

Research on Product Cutting Problem Based on Two-Dimensional Planar Packing Model

Heng Zhang

Shandong University of Science and Technology, Qingdao Shandong
Email: suixinyingtian@163.com

Received: Aug. 26th, 2019; accepted: Sep. 9th, 2019; published: Sep. 16th, 2019

Abstract

Cutting stock problem refers to the way that a given product is reasonably placed in the layout space to achieve the highest utilization rate under the necessary constraints. This paper establishes a two-dimensional plane packing model based on the analysis of the utilization ratio of wood panels. Considering the profit maximization, a two-dimensional plane multi-objective packing model is established to solve the following problems: 1) solving the single product cutting problem by combining the "four-block" programming; 2) solving the double product cutting problem by using greedy algorithm; 3) using the optimal angle-occupying strategy to solve the problem of double product cutting which meets the requirement of task number; 4) solving the problem of multi-product cutting by using hybrid genetic algorithm (GA) and simulated annealing algorithm (SA).

Keywords

2-D Plane Packing Model, Genetic Algorithm, Greedy Algorithm

基于二维平面装箱模型的产品切割问题研究

张 衡

山东科技大学, 山东 青岛
Email: suixinyingtian@163.com

收稿日期: 2019年8月26日; 录用日期: 2019年9月9日; 发布日期: 2019年9月16日

摘 要

下料问题, 是指将给定产品合理地摆放在布局空间中, 在满足必要的约束条件下, 使利用率达到最高的

方式。本文在分析木板利用率的基础上建立二维平面装箱模型，考虑到利润最大化建立二维平面多目标装箱模型解决以下问题：1) 结合“4块法”规划解决单产品切割问题；2) 利用贪心算法解决双产品切割问题；3) 利用最优占角策略解决满足任务数要求的双产品切割问题；4) 利用混合遗传算法(GA)和模拟退火算法(SA)的HGASA算法，解决多产品切个问题。

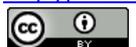
关键词

二维平面装箱模型，遗传算法，贪心算法

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

下料问题，又称为布局问题(Packing Problem)，是指给定一个布局空间和若干待下物体，将待下物体合理地摆放在布局空间中，满足必要的约束条件，并使布局利用率达到最高。下料问题可以指导制造企业的采购计划和生产计划。排样布局的优劣直接与材料成本及经济效益有关[1]。二维矩形条带装箱问题(2DR-SPP)是线性的组合优化问题，也是 NP-Hard 问题，求解难度较大，有重要的研究意义。该问题在计算机辅助设计、图像处理、大规模集成电路逻辑布线设计等领域有广泛应用，有较高的实际使用价值。构建 2DR-SPP 的简单数学模型，即可实现对中小规模问题的有效求解，又可为研究求解大规模问题的快速高精度的数学规划启发式算法奠定基础有重要的研究价值[2]。

P. Yongyingprasert 和 S. Vasupongayya 在处理集群作业调度问题时将其转化为一组二维框打包到大局形容其中的问题，旨在应用二维盒打包算法批处理集群作业调度问题[3]。Lam T. Bui、Stephen Baker 和 Axel Bender 为了解决二维(2D)矩形包装问题，引入了一种基于容器的离散化为具有预定义分辨率的单元网格，在添加项目之前，将检查网格单元格是否可以容纳该项目，如果找到合适的空单元群，则添加该项并将其移向容器的左下角。通过将启发式算法与遗传算法(GA)杂交，可以改善项目的顺序和轮换[4]。Omar, Mohamed K 和 Ramakrishnan, Kumaran 在考虑无取向的二维料仓包装问题涉及一组需要装入相同矩形料箱的矩形件提出了进化粒子群优化算法(EPHO)来解决非定向二维装箱问题来进行广泛的数值研究以确定最有解决方案[5]。在本文中总结已有研究工作的基础上，通过分析切割问题的数学模型和基本属性，说明切割问题属于 NP 完全问题，结合实际应用中的智能算法来逼近最优解。

2. 模型假设

- 1) 假设木板厚度和割缝宽度忽略不计。
- 2) 假设木板厚度与产品厚度一致，即可将其转换成二维装箱问题。
- 3) 假设生产任务完成后对多余的模板不再进行处理，即恰好完成生产任务。
- 4) 假设在同一方案下每块木板的切割方式相同，且不会产生误差或可忽略。

3. 符号说明

对于本文所用的符号，这里只列出通用符号，具体符号在引用时会进行具体说明。

符号	含义
η	单一木板利用率
N	所用木板数量
L	木板长度
W	木板宽度
l_i	需要切割的小木板长
w_i	需要切割的小木板宽
ξ	木板总利用率
n_i	切割木板数量

4. 模型的建立与求解

4.1. 单产品切割问题

4.1.1. 木板利用率的计算

➤ 简介

木板利用率的高低是影响企业经济效益的主要因素之一，随着市场竞争的加剧和木板价格的上涨，应该对切割提出更科学的方法，不是简单的“可以切”，而是要“切得块”、“切得好”、“切得省”，通过模板下料的特点提高木板利用率，降低切割成本，从而能为企业获得更多的经济效益

