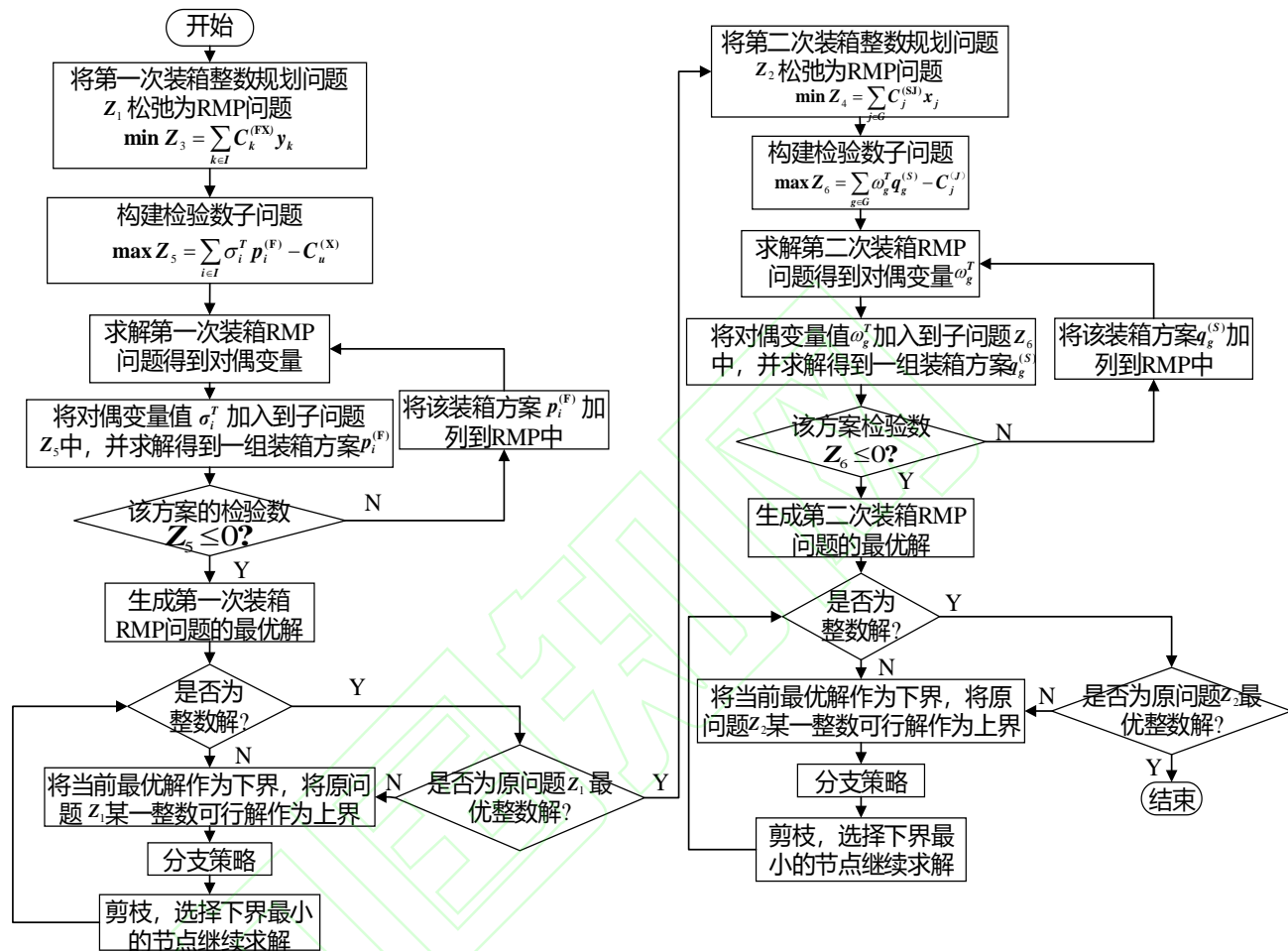


图 2 求解装箱模型算法流程图

3 案例分析

实例运行的计算机的配置如下：Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80，RAM 为 12.00GB，64 位操作系统。软件版本为 MATLAB R2021a64-bit (win64)。对于所提出的一次切割，两次装箱的现实情况来讲，本文考虑存在一个容量无限的木材场，该场根据客户的订单需求对原木板进行切割，在得到满足客户订单需求的小木块后，再通过货运进行运输，运输期间需将小木块进行打包处理（一次装箱），接下来，将运输至中转站的打包小木块装到集装箱中。这里有原木板宽 $W^{(Y)}=500\text{cm}$ ，高 $H^{(Y)}=300\text{cm}$ ，大集装箱规格为宽 $W^{(J)}=1200\text{cm}$ ，高 $H^{(J)}=600\text{cm}$ ，其成本 $C^{(J)}$ 为 720 元，其中客户订单需求的宽度为 $[1/10W^{(Y)}, 3/10W^{(Y)}]$ 内的随机数，订单的高度为 $[1/10H^{(Y)}, 1/3H^{(Y)}]$ 内的随机数，订单需求为 $[200, 300]$ 内的随机数，具体见下表 6 所示：

表 6 客户需求

小木块编号	规格(cm)	个数
C ₁	83*146	284
C ₂	48*105	225
C ₃	65*64	282
C ₄	79*65	224
C ₅	93*76	293

表中第一列为客户所需小木块编号，总共有 5 种客户需求，其对应的矩形小木块的尺寸规格和需求数量如第二列、第三列所示。可以看出，每种订单需求的规格都不超过原木板的规格。

可选择的箱子（小母板）的种类 b 为 36 种，其宽度 $W_u^{(x)}$ ，高度 $H_u^{(x)}$ 及成本 $C_u^{(x)}$ 如表 7 所示：

表 7 小母板规格和成本矩阵（单位/元）

$H_u^{(x)}$ (cm) $W_u^{(x)}$ (cm)	100	200	300	400	500	600
100	1	2	3	4	5	6
200	2.1	4.1	6.1	8	10	12
300	3.1	6.2	9	12.1	15	18
400	4.2	8.1	12.2	16	20	24
