

# 基于智能优化算法的太阳影子确定物体位置技术

## 摘要

文章在分析日晷原理基础上，建立物体在太阳下影子长度和物体所在位置太阳高度角间的关系，再进一步分析太阳高度角和物体位置（经度、纬度）关系式，从而找出了物体在太阳下影子长度和物体所在位置（经度、纬度）以及日期、时间之间的公式。在充分分析问题二、问题三和问题四后，将三个问题进行统一处理，将解非线性方程组问题转化为函数优化问题，并利用遗传算法进行求解，得到了比较好的结果。具体如下：

针对问题一，通过查阅相关文献，建立影子长度关于时间、太阳高度角、杆长之间的函数模型为  $L^2 = \frac{H^2[1 - (\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2]}{(\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2}$ ，将 2015 年 10 月 22 日的相关数据代入，计算得到：在 11 时 55 分 48 秒前，太阳影子长度由 6.585m 慢慢变小直到 3.675m，之后再慢慢增大直到 15 点时的 6.695m，影子曲线如图 4。

针对问题二、三，根据问题一的函数模型，假设经纬度、物体高度、日期为未知数，可以建立有三角函数组成的非线性多元方程组，将每个方程左右做差，再平方，进行求和，得到  $G(L_i, t_i, \lambda, \varphi)$  作为目标函数进行最优化，利用遗传算法进行求解得到：附件 1 数据对应地点为：三亚（经度 109.4642°，纬度 18.2554°）或海南（经度 109.8613°，纬度 18.8127°）；附件 2 数据对应的地点：新疆（经度 79.75°，纬度 39.53°）日期为：2015 年 1 月 13 日或青海省海东市（经度 101.9°，纬度 36.9°），日期为：2015 年 8 月 29 日。附件 3 内蒙古（经度 101.9°，纬度 36.9°），日期为：2014 年 4 月 18 日。

针对问题四：首先对附件 4 中的视频进行处理，处理步骤为：将 50 分钟的视频区时间间隔，时间间隔为 3 分钟一帧。找到三个关键点：影子的端点、杆子的顶点、杆子与影子的焦点（也就是底座的中心点），分别对这三个点周围的图片进行截取，对所截取的图片进行灰度处理，找到阈值估计值，取出界点，让图片转变成真值。在使用 MATLAB 对所处理的图片进行求坐标。求得杆子的像素差与影子的像素差，根据题中给出的杆高与像素差之间进行比例计算，计算出实际视频中所截取出来的 14 帧影子的长度，带入问题二所建立的遗传算法模型中，减少一个杆高的变量，运行得到地点为：经纬度：（72.1995134383368°， 22.6161028834597°）这一组拟合程度最好，地址为：印度

**关键词：**太阳高度角, MATLAB, 遗传算法, 灰度处理

## 一、问题重述

如何确定视频的拍摄地点和拍摄日期是视频数据分析的重要方面，太阳影子定位技术就是通过分析视频中物体的太阳影子变化，确定视频拍摄的地点和日期的一种方法。

1. 建立影子长度变化的数学模型，分析影子长度关于各个参数的变化规律，并应用你们建立的模型画出 2015 年 10 月 22 日北京时间 9:00-15:00 之间天安门广场（北纬 39 度 54 分 26 秒,东经 116 度 23 分 29 秒）3 米高的直杆的太阳影子长度的变化曲线。

2. 根据某固定直杆在水平地面上的太阳影子顶点坐标数据，建立数学模型确定直杆所处的地点。将你们的模型应用于附件 1 的影子顶点坐标数据，给出若干个可能的地点。

3. 根据某固定直杆在水平地面上的太阳影子顶点坐标数据，建立数学模型确定直杆所处的地点和日期。将你们的模型分别应用于附件 2 和附件 3 的影子顶点坐标数据，给出若干个可能的地点与日期。

4. 附件 4 为一根直杆在太阳下的影子变化的视频，并且已通过某种方式估计出直杆的高度为 2 米。请建立确定视频拍摄地点的数学模型，并应用你们的模型给出若干个可能的拍摄地点。

如果拍摄日期未知，你能否根据视频确定出拍摄地点与日期？

## 二、问题分析

### 2.1 问题一分析

问题一要求我们建立影子长度的变化规律模型。首先根据太阳的运动轨迹将立体图画成平面图，从图中找出直杆与影子的位置关系，建立三角函数关系。其中要对太阳高度角进行求解。我们通过查找文献找出太阳高度角的计算公式，最后得出函数关系式。然后将附录一中的数据代入函数关系式，应用 MATLAB 对函数进行求解并画出直杆的太阳影子长度的变化曲线图。

