

# 印章制作切割优化方案设计

## 摘要

印章是人们生活工作中常用的物品，印章的形状各种各样，厚度，尺寸大小不一，印章切割属于三维背包问题，是 NP 完全问题，对于减少原材料浪费、提高原材料利用率，优化下料方案的设计有着十分重要的意义。

对于问题一，要在一块原材料 S 上切割 P5 产品，给出原材料利用率最高的切割方案。首先我们假设 P5 模型水平或竖直放置，将问题转化为在二维矩形面上放椭圆形件的问题，建立横纵向随机切割模型，对不同的长、宽、高进行遍历取最优解，求得原料利用率为 77.97%，利用率较低，因此对模型进行了优化，椭圆以交错的形式排放，建立了见缝插针式模型，此原料利用率为 88.72%，P5 数量为 37236 件，排样效果较好。在此基础上对余料加以利用，最终原材料可切割 37496 件产品，最优利用率为 89.34%，最后做灵敏度分析时，分别改变原材料和产品的尺寸，其中利用率上下波动幅度不超过 0.8%，且厚度的改变对利用率影响较小，说明模型较稳定。

对于问题二，要在一块原材料 S 上切割 6 种产品，给出 S 利用率较优的 5 种切割方案。首先，我们观察到高度可以被最大化利用，同样我们将问题转化为在 4550\*1630 平面上的排料问题。然后我们分块处理不同形状的产品，对于圆形件，使用较为成熟的 ASA 算法，对于椭圆形件，使用问题一中的见缝插针式算法，通过改变分块的横纵坐标得出前 5 种方案的利用率，结果见表 4。由于运算时间过长，对模型做了进一步优化，采用剪冲式排样模型，对原料进行剪冲式切割分块，对每块小矩阵进行见缝插针式排样，得出最优利用率为 89.46%，89.33%，89.30%，89.28%，89.26%，结果较优且数量分布比较均匀，运行速度快。

对于问题三，需要完成六种产品的生产任务，给出原材料利用率最高 5 种方案，并求 S 的数量。因切割 6 种产品，故可以沿用问题二剪冲式模型随机生成了约两万种方法，我们定义了方法搜索的优先级，对方法进行筛选，使之满足产品的生产需求。最终结果：5 种切割方案下，需要 S 原材料 281 块，原料利用率为 88.71%，结果较优。得到最优解之后，我们对 6 种产品的需求、原材料以及产品的尺寸做了灵敏度分析，各个数值的改变使利用率在 1.5% 以内围绕某个值上下波动，故模型效果较好。

对于问题四，在问题三的基础上，新加了三个不同规格的长方体，由于厚度不同所以无法分层考虑排样问题。因为各材料的厚度都可以把高度整除，所以我们沿用了问题三的计算方法，修改并构建了新的程序，再次使用剪冲式下料方法，把三维不同规格的装箱问题转化为二维排件问题，生成两万种方案，沿用第三问的方案选择优先级，对符合要求的优先级最高的方案进行挑取最终结果为：需要原材料 463 块，原材料利用率为 90.62%，模型利用率结果较优。

对于问题五，100 块原材料中挑选出最好的生产方案，首先，这个问题的全局最优解(100 块)的局部(1 块)也为最优解，所以我们将问题转化为求解 1 块原材料的利润最大值，然后对问题四中方案库构造利润函数，得出性价比最高的方案为全部切割产品 P6，其数量为 420.54 万件，原材料利用率为 89.07%，最大利润为 6569.72 万元，原料的利用率较高，模型具有一定的准确性。

最后，我们对建立的模型的优缺点进行了评价，并探讨了模型的推广前景和改进方向。

关键词：ASA 算法，背包问题，三维装箱问题，剪冲式切割，计算机仿真

## 一、问题的背景与重述

### 1.1 问题的背景

印章是人们生活工作中常用的物品，在市场上，印章截面的形状是多种多样的，有圆形、矩形等，印章的厚度（高度）尺寸也是大小不一的。在印章的生产过程中，第一步是需要依照印章产品尺寸从原材料中截取初级产品，这是印章制造的第一道工序，在这道工序中，不同的截取方案具有不同的材料利用率。在原材料尺寸固定的前提下，截取印章的初级产品后产生的废料最少，生产成本最低是企业的追求。面向订单的材料切割问题称为下料问题，技术优化下料方案，对于减少原材料浪费、简化工艺生产有着十分重要的意义，设计一个优秀的切割方案，有助于提高企业生产力、增强企业的行业竞争能力。

国内外许多学者对于 Cutting&Packing 问题进行了大量的研究，此类问题是组合优化里的一个 NP 难问题，其中有传统的数学规划方法，如线性规划、动态规划等，也有现代智能计算方法和启发式算法，如遗传算法、模拟退火、离子群算法等。然而，这些方法只能对较小规模的问题才能有效的求出最优解。当问题的规模增大时，组合具有爆炸性传统精确算法的时间复杂程度会成指数增长，而在实际大规模的工业生产中，特别是切割贵重原材料，哪怕只提高其 1% 的材料利用率，带来的经济效益也是巨大。

由于各种排样问题很复杂，计算复杂度高，并且下料问题的求解过程是结合实际产业的生产环境和企业成本等条件进行考虑的，因而必然存在一定的工艺流程约束，且不同产业之间约束的表现形式也互不相同，因此合理的下料方案可以为企业节约生产成本，提高材料利用率，具有广泛的现实性指导意义。

### 1.2 问题的重述

某印章加工厂新进一种原材料用来加工印章，如表 1 所示。在印章加工的过程中，需要使用切割生产的初级产品如附录 1 所示。请你为该印章厂提供如下问题的原材料最优切割设计方案。其中，原材料的利用率（原材料截取初级产品的总体积与原材料体积之比）直接影响产品的生产成本。

表 1：原材料 S 的相关尺寸

原材料	长度(mm)	宽度(mm)	高度(mm)	购入价格
S	4550	1630	120	10 万元/个

本文建立了合适的数学模型，并解决了如下的问题：

问题 1：在一块长方体的原材料 S 上切割正椭圆柱体 P5 产品，建立数学模型，使切割出的 P5 数量尽可能的多，减少废料，求出原材料利用率最高的切割方案，并将最优方案的结果写入表格中。

问题 2：在一块原材料 S 上切割 P1、P2、P3、P4、P5、P6 产品，6 种产品的厚度均为 20mm，建立相关的数学模型，给出按照原材料利用率由高到低排序的前 5 种切割方案。

问题 3：要完成附录 1 中 P1、P2、P3、P4、P5、P6 产品的生产任务，需要多个原材料 S？由于工艺的缘故，至多只允许采用 5 种切割方案，给出原材料总利用率最高的至多 5 种切割方案，如优化计算中显示不需要 5 种切割的方案，可少于 5 种方案。

问题 4：要完成附录 1 中 P1-P9 产品的生产任务，需要多个原材料 S？设计原材料利用率较高的切割方案，并建立相应的数学模型，给出原材料总利用率最高的至多 5 种切割方案。

问题 5：不考虑产品 P1-P9 的需求数量，给定 100 个 S 原材料，按照附录 1 中给出的每个产品利润，建立数学模型，给出总利润最大的切割方案。

## 二、问题的假设

- (1) 假设：割缝宽度忽略不计。
- (2) 假设：各器件的测量误差在可控范围内。
- (3) 假设：在切割过程中不会出现器件损坏情况。
- (4) 假设：在切割过程中不会出现原材料损坏情况。

## 三、主要符号说明

符号	符号说明
$n_i$	第 $i$ 种规格的圆形件数
$m_j$	第 $j$ 种规格的椭圆形件数
$(x_i, y_i)$	第 $i$ 个圆的圆心坐标
$(u_i, v_i)$	第 $i$ 个椭圆的中心坐标
$L$	原材料 $S$ 的长度
$W$	原材料 $S$ 的宽度
$c_i$	第 $i$ 个方案执行次数
$f_{ij}$	第 $i$ 个方案第 $j$ 个产品个数
$p_j$	第 $j$ 个产品需求量
$s_j$	第 $j$ 个产品单位利益
$F_{ij}$	第 $i$ 个方案第 $j$ 个产品个数

注：其余符号详见文中说明。

## 四、模型的建立与求解

### 4.1 问题一：切割产品 P5 的最优方案

#### 4.1.1 名词介绍

- ①正圆柱体：以矩形的一边所在直线为轴、其余各边绕轴旋转而成的曲面所围成的几何体。
- ②正椭圆柱体：两个椭圆底面积相等，侧面垂直于地面的圆柱。
- ③正椭圆柱体体积公式：底面积（ $\pi$ \*半长轴\*半短轴）\*高。
- ④原材料的利用率：原材料截取初级产品的总体积与原材料体积之比。

#### 4.1.2 问题的分析

用原材料  $S$  为长 4550mm，宽 1630mm，高 120mm，切割横径为 45mm，竖径为 30mm，厚度为 20mm 的正椭圆柱 P5，求原材料利用率最高的方案。切割正椭圆柱问题属于三维问题，是 NP 完全问题，首先我们考虑是否可以转化为二维问题，所以假设 P5 模型水平或 90° 竖直放置，巧妙的将问题转化为在二维矩形面上放椭圆形件的问题，以此对不同的长宽高进行遍历取最优解，对于二维排样我们采用横纵向随机切割方式，计算 P5 的数量以及原材料利用率。

#### 4.1.3.1 模型一：横向竖向随机切割

##### (1) 决策变量的介绍

$p$ : 在二维面中，竖向摆放切割的椭圆数目。

$q$ : 在二维面中，横向摆放切割的椭圆数目。

##### (2) 目标函数的确立

不同截取方案有不同的利用率，原材料的利用率直接影响生产的成本，截取印章后的初级产品所剩废料越少，材料的利用率越高。

$$\text{原材料利用率} = \frac{\text{截取初级产品总体积}}{\text{原材料总体积}},$$

根据此公式，对原材料利用率最大化的求解，转化为求解切割最多数量的 P5 产品，使截取的体积最大。产品数目=层数\*行数\*列数，且都为整数，因此，确立一下目标函数：

$$f = \max(\max f1, \max f2, \max f3)$$

$$f1 = \left( p * \left\lfloor \frac{1630}{30} \right\rfloor + q * \left\lfloor \frac{1630}{45} \right\rfloor \right) * \left\lfloor \frac{120}{20} \right\rfloor$$

$$f2 = \left( p * \left\lfloor \frac{120}{30} \right\rfloor + q * \left\lfloor \frac{120}{45} \right\rfloor \right) * \left\lfloor \frac{1630}{20} \right\rfloor$$

$$f3 = \left( p * \left\lfloor \frac{120}{30} \right\rfloor + q * \left\lfloor \frac{120}{45} \right\rfloor \right) * \left\lfloor \frac{4550}{20} \right\rfloor$$

### (3) 约束条件

假设排入的不互相重叠、不超出材料边界，因此切割的椭圆不超过平面的行长，取不同行长时约束条件不同，即上面 f1, f2, f3 函数的约束条件分别为：

$$45p + 30q \leq 4550$$

$$45p + 30q \leq 4550$$

$$45p + 30q \leq 1630$$

p, q 为切割椭圆的数目，应为正整数：

$$p, q \geq 0$$

模型总结：

$$f = \max(\max f1, \max f2, \max f3)$$

$$f1 = \left( p * \left\lfloor \frac{1630}{30} \right\rfloor + q * \left\lfloor \frac{1630}{45} \right\rfloor \right) * \left\lfloor \frac{120}{20} \right\rfloor$$

$$\text{s.t. } 45p + 30q \leq 4550$$

$$f2 = \left( p * \left\lfloor \frac{120}{30} \right\rfloor + q * \left\lfloor \frac{120}{45} \right\rfloor \right) * \left\lfloor \frac{1630}{20} \right\rfloor$$

$$\text{s.t. } 45p + 30q \leq 4550$$

$$f3 = \left( p * \left\lfloor \frac{120}{30} \right\rfloor + q * \left\lfloor \frac{120}{45} \right\rfloor \right) * \left\lfloor \frac{4550}{20} \right\rfloor$$

$$\text{s.t. } 45p + 30q \leq 1630$$

$$p, q \geq 0$$

### (4) 算法设计

Step1: 确定切割面。

Step2: 确定横放产品可以摆放的列数范围，确定竖放产品可以摆放的列数范围。

Step3: 对不同摆放方式进行遍历。

Step4: 判断所有切割面是否遍历完全，若遍历完全，则输出最优解 P5 的数量和原材料利用率，否则返回 Step1 遍历下一个面。

### (5) 结果展示与分析

根据算法步骤，MATLAB 编写程序，得出此模型下的最优解，分 6 层切割，P5 总个数为 32724 件，材料最优利用率为 0.7797。该模型下材料利用率较低，在原材料价格较贵的情况下，还剩余较多的废料，该切割方案的生产成本比较高，不太符合工艺切割要



