

# 创意平板折叠桌

## 摘要

本文针对创意平板折叠桌的分析和设计问题,建立了基于空间解析几何的动态变化模型并运用了插值拟合的方法,解决了折叠桌的动态变化过程的描述问题以及设计加工参数的确定问题;建立了基于遗传算法的多目标优化模型,解决了折叠桌的最优设计加工参数的确定问题;建立了基于多目标优化的折叠桌设计通用模型,解决了设计软件的模型设计问题,并设计了两个创意平板折叠桌。

针对问题一,建立了基于空间解析几何的动态变化模型并运用了插值拟合的方法,解决了折叠桌的动态变化过程的描述问题以及设计加工参数的确定问题;首先,以木板下表面中心为圆心建立空间直角坐标系;其次利用空间几何的相关知识,建立了基于空间解析几何的动态变化模型,以最外侧木条与桌面的夹角为自变量描述内侧木条与桌面的夹角以及桌脚的边缘位置;然后当木条转动到最终位置时,确定桌腿木条开槽的长度和桌脚边缘的坐标,并通过插值拟合方法得到桌脚边缘线的数学描述。最后检验了模型的灵敏度和稳定性。最终得到了折叠桌的动态变化过程和桌脚边缘线的数学描述,木条开槽长度分别为 20.01、19.60、18.77、17.59、16.01、14.14、11.81、9.00、5.53、0cm。

针对问题二,建立了基于遗传算法的多目标优化模型,解决了折叠桌的最优设计加工参数的确定问题;首先将折叠桌的稳固性、加工系数、用材分别用数学语言描述;其次确定稳固性好为第一目标,加工方便为第二目标,用材最少为第三目标,根据约束条件,建立了多目标优化模型,并用惯序算法进行求解;然后使用遗传算法对已求得的较优解进一步优化。最后检验了模型的灵敏度和稳定性。最终得到当桌高 70cm、桌面直径 80 cm 时,平板长度为 158.0020cm,平板宽度为 79.9000cm,桌腿宽度为 4.7cm,单侧桌腿条数为 17 条,最外侧桌腿钢筋位置比率为 0.4857,最外侧桌腿钢筋位置距铰链距离 39.6027cm,开槽长度由外而内分别为 0、11.8615、18.9448、24.5395、28.9823、32.4182、34.9331、36.5821、37.3986cm,开槽总长度为 827.8432cm。

针对问题三,建立了基于多目标优化的折叠桌设计通用模型,解决了设计软件的模型设计问题,并设计了两个创意平板折叠桌。首先建立了折叠桌设计通用模型,并提出模型的适用条件;其次重新建立基于遗传算法的多目标优化模型,得到参数优化的通用模型;最后设计了会议桌和椭圆桌并得到了优化参数。最终得到高度为 70cm、矩形长度为 60cm、矩形宽度为 40cm 的会议桌的平板长度为 149.7071cm,平板宽度为 59.8cm,桌腿宽度为 4.6cm,单侧桌腿条数为 13 条,最外侧桌腿钢筋位置为 0.3443,最外侧桌腿钢筋位置距铰链距为 47.4788cm,开槽长度由外而内分别为 0、10.3419、15.7401、19.5900、22.3116、24.0656、24.9268cm,开槽总长度为 418.0505cm;高度为 70cm、椭圆短轴为 60cm、椭圆长轴为 80cm 的椭圆桌的平板长度为 153.6617cm、平板宽度为 59.4cm、桌腿宽度为 2.7cm、单侧桌腿条数为 22 条、最外侧桌腿钢筋位置比率为 0.5814、最外侧桌腿钢筋位置距铰链距离 29.7965cm、开槽长度由内而外分别为 0、10.0928、17.1114、22.8413、27.5951、31.5261、34.7319、37.2803、39.2191、40.5821、41.3914cm,开槽总长度: 1209.5cm。

**关键词:** 空间解析几何 插值拟合 遗传算法 多目标优化模型 惯序算法

## 一、问题重述

某公司生产一种可折叠的桌子，桌面呈圆形，桌腿随着铰链的活动可以平摊成一张平板（如图 1-2 所示）。桌腿由若干根木条组成，分成两组，每组各用一根钢筋将木条连接，钢筋两端分别固定在桌腿各组最外侧的两根木条上，并且沿木条有空槽以保证滑动的自由度（见图 3）。桌子外形由直纹曲面构成，造型美观。附件视频展示了折叠桌的动态变化过程。

试建立数学模型讨论下列问题：

1. 给定长方形平板尺寸为  $120\text{ cm} \times 50\text{ cm} \times 3\text{ cm}$ ，每根木条宽  $2.5\text{ cm}$ ，连接桌腿木条的钢筋固定在桌腿最外侧木条的中心位置，折叠后桌子的高度为  $53\text{ cm}$ 。试建立模型描述此折叠桌的动态变化过程，在此基础上给出此折叠桌的设计加工参数（例如，桌腿木条开槽的长度等）和桌脚边缘线（图 4 中红色曲线）的数学描述。

2. 折叠桌的设计应做到产品稳固性好、加工方便、用材最少。对于任意给定的折叠桌高度和圆形桌面直径的设计要求，讨论长方形平板材料和折叠桌的最优设计加工参数，例如，平板尺寸、钢筋位置、开槽长度等。对于桌高  $70\text{ cm}$ ，桌面直径  $80\text{ cm}$  的情形，确定最优设计加工参数。

3. 公司计划开发一种折叠桌设计软件，根据客户任意设定的折叠桌高度、桌面边缘线的形状大小和桌脚边缘线的大致形状，给出所需平板材料的形状尺寸和切实可行的最优设计加工参数，使得生产的折叠桌尽可能接近客户所期望的形状。你们团队的任务是帮助给出这一软件设计的数学模型，并根据所建立的模型给出几个你们自己设计的创意平板折叠桌。要求给出相应的设计加工参数，画出至少 8 张动态变化过程的示意图。



图 1



图 2



图 3



图 4

附件：视频

## 二、问题分析

### 2.1 问题一的分析

该问是一个给定平板尺寸，每根木条宽度，折叠后桌子的高度，要求建立模型描述此折叠桌的动态变化过程，并求出折叠桌的设计加工参数和桌角边缘的数学描述的问题。题中要求折叠后桌子的高度为 53cm，而由于每根木条的厚度为 3cm，假设桌腿铰链固定在桌面下边缘，所以折叠后桌子圆形桌面下底面到地面的高度 50cm。

为了描述其动态过程，对于给定的实物，通常找出其影响运动的特征点，而忽略具体外观形状<sup>[1]</sup>。通过对视频的观察，我们发现该折叠桌的动态过程可以看作是从木板平放时的平板形状开始，随着桌子高度的变化，各桌腿与圆形桌面的夹角，各桌腿端点位置，各桌腿中钢筋到相应铰链的距离也相应发生变化，因此

可以用这些参数随折叠桌高度的变化来描述此折叠桌的动态变化过程。在描述折叠桌的动态变化过程时，可以建立相应空间直角坐标系，然后确定各个坐标点，利用空间几何的相关知识和题目所给条件画出其动态变化的示意图，并结合坐标变化描述折叠桌的动态变化过程。

对于桌腿木条开槽长度的求解问题，可以求出桌子平铺时各桌腿中钢筋到铰链的距离和折叠完成后各桌腿中钢筋到铰链距离之和，在折叠过程中，各桌腿中钢筋到铰链的距离随最外侧桌腿与圆形桌面下底面向圆心延长线的夹角减小而单调递增。

对于桌脚边缘线的数学描述问题，可先求解折叠桌动态变化结束后，各个桌腿末端所对应的坐标，即求出题中图 4 红色曲线上各桌腿端点坐标。由于坐标是离散的点，所以对坐标进行插值拟合<sup>[2]</sup>，拟合出的连续曲线即为桌脚边缘线的数学描述。

## 2.2 问题二的分析

该问要求讨论给定任意折叠桌高度和圆形桌面直径最优设计。折叠桌的设计应做到产品的稳固性好、加工方便、用材最少。此问题是一个多目标规划问题，将稳固性好作为第一目标，加工方便作为第二目标，用材最少作为第三目标，根据约束条件进行求解。对于稳固性，可用最外侧木条与最内侧木条的夹角决定<sup>[3]</sup>；对于加工方便，可用开槽总长度和开槽深度表示；对于用材最少，可用所需木板的体积表示。再在此基础上，求出最优的钢筋位置、开槽长度、木板条数并用智能算法对参数进行优化得到精确解。

## 2.3 问题三的分析

该问要求讨论对于任意高度、任意桌面形状、任意桌脚边缘线的大致形状的情况下，求出平板材料的形状尺寸和切实可行的最优设计加工参数，并设计几种自己设计的创意平板折叠桌。此问题要求做出折叠桌加工过程中的通用模型，并对加工参数进行优化。对于此类问题，首先应约定坐标系并建立折叠桌加工过程中的通用模型；其次应建立相应的约束条件和目标函数；然后在满足约束条件和目标函数值最优的条件下，确定木板的大小和参数；最后优化参数。对于设计创意平板折叠桌问题，首先应根据约束条件大致确定可用的桌面边缘线的形状，其次制定合理的大小并利用通用模型进行求解。

# 三、基本假设

- 1、桌面直径为长方形平板的宽；
- 2、不考虑相邻桌腿之间空隙；
- 3、忽略木条与桌面连接轴承的厚度；
- 4、木板厚度不变。

#### 四、符号说明

$b$	木板的宽度
$n$	木条的个数
$r$	圆形桌面的半径
$\alpha$	最外侧木条与地面的夹角
$\beta$	其他木条与地面的夹角
$l_i$	第 $i$ 根桌腿的长度
$a$	木板的长
$h$	桌面的厚度
$Q_i$	桌脚的边缘点
$f_i$	第 $i$ 个目标函数
$d_i^+$	正偏差变量
$d_i^-$	负偏差变量
$\alpha$	折叠桌与地面之间的摩擦力
$V$	木板的体积
$L$	木板的厚度
$f_i(x)$	待优化的目标函数
$lb$	下限约束
$ub$	上限约束
$l$	开槽总长度

#### 五、模型的建立与求解

##### 5.1 问题一模型的建立与求解

该问要求我们建立描述此折叠桌的动态变化过程的模型，此折叠桌的设计加工参数和桌脚边缘线的数学描述。首先，我们建立空间直角坐标系。其次，由于桌面半径并不等于长方形平板的宽，所以我们先要用勾股定理计算出桌面半径，从而利用几何关系算出桌腿长度。再次，利用相似三角形，旋转角与三角函数的关系计算出动态变化过程中桌脚边缘点与铰链位置的坐标，从而利用距离公式计算出开槽长度。最后对桌脚边缘点进行拟合画图，得出桌脚边缘线的曲线方程与动态过程示意图。