500	5.1	10.1	15.1	20.1	25	30
600	6.3	12.3	18.1	24.1	30.1	36

上表中第一行表示可选箱子（小母板）的高度取值，第一列表示可选箱子（小母板）的宽度取值，两者共同组成了一个 6×6 的矩阵，每种箱子（小母板）规格的成本如矩阵中相应位置所对应，且不同规格小母板对应着不同的成本。由上可知，该案例在装箱过程中，将装箱模型中的可选箱子（小母板）的种类数 b 固定为 36，然后得到简化后的一次装箱模型和二次装箱模型，并在此基础上，对案例进行的求解。

① 对于传统的原木板切割和一次装箱问题而讲，需要确定最佳的切割/装箱方案，以及各方案所使用的次数，并且使得原木板/集装箱使用总数最少。算法对两者求解结果如下表 8、表 9 所示：

表 8 算法对切割方案的求解结果

切割方案	所需原木板（张）	切割零件编号×个数	原木板利用率	原木板综合利用率	消耗原木板数量	运算时间
①	19	C ₁ *6,C ₂ *4,C ₃ *3,C ₄ *5,C ₅ *2	96.77%	94.91%	63 张	45.87s
②	23	C ₁ *6,C ₂ *5,C ₃ *6,C ₅ *3	96.05%			
③	3	C ₂ *5,C ₃ *5,C ₄ *8,C ₅ *8	95.75%			
④	6	C ₁ *6,C ₂ *4,C ₄ *6,C ₅ *3	96.59%			

⑤	12	C ₃ *6,C ₄ *6,C ₅ *12	93.72%
---	----	--	--------

注：原木板综合利用率=切割出小木块的面积和/所使用的原木板的面积和

表 9 算法对传统一次装箱求解结果

装箱方案	所需集装箱（个）	排放零件序号×个数	单箱利用率	集装箱综合利用率	消耗集装箱数量	运算时间
(1)	5	C ₁ *31,C ₂ *15,C ₃ *14,C ₄ *6,C ₅ *24	98.60%			
(2)	4	C ₁ *16,C ₂ *45,C ₃ *17,C ₄ *10,C ₅ *24	98.94%			
(3)	2	C ₁ *11,C ₂ *20,C ₃ *58,C ₄ *44,C ₅ *2	99.37%	98.98%	14 个	1831.81s
(4)	1	C ₂ *50,C ₄ *84,C ₅ *5	99.82%			
(5)	2	C ₁ *22,C ₂ *15,C ₃ *14,C ₄ *6,C ₅ *40	99.16%			