每个式子中，S 的三个长度都要取到，且不重复，其中，  
 $i \neq j \neq k, 0 \leq i, j, k \leq 3$

模型总结：

$$f = \max(\max f1, \max f2)$$

$$\begin{cases} \{x_i - [a * \lfloor (x_i - 22.5) / a \rfloor + 22.5]\} \geq 22.5 \\ f1 = \left\{ \left\lceil \frac{(x_i - 22.5) / a}{2} \right\rceil * \left\lfloor \frac{x_j}{30} \right\rfloor + \left\lceil \frac{(x_i - 22.5) / a}{2} \right\rceil * \left\lfloor \frac{x_j}{30} - 1 \right\rfloor \right\} * \left\lfloor \frac{x_k}{20} \right\rfloor \\ \{x_i - [a * \lfloor (x_i - 22.5) / a \rfloor + 22.5]\} < 22.5 \\ f2 = \left\{ \left\lceil \frac{(x_i - 22.5) / a}{2} \right\rceil * \left\lfloor \frac{x_j}{30} \right\rfloor + \left\lceil \frac{(x_i - 22.5) / a}{2} \right\rceil * \left\lfloor \frac{x_j}{30} - 1 \right\rfloor \right\} * \left\lfloor \frac{x_k}{20} \right\rfloor \end{cases}$$

$$\text{s.t. } i \neq j \neq k, 0 \leq i, j, k \leq 3$$

#### (4) 算法设计

Step1: 求交错排列时上下相邻的两椭圆中心之间的距离，设出相邻的椭圆方程，因为有唯一交点，联立两个方程组，解出两椭圆中心之间的距离，记为 a。

Step2: 设出决策变量 x, y, 任取一 S 长度，除以 P5 的厚度并向下取整，求出可以切割成几层。

Step3: 椭圆交错切割时，取剩余一 S 长度做平面为列长，此长度除以 P5 竖径并向下取整，即为正常切割的列数，见缝插针式切割的列数比正常切割少 1。

Step4: 剩余最后一个 S 长度作为平面的行长，行长先减去椭圆的半长轴再除以 2\*a 并向上取整或向下取整，故需要讨论最后一列能不能放全，判断切割后剩余的材料是否多于 22.5，若多于 22.5mm 时，正常切割等于见缝插针式切割的行数都取向上取整，否则正常式切割多 1 行，见缝插针式使用向下取整运算。

Step5: 将正常切割椭圆、见缝插针式切割的行数列数以及层数进行相乘，求出最多切割 P5 的数量。

Step6: 设约束条件，求出最优解，考虑废料的剩余情况，计算是否可以再切割出一定数量的 P5。

#### (5) 结果显示

表 2: 结果显示

行长	列长	正常放置列数	见缝插针列数	正常放置行数	见缝插针行数	产品数	利用率	1 为横径水平, 2 为竖径水平
4550	1630	58	58	54	53	37236	0.8872	1
4550	1630	21	20	151	150	37026	0.8822	2
4550	120	58	58	4	3	33089	0.7884	1
4550	120	1	1	151	150	24532	0.5845	2
1630	120	21	20	4	3	32760	0.7806	1
1630	120	1	1	54	53	24343	0.5800	2

由上表可知，印章的最佳利用率为 0.8872，正椭圆柱直立切割，且横径水平放置，竖径垂直放置，正常放置 58 列，见缝插针 58 列，接下来还需要考虑产生的废料是否可以再次利用。

整个切割长度为  $45 + 2 * a * 57 + a = 4526.7\text{mm}$ ，总长度为 4550mm，剩余体积近似为

20\*1630\*120 的长方体,采用程序 1 中找出的 1630\*120 为切割面时的最佳切割方案为横径与 1630 边水平放置,可产生  $36*4*1=144$  件,在考虑每个见缝插针列最下面产生的废料,近似为 20\*30\*120 的长方体,每列最多切割  $120/45=2$  件,一共有 58 列则产生  $2*58=116$  件。

所以在废料里总共可以产生 260 件,最终 P5 的数量为 37496,利用率为 89.34%,提高了 0.62%,模型结果较好,比较符合实际切割的工艺要求。

表 3: 原材料利用率最优时 P5 的数量

P5 的数量 (件)	原材料利用率
37496	89.34%

4.1.4 模型的灵敏度分析

灵敏度分析,可以用来研究原始数据不准确或发生变化时最优解的稳定性,通过灵敏度分析还可以决定哪些参数对系统或模型有较大的影响,比如:当这些数据中的一个或几个发生变化时,去判断最优解将会发生怎样的变化。

4.1.4.1 S 尺寸改变的分析

我们改变 S 原材料长、宽、高的尺寸, S 以 10mm 为单位在标准值左右取值,然后编程计算原材料利用率的变化情况,并画出相应的图像,更加直观的进行分析,进而判断得出最优解的模型是否稳定。

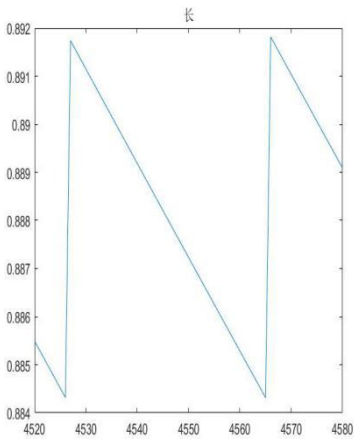


图 2: S 长度改变对原材料利用率的影响情况

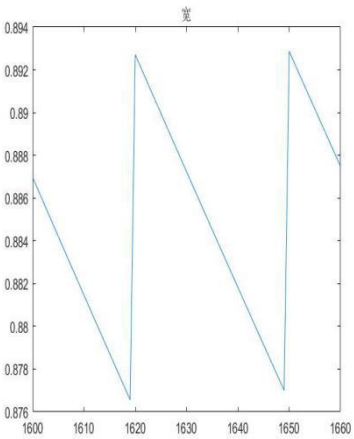


图 3: S 高度变化对原材料利用率的影响

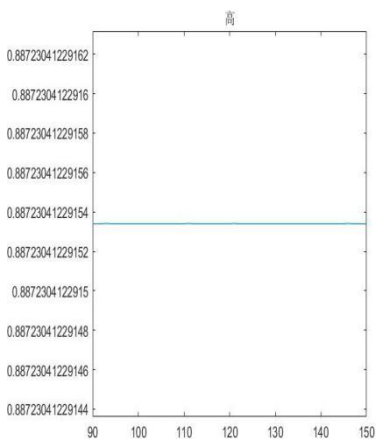


图 4: S 宽度改变对原材料利用率的影响

由上图可知,在一定范围内,长、宽围绕原材料 S 值上下改变时, S 的利用率围绕某个值上下波动,长度最大改变幅度大约为 0.74%,总体浮动变化较小,宽度的改变在 88.5%上下波动 0.8%左右,宽度的改变影响震荡幅度比长度的改变影响稍大一些,从图三可看出,原材料的高度的变化没有影响原材料 S 的利用率,所以总的来说模型比较稳定。

4.1.4.2 P5 尺寸改变的分析

我们分别改变 P5 横径、竖径、厚度的尺寸,以 5mm 为单位在标准值左右改变,求 S 利用率变化情况,并检验该模型是否稳定。

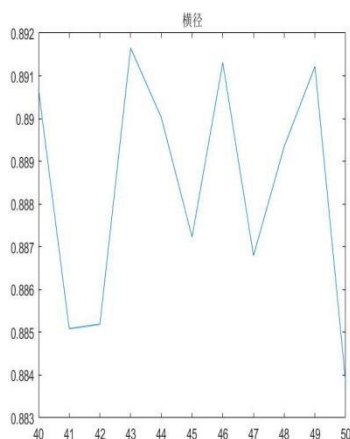


图 5: P5 横径改变对原材料利用率的影响

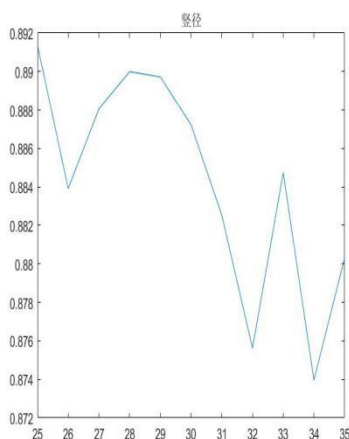


图 6: P5 竖径改变对原材料利用率的影响

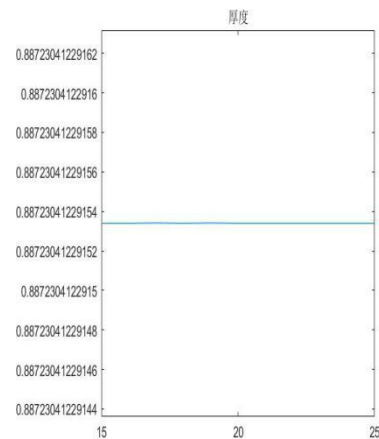


图 7: P5 厚度改变对原材料利用率的影响

由上图可知,在一定范围内,P5 横径的围绕 45mm 左右改变,利用率大约围绕 0.8890 上下波动,最大波动值为 0.65%,改变量较小;P5 竖径在 30mm 以下比 30mm 以上的利用率要高一些,整体呈下降的趋势;P5 厚度的改变没有影响利用率的改变,厚度对模型的影响较小,因此,相对来说,竖径的改变对 S 利用率影响要稍大一些,但改变幅度也比较小,因此得出最优利用率的方案的模型比较稳定。

#### 4.2 问题二: 切割 6 种产品给出 5 种较优方案

##### 4.2.1 问题的分析

由于 P1-P6 厚度为 20 能被  $h=120\text{mm}$  整除,高度被最大化利用,所以本题可以被简化为  $4550*1630$  的二维排样问题,所以建立如下模型:

##### (1) 决策变量的介绍

$n_i$ : 第  $i$  种规格的圆形件数;

$m_j$ : 第  $j$  种规格的椭圆形件数。

##### (2) 目标函数的确立

在原材料高度完全利用的情况下, S 利用率转化为面积之比,要使利用率最大化即要尽可能的排更多的圆形件、椭圆形件,减少原材料的剩余,则要达到的优化目标为:

$$\max f = \frac{\sum_{i=1}^4 n_i * \pi * r_i^2 + \sum_{j=1}^2 m_j * \pi * a_j * b_j}{L * W}$$

##### (3) 约束条件

①排入的圆形件、椭圆形件, 互不重叠。

$$S_1 \begin{cases} (x-x_i)^2 + (y-y_i)^2 = r_i^2 \\ (x-x_j)^2 + (y-y_j)^2 = r_j^2 \end{cases}, \text{len}(\text{solve}(S_1)) \leq 1 \text{ 其中, } i=1,2,\dots, \sum_{l=1}^4 n_l, j=1,2,\dots, \sum_{c=1}^4 m_c, i \neq j$$

$$S_2 \begin{cases} (x-x_i)^2 + (y-y_i)^2 = r_i^2 \\ \frac{(x-u_j)^2}{a_j^2} + \frac{(y-v_j)^2}{b_j^2} = 1 \end{cases}, \text{len}(\text{solve}(S_2)) \leq 1 \text{ 其中, } i=1,2,\dots, \sum_{l=1}^4 n_l, j=1,2,\dots, \sum_{c=1}^2 m_c$$

$(x_i, y_i), (x_j, y_j)$  分别为第  $i, j$  个圆的圆心坐标;

$(u_i, v_i), (u_j, v_j)$  分别为第  $i, j$  个椭圆的中心坐标。



②圆形件和椭圆形件排入卷材不能超过卷材的边界，L, W 为原材料的长度，宽度。

$$\text{圆: } \begin{cases} x_i + r_i \leq L \\ x_i - r_i \geq 0 \\ y_i + r_i \leq W \\ y_i - r_i \geq 0 \end{cases} \text{ 其中, } i = 1, 2, \dots, \sum_{j=1}^4 n_j$$

$$\text{椭圆: } \begin{cases} u_j - a \geq 0 \\ v_j - b \geq 0 \\ u_j + a \leq L \\ v_j + b \leq W \end{cases} \text{ 或 } \begin{cases} u_j - b \geq 0 \\ v_j - a \geq 0 \\ u_j + b \leq L \\ u_j + a \leq W \end{cases}, \text{ 其中, } j = 1, 2, \dots, \sum_{l=1}^2 m_l$$

模型总结:

$$\max f = \frac{\sum_{i=1}^4 n_i * \pi * r_i^2 + \sum_{j=1}^2 m_j * \pi * a_j * b_j}{L * W}$$

$$s.t. \begin{cases} S_1 \begin{cases} (x - x_i)^2 + (y - y_i)^2 = r_i^2 \\ (x - x_j)^2 + (y - y_j)^2 = r_j^2 \end{cases}, \text{len}(\text{solve}(S_1)) \leq 1, i = 1, 2, \dots, \sum_{l=1}^4 n_l, j = 1, 2, \dots, \sum_{c=1}^4 m_c, i \neq j \\ S_2 \begin{cases} (x - x_i)^2 + (y - y_i)^2 = r_i^2 \\ \frac{(x - u_j)^2}{a_j^2} + \frac{(y - v_j)^2}{b_j^2} = 1 \end{cases}, \text{len}(\text{solve}(S_2)) \leq 1, i = 1, 2, \dots, \sum_{l=1}^4 n_l, j = 1, 2, \dots, \sum_{c=1}^2 m_c \\ \begin{cases} x_i + r_i \leq L \\ x_i - r_i \geq 0 \\ y_i + r_i \leq W \\ y_i - r_i \geq 0 \end{cases}, i = 1, 2, \dots, \sum_{j=1}^4 n_j, \begin{cases} u_j - a \geq 0 \\ v_j - b \geq 0 \\ u_j + a \leq L \\ v_j + b \leq W \end{cases} \text{ or } \begin{cases} u_j - b \geq 0 \\ v_j - a \geq 0 \\ u_j + b \leq L \\ u_j + a \leq W \end{cases}, j = 1, 2, \dots, \sum_{l=1}^2 m_l \end{cases}$$