##### 5.1.1 模型的建立

首先，建立空间直角坐标系。为了方便研究折叠桌的动态变化过程，我们将折叠桌抽象为几何图形，本文以圆盘桌面上表面圆心为原点，以圆盘桌面表面所

在平面为  $xOy$  面，垂直于桌面且方向指向地面为  $z$  轴正半轴，建立三维空间直角坐标系，如图 5.2.1.1 所示：

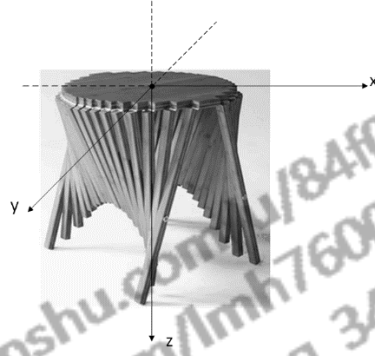


图 5.1.1.1 空间直角坐标系的建立

木板上桌面与木条的分布如图 5.2.1.2 所示，根据视频我们得知该折叠桌是前后左右对称的，具有对称性，所以只需考虑桌子的四分之一即可。在这里，我们选取木板四分之一进行研究。其他部分可通过对称性求出。

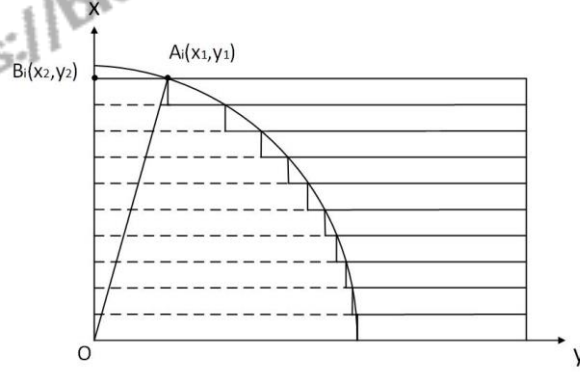


图 5.1.1.2

其中， $A_i (i = 1, 2, \dots, n)$  表示木条与圆的相接点， $A_1$  为最外侧木条； $B_i (i = 1, 2, \dots, 10)$  表示第  $i$  根木条与圆形桌面交点处的延长线与圆中心轴线的交点。

为了保证圆形桌面的对称性及美观性，我们可以认为  $A_n B_n$  的长度近似等于木条的宽度，即：

$$A_1 B_1 = w = \frac{b}{n}$$

其中， $w$  为木条的宽度， $b$  为木板的宽度， $n$  为木条的个数。

在  $\text{Rt}\triangle A_i B_i O$  中，根据勾股定理，得：

$$A_i B_i^2 + B_i O^2 = O A_i^2$$

由此得出圆的半径  $R = O A_1$ ，则圆的方程为：

$$x^2 + y^2 = \left(\frac{b}{2}\right)^2 + (w)^2$$

得到第  $i$  根桌腿的长度  $l_i$ ：

$$l_i = \frac{a}{2} - A_i B_i = \frac{a}{2} - \sqrt{R^2 - B_i O^2} = \frac{a}{2} - \sqrt{R^2 - \left(\frac{R}{2} - (i-1)\frac{R}{n}\right)^2}$$

其中， $a$  为木板的长。



以其中任意一根木条 $A_i$ 为研究对象, 取其折叠过程中的任一时刻, 如图 5.2.1.3 所示:

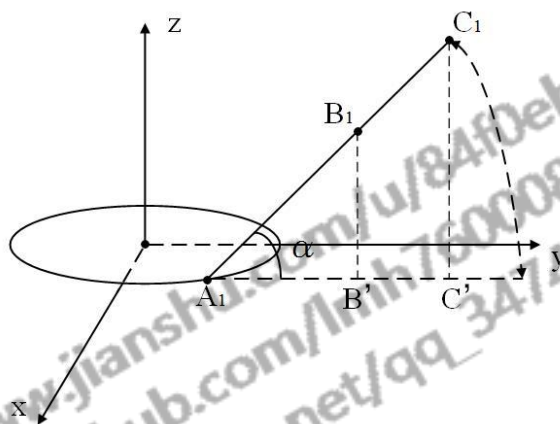


图 5.2.2.3

通过观察,我们发现,折叠过程中,如果以每一根木条为研究对象,则每一根木条的  $x$  坐标保持不变;如果以钢筋为研究对象,则折叠过程中同一条钢筋上的所有几何点的  $x$  坐标、 $y$  坐标都是相同的,且钢筋在最外侧木条的中点位置不发生变化。钢筋的变化过程可以看作是其围绕一条与它平行的直线进行圆周运动,故该钢筋的运动轨迹是以其中一组桌脚最外侧两根桌腿的连线为旋转轴,最外侧即最长木条与桌面连接处到钢筋固定处距离为半径旋转而成的圆柱体。

则 $A_i$ 的坐标为:

$$A_i = (\frac{R}{2} - (i-1)\frac{R}{n}, \sqrt{(\frac{b}{2})^2 + (\frac{b}{n})^2 - [\frac{R}{2} - (i-1)\frac{R}{n}]^2}, h)$$

其中,  $h$  为桌面的厚度

接下来我们对 $C_i$ 点进行研究。通过观察,我们发现 $A_i, B_i, C_i$ 的横坐标是相同的,  $C_i$ 的 z 坐标是桌子的高度 H, 则 $C_i$ 的 y 坐标为 $\sqrt{(\frac{b}{2})^2 + (\frac{b}{n})^2 - [\frac{R}{2} - (i-1)\frac{R}{n}]^2} + \sqrt{(\frac{a}{2} - \frac{b}{n})^2 - H^2}$ , 即:

$$C_i = (\frac{R}{2} - (i-1)\frac{R}{n}, \sqrt{(\frac{b}{2})^2 + (\frac{b}{n})^2 - [\frac{R}{2} - (i-1)\frac{R}{n}]^2} + \sqrt{(\frac{a}{2} - \frac{b}{n})^2 - H^2}, H)$$

易知,  $\Delta A_i B_i B' \sim \Delta A_i C_i C'$ , 根据三角形的相似性得到:

$$\frac{B_i B'}{C_i C'} = \frac{A_i B'}{A_i C'}$$

即：

$$\begin{aligned} B_i B' &= \frac{A_i B'}{A_i C'} \times C_i C' \\ A_1 B' &= \frac{A_i B'}{A_i C'} \times A_i C' \end{aligned}$$

由此求出 $B_i$ 点的  $z$  坐标和  $y$  坐标。则 $B_i$ 点的坐标为:

$$B_i = (\frac{R}{2} - (i-1)\frac{R}{n}, B_i B', A_1 B')$$

在桌子折叠的过程中, 钢筋保持直线状态且始终与  $yOz$  面垂直。

各个阶段折叠桌的变化主要表现为各个点坐标的变化, 其形状由各参数决定, 因此只需通过比较计算出各时间内的点坐标 $C_i(x_i, y_i, z_i)$ , 即可大致描述各个点的整体变化趋势, 通过比较即可描述折叠桌的动态变化过程。

## 2、开槽长度的数学描述

木条开槽在收起和折叠时起到固定作用, 开槽长度决定了桌子高度和稳定性。在桌子收起作为普通木板时, 钢筋刚好处于木板最中间位置; 在完全折叠后, 钢筋则刚好处于该木条的顶端。两次最短木条连接处到钢筋的距离差即为中间木条开槽长度。

由上述过程已经求得 $A_i, B_i$ 点的坐标, 根据两点确定一条直线的数学定理, 可以确定 $A_i, B_i, C_i$ 三点所在的直线方程, 已知 $B_i, C_i$ 两点的坐标, 进而可以求得 $B_i, C_i$ 两点间的距离, 即:

$$\text{开槽长度} = \sum_{i=1}^n B_i C_i$$

其中,  $n$  为木板的根数。

## 3、桌脚边缘线的数学描述

桌脚边缘线即为折叠桌在固定时桌脚下方的离散点所连成的一条边缘线。设桌脚的边缘点为 $Q_i$ , 其空间直角坐标系中的坐标为 $(x_i, y_i, z_i)$ 。根据桌脚边缘点拟合出桌脚边缘线方程。

### 5.1.2 模型的求解

根据题目条件, 给定长方形平板尺寸为  $120\text{cm} \times 50\text{cm} \times 3\text{cm}$ , 每根木条宽  $2.5\text{cm}$ , 连接桌腿木条的钢筋固定在桌腿最外侧木条的中心位置, 折叠后桌子的高度为  $53\text{cm}$ , 由于木板厚度为  $3\text{cm}$ , 所以由地面到桌腿最顶端的垂直距离为  $50\text{cm}$ 。

由于桌子关于  $xoz$  和  $yoz$  平面对称, 所以只需要考虑四分之一桌面的情况。进行计算, 可以得出四分之一桌面内木条的长度如下:

表 5.2.2.1 四分之一桌面内每根木条长度

木条编号	1	2	3	4	5
木条长度 (cm)	35	35.38	36.02	36.95	38.21
木条编号	6	7	8	9	10
木条长度 (cm)	39.84	41.97	44.79	48.82	57.5

木条长度确定后, 可以算出桌腿运动的最终位置:

$$\alpha_{\text{end}} = 1.0543\text{rad}$$

木条运动过程的同时, 钢筋在滑槽内部运动, 始末位置确定后进行距离相减可确定开槽长度如下:

表 5.2.2.2 四分之一桌面内每根木条开槽长度

木条编号	1	2	3	4	5
开槽长度 (cm)	20.01	19.60	18.77	17.59	16.01



木条编号	6	7	8	9	10
开槽长度 (cm)	14.14	11.81	9.00	5.53	0

由此表，我们可以得出，编号越小的木条，即木条所在直线距离圆心越近の木条，其开槽长度越长。

根据前面的分析可知，折叠桌折叠的过程中， $\alpha$ 是由小变大的，通过改变 $\alpha$ 的值可得到不同时刻桌脚边缘点的坐标，再作图即可得出连续的桌角边缘线：

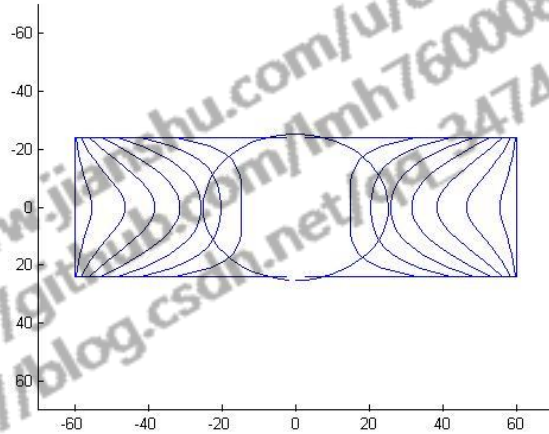


图 5.2.2.1 桌脚边缘线动态过程平面示意图

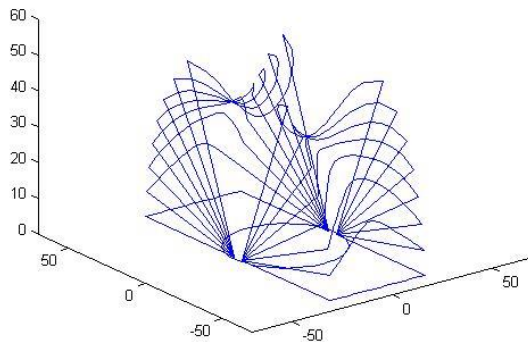


图 5.2.2.2 桌脚边缘线动态过程三维示意图

由桌脚边缘线的变化情况可以分析得出，随着 $\alpha$ 从 0 逐渐变大到极限位置，即折叠桌从平板状态到完全打开的过程中，桌脚边缘线的开口方向先是朝向桌面，然后到快要打开时才朝向地面，并且向桌面内部方向弯曲。