木板利用率越高，即剩余木板面积最小，获得经济效益越高，反之，越低。

➤ 公式

设木板面积为 S ，产品面积为 $s_i (1, 2, \dots, n)$ 。

则某一块木板的利用率为：

$$\eta = \frac{S_i}{S} \times 100\% \quad (1)$$

其中， η 表示单一木板利用率， N 表示所用木板数量， n 表示切割的产品数量。

4.1.2. 平面二维装箱问题“4块法”规划求解模型

➤ “4块法”摆放方式介绍

“4块法”就是把二维平面分成4个区域摆放或放置产品，即A、B、C和D四个区域，设置木板长为 L ，宽为 W ，产品长为 l ，宽为 w ，A区有 X_1 和 Y_4 构成，B区有 X_2 和 Y_1 构成，C区由 X_4 和 Y_3 构成，D区由 X_3 和 Y_2 构成。 X_i 为产品长在木板长或宽方向上摆放个数， Y_i 为产品宽在木板长或宽方向上摆放个数。“4块法”摆放方式如图1所示：

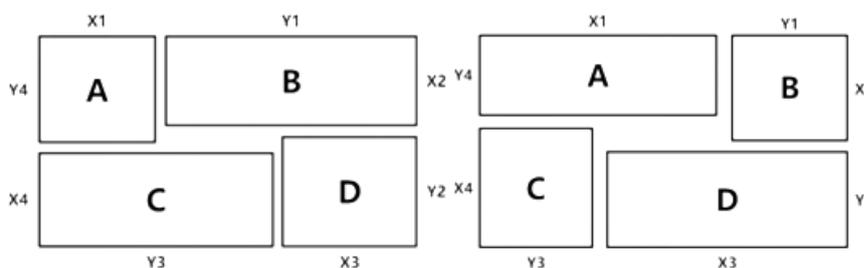


Figure 1. Two layout drawings of “Four blocks method”

图1. “4块法”摆放的两种布置图

► 典型的“4块法”数学模型

“4块法”摆放布置可分为2中类型，一是B和C横向长度之和大于木板长度，此时可能会A和D对角发生矛盾或B和C对角发生矛盾；二是A和D横向程度之和大于木板长度，此时可能会出现B和C对角发生矛盾或A和D对角发生矛盾，矛盾如图2所示。

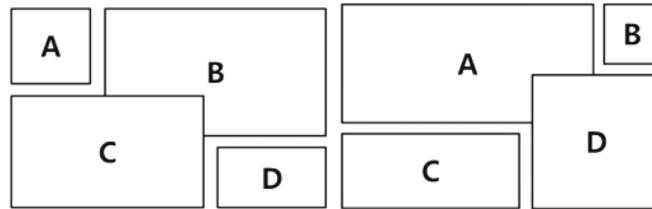


Figure 2. Layout of two contradictory situations
图2. 发生矛盾的两种情况布置图

为了解决这两种矛盾情况，建立以下模型：

目标函数：

$$\max Z = X_1 \times Y_4 + X_2 \times Y_1 + X_3 \times Y_2 + X_4 \times Y_3 \quad (2)$$

约束条件：

$$\begin{cases} X_1 + X_3 \geq N_1 \\ X_2 + X_4 \geq N_2 \\ Y_1 + Y_3 \leq N_3 \\ Y_2 + Y_4 \leq N_4 \\ X_1 \times l + Y_1 \times w \leq L \\ X_2 \times l + Y_2 \times w \leq L \\ X_3 \times l + Y_3 \times w \leq W \\ X_4 \times l + Y_4 \times w \leq W \\ X_i \in N (i = 1, 2, 3, 4) \\ Y_i \in N (i = 1, 2, 3, 4) \end{cases}$$

其中， $N_1 = \left\lceil \frac{L}{l} \right\rceil, N_2 = \left\lceil \frac{L}{l} \right\rceil, N_3 = \left\lfloor \frac{L}{l} \right\rfloor, N_4 = \left\lfloor \frac{L}{l} \right\rfloor$ 对于 $\lceil \cdot \rceil$ 表示取整符号。

4.1.3. “4块法”规划模型求解

通过 Lingo 对该线性规划进行求解得到结果如图3所示：

► 结果分析

通过 Lingo 的求解我们得到 $X_1 = 1, X_2 = 4, X_3 = 7, X_4 = 0, Y_1 = 13, Y_2 = 0, Y_3 = 1, Y_4 = 7$ ， $\max Z = 1 \times 7 + 4 \times 13 + 7 \times 0 + 0 \times 1 = 59$ 在增加的四个约束条件下很好地解决了对角矛盾问题，在利用率最高的要求下，实现了木板的充分利用，其利用率为 98.3%，具体如表1所示：

Table 1. The result of single product cutting problem
表1. 单产品切割问题的结果

P1 的数量	木板利用率
59	98.3%

```

Local optimal solution found.
Objective value:                59.00000
Objective bound:                59.00000
Infeasibilities:                0.000000
Extended solver steps:          88
Total solver iterations:        518
Elapsed runtime seconds:        0.11

Model Class:                    PIQP

Total variables:                8
Nonlinear variables:            8
Integer variables:              8

Total constraints:              9
Nonlinear constraints:          1

Total nonzeros:                24
Nonlinear nonzeros:            4

Variable      Value      Reduced Cost
X1            1.000000   -6.999999
Y4            7.000000    6.000000
X2            4.000000  -13.00000
Y1           13.00000   -3.999999
X3            7.000000  -0.100000E-05
Y2            0.000000    0.000000
X4            0.000000  -0.9999990
Y3            1.000000    0.000000