根据以上建立的目标函数以及约束条件，我们建立了如下的模型和算法。

#### 4.2.2 模型一：ASA 与见缝插针混合模型

##### (1) ASA 算法介绍

圆弧搜索放置算法（ASA 算法），圆的位置关系定位方法如下：首先，根据两圆相切计算待排圆圆心轨迹方程为

$$(x - x_i)^2 + (y - y_i)^2 = (r_i + r_j)^2,$$

其中 (x, y) 为待排圆的圆心坐标，(x<sub>i</sub>, y<sub>i</sub>) 为已放入的第 i 个圆的圆心坐标，(r<sub>i</sub>, r<sub>j</sub>) 为两个圆的半径；然后，将圆心轨迹均匀离散化，取离散后的子集作为搜索空间；最后取这些子集位置从左到右依次搜索，搜索过程中删除非法位置并将可行位置进行保留，选择最左最下位置进行圆形件的放置。

##### (2) 问题分析

由于 p1-p6 各个产品的厚度为 20mm，可以使得高度 120mm 的利用率做到最大化，所以该问题可以转化为在 4550\*1630 面上的二维问题，在解决不同规格的圆形件排样问题在国内外有较为成熟的 ASA, BLP, ABLP 算法,但是 ASA, BLP, ABLP 算法在解决不规则器件在算法实现上会非常困难，由于随着器件的摆放，产生的组合具有爆炸性，很难界定

待放的圆形件与椭圆形件的摆放是否会与已放置的发生重叠，所以我们首先考虑对原材料的 4550\*1630 面进行分割，先在平面上竖切一刀，记为点 a 左面用来对圆形件进行排样记为 A 面，右面横切一刀切点记为点 b，分为上下 B, C 两个面，B, C 面对不同规格的椭圆形件运用问题 1 中见缝插针式的排样策略，A 面用较为成熟的排样算法 ASA 进行排样。对 a 的横坐标，b 的纵坐标进行遍历，最终选取利用率最高的 5 个方案。

### (3) 算法设计

对于 ASA 算法，设计了执行 ASA 的子流程图，对于分块计算圆形件与两种椭圆形件的 S 利用率，设计了子流程图加以说明，以及问题的求解思路，详细算法设计见下表。

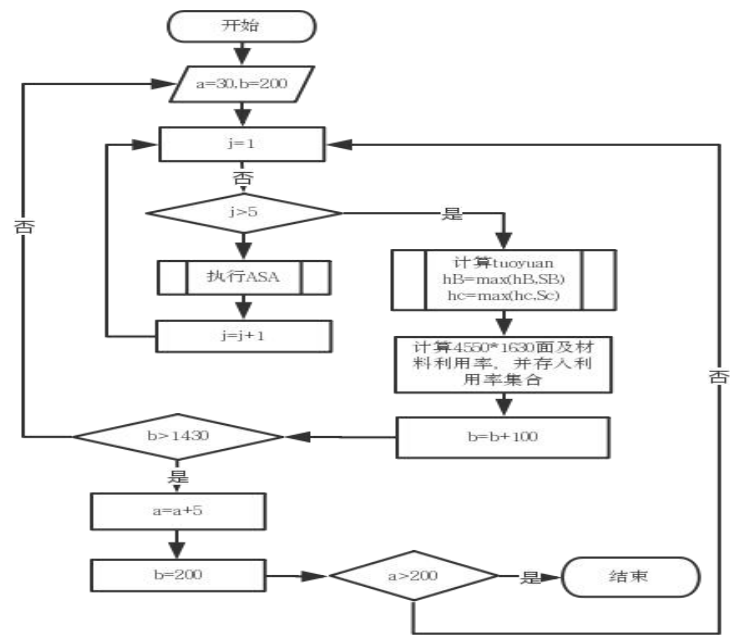


图 8：计算 S 利用率较优的 5 种方案的主流程图

### (4) 结果展示与分析

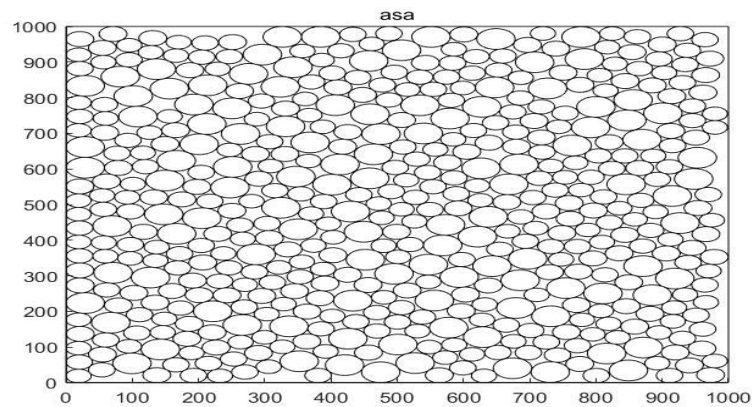


图 9：不同规格球排列情况

由图 9 结合所得数据可以看出，球的排列之间有较为紧密，但在边缘处材料剩余很大的空隙，导致材料的剩余，此排列方法有较大的随机性，此图是对 1000\*1000 的平面进行排料，对于大规模的排料问题，此方法的运算时间会呈指数增长，各规格的球随机排列也许不如问题分块排列利用率高，效率高。

表 4：切割 P1-P6 的数量以及平均利用率

P1 数量	P2 数量	P3 数量	P4 数量	P1-P4 总 利用率	P5 数量	P6 数量	P5 利用率	P6 利用率	P1-P6 平 均利用率
144	78	96	72	0.7064	4617	1623	0.8944	0.8593	0.8818
126	84	60	120	0.7112	2679	3776	0.8734	0.8876	0.8777
114	96	84	90	0.6908	5757	320	0.8955	0.6770	0.8781
102	96	102	108	0.7030	4161	2065	0.8764	0.8737	0.8720
102	114	54	138	0.7015	3098	3264	0.8883	0.8631	0.8723

因函数每次都是随机，所以结果的数量上会有差异。采用分块的方式计算，圆形件用 ASA 圆弧搜索法，得出圆形件的 S 利用率在 70%左右，利用率有待提高，椭圆采用问题一的算法，得出 S 利用率在 88%左右，此结果的利用率较好，并算出产品总平均最高利用率为 88.18%，P1-P6 各件数分别为 144、78、96、72、4617、1623 件，虽然总平均利用率较高，但模型运算时间过长，平均计算每个结果 8 分钟左右，模型还需进一步改进，缩短运行时间。

#### 4.2.3 模型二：剪冲式下料模型

下料问题按工艺划分，分为剪冲下料和非剪冲下料，剪冲下料又称剪切下料，指切成条带再到每个圆形件或椭圆形形件的过程，如图 10 所示；非剪冲下料则是指直接从上原材料上切割所需求产品，如图 11 所示，本文采用剪冲下料方式。

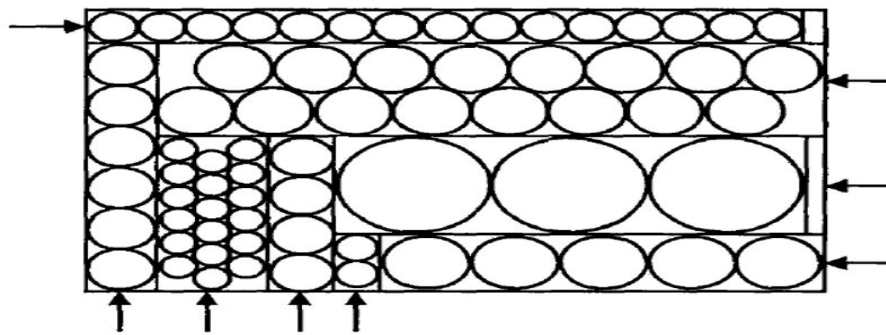


图 10：剪冲下料方式

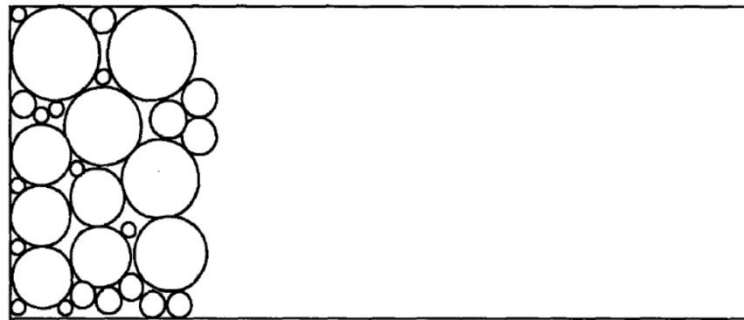


图 11：非剪冲下料式

#### (1) 问题分析

由于模型一运行时间过长，考虑模型二方法，模型二采用随机排布来确定 6 种产品在原材料 4550\*1630 面上所占面积，每次放入的产品种类是随机的，同时为了使随机面积的横长和纵长都发生改变，故每次划分区域改变一次方向（即横竖交替或竖横交替划分）。计算在此随机条件下的总利用率，在对模型模拟 10 万次，取出利用率最高的前 5

种情况。

(2) 算法设计

对于剪切式算法，随机打乱数字序列，随机确定 6 种产品所占面积，在划分区域出运用相应圆形件和椭圆形件的函数，求出组合切割的数量，以及相应的原材料利用率。具体随机切割的编程算法见如下的流程图：

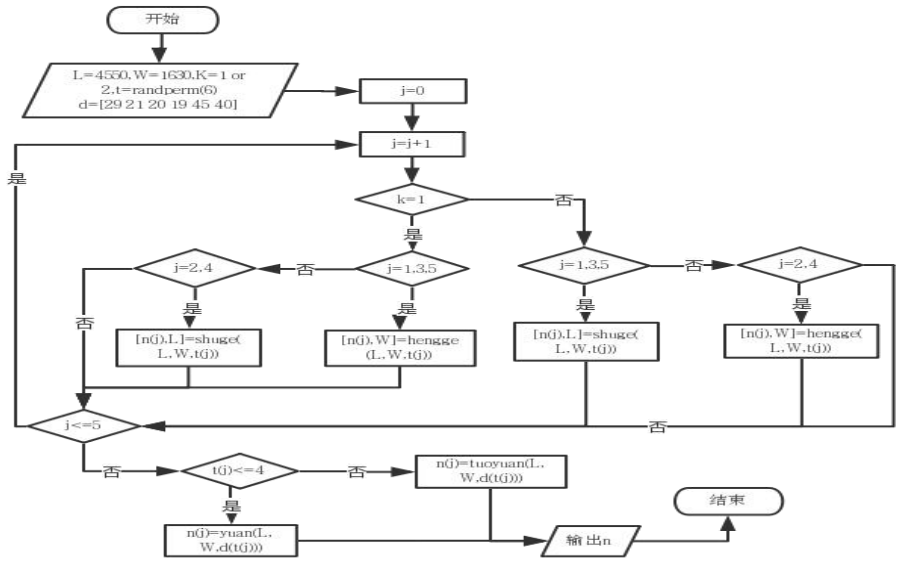


图 12：随机横向竖向切割分块的流程图

(3) 结果展示与分析

表 5：随机产生利用率高的 5 种方案

方案	P1 数目	P2 数目	P3 数目	P4 数目	P5 数目	P6 数目	S 总平均利用率
1	84	36	336	90	606	40710	0.8946
2	66	120	228	558	606	40158	0.8933
3	0	0	7824	0	606	31050	0.8930
4	0	0	438	252	0	41268	0.8928
5	0	0	414	234	36	41268	0.8926

因函数每次都是随机，所以结果的数量上会有差异。明显看出模型二的利用率普遍高于比模型一的利用率，S 总平均利用率最高为 89.46%，结果较好，切割数目分布比较均匀，且运行速度快，可以快速产生不同用户需要的高效率方案。算法时间性能上有了大幅度的提升，很好地解决了模型一的运行速度的问题。

4.2.4 模型灵敏度分析

为了分析数据的变化，对模型最优解有什么影响，我们分别改变了原材料的长度和宽度以及 6 种圆形件半径，椭圆形件的横径、竖径的尺寸，同样以 2mm 为单位在标准值上下改变数值观察对原材料利用率的影响情况。对于原材料尺寸改变对利用率影响图见附录四。

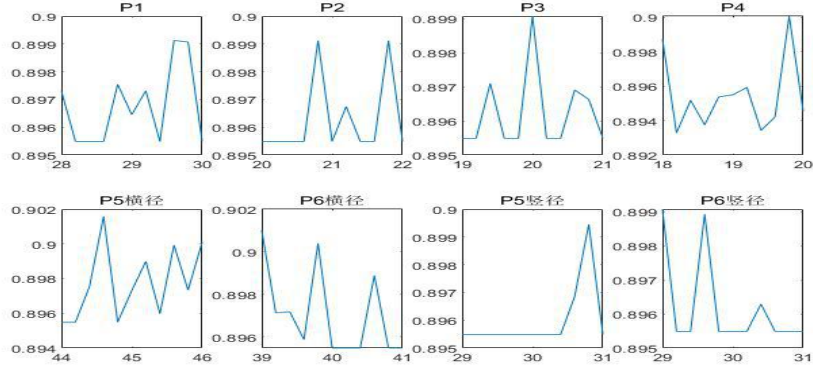


图 13: 改变产品的相关尺寸对 S 利用率的影响

在一定范围内, 原材料长度的改变, 利用率在较小的范围内波动, 大约在 89.34% 上下波动, 比较稳定, 但宽度的改变有一突变值, 在宽度小于 1634mm 时, 模型较稳定。由图 13 可知, 改变产品的尺寸, S 利用率有周期性的变化, 有一定的波动范围, 但对于材料的利用率相对较高, 模型较优。