分别对 $y = f(x)$ ,  $z = g(x)$ 这 2 条曲线进行多项式拟合，可得曲线方程为：

$$y = f(x) = 0.002658x^3 - 0.05273x^2 + 0.2972x + 13.56$$

其中拟合标准差为 0.6227，确定系数为 0.987。

$$z = g(x) = -8.625 \times 10^{-5}x^4 + 0.003593x^3 - 0.01735x^2 + 0.2645x + 32.87$$

其中拟合标准差为 0.09843，确定系数为 0.9999。

拟合标准差越小，确定系数接近 1，表明方程的自变量对因变量的解释能力越强，即说明拟合曲线具有很高的可信度。

因此，四分之一桌面内桌角边缘线为:

$$\begin{cases} 0.002658x^3 - 0.05273x^2 + 0.2972x + 13.56 - y = 0 \\ -8.625 \times 10^{-5}x^4 + 0.003593x^3 - 0.01735x^2 + 0.2645x + 32.87 - z = 0 \end{cases}$$

### 5.1.3 模型的检验

动态变化过程中桌脚边缘点坐标 $C_i(x, y, z)$ 与 $\alpha$ 密切相关。由前面的推理过程知:

$$\begin{cases} x = x_0 \\ y = y_0 + l_i \times \cos\beta \\ z = l_i \times \sin\beta \end{cases}$$

其中,

$$\beta = \arctan \frac{\frac{1}{2}l_i \times \sin\alpha - z_0}{2.5 + \frac{1}{2}l_i \times \cos\alpha - y_0}$$

当 $\alpha \rightarrow \alpha + \Delta\alpha$ 时,

$$\Delta y = l_i \times \left[ \cos \left( \arctan \frac{\frac{1}{2}l_i \times \sin(\alpha + \Delta\alpha) - z_0}{2.5 + \frac{1}{2}l_i \times \cos(\alpha + \Delta\alpha) - y_0} \right) - \cos \left( \arctan \frac{\frac{1}{2}l_i \times \sin\alpha - z_0}{2.5 + \frac{1}{2}l_i \times \cos\alpha - y_0} \right) \right]$$

### 5.1.4 小结

本文建立了基于空间解析几何的动态变化模型并运用了插值拟合的方法,解决了折叠桌的动态变化过程的描述问题以及设计加工参数的确定问题;首先,以木板下表面中心为圆心建立空间直角坐标系;其次利用空间几何的相关知识,建立了基于空间解析几何的动态变化模型,以最外侧木条与桌面的夹角为自变量描述内侧木条与桌面的夹角以及桌脚的边缘位置;然后当木条转动到最终位置时,确定桌腿木条开槽的长度和桌脚边缘的坐标,并通过插值拟合方法得到桌脚边缘线的数学描述。最后检验了模型的灵敏度和稳定性。最终得到了折叠桌的动态变化过程和桌脚边缘线的数学描述,木条开槽长度分别为 20.01、19.60、18.77、17.59、16.01、14.14、11.81、9.00、5.53、0cm。

## 5.2 问题二的模型建立与求解

设计折叠桌的平板尺寸、钢筋位置、开槽长度,使得折叠桌的设计满足产品稳固性好、加工方便、用材最少三个方面的要求,是一个典型的多目标规划问题。本文将三个要求作为要实现的三个目标,根据折叠桌的外部约束条件进行对加工参数的求解。接着在原算法上进行优化,建立基于遗传算法的多目标规划模型,对折叠桌的开槽总长度和桌腿宽度进行优化求解。

### 5.2.1 模型的建立

#### 1、多目标规划模型的建立

想要折叠桌的设计达到产品稳固性好、加工方便、用材最少三个方面的要求，需要从折叠桌的最优设计加工参数来考虑，即平板尺寸、钢筋位置、开槽长度。需要满足的目标为：

- (1) 产品稳固性好，即折叠桌与地面的摩擦力达到最大，以此使折叠桌拥有较高的稳定性；
- (2) 加工方便；
- (3) 用材最少，即所用木板的体积最小。

由于本题解决的是一个设计折叠桌的最优加工参数以保证多个目标实现的问题，因此我们选择建立多目标规划模型。

在多目标规划中，为了更好的解决问题，我们引入以下 4 个定义：

- (1) 正、负偏差变量：

设  $f_i (i = 1, 2, \dots, l)$  为第  $i$  个目标函数，它的正偏差变量  $d_i^+ = \max\{f_i - d_i^0, 0\}$ ，表示决策值超过目标值的部分，负偏差变量  $d_i^- = -\min\{f_i - d_i^0, 0\}$  表示决策值未达到目标值的部分，这里  $d_i^0$  表示  $f_i$  的目标值。决策值不可能既超过目标值同时又未达到目标值，即恒有  $d_i^+ \times d_i^- = 0$ 。

- (2) 绝对约束和目标约束：

绝对约束是指必须严格满足的等式约束和不等式约束，如线性规划问题的所有约束条件，不能满足这些约束条件的解称为非可行解，所以他们是硬约束。在达到此目标值时允许发生正或负偏差，因此在这些约束中加入正、负偏差变量，它们是软约束。线性规划问题的目标函数，在给定目标值和加入正、负偏差后可变换为目标约束。也可根据问题的需要将绝对的约束变为目标约束。

- (3) 优先因子

一个规划问题常常有若干个目标。但决策者在要求达到这些目标时，是有主次或轻重缓急之分的。凡是要求第一位达到的目标赋予优先因子  $P_1$ ，，次位的目标赋予优先因子  $P_2, \dots, P_n$ ，并规定  $p_k \gg p_{k+1}$ ，表示  $p_k$  比  $p_{k+1}$  有更大的优先权。

- (4) 目标规划的目标函数

目标规划的目标函数是按各目标约束的正、负偏差变量和赋予相应的优先因子构造的。当每一目标值确定后，决策者的要求是尽可能的缩小偏离目标值。其基本形式有三种：

第  $i$  个目标要求恰好达到目标值，即正、负偏差变量都要尽可能的小，这时：

$$\min w_i^- d_i^- + w_i^+ d_i^+$$

第  $i$  个目标要求不超过目标值，即允许达不到目标值，即正偏差变量都要尽可能的小，这时：

$$\min w_i^+ d_i^+$$

第  $i$  个目标要求超过目标值，即超过量不限，但偏差变量都要尽可能的小，这时：

$$\min w_i^- d_i^-$$

由此，我们建立多目标线性规划模型。

从以上三个目标进行分析，我们发现在分配合理方案时应先满足第一个目标，其次尽可能满足第二个目标，最后考虑第三个目标。因此我们采用求解目标规划算法中的序贯算法。

序贯算法是求解目标规划的一种早期算法，其核心是根据优先级的先后次序，将目标规划问题分解成一系列的单目标规划问题，再依次进行求解。

根据以上 3 个目标，要尽可能的满足第 1 个和第 2 个目标；对于第 3 个目标，允许实际的木板体积较最小木板体积偏大。根据正负偏差变量的定义，我们建立目标函数如下：

$$\min z = P1(d_1^- + d_1^+) + P2(d_2^- + d_2^+) + P3d_3^-$$

其中，其优先级分别为 P1、P2、P3。

记折叠桌与地面之间的摩擦力为  $\alpha$ ，其计算公式为：

$$\alpha = \sum_{i=1}^n \alpha_i$$

记折叠桌的开槽总长度为  $l$ ，其计算公式为：

$$l = \sum_{i=1}^n l_i \times \frac{b}{n}$$

其中， $l_i$  为第  $i$  根木条的长度， $b$  为木板的宽度。

记折叠桌所用木板的体积为  $V$ ，则：

$$V = L \times a \times b$$

其中， $L$  为木板的厚度， $a$  为木板的长度， $b$  为木板的宽度。

根据上述分析，目标函数为：

$$Z = \max(\sum_{i=1}^n \alpha_i) + \min(\sum_{i=1}^n l_i \times \frac{b}{n}) + \min(L \times a \times b)$$

其约束条件为：

(1)为了让桌子可以平稳地放在水平面上，至少有 4 根桌腿着地，且桌腿顶部距离地面的高度相同,对于本题即最外侧着地，其余不着地；

(2)向桌子内部弯曲的桌腿不碰撞，对于本题即最内部的不碰撞；

(3)钢筋在桌腿内，且每跟开槽长度小于桌腿长度，即钢筋最终的  $y$  坐标小于最短腿的  $y$  坐标；

(4)稳定性还看最外侧木条与最内侧木条的夹角，其夹角必须大于零；

(5)板宽大于直径，木板长大于直径，满足要求高度。

$$\text{s.t.} \begin{cases} y_{Ci左} < y_{Ci右} & i = 2, 3 \dots n \\ y_{Ci} > 0 & i = 2, 3 \dots n \\ y_{Bi} < y_{Ci} & i = 1, 2 \dots n \\ \alpha > 0 \\ L > 2 \times y_{C1} \\ R > R_0 \end{cases}$$

通过以上约束条件设计出折叠桌最优的加工参数方案。

## 2、基于遗传算法的多目标优化算法

为了更精确的算出折叠桌的优化参数,需要对上述多目标规划的算法进行优化,而遗传算法借助生物进化理论,体现了优胜劣汰思想,通过交叉及变异操作保证了种群的多样性,具有较强的全局搜索能力和并行处理能力,因此,本文建立基于遗传算法的多目标规划模型。

#### (1)遗传算法概述

遗传算法是一种基于自然选择原理和自然遗传机制的搜索和优化方法,它是模拟自然界中的生命进化机制,在人工系统中实现特定目标的优化。

①染色体: 在使用遗传算法时, 需要把问题的解编成一个适合的码子。这种具有固定结构的符号串即是染色体。符号串的每一位代表一个基因。符号串的总位数称为染色体的长度。一个染色体就代表问题的一个解, 每个染色体也被称为一个个体。

②群体: 每代所产生的染色体总数称为群体, 一个群体包含了该问题在这一代的一些解的集合。

③适应度: 对群体中每个染色体进行编码后, 每个个体对应一个具体问题的解, 而每个解对应于一个函数值。该函数值即适应函数, 就是衡量染色体对环境适应度的指标, 也是反映实际问题的目标函数。