### 2.2 问题二分析

由第一问可知影长  $L$ ，杆高  $H$ ，经度  $\lambda$ ，纬度  $\alpha$  之间的关系可建立几个位置参数的关系，第二问中这某一根固定直杆在水平地面上的太阳影子顶点坐标数据已经给出，也就可以计算出每一时刻影子的长度  $L$ ，建立数学模型确定直杆所处的地点，也就是找到杆子所在的经纬度  $\lambda$ 、 $\alpha$ ，但是杆子的高度  $H$  依然为未知量。只有一个已知量，有三个未知量，并且几个未知量之间的函数关系是非线性的，无法进行实际的求解，所以通过遗传算法对其进行求解，找到最优解。

### 2.3 问题三的分析

问题三与问题二是类似的，不同之处在于问题三是在不知时间的情况下要求我们求解直杆的经纬度即地点，与此同时也要求出相应的年份与日期。因此，同样适用遗传算法对方程进行求解与优化。

### 2.4 问题四的分析

问题四在问题二、三的基础之上加上了一个视频处理的过程。问题二中给出了影子的  $x, y$  坐标，但是问题四中没有给出，首先就是要对视频中的影子的坐标进行提取。提取出来后适用遗传算法对方程进行求解与优化。

## 三、模型假设与符号系统

### 3.1 模型假设

- 1、假设天气状况良好，没有任何光线阻挠。
- 2、假设视频中的拍摄位置对影子的长短变化造成影响。
- 3、假设视频中的杆子与地面始终垂直。

### 3.2 符号系统

<b>L</b>	影长
<b>H</b>	直杆长
<b>h</b>	太阳高度角
$\lambda$	经度
$\delta$	赤纬角
<b>n</b>	修正日
<b>N</b>	日积数
<b>Y</b>	年份
<b>E<sub>Q</sub></b>	时差
<b>TT</b>	真太阳时
<b>t</b>	北京时间
<b>T</b>	地方时
$\phi$	纬度

## 四、模型一的建立与求解

### 4.1 模型的建立

立竿见影是大家都知的现象，建立适合的模型研究这一现象，我们考虑到影响这一现象的因素：杆长，时间，经纬度，太阳的位置。

地球相对于太阳的公转形成一年四季春夏秋冬，为 365 天，平均一天的转变角度  $\theta$ ，他的改变角度相对于地球的自传来说微不足道，所以我们改变相对运动的参照对象，参照物为地球上的静止的杆子，太阳相对地球产生运动。

太阳在天球上的位置每日、每时都在变化，要想确定它的位置通常使用的方式就是：用赤道坐标系和地平面坐标系从不同的角度来表示。又由于地理经纬度的不同，从地平面观察到的太阳轨迹就不一样，因此太阳准确的位置需要按太阳高度角来确定。太阳高度角是指太阳光的入射方向和地平面之间的夹角，在任何地区日出、日落时，太阳高度角最小；正午即为真太阳时 12 时，太阳高度角最大。在要知道太阳高度角时仍需要一个关键角度即为赤纬角。

由太阳与杆子的顶点连线与地面的焦点即为影子所投到的最终位置，影子、杆子、光线形成一个直角三角形（见图 1），使用三角函数：

$$\sin h = \frac{H}{\sqrt{H^2 + L^2}}$$

$$\text{即： } L^2 = \frac{H^2 [1 - (\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2]}{(\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2}$$