注：集装箱综合利用率=所装载矩形零件的面积和/所使用的集装箱的面积和

对于切割方案的求解结果而言，表中第一列为最优的切割方案数量 $m=5$ ，第二列为不同切割方案所使用原木板的数量，第三列为具体的切割方案，即不同切割方案中切割小木块的种类以及数量，第四列为各切割方案对各自使用原木板的利用率，第五列为原木板综合利用率，其计算公式为：原木板综合利用率=切割出小木块的面积和/所使用的原木板的面积和，第六列为所有切割方案所消耗原木板的总数量，第七列为算法运算时间；对于传统一次装箱求解结果而言，与对切割方案的求解结果描述类似。

由上述两表知，算法分别在 45.87 s 和 1831.81s 内对木材切割问题和传统的一次装箱问题进行了求解，完成了传统的切割方案和装箱方案的制定任务。对于切割方案而言，共消耗原木板 63 张，原木板综合利用率为 94.91%，第三列所示为具体的切割方案，以第一种最优切割方案为例，其包括切割 C₁ 型编号的小木块 6 个，C₂ 型编号的小木块 4 个，C₃ 型编号的小木块 3 个，C₄ 型编号的小木块 5 个，C₅ 型编号的小木块 2 个，其中各小木块编号对应的规格如表 6 所示，该方案共使用原木板 19 张。其余切割方案在表中的表达方式跟这里相同，具体不再展开描述。对于装箱方案而言，最终使用了 14 个集装箱将全部小木块装下，且集装箱的综合利用率为 98.98%，其余各列含义与切割问题类似，具体不在展开描述。

其中，所用原木板张数随算法迭代次数变化曲线，以及集装箱所用个数随算法迭代次数变化曲线如下图 3 所示。对于切割阶段而言，算法在第 6 次迭代后，原木板所用张数趋于稳定，在经过第 20 次迭代后，算法求解结果达到最优，此时所用原材料张数为 63 张；对于装箱阶段而言，算法在第 5 次迭代后，原木板所用张数趋于稳定，在经过第 15 次迭代后，算法求解结果达到最优，此时所用集装箱个数为 14 个。

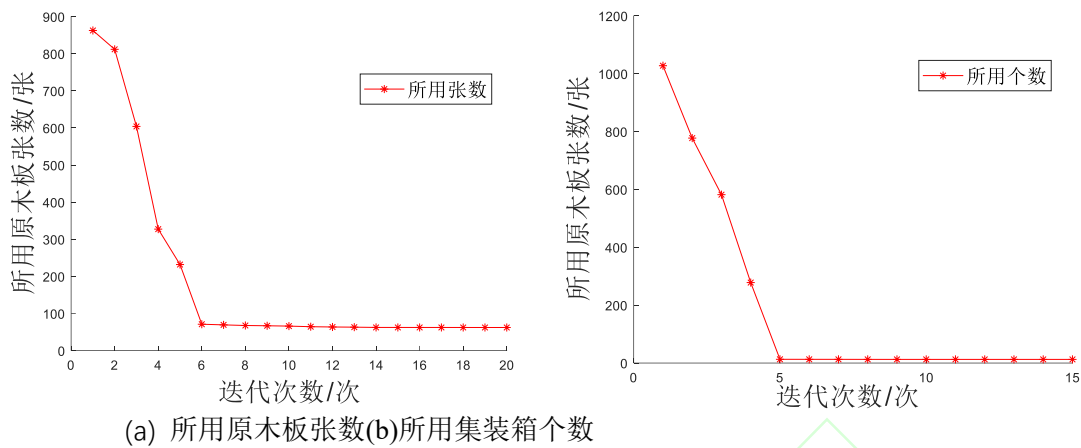


图 3 成本随迭代次数变化曲线

② 对于二次装箱策略而讲：

传统的一次装箱问题当中，各类物流服务商仅仅是单纯地依靠工作人员传统模式下的经验主义来进行装载作业，将货物在箱子中任意的叠放在一起或按照从大到小，从小到大的顺序进行排放，或者在现有的算法的求解结果基础上，将货物按照所求在箱子中进行摆放，并未充分考虑在后续搬运过程中的便捷性，以及搬运所需费用。而我们提出的二次装箱策略相当于对传统的一次装箱进行了分类包装，并在箱子上做上标记，方便管理人员对货物的辨识以及搬卸人员的搬运，可以有效减少货物后续在需求地搬运过程中的费用。在配送中心将所有客户需求的货物运送到各个客户所在地时，传统的一次装箱问题需将所有的货物从集装箱中都卸载下来之后，才能去寻找该客户所需求的货物类型及数量，而我们的二次装箱策略在将所有客户需求的货物运送到各个客户时，因在箱子上已做上标记，其记录着箱子中所装载货物的类型和数量，所以在到达客户地点之后，只需从集装箱中搬运装有该地客户订单需求的箱子后，再直接从箱子中拿出所需货物即可，进而减少一定的搬运成本。且某个箱子中可能会出现不装载某一客户订单需求的情况，此时在该需求地就不用搬运该箱子，进而也会减少一定的搬运成本。