#### 4.3 问题三: 完成生产任务的方案以及 S 的件数

##### 4.3.1 问题的分析

完成六种产品的生产任务, 给出原材料利用率最高 5 种方案, 求需要 S 的数量。因六种产品的高度均为 20mm, 可以被 S 原材料整除, 同问题二转化为二维问题, 基于问题二剪冲式切割的模型, 对求出的近 2 万种切割方式, 然后, 定义方法搜索的优先级, 对方法进行筛选, 使之满足产品的生产需求, 但最多使用 5 种方案。

##### 4.3.2 模型的建立与求解

###### (1) 决策变量的介绍

$c_i$ : 第  $i$  个方案执行次数;

$f_{ij}$ : 第  $i$  个方案第  $j$  个产品个数;

$p_j$ : 第  $j$  个产品需求量。

###### (2) 目标函数的确立

在完成 P1-P6 生产任务的基础上, 为了使生产成本尽可能的低, 需要减少 S 原材料的使用。原材料使用的数量越少, 相当于产品使用  $c$  块原材料后, 完成生产任务剩余的该产品数越小越好, 因此确立以下目标函数:

$$\min z = \sum_{j=1}^6 [(\sum_{i=1}^5 c_i * f_{ij}) - p_j]$$

###### (3) 约束条件

①切割的各个产品数量, 必须满足各自的需求量, 生产的件数要多于生产任务件数:

$$\sum_{i=1}^5 c_i * f_{ij} - p_j \geq 0, j = 1, \dots, 6$$

②其中, 执行次数为正整数:

$$c_i \geq 0$$

模型总结:

$$\min z = \sum_{j=1}^6 [(\sum_{i=1}^5 c_i * f_{ij}) - p_j]$$

$$s.t. \begin{cases} \sum_{i=1}^5 c_i * f_{ij} - p_j \geq 0, j = 1, \dots, 6 \\ c_i \geq 0 \end{cases}$$

#### (4) 算法设计

Step1: 利用问题二中运用剪冲式下料模型, 随机出 2 万种切割方式。先选取切割方法库中利用率最高的方法对原材料进行切割, 直到 A (A 为任意) 产品生产达到需求, 停止此方法切割;

Step2: 对方法库进行筛选, 选取不生产 A 产品的方法, 再找出利用率最大的方法, 对原材料切割直到 B 产品生产达到需求, 停止此方法切割, 直到切割方法达到四种;

Step3: 对于第五种切割方法需使得剩余产品满足需求, 即以不生产已完成产品并且剩余产品都产生为条件对方法库进行筛选, 选取两产品相差幅度不大的为切割方法进行操作, 最后得到题目所需数据。

#### 4.3.3 结果展示与分析

根据算法 MATLAB 编程得到如下结果:

表 6: 切割方案得出的产品数量、使用 S 数量以及相应的利用率

切割方案	1	2	3	4	5	总计
原材料 S 的数量	25	48	51	81	76	281
P1 的数量	0	2880	0	36450	840408	879738
P2 的数量	0	0	0	478062	0	2163186
P3 的数量	12900	49824	1583550	0	0	1646274
P4 的数量	3000	1615968	0	0	0	1618968
P5 的数量	15150	0	29988	109836	701328	856302
P6 的数量	1017750	0	0	0	0	1017750
原材料利用率	89.43	89.09	89.09	89.05	87.64	88.71

因函数每次都是随机, 所以结果的数量上会有差异。由上表可知, 最优解为原材料 S 使用 281 个, 在 5 种切割方案下, 完成了各个产品的生产任务, 原材料利用率达 88.71%, 利用率结果较优。由于产品都超额完成了生产任务, 比如 P1, P2 产品数量多需求量上万件, 所以该切割方案使用 S 原材料稍高, 生产成本会有些高。

#### 4.3.4 模型的灵敏度分析

灵敏度分析是在建立数学模型和求得最优解之后, 针对数据资料变化而做的研究与分析, 在求出最优解 S 原材料 281 块后, 我们对 6 种产品的需求、原材料的长度和宽度以及产品的相关尺寸小幅度的改变, 判断其数据的影响情况。

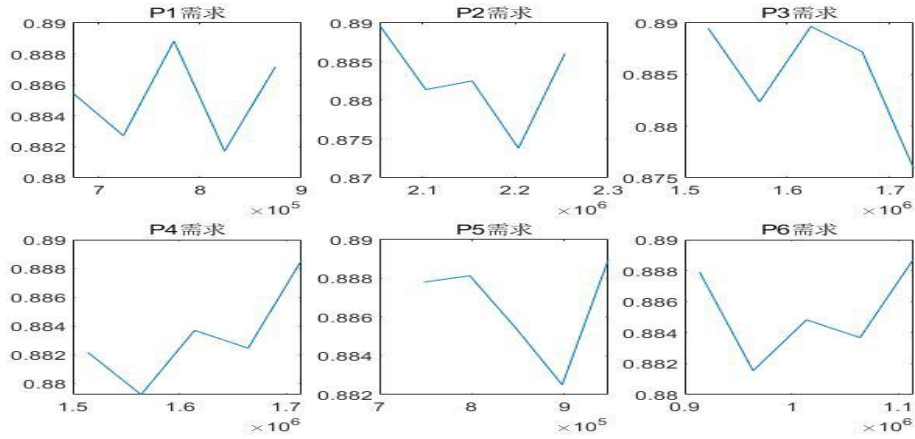


图 14：产品需求的改变对 S 原材料利用率的影响

由上图可知，各个需求量的改变使利用率在 1.5% 以内围绕某个值上下浮动，其中 P6 需求量的改变对原材料的利用率影响较小，说明模型对小椭圆形件的影响较好，原材料长度和宽度，产品个尺寸改变影响见附录 5，对不同大小，形状的产品数据的改变有着不同的影响，但都围绕某个值上下浮动，且该值较优，故模型效果较好。

#### 4.4 问题四：5 种最优方案以及 S 数量的求解

##### 4.4.1 问题的分析

本问在问题三的基础上增加了三种不同规格的矩形印章件，现需要考虑切割出 9 种印章并且使其满足需求量的条件。首先我们查阅了相关解决三维问题方面的论文，结合本题，考虑把圆形，椭圆形均看成矩形，这样问题转化为不同尺寸的长方体的三维排料问题，由于智能算法的求解时间较长以及处理的规模较大，因此我们对问题做了进一步的分析，观察到矩形的尺寸三种矩形的高 [60, 40, 30] 均可以被原材料高度 120mm 整除，我们假定矩形的放置方式为高度边与原材料高度边水平放置，将此问题转化为在 4550\*1630 平面上的摆放问题。然后我们在问题三的程序基础上进行修改并构建新的程序，再次使用剪冲式下料方法，得到数量足够多的方法库，再对从中选出 5 种方案对原材料进行切割使得能够满足产品需求并且利用率最高。

##### 4.4.2 模型的建立与求解

###### (1) 决策变量的介绍

本模型和问题三建立的模型类似。问题三模型中的 (1) 式同样作为本问题的决策变量。

###### (2) 目标函数的确立

目标函数同问题三基本一致，将问题三中 6 件产品改为生产 9 件产品，其它部分一致。

$$\min z = \sum_{j=1}^9 [(\sum_{i=1}^5 c_i * f_{ij}) - p_j]$$

###### (3) 约束条件

切割的各个产品数量，必须满足各自的需求量，生产的件数要多于生产任务件数，对 9 件产品进行循环：

$$\sum_{i=1}^5 c_i * f_{ij} - p_j \geq 0, j=1, \dots, 9, c_i \geq 0$$

模型总结：

$$\min z = \sum_{j=1}^9 [(\sum_{i=1}^5 c_i * f_{ij}) - p_j]$$

$$s.t. \sum_{i=1}^5 c_i * f_{ij} - p_j \geq 0, j=1, \dots, 9, c_i \geq 0$$

#### (4) 算法设计

我们对该问题的算法绘制了流程图，此问算法设计前四种切割方式的选择与第三问设计类似，对于第五种切割方式，发现剩余未完成产品种类较多，改变切割方条件，选取方法库中未完成产品都生产的方法，对每种方法以第一种为 1 构建比例数值，再将其与标准比例数值比较，选取浮动最小的方法完成剩余任务。考虑到随机性，选取最终消耗原材料小于 600 的方案。设计的子流程图见附录 7，算法设计的主流程图如下：

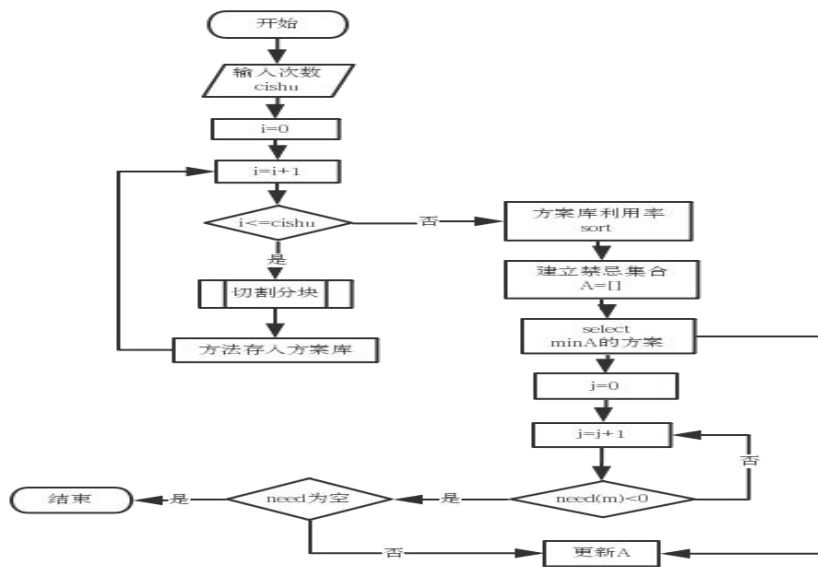


图 15：方案选择生产 9 种印章的主流程图

#### 4.4.3 结果展示与分析

表 7：S, P1-P9 的数量以及 S 利用率的结果展示

切割方案	1	2	3	4	5	总计
原材料 S 的数量	22	73	47	25	302	463
P1 的数量	0	0	0	0	875196	875196
P2 的数量	0	11388	0	0	2569416	2580804
P3 的数量	0	0	0	14850	1781196	1796046
P4 的数量	0	0	0	2250	1614492	1616742
P5 的数量	0	0	0	0	1406112	1406112
P6 的数量	0	0	0	1031700	21744	1053444
P7 的数量	0	2551934	0	0	27180	2579114
P8 的数量	49698	238710	3197880	0	150396	3636684
P9 的数量	2799720	0	0	0	161872	2961592
原材料利用率	99.81	99.52	99.08	89.31	86.6	90.62



因函数每次都是随机的，计算结果可能有误差。5 种切割方案，总计需要原材料 S 的数量为 463 块，原材料利用率为 90.62%，全切割长方体是，原材料的利用率可达到 99%，基本上原材料没有浪费，对于其他形件的切割，原材料利用率也很高，说明模型效果较好，P1-P9 生产的数量各超过了需求不等，虽然 S 需要 463 块，但切割的原材料利用率依旧很高。