(2)多目标优化问题可以描述如下:

$$\begin{aligned} \min & [f_1(x), f_2(x), \dots, f_m(x)] \\ \text{s.t.} & \begin{cases} lb \leq x \leq ub \\ Aeq * x = beq \\ A * x \leq b \end{cases} \end{aligned}$$

其中,  $f_i(x)$  为待优化的目标函数;  $x$  为待优化的变量;  $lb$  和  $ub$  分别为变量  $x$  的下限和上限约束;  $Aeq * x = beq$  为变量  $x$  的线性等式约束;  $A * x \leq b$  为变量  $x$  的线性不等式约束。

以此对上述方程进行优化求解。

### 5.2.2 模型的求解

#### (1)多目标规划模型的求解

根据多目标线性规划的条件, 解得折叠桌平板尺寸、钢筋位置、开槽长度如表 5.3.2.1 所示:

表 5.3.2.3 多目标线性规划结果

平板尺寸			钢筋位置		开槽总长度	桌腿宽度
平板长度	平板宽度	平板厚度	最外侧桌腿钢筋位置比率	最外侧桌腿钢筋位置距铰链距离		
158.0020	79.9000	3	0.4857	39.6027	827.8432	4.7000

根据上表可以知道, 平板的尺寸即平板的长度、宽度、厚度分别为 158.002cm、79.9cm、3cm。钢筋的位置通过最外侧桌腿钢筋位置比率和最外侧桌腿钢筋位置距铰链距离决定, 分别为 0.4857 和 39.6027cm。

通过计算得到的 10 个开槽的长度分别为 0、11.8615、18.9448、24.5395、28.9823、32.4182、34.9331、36.5821、37.3986cm。即开槽总长度为 827.8432cm。

#### (2)基于遗传算法的多目标优化算法的求解

为了对开槽总长度和桌腿宽度进行优化, 本文设置最优前端个体系数为 0.3; 种群大小为 100; 最大进化代数 200; 停止代数也为 200; 适应度函数值偏差

为 $1e^{-100}$ 。计算得到基于遗传算法的多目标优化算法的结果为开槽总长度为826.8432cm，桌腿宽度为4.6000cm，与之前多目标算法的计算结果相比都有较小，即用材较少。

其动态变化过程和最终形态如图 5.3.2.1 和图 5.3.2.2 所示：

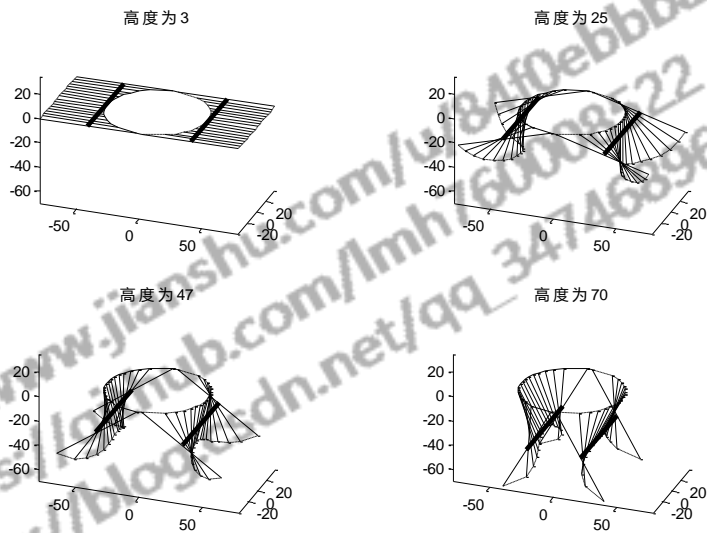


图 5.3.2.1 折叠桌动态变化过程

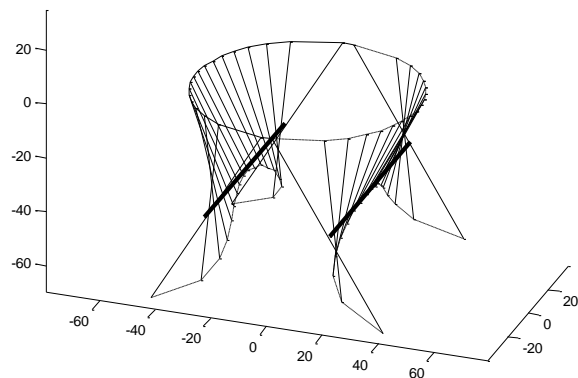


图 5.3.2.4 折叠桌最终形态

### 5.2.3 模型的检验

将多目标规划的计算所得的开槽总长度和桌腿宽度与基于遗传算法的多目标优化算法的计算结果进行比较，如表 5.3.3.1 所示：

表 5.3.3.1 结果对比

	开槽总长度	桌腿宽度
多目标规划	827.8432	4.7000
基于遗传算法的多目标优化	826.8432	4.6000

通过表 5.3.3.1 可以看出，基于遗传算法的多目标优化算法的开槽总长度结果为826.8432cm，多目标规划的开槽总长度结果为827.8432cm，优化后的开槽总长度减小了1cm；基于遗传算法的多目标优化算法的桌腿宽度结果为4.6000cm，多目标规划的桌腿宽度结果为4.7000cm，优化后的桌腿宽度也减小了1cm。由



此,我们发现,利用优化后的算法设计折叠桌的参数可以让折叠桌的用材量减小,即进行优化后的结果更符合折叠桌的设计要求。

#### 5.2.4 小结

本文建立了基于遗传算法的多目标优化模型,解决了折叠桌的最优设计加工参数的确定问题;首先将折叠桌的稳固性、加工系数、用材分别用数学语言描述;其次确定稳固性好为第一目标,加工方便为第二目标,用材最少为第三目标,根据约束条件,建立了多目标优化模型,并用惯序算法进行求解;然后使用遗传算法对已求得的较优解进一步优化。最后检验了模型的灵敏度和稳定性。最终得到当桌高 70cm、桌面直径 80 cm 时,平板长度为 158.0020cm,平板宽度为 79.9000cm,桌腿宽度为 4.7cm,单侧桌腿条数为 17 条,最外侧桌腿钢筋位置比率为 0.4857,最外侧桌腿钢筋位置距铰链距离 39.6027cm,开槽长度由外而内分别为 0、11.8615、18.9448、24.5395、28.9823、32.4182、34.9331、36.5821、37.3986cm,开槽总长度为 827.8432cm。

#### 5.3 问题三的模型建立与求解

该问要求讨论对于任意高度、任意桌面形状、任意桌脚边缘线的大致形状的情况下,求出平板材料的形状尺寸和切实可行的最优设计加工参数,并设计几种自己设计的创意平板折叠桌。此问题要求做出折叠桌加工过程中的通用模型,并对加工参数进行优化。对于此类问题,首先约定坐标系并建立折叠桌加工过程中的通用模型;其次应建立相应的约束条件和目标函数;然后在满足约束条件和目标函数值最优的条件下,确定木板的大小和参数;最后优化参数。对于设计创意平板折叠桌问题,首先应根据约束条件大致确定可用的桌面边缘线的形状,其次制定合理的大小并利用通用模型进行求解。

##### 5.3.1 模型的建立

首先,建立空间直角坐标系如图 5.4.1.1 所示。设木板长度为  $a$ ,木板宽度为  $b$ ,折叠桌的高度为  $H$ ,厚度为  $h$ 。

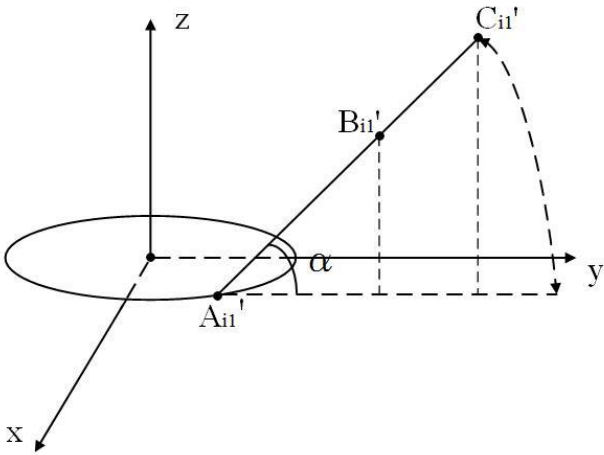


图 5.4.1.5

其次,根据问题一,我们得到 $A'_{i1}, B'_{i1}, C'_{i1}$ 三点的坐标分别为:

$$A_{i1}(\frac{b}{2} - (i-1)\frac{b}{2n}, f(\frac{b}{2}, y_{Ai}), h)$$

$$B_{i1}(\frac{b}{2} - (i-1)\frac{b}{2n}, y_{Bi}, h)$$

$$C_{i1}(\frac{b}{2} - (i-1)\frac{b}{2n}, y_{Ci}, h)$$

求解方法同问题一。由此，计算得到桌腿长度为 $y_{Ci} - y_{Ai}$ ，其目标函数和优化参数同问题二，具体求解步骤如下：

- (1) 设桌面边缘线形状为  $f(x, y)$ ;
- (2) 确定分割角度, 尽量让图形对称的分布在  $x$  轴或  $y$  轴两侧;
- (3) 同问题一建立空间直角坐标系;
- (4) 同问题一，利用勾股定理和三角形的相似性求解 $A_i, B_i, C_i$ 三点坐标;

$$A_i' = (\frac{b}{2} - (i-1)\frac{b}{2n}, \sqrt{(\frac{b}{2})^2 + (\frac{b}{n})^2 - [\frac{R}{2} - (i-1)\frac{R}{n}]^2}, h)$$

易知， $\Delta A_i B_i B' \sim \Delta A_i C_i C'$ ，根据三角形的相似性得到：

$$\frac{B_i B'}{C_i C'} = \frac{A_i B'}{A_i C'}$$

$$\frac{A_i B'}{A_i C'} = \frac{A_i B'}{A_i C'}$$

即：

$$B_i B' = \frac{A_i B'}{A_i C'} \times C_i C'$$

$$A_1 B' = \frac{A_i B'}{A_i C'} \times A_i C'$$

由此求出 $B_i$ 点的  $z$  坐标和  $y$  坐标。则 $B_i$ 点的坐标为：

$$B_i' = (\frac{b}{2} - (i-1)\frac{b}{2n}, B_i B', A_1 B')$$

$$C_i' = (\frac{b}{2} - (i-1)\frac{b}{2n}, \sqrt{(\frac{b}{2})^2 + (\frac{b}{n})^2 - [\frac{R}{2} - (i-1)\frac{R}{n}]^2} + \sqrt{(\frac{a}{2} - \frac{b}{n})^2 - H^2}, H)$$

- (5) 调节桌脚的长度，使得其满足：

- ① 桌脚四点共面，即折叠桌的稳定性好；
- ② 满足桌面高度要求，即 $y_{C1左} = y_{C1右} = H$ ；
- ③ 使得 $C_i$ 点的坐标大致满足方程  $g(x, y, z)$ ，即桌脚边缘线的方程。