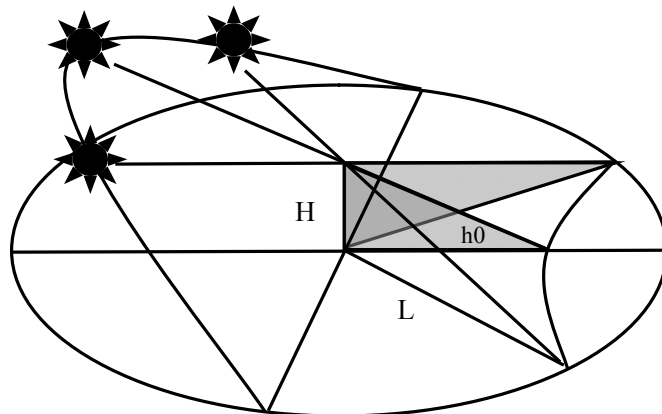


图 1

#### 4.1.1 太阳高度角

计算太阳高度  $h$ ，太阳高度角是指太阳光的入射方向和地平面之间的夹角，它的取

值范围是 $[0^{\circ}, 90^{\circ}]$ ，高度角  $h$  的计算公式是：

$$\sin h = \sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T \quad (1)$$

其中  $\alpha$  是纬度； $T$  是太阳时角， $\delta$  是赤纬角。

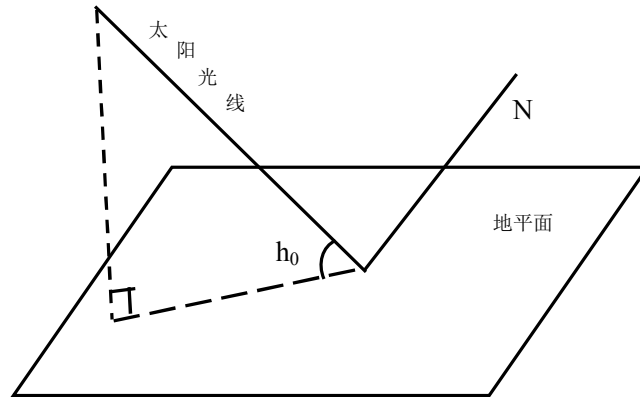


图 2

#### 4.1.2 赤纬角

计算太阳赤纬角  $\delta$ ，太阳赤纬角是地球赤道平面与太阳和地球中心连线之间的夹角<sup>[1]</sup>。其实也就是太阳和地球中心连线与南北极极轴之间的夹角的余角，因为极轴是赤道平面的法线。赤纬角是以年为周期，取值范围  $\delta$  是 $[-23^{\circ}27', +23^{\circ}27']$ ，最大值最小值在夏至冬至取到，春分秋分均为 0。（见图 3）

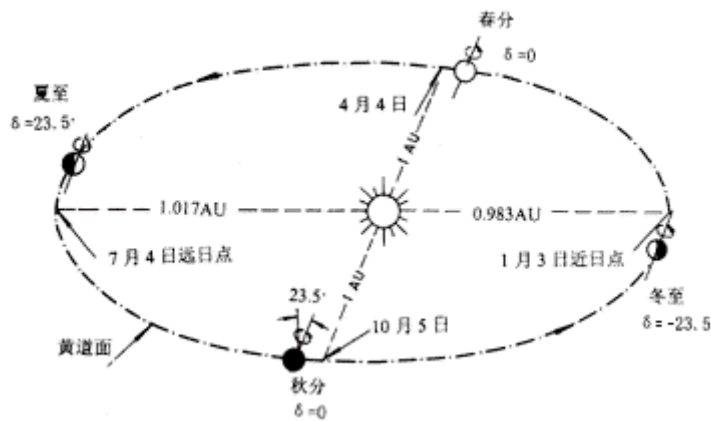


图 3

地球绕太阳公转的轨道是椭圆形而不是圆形，并且太阳的位置不是在圆形的中心而是在椭圆的一个焦点上，所以日地之间的距离在不同的位置就不会相同，但是要知道准确每日的日地距离就要知道日地平均距离，日地平均距离是  $RO$  又称天文单位<sup>[2]</sup>，

$$1 \text{ 天文单位} = 1.496 \times 10^8 \text{ km}$$

日地最近距离时叫做近日点为 0.983 天文单位，日期大概是 1 月 3 日，最远距离时叫做远日点为 1.017 天文单位，日期大概是 7 月 4 日，地球与太阳的距离为一天文单位时

期大概是 4 月 4 日和 10 月 5 日。这样局可以推知其实每一天的日地距离都是可以确定可知的，可以用一个数学表达式来表示，又由于数字的巨大的特点我们一般都以与日地

平均距离比例的平方来表示： $D_R = \left(\frac{r}{r_0}\right)^2$

则我们得到的数学表达式为：

$$D_R = 1.000423 + 0.032359 \sin \theta + 0.000086 \sin 2\theta - 0.008649 \cos \theta + 0.000115 \cos 2\theta \quad (2)$$