#### 4.4.4 灵敏度分析

该模型的灵敏度分析与问题三类似，我们改变 9 种产品的需求，原材料以及 9 种产品尺寸，得如下折线图，其中产品尺寸的改变见附录 6。

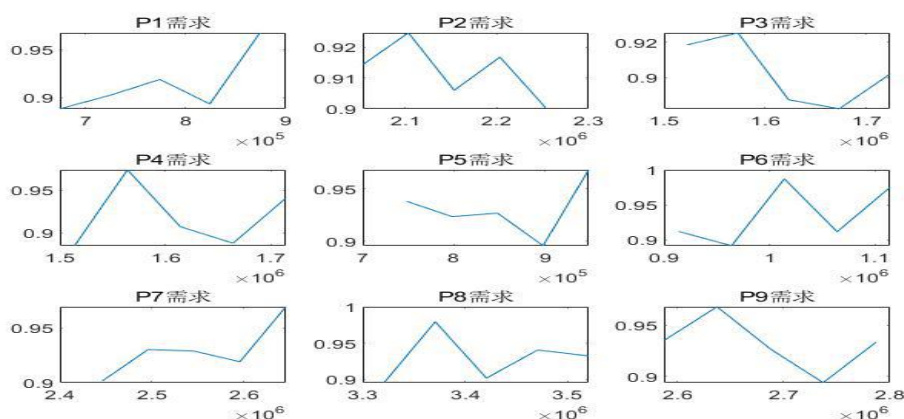


图 15: 需求的改变对 S 利用率的影响

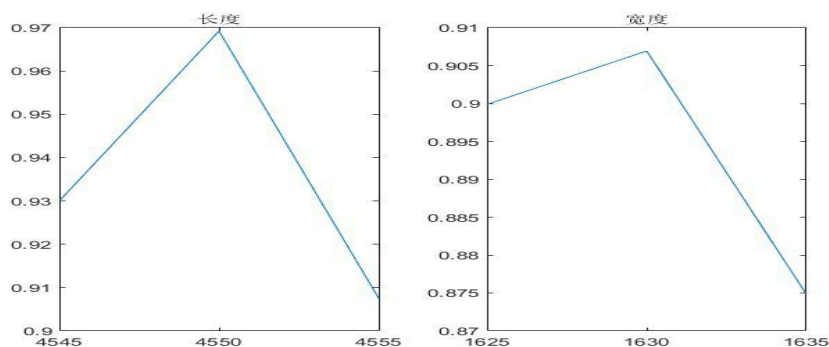


图 16: S 原材料长度和宽度的改变对 S 利用率的影响

根据图 5 可知，产品需求量的改变对 S 利用率的影响较大，大约有 5% 的波动范围，但数值均在 90% 以上，原材料的利用程度较高，对比产品和原材料尺寸的改变对原料利用的影响可知，产品对其影响更大一些，波动更明显，但都有较高的原料利用率，因此，模型解得效果比较好。

### 4.5 问题五：总利润最大的切割方案

#### 4.5.1 问题的分析

根据问题五可得知目标函数进行了改变，即现在需要总利润最大的方案，且无需考虑各产品的需求数量。我们考虑对问题四中一开始建立方法库构造利润函数，使得方法库中的每一个产品和它自身的价格相乘再累加，再减去原材料的费用，找出这些方法中产生利润的最大值，使用此方法对一块原材料进行切割，题目中给出了 100 块原材料，因此方法为我们找到的利润最大的方案，故对 100 个原材料都进行同样的切割方式，使得整体的利润也达到最大。

#### 4.5.2 模型的建立与求解

##### (1) 决策变量的介绍

$s_j$ : 第 j 各产品单位利益;

$F_{ij}$ : 第 i 各方案第 j 各产品数量。

##### (2) 目标函数的确立

求总利润最大的方案，且无需考虑各产品的需求数量，即考虑切割的产品件数乘以该产品的单位利益再减去成本的最大值情况。

$$f = \max \sum_{j=1}^9 F_{ij} * s_j - 1 * 10^6$$

##### (3) 约束条件

对所有的 i, j 来说,

$$F_{ij} \geq 0$$

模型总结:

$$\begin{aligned} f &= \max \sum_{j=1}^9 F_{ij} * s_j - 1 * 10^6 \\ s.t. & F_{ij} \geq 0 \end{aligned}$$

##### (4) 算法设计

Step1: 根据问题四建立方法库构造利润函数，利润=销售金额-生产成本，使得方法库中的每一个产品和它自身的价格相乘再累加，再减去原材料的费用。

Step2: 使用此方法对一块原材料进行切割，找出这些方法中产生利润的最大值。

Step3: 局部最优代替整体最优，找到一块最优的切割方案即找到 100 块最优的切割方案。

Step4: 输出最优方案和原料的利用率。

#### 4.5.3 结果展示与分析

表 8: 总利润最高的切割方案

切割方案	1
原材料 S 的数量	100
P1 的数量	0
P2 的数量	0
P3 的数量	0
P4 的数量	0
P5 的数量	0
P6 的数量	4205400
P7 的数量	0
P8 的数量	0
P9 的数量	0
利润	65697200
原材料利用率	89.07%

切割一百块原材料 S，当利润最大时，最优方案为全部切割产品 P6，总共切割 P6 数量为 4205400 件，原材料利用率为 89.07%，最大利润为 6569.72 万元，该切割方案对材料的利用率较高，分析题中各产品的利润可知，型号 P1、P5、P6 的利润较高，总利润最高的取得应切割较多此三种型号，由上表可知，全部切割型号 P6 利润最高，符合

分析的情况，模型的求解具有一定的准确性。

## 五、模型的评价与推广

### 5.1 模型的评价

#### 5.1.1 模型的优点

(1) 由于各种排样问题很复杂，计算复杂度高，而且三维背包问题属于 NP 难题，我们根据材料高度的特点转化为二维问题，极大的降低了解决问题的难度，三维问题转化为二维问题，由难化简，想法较好。

(2) 根据椭圆与圆形的形状结合图像，得到交叉排放的方式比较节省空间，而且建立的模型简单，运行时间短，对于解决产品件数较多的情况也能保持较为快速的运行速度，适合用于对时间要求较为敏感的装箱运输行业。

(3) ASA 算法可以解决已知各件数比例数量的排样问题，运行解较为优质，模型简单，效率较高。

(4) 剪冲式下料模型可以生成多组解决方案，且解决方案较为优质，方案库任意组合适合不同的装箱情况。模型效率较高，运行速度与智能算法与三维排样算法相比较为快速，运行结果较好，符合实际的情况，现实生活中实现该方案也较为容易。

#### 5.1.2 模型的缺点

(1) ASA 虽然属于已经发展较为完善的排样算法，但由于物件组合具有爆炸性，很难界定待放的圆形件与椭圆形件的摆放是否会与已放置的发生重叠，时间复杂程度成指数增长，运行时间过长。

(2) 模型简单，牺牲了部分利用率来换取运行时间，问题二、三、四模型的运行结果具有随机性，每次的运行结果不同，可能每次运行输出的利用率会随着运行次数的增加而上下浮动。

### 5.2 模型的推广

剪冲式切割方法可以结合遗传算法对问题求解，对切的刀数切割种类进行编码，挑选出来的解比较优质，会对该问题进行较好的求解；若解决件数较少的问题可以将 ASA 与遗传算法进行结合，可以挑选出近似最优解的优质解；若解决件数较小的三维装箱问题的算法可以使用八叉树布局方法进行对器件的空间模型进行模拟。下料问题可应用于装箱运输行业、航空航天领域、造纸行业、玻璃切割行业等，设计合理的下料方案可以为企业节约生产成本，提高材料利用率，因此对实际生产应用有一定的指导意义。

## 六、参考文献

- [1] 李尚芳. 冲裁用条带的剪切排样算法研究[D]. 广西师范大学, 2011.
- [2] 刘春霞. 关于装箱问题的若干研究[D]. 大连理工大学, 2006.
- [3] 季君. 基于同形块的剪切下料布局算法研究[D]. 北京交通大学, 2012.
- [4] 赵晓东. 矩形件优化排样算法的研究与实现[D]. 大连交通大学, 2008.
- [5] 吴阳. 并行遗传退火算法的圆形件下料问题求解[D]. 广西大学, 2018.
- [6] 亓晓莹. 二维矩形条带装箱问题的左下角定位模型[D]. 武汉科技大学, 2018.
- [7] 谢琪琦. 圆形件下料优化算法的设计与实现[D]. 广西大学, 2017.
- [8] 王岩, 潘卫平, 张俊晖. 单一尺寸长方体三维装箱问题的一种求解算法[J]. 包装工程, 2015, 36(11): 96-99.
- [9] 陈燕, 刘咏, 谢琪琦, 崔耀东. 基于梯形和平行四边形的圆片剪冲下料算法设计与实现[J]. 图学学报, 2016, 37(05): 661-667.
- [10] 杨剑, 黄少丽, 侯桂玉, 崔耀东. 圆片剪冲下料排样算法[J]. 计算机工程与设计, 2010, 31(23): 5139-5142.
- [11] 郑巧仙, 李明, 郭天姣, 亓晓莹. 二维矩形条带装箱问题的改进左下角定位模型[J].

数学的实践与认识, 2019, 49 (02) :176-183.

[12]熊慧. 基于混沌粒子群算法的多目标二维切割问题[J]. 电子技术与软件工程, 2017(18):156.

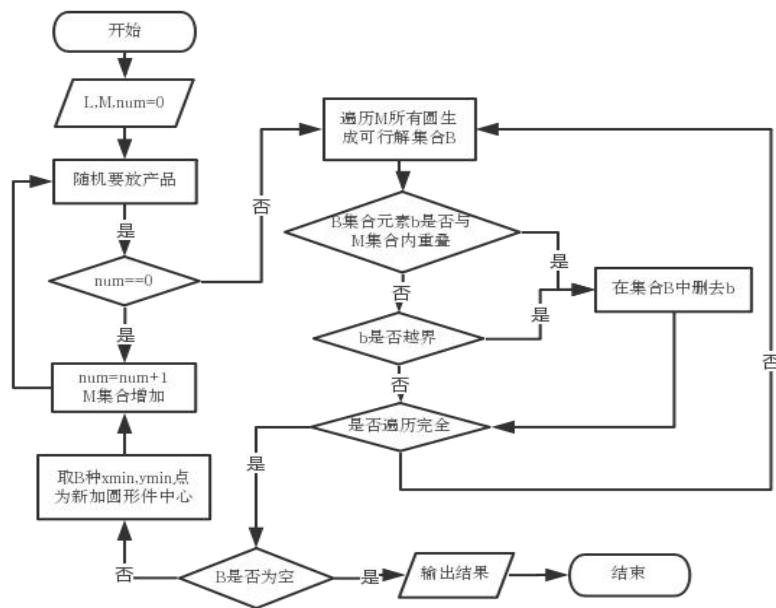
[13]熊慧, 黄菊永. 基于贪心启发式算法的多目标二维切割问题[J]. 电子技术与软件工程, 2016(24):154-155.

## 七、附录

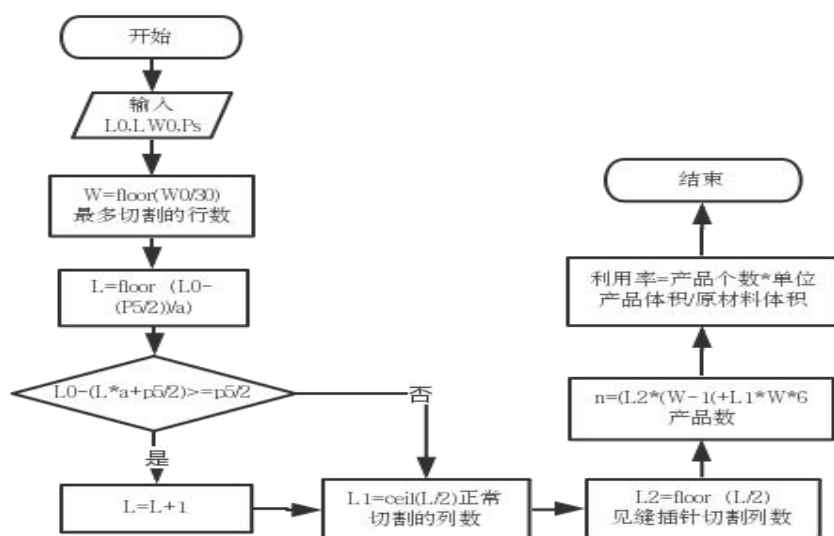
附录 1： 初级产品尺寸、生产任务及利润

型号	印章形状	尺寸(mm)	生产任务 (件)	利润(元/ 件)
P1	正圆柱体	截面直径 58，厚度 20	774000	12
P2	正圆柱体	截面直径 42，厚度 20	2153000	8
P3	正圆柱体	截面直径 40，厚度 20	1623000	7
P4	正圆柱体	截面直径 38，厚度 20	1614000	6
P5	正椭圆柱体	横径为 45，竖径为 30，厚度 20	848000	20
P6	正椭圆柱体	横径为 40，竖径为 30，厚度 20	1014000	18
P7	长方体	截面长 20，宽 20，厚度 60	2546000	4
P8	长方体	截面长 18，宽 18，厚度 40	3421000	2
P9	长方体	截面长 15，宽 15，厚度 30	2687000	1