本节在运输成本一致的情况下，即传统一次装箱所采用的集装箱数量与我们所提出二次装箱策略所采用集装箱数量相同的情况下，对传统的一次装箱和二次装箱策略在空间利用率和搬运成本上进行比较。

假定存在以下的路网，且规定在后续货物运输过程中，配送路线只能为逆时针配送。虽然我们在模型中并未涉及到配送的路径问题约束，但我们是实际出发进行考虑的。设在客户地点搬卸一次货物花费为 1 元，搬卸一次箱子花费为 3 元。下面对该实例分别进行运输成本、空间利用率和搬运成本的分析。小木块编号右下角的数字 i 代表需将该种类小木块送往 i 地。

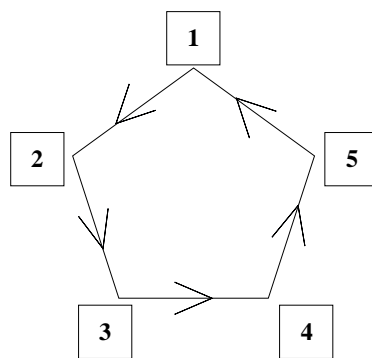


图 4 路网形式

(a) 运输成本和空间利用率:

对于二次装箱策略而言，其第一次装箱阶段和第二次装箱阶段的求解如下表 10、表 11 所示:

表 10 算法对第一次装箱阶段的求解结果

装箱方案	所需箱子 (规格×个)	排放零件序号*个数	单板利用率 (%)	小 motherboard 综合利用率	运算时间/s	总成本 (元)
(6)	(500*600)×1	C ₁ *16,C ₂ *3,C ₃ *3,C ₄ *3,C ₅ *8	97.81			
(7)	(600*600)×4	C ₁ *7,C ₂ *24,C ₃ *10,C ₄ *10,C ₅ *8	98.69			
(8)	(600*600)×8	C ₁ *7,C ₂ *6,C ₃ *26,C ₄ *8,C ₅ *13	98.94	98.48%	791.12	936
(9)	(500*600)×11	C ₁ *16,C ₂ *5,C ₄ *9,C ₅ *4	97.86			
(10)	(600*600)×4	C ₁ *4,C ₂ *6,C ₃ *8,C ₄ *9,C ₅ *28	98.92			

注：小 motherboard 综合利用率=所装载矩形零件的面积和/所使用的小 motherboard 的面积和

表 11 算法对第二次装箱阶段求解结果

装箱方案	所需集装箱 (个)	排放小 motherboard 规格×个数	单箱利用率 (%)	集装箱综合利用率	集装箱数量 (个)	运算时间
(11)	6	(500*600)×2	83.33			
(12)	8	(600*600)×2	100	91.45%	14	2.20

注：集装箱综合利用率=所装载矩形零件的面积和/所使用的集装箱的面积和

由上述两表知，算法分别在 791.12 s 和 2.20s 内对二次装箱策略的第一次装箱阶段和第二次装箱阶段进行了求解，完成了第一次装箱方案和第二次装箱方案的制定任务。对于第一次装箱阶段而言，共消耗规格为（600cm*600cm）的箱子 16 个，消耗规格为（500cm*600cm）的箱子 12 个，箱子综合利用率为 98.48%；对于第二次装箱阶段而言，最终也是使用了 14 个集装箱将所有小 motherboard 全部装箱，集装箱的综合利用率为 91.45%。