其中  $\theta$  为日角，即

$$\theta = \frac{2\pi z}{365.2422} \quad (3)$$

这里  $t$  又由两部分组成，即

$$z = N - n \quad (4)$$

$$\text{最终： } \theta = \frac{2\pi(N - n)}{365.2422}$$

式子中的  $N$  是某一天在这一年当中的序号，如 1 月 3 日的  $N$  就是 3，一般一年 12 月 31 日的  $N=365$  但是那一年是闰年就是  $N=366$ 。

式子中的  $n$  是修正日

$$n = 79.6764 + 0.2422 \times (Y - 1985) - INT\left(\frac{(Y - 1985)}{4}\right) \quad (5)$$

$Y$  是今年的年份， $INT$  是为取整

太阳赤纬角  $\delta$  在周年运动中的任何时刻的具体值就如日地距离一样可以严格的知道，使用数学公式表达就与  $D_R$  的公式 (2) 类似，即：

$$\begin{aligned} \delta = & 0.3723 + 23.2567 \sin \theta + 0.1149 \sin 2\theta - 0.1712 \sin 3\theta - 0.758 \cos \theta \\ & + 0.3656 \cos 2\theta + 0.0201 \cos 3\theta \end{aligned} \quad (6)$$

其中  $\theta$  的含义与 (2) 式相同，为日角。

#### 4.1.3 时差

又由于真太阳的运动不是匀速的，他的变化与日地距离之间的变化  $D_R$  (2) 又是相似相似，其表达式是：

$$\begin{aligned} E_Q = & 0.028 - 1.9857 \sin \theta + 9.9059 \sin 2\theta \\ & - 7.0924 \cos \theta - 0.6882 \cos 2\theta \end{aligned} \quad (7)$$

$$TT = \left(t - \frac{(120 - \lambda) \times 4}{60} + E_Q\right) \quad (8)$$

$TT$  是代表着真太阳时，为了从北京时求出真太阳时需要两个步骤：首先需要将北京时间  $t$  换成地方时  $S_D$ ：

$$S_D = t - \frac{(120^\circ - \lambda) \times 4}{60} \quad (9)$$

式子中， $\lambda$  为经度， $120^\circ$  是北京时的标准经度，乘 4 是因为每一度相差四分钟，除以 60 就是将分钟换成小时。

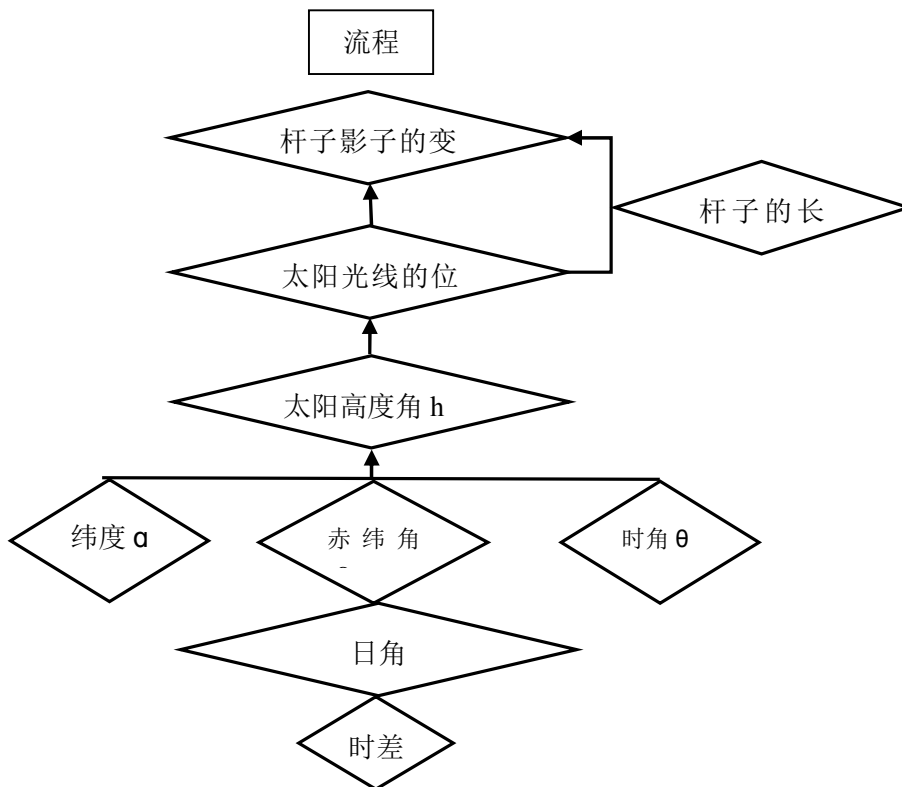
其次就是进行时差修订，这里是为了指出，时角是以太阳正午刻为 0 点的，上午为负下午为正。修正公式，即

$$\begin{aligned} T &= (TT - 12) \times 15^\circ \\ &= \left( t - \frac{(120 - \lambda) \times 4}{60} + E_Q - 12 \right) \times 15^\circ \\ &= 15t - 300 + \lambda - 15E_Q \end{aligned} \quad (10)$$