- (6) 在确定桌脚的长度后，确定木板的尺寸，即长为 $\max(y_{Ci右}) -$

$\min(y_{Ci左})$ ，宽为 $x_{C1} - x_{Cn}$ ；

- (7) 画出折叠桌的动态变化过程。

最后, 仍然采用基于遗传算法的多目标优化算法对上述多目标规划模型进行加工参数的优化, 得出相应的设计加工参数。

### 5.3.2 模型的求解

设计 1:

考虑到美观与实用性, 我们设计一个会议桌, 其桌面形状由一个矩形与两个半圆形构成, 半圆形的半径等于矩形的宽。确定桌面的高度为 70cm, 矩形长度为 60cm, 矩形宽度为 40cm, 边缘线近似抛物线。其动态变化过程和最终形态分别如图 5.4.2.1 和图 5.4.2.2 所示:

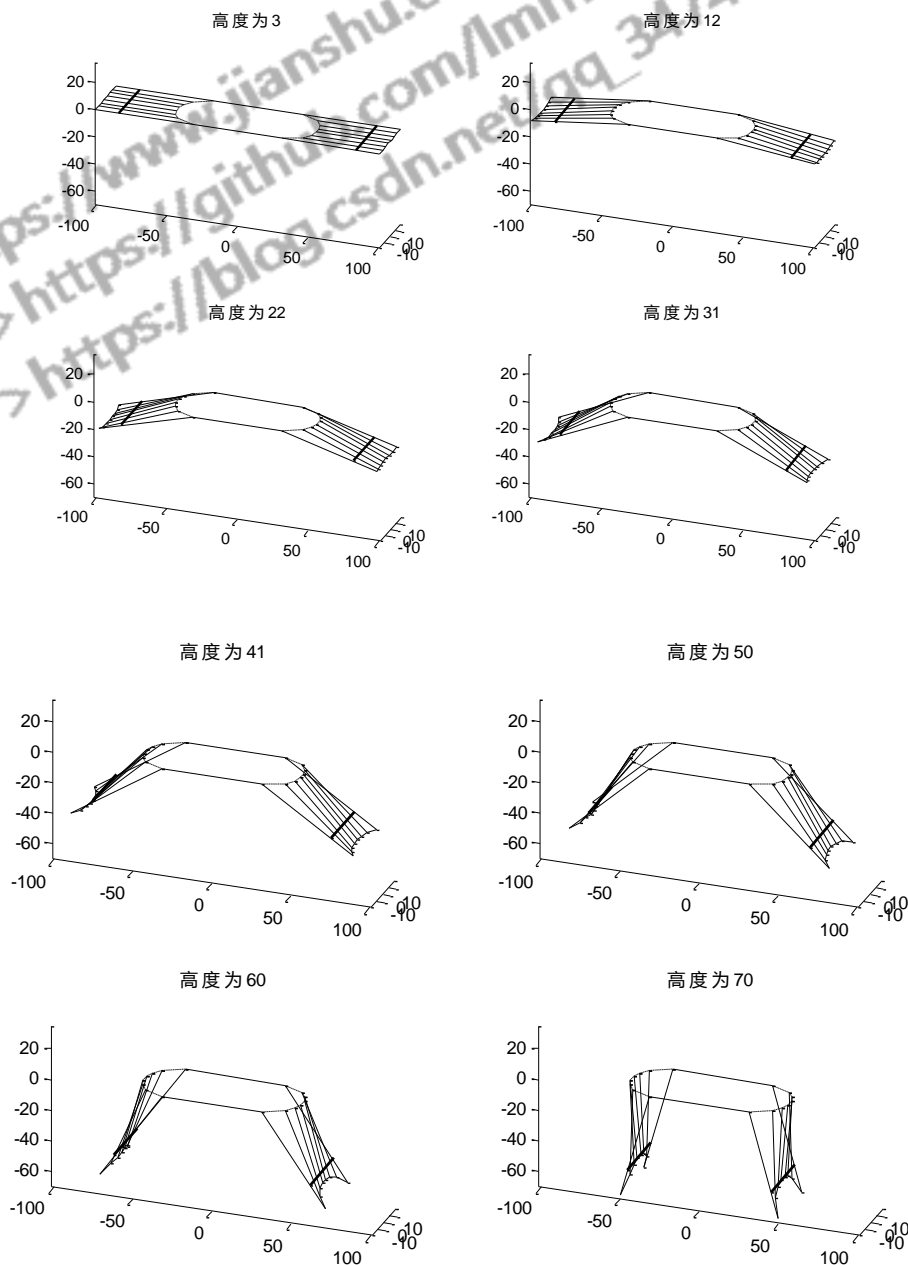


图 5.4.2.6 会议桌动态变化过程

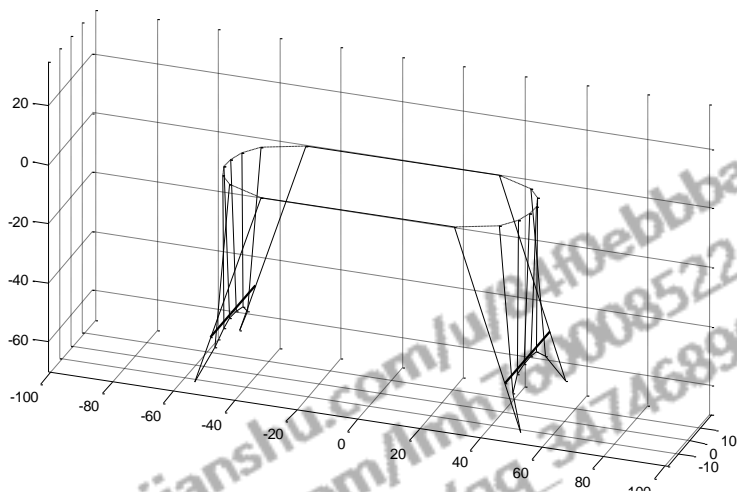


图 5.4.2.7 会议桌的最终形态

计算得到其设计加工参数如表 5.4.2.1 所示：

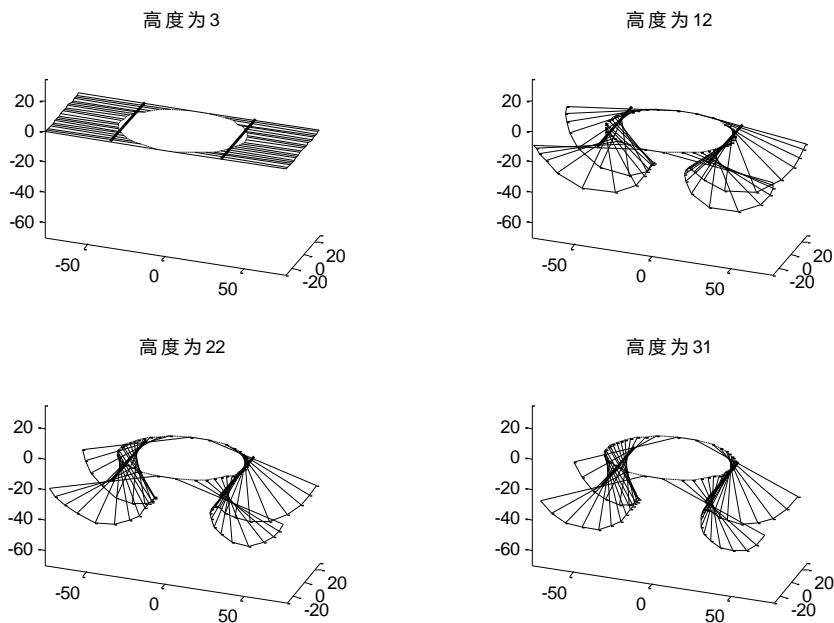
表 5.4.2.1 会议桌设计加工参数

参数	平板尺寸/cm			桌腿宽度	单侧桌腿条数	最外侧桌腿钢筋位置比率	最外侧桌腿钢筋距铰链距离	开槽总长度
	平板长度	平板宽度	平板厚度					
参数	149.707	59.8	3	4.6000	13	0.3443	47.4788	418.0505

通过上表我们得到，平板的长度、宽度、厚度分别为 149.707cm、59.8cm、3cm；桌腿宽度为 4.6cm；单侧桌腿条数为 13 条；最外侧桌腿钢筋位置比率为 0.3443；最外侧桌腿钢筋距铰链距离为 47.4788cm；开槽总长度为 418.0505cm。

设计二：

桌面形状为椭圆形，桌面的高度为 70cm，椭圆短轴 60cm，椭圆长轴 80cm，边缘线近似抛物线。其动态变化过程和最终形态分别如图 5.4.2.3 和图 5.4.2.4 所示：



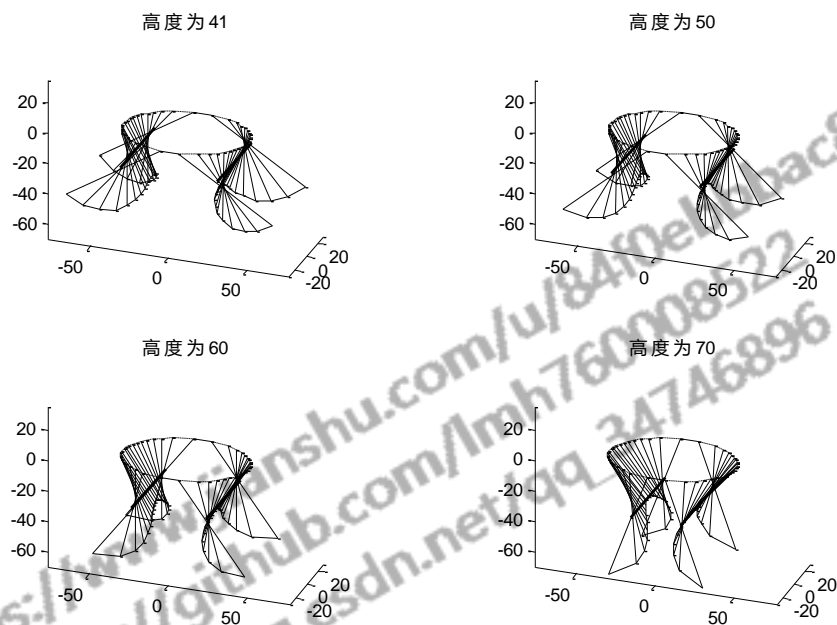


图 5.4.2.3 椭圆桌动态变化过程

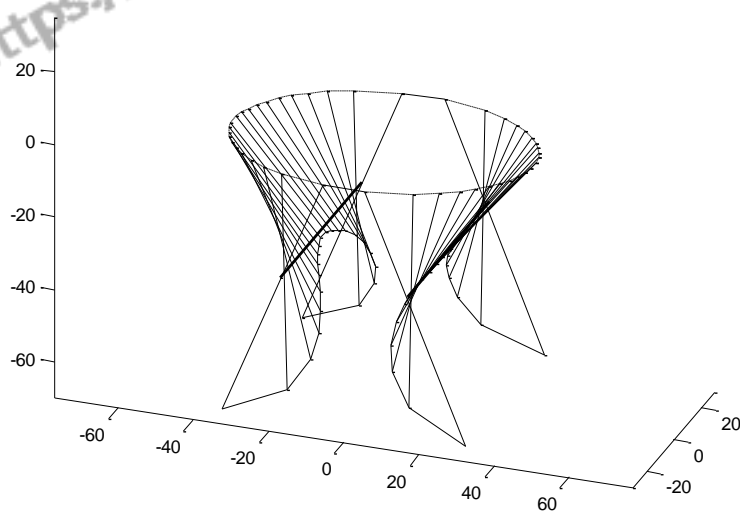


图 5.4.2.4 椭圆桌最终形态

计算得到其设计加工参数如表 5.4.2.2 所示：

表 5.4.2.2 会议桌设计加工参数

	平板尺寸/cm			桌腿 宽度	单侧 桌腿 条数	最外侧桌 腿钢筋位 置比率	最外侧桌 腿钢筋距 铰链距离	开槽总 长度
	平板 长度	平板 宽度	平板 厚度					
参 数	153.662	59.4	3	2.7	22	0.5814	29.7965	1.21e+03