```

Figure 3. Lingo solution of “four block method”
图 3. “4 块法” Lingo 求解结果

4.2. 双产品切割问题

4.2.1. 组合方式的确定

在进行归纳推理时，通过枚举法我们逐个考察某类方案的所有可能发生值，并根据条件判断答案是否合适，保留合适的，删去不合适的，求解并验证。

Step 1: 确定枚举对象

首先，通过单产品切割问题中的模型，在考虑到木板利用率最高的模型下，该木材只切割 P1 可以切割 59 块，同样的，该木材只切割 P3 可以切割 48 块，为减少枚举的次数，我们以 P3 的数量作为枚举对象。

Step 2: 确定枚举范围

对于产品 P1 的范围为[0, 59]，产品 P2 的范围为[0, 48]，且根据实际情况我们知道，产品 P1 和产品 P2 的数量都为整数。

Step 3: 确定判定条件

对逐一枚举的可能的解要满足二维多产品装箱模型，如果满足，继续进行 Step 4,如果不满足，将 P1 的数量减一输出并返回到 Step 2。

Step 4: 计算木板利用率

通过公式(1)对木板利用率进行求解，并比较结果大小，依次选取最大的结果进行填表。

4.2.2. 建立二维多产品装箱模型

二维多产品装箱问题，是指一系列二维平面图形互不交叉的放置在某一板材上，使得所覆盖的板材面积最大，对于企业而言，板材的利用率越高，越节约材料，所用成本越少，对提高企业经济效益起到

关键作用。

► 确定优化目标

- 对于企业来说，为使经济效益达到最高，要使木材利用率达到最高，即生于模板面积最小。
- 对于模板来说，产品 P1 和 P2 的组合要能在一张木板上实现。

► 确定排放原则

将所需产品排放在定长定宽的木板中，产品的长或宽与木板平行，产品之间不可交叉，且不得超过木板的长或宽。

► 二维多产品装箱模型的建立

假设木板的长度为 L ，宽为 W ，面积为 S ，需要切割的小板材长为 l_i ，宽为 y_i ， i 表示第 i 种需要切割的小板材， $i=1,2,3,4$ ，板材剩余面积为 A_{real} 。当零件横放时占用的空间为 h_i 竖放占用的空间是 k_i 。 $l_i h_i$ 表示在原料板上第 j 块区域内横放零件的行数； $w_i k_i$ 表示板材上第 j 块区域内小板材竖放的行数。 n_j 表示板材上第 j 块区域内零件横放时每行的个数； m_j 表示大板材上第 j 块区域内小板材竖放时每行的个数，则：

目标函数：

$$\max \xi = \frac{\sum_{i=1}^n (n_i + m_i) \times s_i}{\sum_{i=1}^N S_i} \quad (4)$$

限制条件：

$$\begin{cases} L - \sum_{j=1}^4 (n_j + m_j) \geq 0 \\ K - \sum_{j=1}^n (l_j h_j + w_j k_j) \geq 0 \\ w_{\min} - (n_1 l_1 h_1 + m_1 w_1 k_1) \geq 0 \\ l_{\min} - (n_2 l_2 h_2 + m_2 w_2 k_2) \geq 0 \\ m_3 w_3 k_3 + m_3 w_3 k_3 - w_{\max} \geq 0 \\ n_4 l_4 h_4 + m_3 w_2 k_2 - j_{\max} \geq 0 \end{cases} \quad (5)$$

4.2.3. 基于贪心算法求解二维装箱问题

► 贪心策略的基本思想

定义：贪心法是一种解决最优问题的策略。它是从问题的初始解出发，按照当前最佳的选择，把问题归纳为更小的相似的子问题，并使子问题最优，再由子问题来推导出全局最优解。

使用贪心方法需要注意局部最优与全局最优的关系，选择当前状态的局部最优并不一定能推导出问题的全局最优。

► 贪心算法求解的三个步骤

Step 1: 确定贪心策略；

Step 2: 根据贪心策略，迭代得到局部最优解；

Step 3: 将局部最优解合并起来就得到全局最优解。

► 贪心策略的特点

- 贪心选择性质：所求问题的整体最优解可以通过一系列局部最优的选择来达到，这样的选择称为贪心选择。这些选择只依赖于以往所做过选择，决不依赖于将来的选择。

- 最优子结构性性质：当一个问题最优解包含其子问题的最优解时，称此问题具有最优子结构性性质。

4.2.4. 结果分析与结论

通过枚举得到相关组合为如表 2 所示：

Table 2. Enumeration combination scheme

表 2. 枚举组合方案

P1 数量	P2 数量	木板利用率/%	P1 数量	P2 数量	木板利用率/%
0	48	99.17	30	22	95.44
1	47	97.11	32	21	96.70
1	46	96.71	33	20	96.30
3	45	97.97	35	19	97.57
4	44	97.57	36	18	97.17
5	43	97.17	37	17	96.77
6	42	96.77	39	16	98.03
8	41	98.04	40	14	95.57
9	39	95.57	41	13	95.17
12	38	98.50	44	12	98.10
13	36	96.04	45	11	97.70
14	35	95.64	46	10	97.30
16	34	96.90	47	9	96.90
18	32	96.10	49	8	98.17
20	31	97.37	50	7	97.77
21	29	94.90	51	6	97.37
22	28	94.50	52	5	96.97
24	27	95.77	54	4	98.23
26	26	97.04	55	3	97.83
27	25	96.64	56	2	97.43
28	24	96.24	57	1	97.03
29	23	95.84	59	0	98.30