其中，对于第一次装箱阶段的成本随算法迭代次数变化曲线以及第二次装箱阶段的所用集装箱个数随算法迭代次数变化曲线如下图 5 所示：

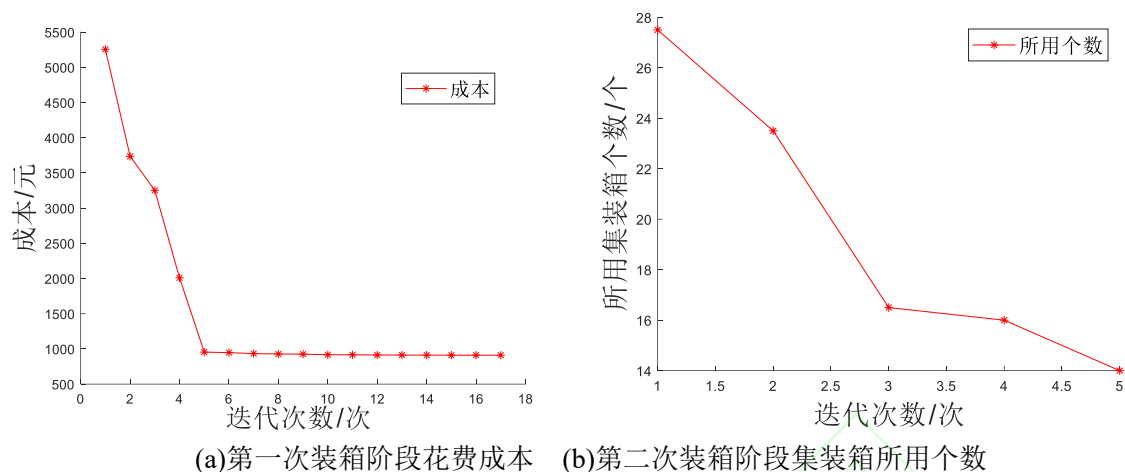


图 5 装箱阶段成本变化曲线

由上图可知，对于第一次装箱阶段而言，算法在第 6 次迭代后，成本趋于稳定，在经过第 17 次迭代后，算法求解结果达到最优，花费成本最优值为 936 元。对于第二次装箱阶段而言，在第 5 次迭代后，算法求解结果达到最优，此时所用集装箱个数为 14 个，其总成本为 10080 元。

综上所述，通过对比上述表格可知，此算例在使用传统的一次装箱策略下所用的集装箱的数量与二次装箱策略所用的集装箱的数量相同，即两种情况下的运输成本一致。在此条件下，对两种情况的空间利用率进行对比可知，传统的一次装箱中，集装箱的综合空间利用率为 98.98%，二次装箱策略集装箱的综合空间利用率为 91.45%，出现这种情况的原因在于我们的二次装箱策略在第一阶段中先将小木块装入到我们给出的箱子（小母板）中，将其进行分类打包，并在箱子上做上了标记，此时箱子（小母板）中就会出现不会被完全填满的情况，即此时会出现一定的空间损失，然后再将我们选择的箱子（小母板）装入到集装箱中，此时，也会出现箱子（小母板）的排放没有将集装箱排满的情况，此时也会出现一定的空间损失。因此，二次装箱策略的集装箱综合利用率相较于传统的一次装箱会低一些，但此时两种情况所用的集装箱数量是相同的，即运输成本是一致的，二次装箱策略也不会出现额外的运输成本支付。

(b) 搬运成本：

在运输成本一致的情况下，传统一次装箱和二次装箱在分别先去路网中各需求地的搬运花费成本如下表：

表 12 两种情况搬运花费成本对比

		搬运花费成本/元
传统一次装箱	先去 1 地	$1308+1024+799+517+293=3941$
	先去 2 地	$1308+1083+801+577+284=4053$
	先去 3 地	$1308+1026+802+509+225=3870$
	先去 4 地	$1308+1084+791+507+282=3972$
	先去 5 地	$1308+1015+731+506+224=3784$

搬运花费成本/元		
二次装箱	先去 1 地	936+1308+84+84+51+84+84=2631
	先去 2 地	2631
	先去 3 地	2631
	先去 4 地	2631
	先去 5 地	2631

注：传统一次装箱的搬运花费成本计算公式=遍历每个客户点所搬运小木块费用的总和。
 二次装箱的搬运花费成本计算公式=箱子（小母板）所花费成本+搬运小木块所花费成本之和+搬运箱子（小母板）所花费成本之和。

其中箱子（小母板）所花费成本以及各类型箱子所使用的数量由表 10 二次装箱策略的第一次装箱阶段求解结果得出，搬运箱子所花费成本为在某地需搬运箱子的数量与搬运箱子单价的乘积。