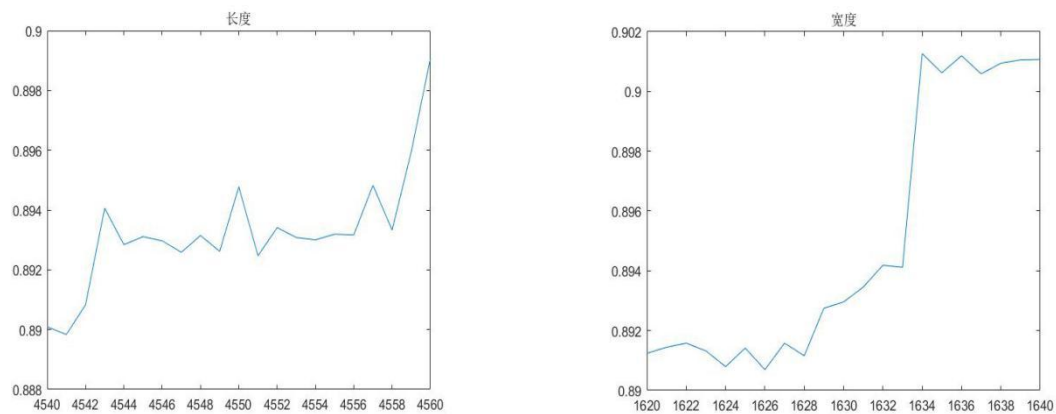
附录 2： 执行 ASA 的子流程图



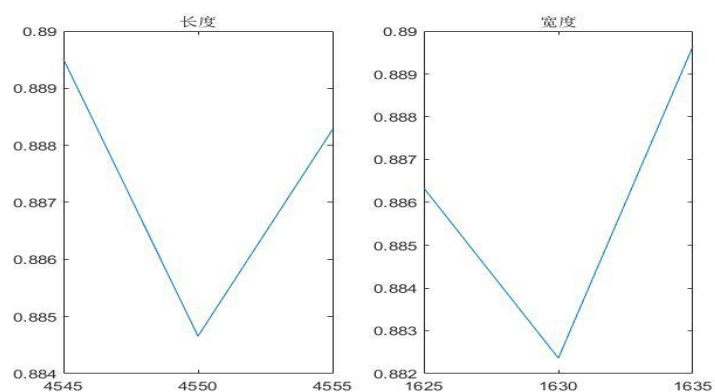
附录 3： 圆形件和椭圆形件利用率的子流程图

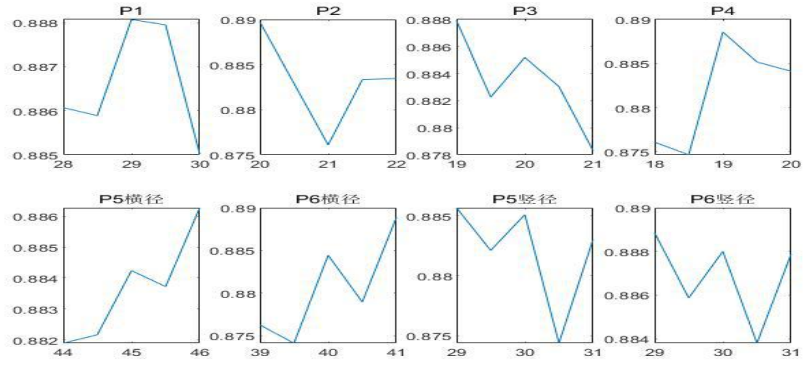


附录 4：问题二模型二灵敏度分析图，原材料 S 长度、宽度改变对利用率的影响

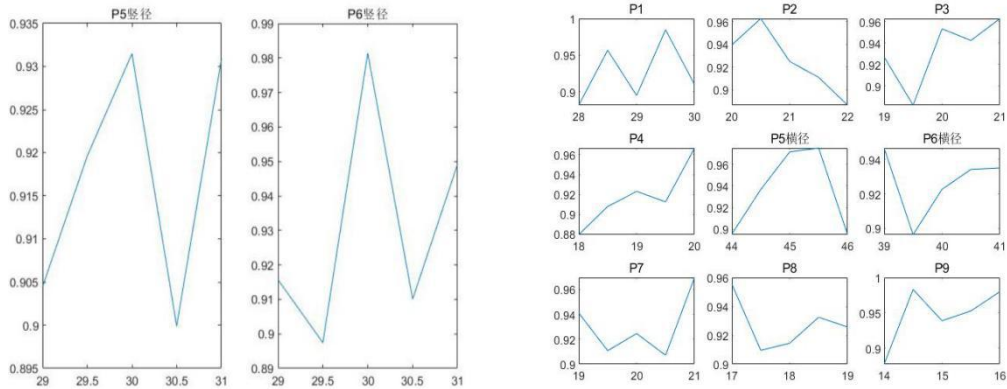


附录 5：问题三灵敏度分析图

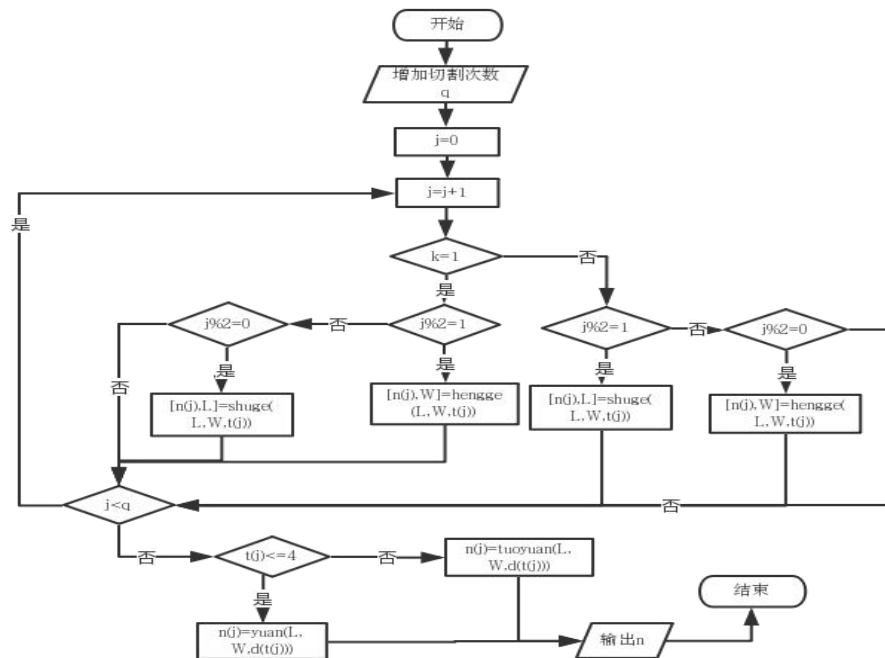




附录 6：问题四的灵敏度分析



附录 7：问题四算法子流程图



附录 8：横向竖向随机切割模型所用 matlab 程序

%计算横纵不同排列方式利用率

```

clear
clc
%原材料
A=[4543, 52, 120];
%p5 材料尺寸
p5=[40, 30, 20];
s1=4550*1630*120;%原材料体积
s=(45/2)*15*pi*20;%p5 的体积
%3 个面横轴联合
t=0;%便于储存 无实际意义
for i=1:2
    for j=i+1:3%两个 for 表示三种切割面
        n1=0;m1=0;n2=0;m2=0;%初始化
        %椭圆横径水平放置
        n1=floor(A(i)/p5(1));%最多排列行数
        m1=floor(A(j)/p5(2));%最多排列列数
        %椭圆竖径水平放置
        n2=floor(A(i)/p5(2));%最多排列行数
        m2=floor(A(j)/p5(1));%最多排列列数
        A1=A;
        A1([i, j])=[];%不同切割面时的高度
        for k=0:n1
            for L=0:n2%两种摆放方式遍历
                %确立限制范围
                if k*p5(1)+L*p5(2)<=A(i)&& k*p5(1)+L*p5(2)>=A(i)-10
                    t=t+1;
                    D(t, 1:4)=[A(i), A(j), k, L];%记录行长 列长 横纵方式个数
                    D(t, 5)=((k*m1)+(L*m2))*floor(A1/p5(3));%产生产品个数
                    D(t, 6)=D(t, 5)*s/s1;%记录利用率
                end
            end
        end
        for k=1:m1
            for L=1:m2%以列为轴来确立排放方式
                %确立限制条件
                if k*p5(1)+L*p5(2)==A(j)&& k*p5(1)+L*p5(2)>=A(j)-30
                    t=t+1;
                    D(t, 1:4)=[A(j), A(i), k, L];%记录行长 列长 横纵方式个数
                    D(t, 5)=((k*n1)+(L*n2))*floor(A1/p5(3));%产生产品个数
                    D(t, 6)=D(t, 5)*s/s1;%记录利用率
                end
            end
        end
    end
end
end

```



```

end
max(D(:,5))%整整齐齐排列情况下最大数量
ans*s/s1%整整齐齐排列情况下最大利用率
%计算见缝插针情况下利用率
D1=shiti1(A,p5);
%显示结果
fprintf('-----' );
-----');
fprintf('\n');
fprintf('%8s%8s%10s%10s%8s%8s%8s%8s%15s','行长','列长','正常放置列数','见缝插针列数','正常放置行数','见缝插针行数','产品数','利用率','1 为横径水平 2 为竖径水平');
fprintf('\n');
for i=1:6
    fmt='%10.f%10.f%10.f%15.f%15.f%12.f%15.f%11.4f%12.f';

    fprintf(fmt,D1(i,1),D1(i,2),D1(i,3),D1(i,4),D1(i,5),D1(i,6),D1(i,7),D1(i,8),D1(i,9));
    fprintf('\n');
end
附录 9: 见缝插针式模型灵敏度分析所用 matlab 程序
%对问题一的灵敏度分析
clear
clc
close all
A=[4550,1630,120];
p5=[45,30,20];
a={'长','宽','高'};
b={'横径','竖径','厚度'};
%对原材料灵敏度分析
for k=1:3
    t=0;
    for i=-50:1:50%检查范围
        B=A;
        t=t+1;
        B(k)=B(k)+i;
        D=shiti1(B,p5);
        y1(t)=D(1,8);
    end
    %绘制图像
    figure
    x=[A(k)-50:A(k)+50];
    plot(x,y1)
    title(a(k));

```

```

end
%对椭圆灵敏度分析
for k=1:3
    t=0;
    for i=-5:5%检查范围
        P=p5;
        t=t+1;
        P(k)=P(k)+i;
        D=shiti1(A,P);
        y2(t)=D(1,8);
    end
    %绘制图像
    figure
    x=[p5(k)-5:p5(k)+5];
    plot(x,y2)
    title(b(k));
end
附录 10: 椭圆见缝插针利用率函数
%计算椭圆见缝插针利用率函数
function D=shiti1(A,p5)
%A 为原材料 p5 为所求材料
s1=A(1)*A(2)*A(3);%原材料体积
s=(p5(1)/2)*(p5(2)/2)*pi*p5(3);%p5 的体积
t=0;%便于储存 无实际意义
a=p5(1)*sqrt(3)/2;%第一列和第二列椭圆中点的横坐标距离

for i=1:2
    for j=i+1:3%两个 for 表示三种切割面
        A1=A;
        A1([i,j])=[]; %储存不同切割面时高度
        for k=1:2 %横纵两种切割方式
            if k==1 %A(i)为行长 A(j)为列长
                t=t+1;
                W=floor(A(j)/p5(2));%最多摆放行数
                L=floor((A(i)-p5(1))/(2*a));
                L1=L+1;L2=L;%L1 表示正常摆放列数 L2 表示见缝插针形的列数
                if A(i)-(L*2*a+p5(1))>=a %判断是否可以再加入一列
                    L2=L2+1;
                end
            end
            if k==2 %A(i)为列长 A(j)为行长
                t=t+1;
                W=floor(A(i)/p5(2));%最多摆放行数
                L=floor((A(j)-p5(1))/(2*a));
            end
        end
    end
end

```

```

        L1=L+1;L2=L;%L1 表示正常摆放列数 L2 表示见缝插针形的列数
        if A(j)-(L*2*a+p5(1))>=a%判断是否可以再加入一列
            L2=L2+1;
        end
    end
    W1=W;W2=W-1;%W1 表示正常排列行数 W2 表示见缝插针方式行数
    n=(L2*W2+L1*W1)*A1/p5(3);%产生产品个数
    p=n*s/s1;%计算利用率
    D(t,:)=[A(i),A(j),L1,L2,W1,W2,n,p,k];
    %储存结果 行长 列长 正常放置行列数 见缝插针行列数 产品个数 利用率 摆放方式
end
end
end

```

#### 附录 11：采用横切的函数

```

function [n12,W]=heng(L,W,k,pp,d)
%横切时采用方法
%输入 k 种产品所占据的区域 L*W
%输出产品数量和裁剪后的原材料宽度
%d 为产品尺寸
%pp 为椭圆横径
%d=[29,21,20,19,45,40];
if k<=4
    n1=floor(W/(d(k)*2));%最大行数
    k1=ceil(rand()*n1);%随机行数
    n12=yuan(L,k1*d(k)*2,d(k));%数量
    W=W-k1*d(k)*2;
else
    n1=floor(W/pp(k-4));%最大行数
    k1=ceil(rand()*n1);%随机行数
    n12=tuoyuan(L,k1*pp(k-4),d(k),pp(k-4));%数量
    W=W-k1*pp(k-4);
end

```

#### 附录 12：ASA 算法所用 matlab 程序

```

%此为以 asa 算法进行一次运算
%此程序为加快程序运行速度使得 a 只在 1-200 随机
%若使得 a 在 1-4550 随机计算时间会很长，对时间浪费大
%故考虑了 shiti_2_yunxin 程序 加快运行速度
clear all
clc
a=ceil(rand()*200);%随机范围
b=ceil(rand()*1630);%随机范围
[p1,n]=asa(a,1630);%对圆以 asa 算法排布
P2=shiti1([4550-a,1630-b,120],[45,30,20]);%椭圆以第一问算法排布