➤ 结果分析

通过对上述结果进行分析，我们可以知道在一块木板上切割 P1 和 P3 产品的优化组合有 44 种，且满足二维多产品装箱模型，通过贪心算法对二维多产品装箱模型进行求解得到相关组合方式，得出给出按照木板利用率由高到低排序的前 3 种切割方案，利用率由高到低的排列为：99.17%、98.50%以及 98.30%，具体结果如表 3 所示：

Table 3. Result of multi-product cutting problem results of multi-product cutting problem

表 3. 多产品切割问题的结果

方案编号	P1 的数量	P3 的数量	木板利用率
1	0	48	99.17%
2	12	38	98.50%
3	59	0	98.30%

4.3. 满足任务数要求的双产品切割问题

4.3.1. 总利用率的计算

► 简介

木板利用率的高低是影响企业经济效益的主要因素之一，随着市场竞争的加剧和木板价格的上涨，应该对切割提出更科学的方法。在实际生产中，往往不是对单一木板进行切割，而是在不同组合方式的切割下要求总利用率尽可能高。

木板总利用率越高，即剩余木板面积最小，获得经济效益越高，反之，越低。

► 公式

设木板面积为 S ，产品面积分别为 s_1, s_2, s_3, s_4 。

则木板总利用率为：

$$\xi = \frac{s_1 * n_1 + s_2 * n_2 + s_3 * n_3 + s_4 * n_4}{S * N} \times 100\% \quad (6)$$

其中， ξ 表示木板总利用率， N 表示所用木板数量， n_i 表示切割的产品数量。

4.3.2. 构建基于生产任务下的二维装箱模型

因此，矩形件套料问题的数学模型如下：

目标函数：

$$\max f(\eta) = \frac{S_n}{S_{\text{大木板}}} = \frac{\sum_{i=1}^R w_i l_i}{W \cdot L} \quad (7)$$

约束条件：

$$x_i + w_i \leq W, \forall i \in n \quad (8)$$

$$y_i + l_i \leq L, \forall i \in n \quad (9)$$

$$(x_i + w_i \leq x_j, \text{ or } x_j + w_j \leq x_i) \text{ or } (y_i + l_i \leq y_j, \text{ or } y_j + l_j \leq y_i), \forall (i, j) \in n, i \neq j \quad (10)$$

$$x_i + y_i \geq 0, \forall i \in n \quad (11)$$

其中，(8)、(9)表示所有木板均能在大木板完整割取，没有超出大木板的边界；式(10)表示各木板之间互不重叠；式(11)表示所有矩形件虽大小不一，但都有一定的面积。所以，最好的套料结果就是各木板之间既不重叠也无空隙，即

$$\sum_{i=1}^n w_i l_i = W \cdot L \quad (12)$$

4.3.3. 最左下占角动作放置策略

► 基本思想

在某一时刻，已经按分割规则把大木板矩形分了若干块，那么对还未分割的部分，最好是占据某个角，其次是贴边，最差的是悬空。我们采用的算法即在分割大木板时，总是先分某个角，这样分割就可减小对空间的浪费。

同时，也避免了基于遗传算法的 BL 启发式策略中当较大矩形产品在连续移动时受阻，会在模板中产生许多空区域以及产品可与之前排好的产品分别形成一个角区的缺点。

► 基本概念

① 格局：在某一时刻，大木板已经互不重叠地分割了若干木板，这种状态，称为一个格局。偌大木

板还未分割，此时的格局称为初始格局；若大木板无法再分割，此时的格局称为终止格局。

② 占角动作：在某一格局下，若即将分割的木板 R 与大木板剩余部分的边有重叠(并且重叠的长度大于 0)，则称分割木板的这种作法为一个占角动作。

③ 两矩形块间的距离：在两木板 R_i 和 R_j ，中各取一点 $D_i \in R_i$ ， $D_j \in R_j$ ，设这两点间的曼哈顿距离为 L_{ij} ，则两木板间的距离 $l = \min_{D_i \in R_i, D_j \in R_j} (L_{ij})$ 。

④ 占角动作的穴度：如图 4 所示，若将木板 R_i 按合法占角动作分割走后， R_i 与形成这个角以外的其它部分的距离的最小值为 l_{\min} ， w_i 和 L_i 分别为木板的宽和长。则此占角动作的穴度 C_i 定义为

$$C_i = 1 - \frac{l_{\min}}{\sqrt{w_i \cdot L_i}} \quad (13)$$

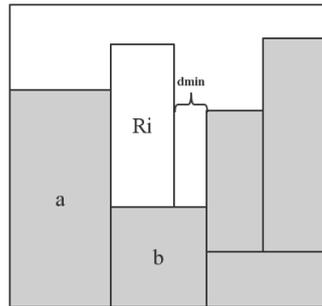


Figure 4. Points of angular action
图 4. 占角动作的穴度

占角动作的穴度反映了即将分割的木板和大木板已有分走的抱紧程度，穴度越大，就抱得越紧，在选择占角动作时，就选穴度最大的占角动作。

⑤ 占角动作的贴边数：设某一占角动作所关联的木板为 R ，那么与 R 贴边的木板块数称为该占角动作的贴边数。

► 占角动作选择原则

为了将木板在大木板中一个“好”的位置和方向上割取，则必须制定一些好的规则，BLCO 放置策略中的主要规则如下：

规则 1：选择最向下向左的动作。

规则 2：选择对应动作矩形件占领角区面积最大的占角动作。

规则 3：选择对应动作矩形件面积最大的占角动作。

规则 4：若对应动作矩形件既可横放，也可竖放，则选择横放的占角动作。

规则 5：选择对应动作矩形件序号最小的占角动作。

BLCO 策略的核心体现在规则 1 和 2 上，即选择最向下向左的动作和选择角区面积最大的占角动作，若规则 1 和 2 无法再起作用，则采用规则 3、4 和 5 打破僵局，从而选出唯一的占角动作。