在运输成本一致的情况下，将表中二次装箱策略和传统的一次装箱搬运花费成本进行对比可知，在先去 1 地的情况下，二次装箱相比于一次装箱能节省 1310 元，在先去 2 地的情况下，二次装箱相比于一次装箱能节省 1422 元，在先去 3 地的情况下，二次装箱相比于一次装箱能节省 1239 元，在先去 4 地的情况下，二次装箱相比于一次装箱能节省 1241 元，在先去 5 地的情况下，二次装箱相比于一次装箱能节省 1153 元。因此，在先去路网中的任意一地的情况下，我们所提出的二次装箱策略都能够将传统的一次装箱进行了分类包装，方便管理人员对货物的辨识以及搬卸人员的搬运，并可以在一定程度上减少货物在搬运过程中的费用。

通过对比上述所有表格可知，在运输成本一致的情况下，我们提出的二次装箱策略在集装箱综合利用率上相较于传统的一次装箱而言可能会低一些，但此时两种情况所用的集装箱数量是相同的，即二次装箱策略也不会出现额外的运输成本支付，所以此时的集装箱综合利用率的高低就显得不是那么重要；并且在搬运花费的成本对比中发现，我们所提出的二次装箱策略可以对货物进行分类包装，能够在一定程度上减少货物在后续搬运过程中辨识工作进而降低搬运费用，从而减少企业物流在路网中各需求地的搬运成本。

4 结束语

本文以储存场切割原材料得到各种类型客户所需求的半成品后，再将其移交物流公司进行装箱操作这一完整的生产运输链条为研究对象，构建了基于列生成和分支定界算法求解的两阶段二次装箱模型。其中，我们将切割和装箱问题进行综合考虑，引入两阶段法对新模型进行求解。在得到储存场切割的半成品后，采用二次装箱策略对半成品进行分类打包、装载运输操作。二次装箱策略能够对货物进行分类打包处理，并在箱子上做上标记，记录箱子内装载货物的规格和数量位置，其还能够对货物进行保护，也便于管理员对货物的辨识以及各种装卸搬运人员或机械的工作。通过列生

成和分支定界算法验证了两阶段二次装箱各阶段模型的有效性。仿真结果表明, 我们所提出的二次装箱策略在运输成本一致的情况下, 能够在一定程度上减少货物在后续搬运过程中的辨识工作进而降低物流企业在路网中各需求地的搬运成本, 帮助物流企业在后续配送过程中提高工作效率。

未来的研究方向应更符合实际工业条件。对于二维平面切割问题, 可以就切割回收的残余材料进行深入讨论, 且本研究仅讨论 Guillotine 约束的切割模式, 例如瓷砖规划或者木材场切割木材, 钢铁厂裁剪钢铁。然而, 非 Guillotine 约束的切割模式如激光切割等许多切割问题也都值得进一步研究。对于二维平面的装箱问题, 二维不规则装箱问题的应用范围也很广泛, 且理论研究难度更大。因此, 建议后续研究人员继续探索上述或不规则形状切割及装箱中的二维问题, 此外对于如何进一步提高算法求解的准确性并提高收敛速度也是研究的重点。

参考文献:

- [1] YAO Yi. Research on Two-dimension Bin Packing Problem of Heuristic Algorithms [D]. Guangzhou: South China University of Technology, 2016. [姚怡. 二维装箱问题的启发式算法研究[D]. 广州: 华南理工大学, 2016.]
- [2] CHENG M Y, FANG Y C, WANG C Y. Auto-tuning SOS Algorithm for Two-Dimensional Orthogonal Cutting Optimization[J]. KSCE Journal of Civil Engineering, 2021, 25(10), 3605–3619.
- [3] Gilmore P, Gomory R. A linear programming approach to the cutting stock problem[J]. Operations Research, 1961, 9: 849–859.
- [4] Gilmore P, Gomory R. A linear programming approach to the cutting stock problem[J]. Operations Research, 1963, 11: 863–888.
- [5] Gilmore P, Gomory R. A Multistage cutting stock problems of two and more dimensions[J]. Operations Research, 1965, 13: 94–120.
- [6] JI Bin, ZHOU Saiqi, ZHANG Zheng. Bifurcated Pricing method for Vehicle routing Problem with two-dimensional packing constraints. Control theory and applications[J], 2021, 38: 1–10 (in Chinese). [季彬, 周赛琦, 张政. 分支定价方法求解带二维装箱约束的车辆路径问题[J]. 控制理论与应用, 2021, 38: 1–10.]
- [7] Alvarez-Valdes R, Parajon A, Tamarit JM. A computational study of LP-based heuristic algorithms for two-dimensional guillotine cutting stock problems[J]. OR Spectrum, 2002, 24(2): 179–192.
- [8] Cintra G F, Miyazawa F K, Wakabayashi Y, et al. Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation[J]. European Journal of Operational Research, 2008, 191(1): 61–85.
- [9] ZHANG Le. Research on two-dimensional cutting of steel plate[D]. Shenyang: Northeastern University, 2005. [张乐. 钢板二维切割问题的研究[D]. 沈阳: 东北大学, 2005.]
- [10] Belov G, Scheithauer G. A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting[J]. European Journal of Operational Research, 2006, 171(1): 85–106.
- [11] Melega G M, de Araujo S A, Jans R. Classification and literature review of integrated lot-sizing and cutting stock problems. European Journal of Operational Research[J]. 2018, 271(1): 1–19.
- [12] Caprara A, Toth P. Lower bounds and algorithms for the two-dimensional vector packing problem[J]. Discrete Applied Mathematics, 2001, 111(3): 231–262.
- [13] Martello S, Pisinger D, Vigo D. The three-dimensional bin packing problem[J]. Operations Research, 2000, 48(2): 256–267.

-
- [14] Vanderbeck F. Computational study of a column generation algorithm for bin packing and cutting stock problems[J]. Mathematical Programming, 1999,86(3): 565-594
- [15] Elhedhli S, Li L, Naoum-Sawaya J, et al. A branch-and-price algorithm for the bin packing problem with conflicts[J]. INFORMS Journal on Computing, 2011,23(3): 404–415.
- [16] REN Jiashuo. Design optimization algorithm and system for thick plate under mass customization [D]. Shenyang: Northeastern University, 2005. [任家硕. 大规模定制下中厚板母板设计优化算法及系统[D]. 沈阳: 东北大学, 2005.]
- [17] Pisinger D, Sigurd M. Using decomposition techniques and constraint programming for solving the two-dimensional bin-packing problem[J]. INFORMS Journal on Computing, 2007,19(1): 36–51.
- [18] Iori M, Salazar-Gonzalez J J, Vigo D. An exact approach for the vehicle routing problem with two-dimensional loading constraints[J]. Transportation Science, 2007,41(2): 253–264.
- [19] Lodi A, Martello S, Monaci M. Two-dimensional packing problems: A survey[J]. European Journal of Operational Research, 2002, 141(2): 241–252.
- [20] LIU Qian. Research on comparison and integration of rectangular optimal layout algorithms under constraint of guillotine [D]. Tianjin: Hebei University of Technology, 2012. [刘倩. “一刀切”约束下的矩形件优化排样算法比较与整合研究[D]. 天津: 河北工业大学, 2012.]
- [21] SHANG ZhengYang, GU JiNan, DING Wei, et al. Heuristic algorithm for 2D rectangle packing problem. Computer Integrated Manufacturing systems, 2018, 24(03): 583 – 590. [尚正阳, 顾寄南, 丁卫, 等. 求解二维矩形装箱问题的启发式算法[J]. 计算机集成制造系统, 2018, 24(03): 583-590.]
- [22] CAO Dayong, YANG Mei, KOTOV VM, et al. Two-stage heuristic algorithm for two-dimensional guillotine bin packing problem[J]. Computer Integrated Manufacturing systems, 2012,18(9):1954-1963. [曹大勇, 杨梅, 科托夫·弗拉基米尔·米哈伊拉维奇. 二维一刀切装箱问题的两阶段启发式算法[J]. 计算机集成制造系统, 2012,18(9):1954-1963.]
- [23] Iori M, de Lima VL, Martello S, et al. Exact solution techniques for two-dimensional cutting and packing[J]. European Journal of Operational Research, 2021,289(2): 399-415.

作者简介:

杨玉冰(1999-), 男, 山东菏泽人, 硕士研究生, 研究方向: 交通运输工程, E-mail: ybyang12138@163.com;
+杨松坡(1987-), 男, 河北廊坊人, 博士研究生, 研究方向: 城市多模式交通、城市轨道交通管理与优化、轨道交通能源管理与优化、智能优化算法等, 通讯作者, E-mail: s.yang1@bjut.edu.cn。