```

```

[p2, n2]=max(P2(:, 8));
P3=shiti1([4550-a, b, 120], [40, 30, 20]);
[p3, n3]=max(P3(:, 8));

p=(p1*a*1630+p2*(4550-a)*(1630-b)+p3*(4550-a)*b)/(4550*1630)%该方法利用率
N(1:4)=n';N(5)=P2(n2, 7);N(6)=P3(n3, 7)%该方法每一种的个数
附录 13: 剪冲式模型所用 matlab 程序
%问题二模型运行程序与结果输出
clear
clc
L=4550;W=1630;%原材料长宽
d=[29, 21, 20, 19, 45, 40];%半径和横径
pp=[30, 30];
D=moxin2(L, W, d, pp);
D
fprintf('----- 结 果 显 示
-----');
fprintf('\n');
fprintf('%5s%8s%8s%8s%8s%8s%12s%8s%8s%8s%8s', '利 用 率
', 'A', 'B', 'C', 'D', 'E', 'F', 'A 类产生数量', 'B 类产生数量', 'C 类产生数量', 'D 类产
生数量', 'E 类产生数量', 'F 类产生数量');
fprintf('\n');
for i=1:5
    fmt='%8.4f%8. f%8. f%8. f%8. f%8. f%8. f%12. f%12. f%12. f%12. f%12. f%12. f';
    fprintf(fmt, D(i, 1), D(i, 2), D(i, 3), D(i, 4), D(i, 5), D(i, 6), D(i, 7), D(i, 8), D(i, 9),
    D(i, 10), D(i, 11), D(i, 12), D(i, 13));
    fprintf('\n');
end

```

%因随机排布，每种生产个数运行出答案与结果展示有差异

附录 14: 剪冲式模型的灵敏度检验所用 matlab 程序

%对问题二模型二的灵敏度检验

%模型全部运行时间约为 2 分钟

```

clear
clc
close all
L=4550;W=1630;%原材料长宽
d=[29, 21, 20, 19, 45, 40];%半径和横径
pp=[30, 30];
t=0;
for i=-5:5%检验范围
    L0=L+i;%改变 L 的长度
    t=t+1;

```

```

        D=moxin2(L0,W,d,pp);
        p(t)=D(1,1);%记录最大利用率
    end
    x=L-5:1:L+5;
    %绘制图像
    plot(x,p)
    title('长度');
    t=0;
    for i=-5:5%检验范围
        W0=W+i;%改变 W 的宽度
        t=t+1;
        D=moxin2(L,W0,d,pp);
        p(t)=D(1,1);%记录最大利用率
    end
    x=[W-5:1:W+5];
    figure%绘制图像
    plot(x,p)
    title('宽度');
    a={'P1','P2','P3','P4','P5 横径','P6 横径'};
    b={'P5 竖径','P6 竖径'};
    %改变产品尺寸绘制灵敏度
    figure
    for i=1:6%对 d 中尺寸进行遍历
        subplot(2,4,i)%都绘制在一个窗口
        t=0;
        p=[];
        for j=-1:0.5:1
            d0=d;
            d0(i)=d0(i)+j;
            D=moxin2(L0,W,d0,pp);
            t=t+1;
            p(t)=D(1,1);%记录最大利用率
        end
        %绘制图像
        x=d(i)-1:0.5:d(i)+1;
        plot(x,p)
        title(a(i));
    end
    for i=1:2
        subplot(2,4,i+6)
        t=0;
        p=[];
        for j=-1:0.5:1
            pp0=pp;

```

```

        pp0(i)=pp0(i)+j;
        D=moxin2(L0,W,d,pp0);
        t=t+1;
        p(t)=D(1,1);%记录最大利用率
    end
    %绘制图像
    x=pp(i)-1:0.5:pp(i)+1;
    plot(x,p)
    title(b(i));
end

```

#### 附录 15：剪冲式模型函数

```

function D=moxin2(L0,W0,d,pp)
%返回切割方式利用率最大的五个数组
%输入原材料长 宽，印章的尺寸
y=6;%裁剪种类
%d=[29,21,20,19,45,40];%半径和横径
%pp=[30,30];
for i=1:4
    V(i)=pi*d(i)^2*20;%圆体积
end
for i=5:6
    V(i)=pi*d(i)*pp(i-4)*20/4;%椭圆体积
end
V0=4550*1630*120;%总体积
p=[];n=[];
for i=1:20000%随机次数
    L=L0;W=W0;
    t=randperm(y);%随机排列插入顺序
    k=ceil(rand()*2);%随机第一次切割方向
    t1(i,:)=t;%无意义 储存数据
    if k==1%第一刀横切
        for j=1:6
            if j==1||j==3||j==5%横切
                [n(i,j),W]=heng(L,W,t(j),pp,d);
            elseif j==2||j==4%竖切
                [n(i,j),L]=shu(L,W,t(j),pp,d);
            else%第六块不需切刀
                if t(j)<=4
                    n(i,j)=yuan(L,W,d(t(j)));
                else
                    n(i,j)=tuoyuan(L,W,d(t(j)),pp(j-4));
                end
            end
        end
    end
end
end

```

```

elseif k==2%第一刀竖切
    for j=1:6
        if j==1||j==3||j==5%竖切
            [n(i,j),L]=shu(L,W,t(j),pp,d);
        elseif j==2||j==4%横切
            [n(i,j),W]=heng(L,W,t(j),pp,d);
        else%第六块不需切刀
            if t(j)<=4
                n(i,j)=yuan(L,W,d(t(j)));
            else
                n(i,j)=tuoyuan(L,W,d(t(j)),pp(j-4));
            end
        end
    end
end
end
V1=0;%初始化
for o=1:6
    V1=n(i,o)*V(t(o))+V1;%生产产品总体积
end
p(i)=V1/V0;%计算利用率
end
for i=1:5
    [D(i,1),n1]=max(p);
    D(i,2:7)=t1(n1,:);
    D(i,8:13)=n(n1,:);
    p(n1)=[];t1(n1,:)=[];n(n1,:)=[];
end

```

附录 16: ASA 算法所用 matlab 程序

```

% %ASA 算法
% clear all
% clc
% close
function [t,n]=asa(L,W)
% A=[4550,1630,120];
% L=445;W=1000;
rectangle('Position',[0,0,L,W]);
hold on;
title('asa')
axis([0 L 0 W])
axis equal
%p5 材料尺寸
d=[];

%for k=1:100

```

```

D=[58, 42, 40, 38];
r=D./2;
S=4550*1630*120;%原材料体积
n=zeros(4,1);
xi=D(4)/2;yi=D(4)/2;ri=D(4)/2;n(4)=n(4)+1;
data(1,:)=[xi,yi,ri];
p=ceil(rand()*4);
n(p)=n(p)+1;
s1=pi*29^2*20;
s2=pi*21^2*20;
s3=pi*20^2*20;
s4=pi*19^2*20;
[f,x,y]=panduan(r(p),data(:,1),data(:,2),data(:,3),L,W);

t=2;
while f==1
    if x>0&&y>0
        data(t,:)=[x,y,r(p)];
        t=t+1;
    end
    p=ceil(rand()*4);
    n(p)=n(p)+1;
    [f,x,y]=panduan(r(p),data(:,1),data(:,2),data(:,3),L,W);
end

%%%添加
for i=1:4
    f=1;
    while f==1
        [f,x,y]=panduan(r(i),data(:,1),data(:,2),data(:,3),L,W);
        if f==0
            continue
        end
        data(t,:)=[x,y,r(i)];
        t=t+1;
    end
end
n=n*6;
t=(n(1)*s1+n(2)*s2+n(3)*s3+n(4)*s4)/(L*W*120);
%end
[m,~]=size(data);
for i=1:m

```



```

rectangle('Position',[data(i,1)-data(i,3),data(i,2)-data(i,3),data(i,3)*2,data(i,3)*2], 'Curvature',[1,1]);
end

```

#### 附录 17：圆的见缝插针利用率函数

```

%计算圆的见缝插针利用率
function n=yuan(L,W,r)
%L W 为原材料长宽 r 为圆的半径
%
if L==0||W==0
    n=0;
end
if L~=0&&W~=0
    b=r*sqrt(3);%第一列和第二列横坐标距离
    W0=floor(W/(2*r));%最多摆放行数
    if W0==0
        n=0;
    end
    if W0==1
        n=floor(L/(2*r))*6;
    end
    if W0>1
        if L<2*r
            n=0;
        elseif L>=2*r
            L0=floor((L-2*r)/(2*b));%列数
            L1=L0+1;L2=L0;%L1 表示正常摆放列数 L2 表示见缝插针形的列数
            if L-(L0*2*b+2*r)>=b %判断是否可以再加入一列
                L2=L2+1;
            end
            W1=W0;
            W2=W0-1;%W1 表示正常排列行数 W2 表示见缝插针方式行数
            n=(L2*W2+L1*W1)*6;%产生产品个数
        end
    end
end
end

```

#### 附录 18：剪冲式改进模型所用函数

```

function [t1,n,p]=shiti3(Ly,Wy,cishu,d,pp)
%返回各种类排序 数目 和 方法的利用率
%输入原材料长宽,随机次数
y=6;%裁剪种类
%d=[29,21,20,19,45,40];%半径和横径
for i=1:4
    V(i)=pi*d(i)^2*20;%圆体积
end

```

```

for i=5:6
    V(i)=pi*d(i)*pp(i-4)*20/4;%椭圆体积
end
V0=Ly*Wy*120;%总体积
p=[];n=[];
for i=1:cishu%随机次数
    L=Ly;W=Wy;
    t=randperm(y);%随机排列插入顺序
    k=ceil(rand()*2);%随机第一次切割方向
    t1(i,:)=t;%无意义 储存数据
    if k==1%第一刀横切
        for j=1:6
            if j==1||j==3||j==5%横切
                [n(i,j),W]=heng(L,W,t(j),pp,d);
            elseif j==2||j==4%竖切
                [n(i,j),L]=shu(L,W,t(j),pp,d);
            else%第六块不需切刀
                if t(j)<=4
                    n(i,j)=yuan(L,W,d(t(j)));
                else
                    n(i,j)=tuoyuan(L,W,d(t(j)),pp(j-4));
                end
            end
        end
    end
    elseif k==2%第一刀竖切
        for j=1:6
            if j==1||j==3||j==5%竖切
                [n(i,j),L]=shu(L,W,t(j),pp,d);
            elseif j==2||j==4%横切
                [n(i,j),W]=heng(L,W,t(j),pp,d);
            else%第六块不需切刀
                if t(j)<=4
                    n(i,j)=yuan(L,W,d(t(j)));
                else
                    n(i,j)=tuoyuan(L,W,d(t(j)),pp(j-4));
                end
            end
        end
    end
end
V1=0;%初始化
for o=1:6
    V1=n(i,o)*V(t(o))+V1;%生产产品总体积
end
p(i)=V1/V0;%计算利用率

```

end

#### 附录 19：问题三所用函数

%第三问

%运行时间约为 1 秒，因随机排布，每种生产个数运行出答案与结果展示有差异  
%利用率无明显变化。

function [p0,K0,K11]=shiti32(L,W,d,pp,need)

%输出总利用率 方案的生产产品个数 切割原材料数

%输入原材料长宽 产品尺寸 需求量

% clear

% clc

% L=4550;W=1630;

% d=[29, 21, 20, 19, 45, 40];%半径和横径

% pp=[30, 30];

% need=[774000 2153000 1623000 1614000 848000 1014000];%需求量

K0=[];

for i=1:4

    V(i)=pi\*d(i)^2\*20;%圆体积

end

for i=5:6

    V(i)=pi\*d(i)\*pp(i-4)\*20/4;%椭圆体积

end

V0=L\*W\*120;%总体积

need\_1=need;%储存需求量数据

% for o=1:20%随机次数

    tc=0;%记录使用切割方案数

    [t,N,p]=shiti3(L,W,20000,d,pp);%方法利用率和产品数量

    need=need\_1;%需求量

    p=p';%转置

    t1=0;%计算所需总原材料数

    n=[];

    %%进行排序 使得 n 中按[1 2 3 4 5 6]种类产品排序

    for i=1:20000

        [~,b]=sort(t(i,:));

        for j=1:6

            T=N(i,:);

            n(i,j)=T(b(j));

        end

    end

    [~,n1]=max(p);%最大利用率的行数

    D=[p,n];%储存利用率和产品数

    t2=n(n1,:);%第一种切割方式产品生产数目

    P=ones(1,6);%用于判断是否满足需求 产生 6 个 1

    P1=P;P2=P;

    while ismember(0,P1)==0%当 P1 中有 0 是跳出循环 即有一种产品已到达需求量

```

    need=need-t2;%需求量减去产生量
    t1=t1+1;%消耗原材料加一
    for i=1:6%判断哪一种物料达到需求
        if need(i)<0
            P1(i)=0;%若 i 种物料达到需求则 P1 (i) 为 0
        end
    end
end
a=find(P1==0);%找出满足需求的数量
len=length(a);
D1=D;
for i=1:len
    D1=D1(find(D1(:,a(i)+1)==0),:);
end
tc=tc+1;%采用切割方法个数
K(tc,:)= [t2,t1];%储存切割方法的生产个产品数，切割原材料数。

while len~=6%如果全都满足需求则跳出循环
    [~,n1]=max(D1(:,1));%最大利用率的行数
    t2=D1(n1,2:7);%产品生产数目
    P1=P;
    tt=0;%计算切割方案所需原材料数
    while ismember(0,P1)==0%当 P1 中有 0 是跳出循环
        P1=P;P2=P;
        need=need-t2;%需求数-生产数
        t1=t1+1;%切割原材料总数
        tt=tt+1;%该切割方式切割原材料数
        for i=1:6%判断哪一种物料达到需求
            if need(i)<0
                P1(i)=0;P2(i)=0;
            end
        end
        end
        P1(a)=[];
    end
    a=find(P2==0);%找到满足条件的产品种类
    D1=D;
    for i=1:length(a)%D1 为不产生以满足条件产品的切割方式
        D1=D1(find(D1(:,a(i)+1)==0),:);
    end
    tc=tc+1;
    K(tc,:)= [t2,tt];%储存切割方法的生产个产品数，切割原材料数。
    if tc==4%当切割方法达到四种时 则最后一种方法需满足剩余要求
        a0=[1 2 3 4 5 6];
        a0(a)=[];
    end
end