通过上表我们得到，平板的长度、宽度、厚度分别为 153.662cm、59.4cm、3cm；桌腿宽度为 2.7cm；单侧桌腿条数为 22 条；最外侧桌腿钢筋位置比率为 0.5814；最外侧桌腿钢筋距铰链距离为 29.7965cm；开槽总长度为 1.21e+03cm。

### 5.3.3 小结

本文建立了基于多目标优化的折叠桌设计通用模型,解决了设计软件的模型设计问题,并设计了两个创意平板折叠桌。首先建立了折叠桌设计通用模型,并提出模型的适用条件;其次重新建立基于遗传算法的多目标优化模型,得到参数优化的通用模型;最后设计了会议桌和椭圆桌并得到了优化参数。最终得到高度为70cm、矩形长度为60cm、矩形宽度为40cm的会议桌的平板长度为149.7071cm,平板宽度为59.8cm,桌腿宽度为4.6cm,单侧桌腿条数为13条,最外侧桌腿钢筋位置为0.3443,最外侧桌腿钢筋位置距铰链距为47.4788cm,开槽长度由外而内分别为0、10.3419、15.7401、19.5900、22.3116、24.0656、24.9268cm,开槽总长度为418.0505cm;高度为70cm、椭圆短轴为60cm、椭圆长轴为80cm的椭圆桌的平板长度为153.6617cm、平板宽度为59.4cm、桌腿宽度为2.7cm、单侧桌腿条数为22条、最外侧桌腿钢筋位置比率为0.5814、最外侧桌腿钢筋位置距铰链距离29.7965cm、开槽长度由内而外分别为0、10.0928、17.1114、22.8413、27.5951、31.5261、34.7319、37.2803、39.2191、40.5821、41.3914cm,开槽总长度:1209.5cm。

## 六、模型的评价与推广

### 6.1 模型的优点与缺点

问题一中基于空间解析几何的动态变化模型的插值拟合方法将离散的点连成所需曲线,较好的描述了实物的动态变化过程;问题二中的基于遗传算法的多目标优化模型在原始的多目标模型上进行改进,使得折叠桌的开槽总长度和桌腿宽度较小,更好的满足了用材少的要求。

本文在建立模型的过程中对摩擦力的定义不够准确,不能完全符合现实情况。

### 6.2 模型的推广

本文设计的求解折叠桌加工参数的方法,不仅适用于圆形桌面的设计,还适用于椭圆形桌面与会议桌等其他类型折叠桌的设计。问题二中采用的基于遗传算法的多目标优化算法,可以应用于处理生活中的其他方面,如生产销售中既要满足生产成本最小又要考虑到消费者满意度最大等。

## 参考文献

- [1]王力纬.原子的电子结构以及原子核外电子运动状态研究[J]. 量子化学理论与计算方法.2014(1):243-244
- [2]曾双双.几何形态在室内设计中的应用研究[D].中南林业科技大学,2014.
- [3]Li-Yong Shen.Computing  $\mu$  bases from algebraic ruled surfaces[J]. Computer Aided Geometric Design,2016
- [4]张余华.面向客户需求的大规模定制策略[M].清华大学出版社.2010
- [5]戴若夷.面向大规模定制的广义需求建模方法与实现技术的研究及应用[D].浙江大学,2004.
- [6]王健.模数化组合家具设计研究[D].西南交通大学,2017.
- [7]张昱宏,辛一杰,叶喜.多功能家具万变的本源——折叠[J].家具与室内装



饰,2016(07):18-21.

[8] 林舒瑶,徐燕君.家居设计中“微空间”的折叠设计[J].现代装饰(理论),2014(04):42-43.

[9] 朱伟.基于收启式结构的木质折叠椅系列产品设计[D].中南林业科技大学,2016.

[10] 周雪冰.折叠式多功能家具设计研究[D].中南林业科技大学,2012.

大美mixer  
简书>>

Github>>

CSDN>>

<https://www.jianshu.com/u/84f0ebb2ac07>  
<https://github.com/lmh760008522>  
[https://blog.csdn.net/qq\\_34746896](https://blog.csdn.net/qq_34746896)

## 附录

附录一：

```
clc,clear
```

```
%tc = 1; %xyz1 = 1; %xyz2 = 1; %xyz3 = 1; %alp = 0; %jiaolian = 0; %w = 2.5; %l = 120; %r = sqrt(2.5^2+25^2); %h = 1; %beta = 0;
```

```
l=120;
```

```
r=sqrt(2.5^2+25^2);
```

```
w=2.5;
```

```
for h=0:7:50;
```

```
n=10;
```

```
tc=zeros(n,1);
```

```
xyz1=zeros(n,3);
```

```
xyz1(:,2)=1/2*l;
```

```
xyz1(:,3)=0;
```

```
xyz2=zeros(n,3);
```

```
for i=1:n
```

```
tc(i)=1/2*l-sqrt(r^2-(w*(i))^2);
```

```
xyz1(i,1)=-w/2+w*i;
```

```
xyz2(i,1)=-w/2+w*i;
```

```
end
```

```
xyz2(:,2)=1/2*l-tc;
```

```
xyz2(:,3)=0;
```

```
alp=asin(h/tc(n));
```

```
jiaolian1=zeros(n,3);
```

```
jiaolian1(:,1)=xyz2(:,1);
```

```
jiaolian1(:,2)=1/2*l-1/2*tc(n);
```

```
jiaolian1(:,3)=0;
```

```
jiaolian2=zeros(n,3);
```

```
jiaolian2(:,1)=xyz2(:,1);
```

```
jiaolian2(:,2)=xyz2(n,2)+1/2*tc(n)*cos(alp);
```

```
jiaolian2(:,3)=1/2*tc(n)*sin(alp);
```

```
beta=zeros(10,1);
```

```
beta(:,1)=atan((jiaolian2(:,3)-xyz2(:,3))./(jiaolian2(:,2)-xyz2(:,2)));
```

```
for i=1:n
```

```
if beta(i)<0
```

```
beta(i)=beta(i)+pi;
```

```
end
```

```

end
xyz3=zeros(n,3);
xyz3(:,1)=xyz2(:,1);
xyz3(:,2)=xyz2(:,2)+tc.*cos(beta);
xyz3(:,3)=tc.*sin(beta);
huacao=zeros(n,1);
huacao=sqrt((jiaolian1(:,1)-xyz1(:,1)).^2+(jiaolian1(:,2)-xyz1(:,2)).^2+(jiaolian1(:,3)-xyz1(:,3)).^2)-
sqrt((jiaolian2(:,1)-xyz3(:,1)).^2+(jiaolian2(:,2)-xyz3(:,2)).^2+(jiaolian2(:,3)-xyz3(:,3)).^2);
zb1=zeros(2*n,3);
zb2=zeros(2*n,3);
zb1(n+1:2*n,:)=xyz3;
for i=1:10
    zb1(n-i+1,:)=xyz3(i,:); zb1(n-i+1,1)=-xyz3(i,1);
end
zb2(:,1:2)=-zb1(:,1:2); zb2(:,3)=zb1(:,3);
axis([-70 70 -70 70 0 60]);
%grid on
%axis
A1=[-23.75,2.5,0;zb1(1,1),zb1(1,2),zb1(1,3)];
A2=[23.75,2.5,0;zb1(2*n,1),zb1(2*n,2),zb1(2*n,3)];
plot3(A1(:,1),A1(:,2),A1(:,3));
hold on
plot3(A2(:,1),A2(:,2),A2(:,3));
hold on
plot3(zb1(:,1),zb1(:,2),zb1(:,3));
hold on
B1=[23.75,-2.5,0;zb2(1,1),zb2(1,2),zb2(1,3)];
B2=[-23.75,-2.5,0;zb2(2*n,1),zb2(2*n,2),zb2(2*n,3)];
plot3(B1(:,1),B1(:,2),B1(:,3));
plot3(B2(:,1),B2(:,2),B2(:,3));
plot3(zb2(:,1),zb2(:,2),zb2(:,3));
hold on
end
huayuan(0,0,r);

%huanyuan.m
function []=huayuan(x,y,r)
theta=0:0.1:2*pi;
Circle1=x+r*cos(theta);
Circle2=y+r*sin(theta);
%c=[123,14,52];
z=zeros(63,1);
plot3(Circle1,Circle2,z,'linewidth',1);
end