4.3.4. 全局搜索的 GA 算法

► 简介

遗传算法(Genetic algorithm, GA)是智能优化方法中应用最广泛的算法，也是目前发展最为成功的智能算法之一。GA 算法是由 Holland 教授 1970 年代提出来的，其中最为重要的就是模式定理(schema theorem)，所借鉴的生物学基础就是生物的进化和遗传。下面从 GA 算法的数学基础与收敛理论两个方面进行描述，并总结出 GA 算法的特点、不足及改进方法。

► 原理

① 染色体编码与解码：染色体编码是遗传算法最关键的步骤，其是一种可以将实际待优化问题的可行解转化到遗传算法可以处理的搜索空间中的一种方法。

② 种群初始化：随机产生 N 个染色体作为初始种群，遗传过程中的收敛速度会受到初始种群大小的影响，因此 N 的取值应选择适当，本文 N 选取 60。

③ 适应度函数：适应度函数指引着遗传算法的搜索方向，适应度函数的选择应该根据具体的问题而定，本研究中适应度函数为木板总利用率，即为目标函数。

④ 遗传算子

- 选择算子：轮盘赌选择算法是遗传算法中最常用的选择算法，其根据群体中每个染色体的适应值得到群体所有染色体的适应值总和，并分别计算每个染色体适应值与群体适应值总和的比；较优染色体的值较大，被选择的概率就相对较大。
- 交叉算子：交叉操作是遗传算法中的核心操作步骤，是种群中产生新个体的主要途径之一。一般情况下，设置交叉概率为 0.4~0.9 之间。
- 变异算子：在遗传算法中，产生新一代的个体主要依靠的是选择和交叉，变异操作是一种产生多样性个体的辅助手段，变异算子设计的好坏能决定算法的局部搜索能力。变异概率较小，一般设为 0.0001~0.1 之间。

在遗传算子的操作下被选择、交叉与变异，形成高阶的、高于群体平均适应值的模式(较优解)，最终接近全局最优解。所以，GA 算法是一种基于自然法则和遗传机制的智能优化算法，并且基于群体进行搜索，其基本流程如图 5 所示。

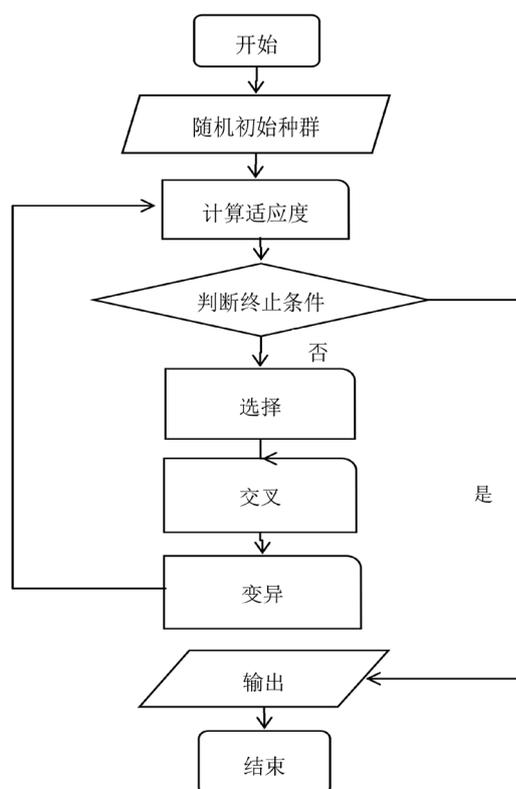


Figure 5. The basic flow chart of the algorithm GA
图 5. GA 算法基本流程图

从上图可知, GA 算法从随机产生的初始种群开始搜索, 分别计算每个个体的适应值, 进而判断个体的优劣, 通过选择算子选出较优秀的个体, 通过交叉运算和变异运算产生后代, 再重新计算每个新个体的适应值, 从而判断是否收敛到最优解。如果满足终止条件, 则搜索到最优个体。

4.3.5. 模型的求解与结果分析

基于多产品切割问题中对 P1 和 P3 排列组合的 44 种方式, 通过遗传算法(GA)对全局最优解进行搜索, 在木板总利用率最高的目标前提下, 从初代种群开始搜索, 分别计算每个个体的适应值, 进而判断个体的优劣, 通过选择算子选出较优秀的个体, 通过交叉运算和变异运算产生后代, 再重新计算每个新个体的适应值, 从而判断是否收敛到最优解, 从而搜索得到最优个体, 其组合结果如图 6 所示:

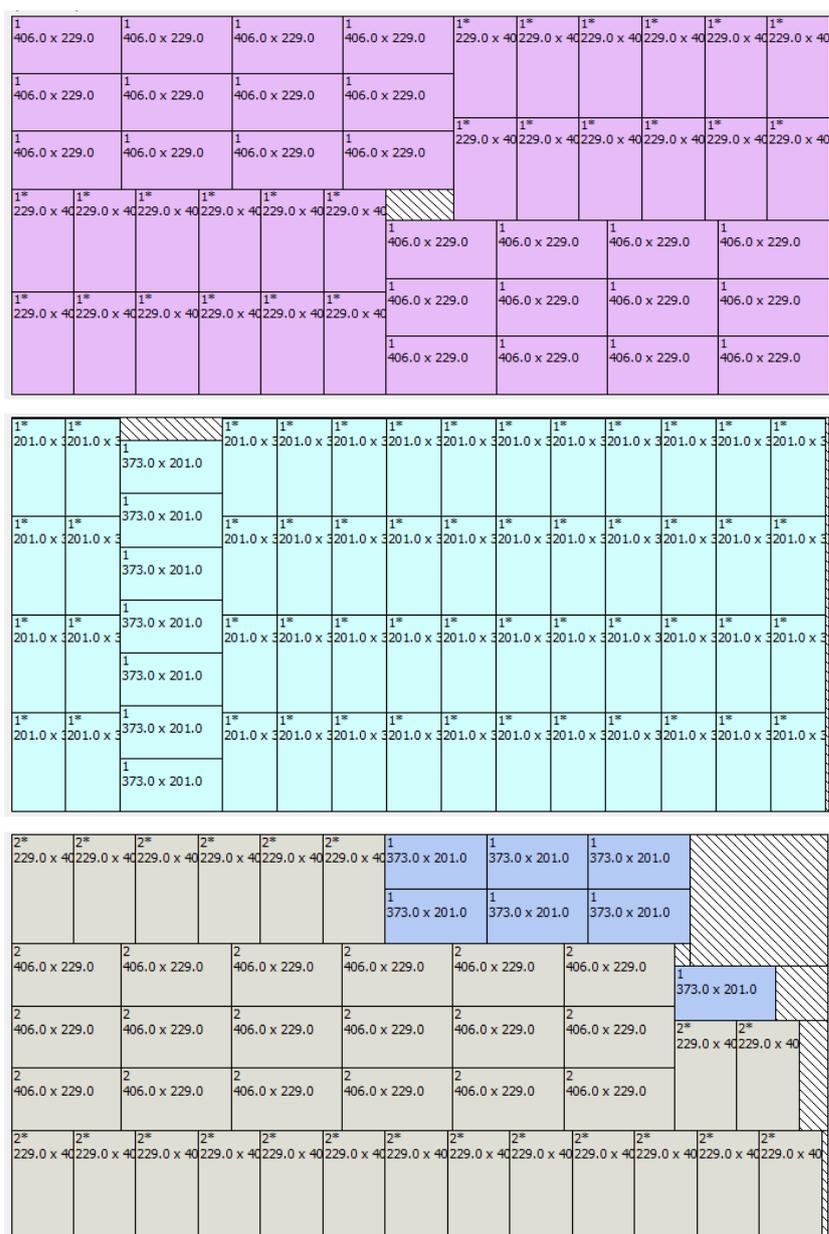


Figure 6. Optimal cutting scheme for multiproduct cutting problem

图 6. 多产品切割问题的切割最优方案

► 结果分析

通过对上述结果进行分析, 为完成 P1 和 P3 的生产任务, 加入生产任务的限制条件, 我们得到木板利用率最高的前提下的三种排列组合方案, 每种组合只需用一块木板, 且第一块木板利用率为 99.17%, 第二块木板的利用率为 98.29%, 第三块模板的利用率为 92.23%, 得出总利用率, 具体结果如表 4 所示。

Table 4. Result of double product cutting problem meeting task number requirements

表 4. 满足任务数要求的双产品切割问题的结果

木板 S1 的数量	P1 的数量	P3 的数量	木板利用率
33	0	48	99.17%
13	59	0	98.29%
1	7	39	92.23%
合计数量: 47	774	1623	木板总利用率: 98.78%

4.4. 多产品切个问题

4.4.1. 多产品切割模型的准备

遗传算法在小范围内求解全局最优解具有效率高、准确性高的特点, 但是对于大范围的求解显示遗传算法的效率不那么明显, 在满足任务数要求的双产品切割问题中只有 44 种排列组合方式, 运用遗传算法(GA)可以满足效率要求, 但是对于四种产品的组合数量 5355312 ($59 \times 31 \times 48 \times 61$)显然已经不满足要求, 在此基础上通过分析遗传算法(GA)和求解局部最优解所用方法模拟退火算法(SA)的缺点和不足, 提出一种混合 GA 算法和 SA 算法的 HGASA 算法。

4.4.2. 混合 GA 算法和 SA 算法的 HGASA 算法

► 简介

HGASA 算法: 借鉴 Memetic 算法的思想, 混合 GA 算法和 SA 算法的 HGASA 算法的一种新算法, 其最主要的思想就是融合了 GA 算法的全局搜索 Negligence 和 SA 算法的局部搜索能力。

► 原理

矩形套料中, 产品的套料次序和各自选装角度是影响套料切割的重要因素。采用 HGASA 算法, 通过全局优化概率产生最佳的套料次序和每个举行见的旋转角度, 然后采用 BLOC 放置策略进行排放和定位, 再通过局部搜索过程, 循环往复, 最终实现自动套料, 其基本流程如图 7 所示:

4.4.3. 模型的求解与结果分析

基于遗传算法(GA)改进的混合 GA 算法和 SA 算法的 HGASA 算法能够更加高效的找到四种产品的最优排列组合, 该算法首先通过全局优化概率产生最佳矩形件排列顺序和每个矩形件对应的旋转角度, 然后利用 BLOC 放置策略进行排放和定位, 并计算其适应函数值, 再利用 GA 算法对当前方案进行交叉变异, 执行最优方案的局部搜索过程, 循环迭代直到满足终止条件为止, 最终实现对最优切割方案的求解, 因篇幅所限此处只列出一种排版方案, 如图 8 所示:

► 结果分析

通过对上述结果进行分析, 为完成四种产品的生产任务, 我们得到木板利用率最高的前提下的排列组合方案, 方案 1 需要 41 块木板, 每块木板的利用率为 98.51%; 方案 1 需要 41 块木板, 每块木板的利用率为 98.51%; 方案 2 需要 5 块木板, 每块木板的利用率为 98.29%; 方案 3 需要 1 块木板, 每块木板的利用率为 98.10%; 方案 4 需要 11 块木板, 每块木板的利用率为 97.55%; 方案 5 需要 71 块木板, 每块木

板的利用率为 96.86%；方案 6 需要 1 块木板，每块木板的利用率为 96.53%；方案 7 需要 1 块木板，每块木板的利用率为 95.06%；方案 8 需要 8 块木板，每块木板的利用率为 92.66%；方案 9 需要 1 块木板，每块木板的利用率为 76.40%，进而得出总利用率，具体结果如表 5 所示：