```

```

for i=1:length(a0)
    D1(find(D1(:,a0(i)+1)==0),:)=[];
end
if length(a0)==2
    deta=abs(D1(:,a0(1)+1)-D1(:,a0(2)+1));%计算产品生成数之差
    [~,n1]=min(deta);%差值最小的行数
    t2=D1(n1,2:7);%产品生产数目
elseif length(a0)==1
    deta=D1;
    [~,n1]=max(D1(:,a0));%a1 最大的行数
    t2=D1(n1,2:7);;%产品生产数目
end
P1=P;tt=0;
while sum(P1)~=0
    P1=P;
    need=need-t2;%需求数-生产数
    t1=t1+1;tt=tt+1;
    for i=1:6
        if need(i)<0
            P1(i)=0;
        end
    end
    end
    tc=tc+1;
    K(tc,:)= [t2,tt];
    break
end
end
%    pval(o)=need_1*V'/(V0*t1);%记录每一次的总利用率
%    if o==1
%        K0=K;
%    else
%        if pval(o)>pval(o-1)%选取总利用率最大的方案，并记录其数据
%            K0=K;
%        end
%    end
%end
for i=1:5
    K0(i,1:6)=K0(i,1:6).*K0(i,7);
    K0(i,8)=(K0(i,1:6)*V')/(K0(i,7)*V0);
end
K0;%输出每一种切割方案的切割数，所切原材料数，利用率。
K11=sum(K0(:,1:7));%输出每种印章一共产生多少个。
p0=sum(K0(:,1:6))*V'/(V0*(K11(7)));%总利用率

```

附录 20: 问题三模型运行程序与结果输出所用 matlab 程序

%问题三模型运行程序与结果输出

%运行时间约半分钟

clear

clc

L=4550;W=1630;%原材料长宽

d=[29, 21, 20, 19, 45, 40];%半径和横径

pp=[30, 30];

need=[774000 2153000 1623000 1614000 848000 1014000];%需求量

for i=1:20%因随机故运行 20 次取最大利用率的情况

[p(i), k1, k2]=shiti32(L, W, d, pp, need);

if i>=2

if p(i)>p(i-1)

K1=k1;K2=k2;P=p(i);

end

end

end

P

K1

K2

%P 为总利用率

%K1 中 1 至 6 列为不同切割方法生产产品总数 第一行为第一种切割方式 第一列为 P1 的生产数

%K1 的第 7 列为不同切割方法各消耗的原材料数

%K1 的第 8 列为不同切割方法的利用率

%K2 为 P1...P6 的生产总数 和总消耗原材料数

%因随机排布, 每种生产个数运行出答案与结果展示有差异

附录 21: 问题三灵敏度分析所用 matlab 程序

%对问题三进行灵敏度分析

%此程序全部运行时间约 1 分钟

clear

clc

L=4550;W=1630;

d=[29, 21, 20, 19, 45, 40];%半径和横径

pp=[30, 30];

need=[774000 2153000 1623000 1614000 848000 1014000];%需求量

%对原材料长宽灵敏度

t=0;P=[];

for i=-5:5:5%检验范围

L0=L;

L0=L+i;%改变 L 的长度

t=t+1;p=0;

[p, ~, ~]=shiti32(L0, W, d, pp, need);

P(t)=p;%记录最大利用率

```

end
x=L-5:5:L+5;
%绘制图像
figure
subplot(1,2,1)
plot(x,P)
title('长度');
t=0;P=[];
for i=-5:5:5%检验范围
    W0=W;
    W0=W+i;%改变 W 的宽度
    t=t+1;
    p=shiti32(L,W0,d,pp,need);
    P(t)=p;%记录最大利用率
end
x=W-5:5:W+5;
subplot(1,2,2)%绘制图像
plot(x,P)
title('宽度');

a={'P1','P2','P3','P4','P5 横径','P6 横径'};
b={'P5 竖径','P6 竖径'};
c={'P1 需求','P2 需求','P3 需求','P4 需求','P5 需求','P6 需求'};
%改变产品尺寸绘制灵敏度
figure
for i=1:6%对 d 中尺寸进行遍历
    subplot(2,4,i)%都绘制在一个窗口
    t=0;
    P=[];
    for j=-1:0.5:1
        d0=d;
        d0(i)=d0(i)+j;
        p=shiti32(L,W,d0,pp,need);
        t=t+1;
        P(t)=p;%记录最大利用率
    end
    %绘制图像
    x=d(i)-1:0.5:d(i)+1;
    plot(x,P)
    title(a(i));
end
for i=1:2
    subplot(2,4,i+6)

```

```

t=0;
P=[];
for j=-1:0.5:1
    pp0=pp;
    pp0(i)=pp0(i)+j;
    p=shiti32(L,W,d,pp0,need);
    t=t+1;
    P(t)=p;%记录最大利用率
end
%绘制图像
x=pp(i)-1:0.5:pp(i)+1;
plot(x,P)
title(b(i));
end
%改变需求量灵敏度
figure
for i=1:6
    subplot(2,3,i)
    t=0;
    P=[];
    for j=-100000:50000:100000
        need0=need;
        need0(i)=need0(i)+j;
        p=shiti32(L,W,d,pp,need0);
        t=t+1;
        P(t)=p;%记录最大利用率
    end
    %绘制图像
    x=need(i)-100000:50000:need(i)+100000;
    plot(x,P)
    title(c(i));
end

```

附录 22: 计算矩形利用率所用函数

```

function n=juxin(L,W,d,D)
a=find(D(7:9)==d);
num=[2 3 4];
L0=floor(L/d);
W0=floor(W/d);
n=L0*W0*num(a);

```

附录 23: 问题四模型运行程序与结果输出所用 matlab 程序

%问题四模型运行程序与结果输出

%运行时间约 1-5 分钟

clear

clc



```

L=4550;W=1630;%原材料长宽
d=[29, 21, 20, 19, 45, 40, 20, 18, 15];%半径和横径
pp=[30, 30];%竖径
need=[774000 2153000 1623000 1614000 848000 1014000 2546000 3421000 2687000];%
需求量
[p, K1, K2]=shiti42(L, W, d, pp, need);
while K2(end)>600
    [p, K1, K2]=shiti42(L, W, d, pp, need);
end
p
K1
K2
%p 为总利用率
%K1 中 1 至 6 列为不同切割方法生产产品总数 第一行为第一种切割方式 第一列为 P1
的生产数
%K1 的第 7 列为不同切割方法各消耗的原材料数
%K1 的第 8 列为不同切割方法的利用率
%K2 为 P1...P6 的生产总数 和总消耗原材料数
%因随机排布，每种生产个数运行出答案与结果展示有差异
附录 24：问题四进行灵敏度分析所用 matlab 程序
%对问题四进行灵敏度分析
%此程序全部运行时间约 4 分钟 可分图像进行输出
clear
clc
L=4550;W=1630;
d=[29, 21, 20, 19, 45, 40, 20, 18, 15];%半径和横径
pp=[30, 30];%竖径
need=[774000 2153000 1623000 1614000 848000 1014000 2546000 3421000 2687000];%
需求量
%对原材料长宽灵敏度
t=0;P=[];
for i=-5:5:5%检验范围
    L0=L;
    L0=L+i;%改变 L 的长度
    t=t+1;p=0;
    [p, ~, ~]=shiti42(L0, W, d, pp, need);
    P(t)=p;%记录最大利用率
end
x=L-5:5:L+5;
%绘制图像
figure
subplot(1, 2, 1)
plot(x, P)
title('长度');

```

```

t=0;P=[];
for i=-5:5:5%检验范围
    W0=W;
    W0=W+i;%改变 W 的宽度
    t=t+1;
    p=shiti42(L, W0, d, pp, need);
    P(t)=p;%记录最大利用率
end
x=W-5:5:W+5;
subplot(1,2,2)%绘制图像
plot(x,P)
title('宽度');

a={'P1','P2','P3','P4','P5 横径','P6 横径','P7','P8','P9'};
b={'P5 竖径','P6 竖径'};
c={'P1 需求','P2 需求','P3 需求','P4 需求','P5 需求','P6 需求','P7 需求','P8 需求','P9 需求'};
%改变产品尺寸绘制灵敏度
figure
for i=1:9%对 d 中尺寸进行遍历
    subplot(3,3,i)%都绘制在一个窗口
    t=0;
    P=[];
    for j=-1:0.5:1
        d0=d;
        d0(i)=d0(i)+j;
        p=shiti42(L, W, d0, pp, need);
        t=t+1;
        P(t)=p;%记录最大利用率
    end
    %绘制图像
    x=d(i)-1:0.5:d(i)+1;
    plot(x,P)
    title(a(i));
end
figure
for i=1:2
    subplot(1,2,i)
    t=0;
    P=[];
    for j=-1:0.5:1
        pp0=pp;
        pp0(i)=pp0(i)+j;

```

```

        p=shiti42(L, W, d, pp0, need);
        t=t+1;
        P(t)=p;%记录最大利用率
    end
    %绘制图像
    x=pp(i)-1:0.5:pp(i)+1;
    plot(x, P)
    title(b(i));
end
%改变需求量灵敏度
figure
for i=1:9
    subplot(3, 3, i)
    t=0;
    P=[];
    for j=-100000:50000:100000
        need0=need;
        need0(i)=need0(i)+j;
        p=shiti42(L, W, d, pp, need0);
        t=t+1;
        P(t)=p;%记录最大利用率
    end
    %绘制图像
    x=need(i)-100000:50000:need(i)+100000;
    plot(x, P)
    title(c(i));
end

```

#### 附录 25： 问题五所用 matlab 程序

```

clear
clc
L=4550;W=1630;%原材料长宽
d=[29, 21, 20, 19, 45, 40, 19, 18, 15];%半径和横径
pp=[30, 30];%竖径
t=[12 8 7 6 20 18 4 2 1];%产品单价
cishu=50000;
[t1, N, p]=shiti5(L, W, cishu, d, pp);
p=p';%转置
%%进行排序 使得 n 中按[1 2 3 4 5 6 7 8 9]种类产品排序
for i=1:cishu
    [~, b]=sort(t1(i, :));
    for j=1:9
        T=N(i, :);
        n(i, j)=T(b(j));
    end
end

```

```

end
d=(n*t'-100000)*100;%计算 100 个时的利润
[num,a]=max(d);%num 为最大利润值, a 为该利润值时的行数
D=[p(a),n(a,:)*.100,num]%记录利润最大时的方案
附录 26: 问题五所用函数
function [t1,n,p]=shiti5(Ly,Wy,cishu,d,pp)
%返回 t1 各种类排序 n 数目 和 p 方法的利用率
%输入原材料长宽 LW, 随机次数 cishu, 产品的尺寸 d pp
y=length(d);%裁剪种类
%d=[29,21,20,19,45,40];%半径和横径
for i=1:4
    V(i)=pi*d(i)^2*20;%圆体积
end
for i=5:6
    V(i)=pi*d(i)*pp(i-4)*20/4;%椭圆体积
end
for i=7:9
    VV=[60 40 30];
    V(i)=d(i)^2*VV(i-6);%矩形体积
end
V0=Ly*Wy*120;%总体积
p=[];n=[];
for i=1:cishu%随机次数
    L=Ly;W=Wy;
    t=randperm(y);%随机排列插入顺序
    k=ceil(rand()*2);%随机第一次切割方向
    t1(i,:)=t;%无意义 储存数据
    if k==1%第一刀横切
        for j=1:9
            if j==1||j==3||j==5||j==7%横切
                [n(i,j),W]=heng(L,W,t(j),pp,d);
            elseif j==2||j==4||j==6||j==8%竖切
                [n(i,j),L]=shu(L,W,t(j),pp,d);
            else%第九块不需切刀
                if t(j)<=4
                    n(i,j)=yuan(L,W,d(t(j)));
                elseif t(j)<=6
                    n(i,j)=tuoyuan(L,W,d(t(j)),pp(t(j)-4));
                else
                    n(i,j)=juxin(L,W,d(t(j)),d);
                end
            end
        end
    end
    elseif k==2%第一刀竖切

```

```

for j=1:9
    if j==1||j==3||j==5||j==7%竖切
        [n(i,j),L]=shu(L,W,t(j),pp,d);
    elseif j==2||j==4||j==6||j==8%横切
        [n(i,j),W]=heng(L,W,t(j),pp,d);
    else%第九块不需切刀
        if t(j)<=4
            n(i,j)=yuan(L,W,d(t(j)));
        elseif t(j)<=6
            n(i,j)=tuoyuan(L,W,d(t(j)),pp(t(j)-4));
        else
            n(i,j)=juxin(L,W,d(t(j)),d);
        end
    end
end
end
V1=0;%初始化
for o=1:9
    V1=n(i,o)*V(t(o))+V1;%生产产品总体积
end
p(i)=V1/V0;%计算利用率
end

```