```

附录二：

clc

clear

tic

h = input('×Ã ß:');  
 R = input('×Ã ÆÖ±¼ ¶£¶);  
 c = 3; %Æ½ °ã¶  
 i = 1; %¼ÇÃ¼×ø±ê  
 for d=1:0.1:10  
 n1 = R/d;  
 if(n1 == floor(R/d))  
 n = n1-1;  
 else  
 n = floor(R/d);  
 end  
 syms a  
 u1=solve(sqrt((a/2 - sqrt((R/2)^2 - (n/2\*d)^2))^2 - (h - c)^2) + sqrt((R/2)^2 - (n/2\*d)^2) -  
 n/2\*d,a);  
 u1 = vpa(u1);  
 u(i) = eval(u1(1));  
 i = i + 1;  
 end  
 u2 = find(u(1,:) == min(u));  
  
 while(1)  
 a = u(max(u2));  
 d = 1 + (max(u2) - 1)\*0.1;  
 n1 = R/d;  
 if(n1 == floor(R/d))  
 n = n1-1;  
 else  
 n = floor(R/d);  
 end  
  
 b1 = n\*d; %Æ½ °ã¶  
 k = sqrt((R/2)^2 - (n/2\*d)^2); %×íâ²×Ã ÆÖ±¼  
 o = asin((h - c)/(a/2 - k)); %×íâ²×Ã ÆÖ±¼Ã¼Ð½Ç  
 j = 1;  
 for v = 0:0.00001:1  
 s8 = 0;  
 for i = 1:ceil(n/2)

```

b5(i) = sqrt((R/2)^2 - ((n/2 - i + 1)*d)^2);    % ×ÀÃæ Æ
f5(i) = a/2 - b5(i);                            % Æ ð×À Æ
s5(i) = f5(i) - v*f5(1);                        % Æ×µ½×À Æ×½
if i == 1
    s6(1) = (1 - v)*f5(1);                      % Æ×µ½×À Æ×½
else
    s6(i) = sqrt(((1 - v)*f5(1))^2 + (b5(i) - b5(1))^2 - 2*(b5(i) - b5(1))*(1 -
v)*f5(1)*cos(o));
end
s7(i) = s6(i) - s5(i);    % Æ
if s6(i) - s5(i) > f5(1)*v
    s8 = inf;
else
    s8 = s8 + s7(i);    % Æ×Û
end
end
if(n/2 == floor(n/2))
    s8 = 4*s8;
else
    s8 = 4*s8 - 2*s7(i);
end
s9(j) = s8;
j = j+1;
end
v = (find(min(s9) == s9(1,:))-1)*0.00001;    % ±ÈÊ
v = v(1);
if inf ~= s8
    break;
else
    u2 = u2 + 1;
end
end
%% ÒÅ ´«Ëä·ÓÅ»-
% {
% kc=min(s9);% Æ×Û
% sd=d;% É
% cd=a;%
% kd=b1;%
fitnessfcn = @my_multi;    % Function handle to the fitness function
nvars = 4;                % Number of decision variables
lb = [min(s9)-1,d-0.1,a,b 1];    % Lower bound
ub = [min(s9)+1,d+0.1,a,b 1];    % Upper bound
A = []; b = [];           % No linear inequality constraints
Aeq = []; beq = [];       % No linear equality constraints

```

```

options =
gaoptimset('ParetoFraction',0.3,'PopulationSize',100,'Generations',200,'StallGenLimit',200,'ToIFun',1e-
100,'PlotFcns',@gaplotpareto);
[x,fval] = gamultiobj(fitnessfcn,nvars, A,b,Aeq,beq,lb,ub,options)
% }
%%  $\times \div \frac{1}{4}$ 
h1 = h;
N = 4; %  $\times \frac{1}{4}$   $\div \frac{1}{4}$ 
I = 1; %  $\mu \frac{1}{4}$   $\div \frac{1}{4}$ 

for h = 3 :1:h1
    h2 = 3:(h1-3)/(N-1):h1; %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
    figure(1);
    if floor(h2(I)) == h
        figure(2);
        subplot(2,2,I);
        view(25,25);
        if I < N %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
            I = I + 1;
        end
        title([' $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ ',num2str(h)]);
    end

    o = asin((h - c)/(a/2 - k)); %  $\times \frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
    for i = 1:ceil(n/2)

        b(i) = sqrt((R/2)^2 - ((n/2 - i + 1)*d)^2); %  $\times \frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
        f(i) = a/2 - b(i); %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
        s(i) = f(i) - v*f(1); %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
        if i == 1
            s1(1) = (1 - v)*f(1); %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
        else
            s1(i) = sqrt(((1 - v)*f(1))^2 + (b(i) - b(1))^2 - 2*(b(i) - b(1))*(1 - v)*f(1)*cos(o));
        end
        s2(i) = s1(i) - s(i); %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 

        if i == 1
            q(i) = pi - o;
        else
            q(i) = acos(((b(i) - k)^2 + s1(i)^2 - s1(1)^2)/(2*(b(i) - k)*s1(i))); %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
        end

        if h == 3 %  $\frac{1}{4}$   $\frac{1}{4}$   $\frac{1}{4}$ 
            if(q(i) < pi/2)

```



```

        q(i) = pi - q(i);
    end
end

if(q(i) > pi/2)
    g(i) = cos(q(i) - pi/2)*f(i);           % Ã ĭõ×À ÊµÄ ,ß¶Ë
    w(i) = b(i) + sqrt(f(i)^2 - g(i)^2);    % ¢×ø±ê
else
    g(i) = sin(q(i))*f(i);                 % Ã ĭõ×À ÊµÄ ,ß¶Ë
    w(i) = b(i) - sqrt(f(i)^2 - g(i)^2);
end
end

x = [w,flipr(w)];
y = [-b1/2:d:-d/2,d/2:d:b1/2];
z = -fix(lg,flipr(g));

x1 = [b,flipr(b)];
y1 = y;
z1 = zeros(1,2*ceil(n/2));

n1 = 2*ceil(n/2);

cla

axis([-a/2 a/2 -b1/2 b1/2 -h1 h1/2]);
axis equal
set(gcf, 'position', [200 200 800 500]);
hold on
% » -ö×ÀÃæ
for i = 1:n1
    plot3([x(i) x1(i)],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

plot3(x,y,z,'k-');
plot3(x1,y1,z1,'k-.');

for i = 1:n1
    plot3([-x(i) -x1(i)],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

plot3(-x,y,z,'k-');
plot3(-x1,y1,z1,'k-.');

```

```

% ×ÀÃæÁ¬Ïß
plot3([x1(1) -x1(1)],[y1(1) y1(1)],[z1(1) z1(1)],'k-');
plot3([x1(n1) -x1(n1)],[y1(n1) y1(n1)],[z1(n1) z1(n1)],'k-');

% Ö½î
x3 = (1 - v)*(w(1) - k) + k;
y3 = b1/2;
z3 = -fix((1 - v)*g(1));
plot3([x3 x3],[y3 -y3],[z3 z3],'k','LineWidth',2);
plot3([-x3 -x3],[y3 -y3],[z3 z3],'k','LineWidth',2);

view(20,25);
pause(0.05);
end

disp('Æ½°ã¶Ë'); a
disp('Æ½°ã¶Ë'); b1
disp('Æ½°ã¶Ë'); c
disp('×ÀÊ¶Ë'); d
disp('μ×ÀÊË'); n
disp('×îâ×ÀÊÖ½îÖÃ±ÊÊ'); v
disp('×îâ×ÀÊÖ½îÖÃ¾½ÁÁ¾àÀ'); f(1)*(1 - v)
disp('ÃÊ×ÀÊ'); f
disp('ÃÊ×ÀÊ°¶Ë¾½ÁÁ¾àÀ'); s
disp('ÃÊ×ÀÊ½ÁÁ¾àÆ½°ÖÏË¾àÀ'); b
disp('°¶Ë'); s2
disp('°¶×Û¶Ë'); min(s9)

```

附录三：

```

clc
clear

tic
h = input('×À,ß:');
R = input('×ÀÃæ¶Öáf');
A = input('×ÀÃæ°Öáf');
c = 3; %Æ½°ã¶Ë
i = 1; %¼ÇÂ¼×ø±ê
for d=1:0.1:10
    n1 = R/d;
    if(n1 == floor(R/d))
        n = n1-1;
    else
        n = floor(R/d);
    end
end

```

```

end
syms a
u1=solve(sqrt((a/2 - sqrt((A/2)^2 - (A/R*(n/2)*d)^2))^2 - (h - c)^2) + sqrt((A/2)^2 -
(A/R*(n/2)*d)^2) - n/2*d,a);
u1 = vpa(u1);
u(i) = eval(u1(1));
i = i + 1;
end
u2 = find(u(1,:) == min(u));

while(1)
a = u(max(u2));
d = 1 + (max(u2) - 1)*0.1;
n1 = R/d;
if(n1 == floor(R/d))
n = n1-1;
else
n = floor(R/d);
end
b1 = n*d; % 计算b1
k = sqrt((A/2)^2 - (A/R*(n/2)*d)^2); % 计算k
o = asin((h - c)/(a/2 - k)); % 计算o
j = 1;
for v = 0:0.00001:1
s8 = 0;
for i = 1:ceil(n/2)

b5(i) = sqrt((A/2)^2 - (A/R*(n/2 - i + 1)*d)^2); % 计算b5(i)
f5(i) = a/2 - b5(i); % 计算f5(i)
s5(i) = f5(i) - v*f5(1); % 计算s5(i)
if i == 1
s6(1) = (1 - v)*f5(1); % 计算s6(1)
else
s6(i) = sqrt(((1 - v)*f5(1))^2 + (b5(i) - b5(1))^2 - 2*(b5(i) - b5(1))*(1 -
v)*f5(1)*cos(o));
end
s7(i) = s6(i) - s5(i); % 计算s7(i)
if s6(i) - s5(i) > f5(1)*v
s8 = inf;
else
s8 = s8 + s7(i); % 计算s8
end
end
end

```

```

if(n/2 == floor(n/2))
    s8 = 4*s8;
else
    s8 = 4*s8 - 2*s7(i);
end
s9(j) = s8;
j = j+1;
end
v = (find(min(s9) == s9(1,:))-1)*0.00001; % ±ÊÂÊ
v = v(1);
if inf ~= s8
    break;
else
    u2 = u2 + 1;
end
end
h1 = h;
N = 8; % ×Ó Í¼ òÊý
I = 1; % uÍî ò×Ó Í¼
for h = 3 :1:h1
    h2 = 3:(h1-3)/(N-1):h1; % È;ö Í¼ ĩñ
    figure(1);
    if floor(h2(I)) == h
        if I <= N/2 % Ö;ª» -Í¼
            figure(2);
            subplot(2,2,I);
        else
            figure(3);
            subplot(2,2,I-N/2);
        end
        view(25,25);
        if I < N % ÀÖ Ô½½½ ç
            I = I + 1;
        end
        title(['ß¶ÊÎ',num2str(h)]);
    end
end

o = asin((h - c)/(a/2 - k)); % ×íâ²a×À ÊÓë×ÀÃæ¼⁄⁄⁄⁄Ç
for i = 1:ceil(n/2)

```

```

    b(i) = sqrt((A/2)^2 - (A/R*(n/2 - i + 1)*d)^2); % ×ÀÃæ ð æ
    f(i) = a/2 - b(i); % Æ;ö×À Ê æ¶Ê