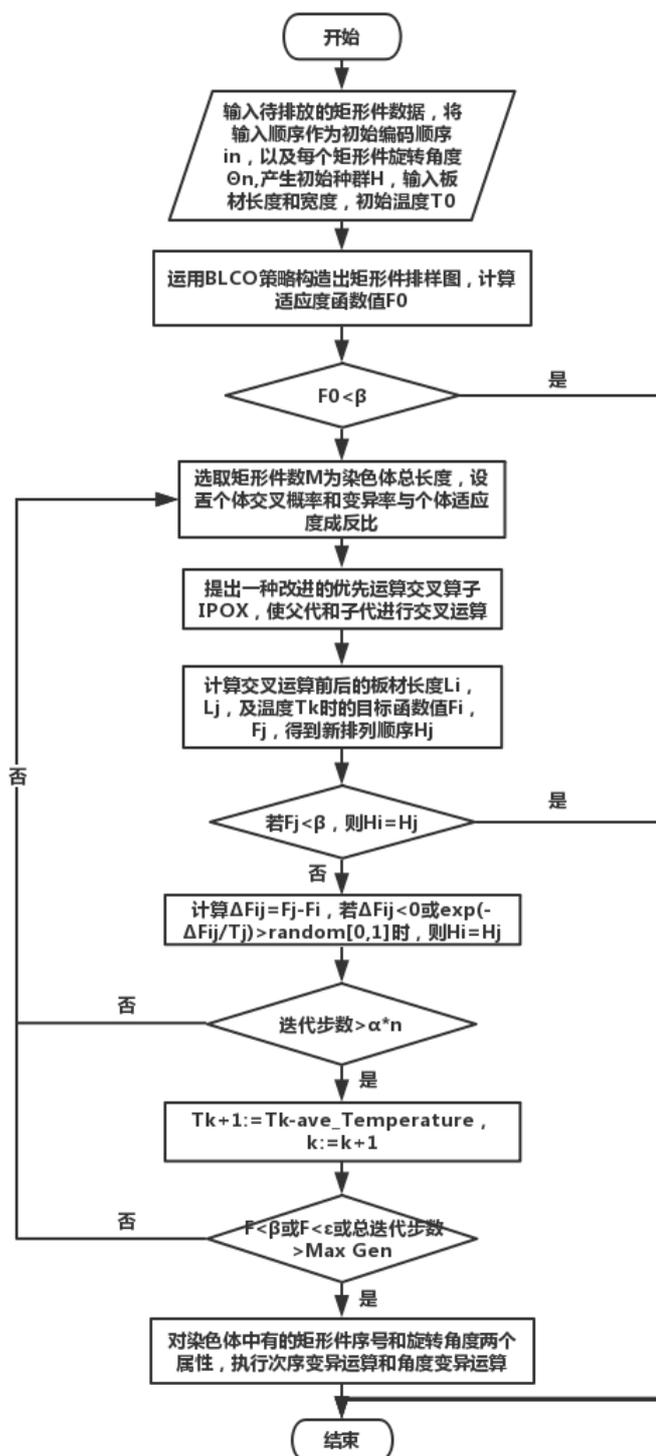


Figure 7. Flow chart of algorithm HGASA

图 7. HGASA 算法流程图

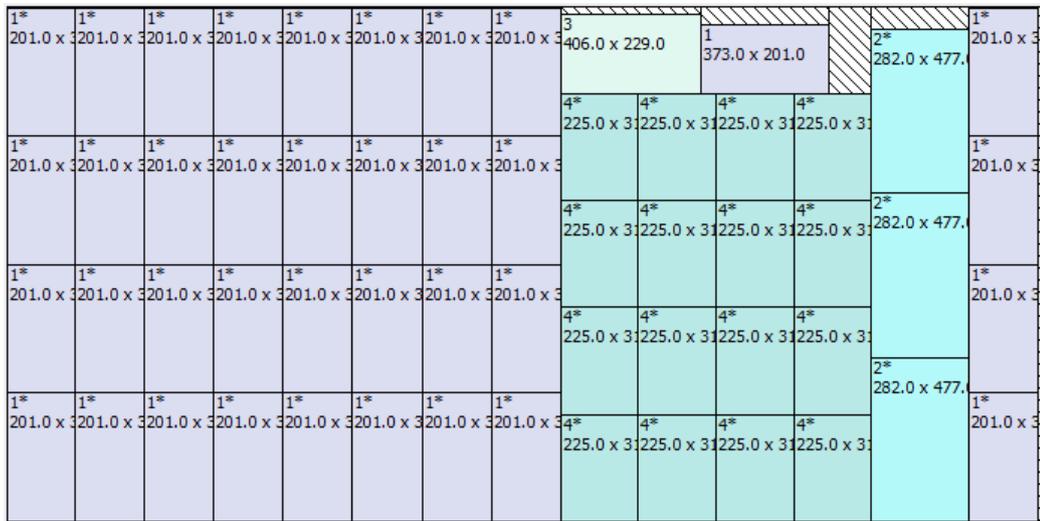


Figure 8. Optimal cutting scheme for multiproduct cutting problem
图 8. 多产品切割问题的切割最优方案

Table 5. Result of multi-product cutting problem
表 5. 多产品切割问题的结果

木板 S1 的数量	P1 的数量	P2 的数量	P3 的数量	P4 的数量	木板利用率
41	0	6	39	0	98.51
5	59	0	0	0	98.29
1	32	6	13	0	98.10
11	37	3	1	16	97.55
71	0	22	0	20	96.86
1	40	10	0	0	96.53
1	0	24	0	15	95.06
8	0	31	0	0	92.66
1	0	24	0	3	76.40
合计数量: 140	774	2153	1623	1614	木板总利用率: 97.06%

5. 模型的评价与推广

5.1. 模型的评价

5.1.1. 模型的优点

- 1) 本文所运用的混合 GA 算法和 SA 算法的结合了 GA 算法和 SA 算法的优点, 在 SA 算法的基础上那个找到局部最优解, 通过 GA 算法得到全局最优解, 能够快速高效的得到方案在整体上的全局最优解。
- 2) 在选择组合方案时, 通过 MATLAB 对与单一产品的利润进行对比, 在利润最大化的基础上对组合方案进行判定, 对模型进行简化, 大大减少了运行双目标二维平面装箱模型的时间。
- 3) 在“4 块法”摆放的对角矛盾中, 通过增加限制条件, 解决对角矛盾。

5.1.2. 模型的缺点

在“4 块法”摆放中为解决对角矛盾增加的限制条件, 缩小了解决范围, 影响力全局优化程度。

5.2. 模型的推广

在三维集装箱装载问题中, 设集装箱的尺寸分别为 L (长)、 W (宽)、 H (高), 同时其最大装载容积和最大装载质量分别用 V 和 G 表达。待装载的货物数量设为 n , 其对应的尺寸参数与质量分别用 l_i, w_i, h_i, g_i 表示。

在关于装箱问题的描述中, 装载利用率受各种复杂因素的影响与约束, 根据现在关于装箱问题的研究, 本文对复杂的影响因素总结如下:

- 货物放置方向约束: 即货物在装载过程中受到装载方向的约束。
- 货物装载容积约束: 即货物总体积不能超过集装箱的最大装载容积。
- 货物装载质量约束: 即货物在装载过程中其总质量不能超过集装箱的最大允许装载质量。
- 货物的装载顺序约束: 即须根据一定的优先性来装载货物, 例如先进后出。
- 稳定性约束: 即货物的重心必须保持在一定的范围内, 保证货物的稳定性。

参考文献

- [1] 郑超越, 冯爱兰. ERW 焊管二维下料指导采购计划的研究[J]. 物流技术, 2010, 29(z2):125-128.
- [2] 元晓莹. 二维矩形条带装箱问题的左下角定位模型[D]: [硕士学位论文]. 武汉: 武汉科技大学, 2018.
- [3] Yongyingprasert, P. and Vasupongayya, S. (2014) Evaluating a Two Dimensional Box Packing Algorithm on Batch Processing Cluster Job Scheduling Problem. 2013 *International Computer Science and Engineering Conference (ICSEC)*, Nakorn Pathom, 4-6 September 2013, 79-84. <https://doi.org/10.1109/ICSEC.2013.6694757>
- [4] Bui, L.T., Baker, S., Bender, A., *et al.* (2011) A Grid-Based Heuristic for Two-Dimensional Packing Problems. 2011 *IEEE Congress of Evolutionary Computation (CEC)*, New Orleans, LA, 5-8 June 2011, 2329-2336. <https://doi.org/10.1109/CEC.2011.5949905>
- [5] Omar, M.K. and Ramakrishnan, K. (2011) EPSO for Solving Non-Oriented Two-Dimensional Bin Packing Problem. 2011 *IEEE International Conference on Industrial Engineering and Engineering Management*, Singapore, 6-9 December, 2011, 106-110. <https://doi.org/10.1109/IEEM.2011.6117888>