```

```

s(i) = f(i) - v*f(1); % 计算 f(i) 与 f(1) 的差值
if i == 1
    s1(1) = (1 - v)*f(1); % 计算 s1(1)
else
    s1(i) = sqrt(((1 - v)*f(1))^2 + (b(i) - b(1))^2 - 2*(b(i) - b(1))*(1 - v)*f(1)*cos(o));
end
s2(i) = s1(i) - s(i); % 计算 s2(i)

if i == 1
    q(i) = pi - o;
else
    q(i) = acos(((b(i) - k)^2 + s1(i)^2 - s1(1)^2)/(2*(b(i) - k)*s1(i))); % 计算 q(i)
end

if h == 3 % 计算 h 为 3 时的 q(i)
    if(q(i) < pi/2)
        q(i) = pi - q(i);
    end
end

if(q(i) > pi/2)
    g(i) = cos(q(i) - pi/2)*f(i); % 计算 g(i)
    w(i) = b(i) + sqrt(f(i)^2 - g(i)^2); % 计算 w(i)
else
    g(i) = sin(q(i))*f(i); % 计算 g(i)
    w(i) = b(i) - sqrt(f(i)^2 - g(i)^2);
end
end

x = [w, flipr(w)];
y = [-b1/2:d:-d/2, d/2:d:b1/2];
z = -fix([g, flipr(g)]);

x1 = [b, flipr(b)];
y1 = y;
z1 = zeros(1, 2*ceil(n/2));

n1 = 2*ceil(n/2);

cla

axis([-a/2 a/2 -b1/2 b1/2 -h1 h1/2]);
axis equal
set(gcf, 'position', [200 200 800 500]);

```

```

hold on
% » -ÿ×ÀÃæ
for i = 1:n1
    plot3([x(i) x1(i)],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

plot3(x,y,z,'k-');
plot3(x1,y1,z1,'k-.');

for i = 1:n1
    plot3([-x(i) -x1(i)],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

plot3(-x,y,z,'k-');
plot3(-x1,y1,z1,'k-.');

% ×ÀÃæÃ-ÿß
plot3([x1(1) -x1(1)],[y1(1) y1(1)],[z1(1) z1(1)],'k-');
plot3([x1(n1) -x1(n1)],[y1(n1) y1(n1)],[z1(n1) z1(n1)],'k-');

% Ö½ î
x3 = (1 - v)*(w(1) - k) + k;
y3 = b1/2;
z3 = -fix((1 - v)*g(1));
plot3([x3 x3],[y3 -y3],[z3 z3],'k','LineWidth',2);
plot3([-x3 -x3],[y3 -y3],[z3 z3],'k','LineWidth',2);

view(20,25);
pause(0.05);
end

disp('Æ½ °ã¶È£'); a
disp('Æ½ °ã¶È£'); b1
disp('Æ½ °ã¶È£'); c
disp('×À Ê½¶È£'); d
disp('µ¥²×À Ê½ËÝ£'); n
disp('×í²×À Ê½ Ö½ îÖÃ±ÊÊ'); v
disp('×í²×À Ê½ Ö½ îÖÃ¾²ÁÁ ¾àÀ'); f(1)*(1 - v)
disp('Ã½×À Ê³'); f
disp('Ã½×À Ê³Ö¾²ÁÁ ¾àÀ'); s
disp('Ã½×À Ê½ÁÁ ¾àÆ½ °ÖÐË¾àÀ'); b
disp('½³¶È£'); s2
disp('½³×Ü³¶È£'); min(s9)
toc

```



附录四：

clc

clear

tic

h = input('×À ß:');

R = input('×À ÃæÖ÷¼ ¶£¶);

L = input('¼ØÐ Îæ ¶£¶);

c = 3; %Æ½ °â ¶£¶

i = 1; %¼ÇÂ¼¼ ×ø±ê

for d=1:0.1:10

n1 = R/d;

if(n1 == floor(R/d))

n = n1-1;

else

n = floor(R/d);

end

syms a

u1=solve(sqrt((a/2 - sqrt((R/2)^2 - (n/2\*d)^2))^2 - (h - c)^2) + sqrt((R/2)^2 - (n/2\*d)^2) -  
n/2\*d,a);

u1 = vpa(u1);

u(i) = eval(u1(1));

i = i + 1;

end

u2 = find(u(1,:) == min(u));

while(1)

a = u(max(u2));

d = 1 + (max(u2) - 1)\*0.1;

n1 = R/d;

if(n1 == floor(R/d))

n = n1-1;

else

n = floor(R/d);

end

b1 = n\*d; %Æ½ °â ¶£¶

k = sqrt((R/2)^2 - (n/2\*d)^2); % ×îâ²×À Ãæ Ææ

o = asin((h - c)/(a/2 - k)); % ×îâ²×À ÆÖë×À Ãæ¼Ð½Ç

j = 1;

for v = 0:0.00001:1

s8 = 0;

for i = 1:ceil(n/2)

```

b5(i) = sqrt((R/2)^2 - ((n/2 - i + 1)*d)^2);    % ×ÀÃæ Æ
f5(i) = a/2 - b5(i);                            % Æ Æ×À Æ
s5(i) = f5(i) - v*f5(1);                        % Æ Æ×À Æ
if i == 1
    s6(1) = (1 - v)*f5(1);                      % Æ Æ×À Æ
else
    s6(i) = sqrt(((1 - v)*f5(1))^2 + (b5(i) - b5(1))^2 - 2*(b5(i) - b5(1))*(1 -
v)*f5(1)*cos(o));
end
s7(i) = s6(i) - s5(i);    % Æ Æ
if s6(i) - s5(i) > f5(1)*v
    s8 = inf;
else
    s8 = s8 + s7(i);    % Æ Æ
end
end
if(n/2 == floor(n/2))
    s8 = 4*s8;
else
    s8 = 4*s8 - 2*s7(i);
end
s9(j) = s8;
j = j+1;
end
v = (find(min(s9) == s9(1,:))-1)*0.00001;    % ±ÊÊ
v = v(1);
if inf ~= s8
    break;
else
    u2 = u2 + 1;
end
end
%% Æ Æ·Ö»-
%kc=min(s9);% Æ Æ
%sd=d;% Æ
%cd=a;% Æ
%kd=b1;% Æ
fitnessfcn = @my_multi;    % Function handle to the fitness function
nvars = 4;                % Number of decision variables
lb = [min(s9)-1,d-0.1,a,b 1];    % Lower bound
ub = [min(s9)+1,d+0.1,a,b 1];    % Upper bound
A = []; b = [];            % No linear inequality constraints
Aeq = []; beq = [];        % No linear equality constraints

```

```

options =
gaoptimset('ParetoFraction',0.3,'PopulationSize',100,'Generations',200,'StallGenLimit',200,'ToIFun',1e-
100,'PlotFcns',@gaplotpareto);
[x,fval] = gamultiobj(fitnessfcn,nvars, A,b,Aeq,beq,lb,ub,options)
%% x = 1/4
h1 = h;
N = 8; % 1/4, 1/2
I = 1; % 1/4, 1/2

for h = 3:1:h1
    h2 = 3:(h1-3)/(N-1):h1; % 1/4, 1/2
    figure(1);
    if floor(h2(I)) == h
        if I <= N/2 % 1/4, 1/2
            figure(2);
            subplot(2,2,I);
        else
            figure(3);
            subplot(2,2,I-N/2);
        end
        view(25,25);
        if I < N % 1/4, 1/2
            I = I + 1;
        end
        title(['1/4', num2str(h)]);
    end

o = asin((h - c)/(a/2 - k)); % 1/4, 1/2
for i = 1:ceil(n/2)

    b(i) = sqrt((R/2)^2 - ((n/2 - i + 1)*d)^2); % 1/4, 1/2
    f(i) = a/2 - b(i); % 1/4, 1/2
    s(i) = f(i) - v*f(1); % 1/4, 1/2
    if i == 1
        s1(1) = (1 - v)*f(1); % 1/4, 1/2
    else
        s1(i) = sqrt(((1 - v)*f(1))^2 + (b(i) - b(1))^2 - 2*(b(i) - b(1))*(1 - v)*f(1)*cos(o));
    end
    s2(i) = s1(i) - s(i); % 1/4, 1/2
    if i == 1
        q(i) = pi - o;
    else
        q(i) = acos(((b(i) - k)^2 + s1(i)^2 - s1(1)^2)/(2*(b(i) - k)*s1(i))); % 1/4, 1/2
    end
end

```

```

if h == 3 % 'À 0Ê±½Ç¶È
    if(q(i) < pi/2)
        q(i) = pi - q(i);
    end
end

if(q(i) > pi/2)
    g(i) = cos(q(i) - pi/2)*f(i); % 'Ã ÿ×À ÊµÃ ß¶È
    w(i) = b(i) + sqrt(f(i)^2 - g(i)^2); % 'á×ø±ê
else
    g(i) = sin(q(i))*f(i); % 'Ã ÿ×À ÊµÃ ß¶È
    w(i) = b(i) - sqrt(f(i)^2 - g(i)^2);
end
end

x = [w,flipr(w)];
y = [-b1/2:d:-d/2,d/2:d;b1/2];
z = -fix([g,flipr(g)]);

x1 = [b,flipr(b)];
y1 = y;
z1 = zeros(1,2*ceil(n/2));

n1 = 2*ceil(n/2);

cla

axis([-a/2-L/2 a/2+L/2 -b1/2 b1/2 -h1 h1/2]);
axis equal
set(gcf, 'position', [200 200 800 500]);
hold on

% »-ÿ×ÀÃæ
for i = 1:n1
    plot3([x(i)+L/2 x1(i)+L/2],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

plot3(x+L/2,y,z,'k-');
plot3(x1+L/2,y1,z1,'k-');

for i = 1:n1
    plot3([-x(i)-L/2 -x1(i)-L/2],[y(i) y1(i)],[z(i) z1(i)],'k-')
end

```

```

plot3(-x-L/2,y,z,'k-');
plot3(-x1-L/2,y1,z1,'k-');

% ×ÀÃæÁ¬Ë
plot3([x1(1)+L/2 -x1(1)-L/2],[y1(1) y1(1)],[z1(1) z1(1)],'k-');
plot3([x1(n1)+L/2 -x1(n1)-L/2],[y1(n1) y1(n1)],[z1(n1) z1(n1)],'k-');

% Ö½î
x3 = (1 - v)*(w(1) - k) + k;
y3 = b1/2;
z3 = -fix((1 - v)*g(1));
plot3([x3+L/2 -x3-L/2],[y3 -y3],[z3 z3],'k','LineWidth',2);
plot3([-x3-L/2 -x3-L/2],[y3 -y3],[z3 z3],'k','LineWidth',2);

view(20,25);
pause(0.05);
end
disp('Æ½°ã¶È£'); a
disp('Æ½°ä¿¶È£'); b1
disp('Æ½°ã¶È'); c
disp('×ÀÊ¿¶È£'); d
disp('μ¥²×ÀÊ¿Êý£'); n
disp('×îâ²×ÀÊÖ½î»ÖÃ±ÈÂÊ'); v
disp('×îâ²×ÀÊÖ½î»ÖÃ¼¿½ÂÁ¼àÀ£'); f(1)*(1 - v)
disp('Ã¿ì×ÀÊ¿'); f
disp('Ã¿ì×ÀÊ¿°Û¶¶Ë¼¿½ÂÁ¼àÀ£'); s
disp('Ã¿ì×ÀÊ½ÂÁ¼àE½°äÖË¼àÀ£'); b
disp('¿°Û¶¶È£'); s2
disp('¿°Û×Û¶¶È£'); min(s9)

```