#### 4.1.4 综合

计算出太阳高度角  $h$ ，最终计算影长其：

$$L^2 = \frac{H^2 [1 - (\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2]}{(\sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T)^2}$$



## 4.2 模型一的求解

当日即为 2015 年 10 月 22 日的日积数为 295，经度为 116.30°，纬度 39.91°，杆高 3 米带入模型中求解。（见图 4）

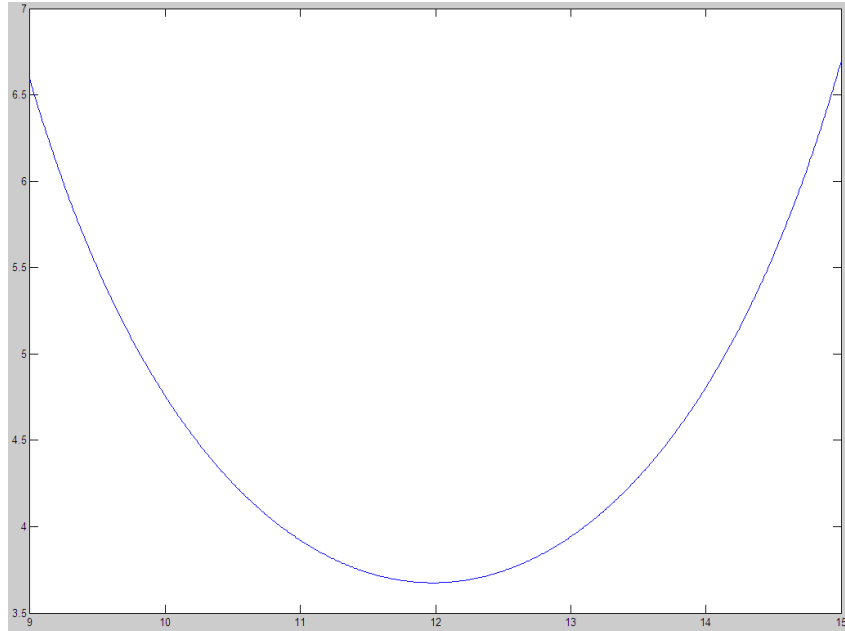


图 4

## 五、模型二的建立与求解

### 5.1 模型二的建立:基于遗传算法的数学模型

遗传算法<sup>[3]</sup>是一类依据生物界的进化规律而演化而来的随机进行探索的方法，它来自达尔文的生物进化论，它可以直接对研究对象的结构进行操作，不存在求导和函数连续性的限定；具有内在的隐并行性和更好的全局寻优性。它提供了一个求解复杂系统优化问题的模式，不依赖于问题的领域、种类。它是采用概率化来寻找方法，能自动获取和指导优化的搜索空间，自适应地调整搜索方向。

根据问题一中的公式  $L^2 = \frac{H^2}{\sin^2 h} - H^2 = \frac{H^2(1 - \sin^2 h)}{\sin^2 h}$  (12) 来建立目标函数：

$$1) L^2_1 = \frac{H^2(1 - \sin^2 h_1)}{\sin^2 h_1}$$

$$2) L^2_2 = \frac{H^2(1 - \sin^2 h_2)}{\sin^2 h_2}$$

.....

$$n) L^2_n = \frac{H^2(1 - \sin^2 h_n)}{\sin^2 h_n}$$

左右相减得出下式



$$L^2_1 - \frac{H^2(1 - \sin^2 h_1)}{\sin^2 h_1} = 0$$

$$L^2_2 - \frac{H^2(1 - \sin^2 h_2)}{\sin^2 h_2} = 0$$

...

$$L^2_n - \frac{H^2(1 - \sin^2 h_n)}{\sin^2 h_n} = 0$$

建立目标函数对上式的方程进行求和得出下式

$$G(L_i, t_i, \lambda, \varphi) = \sum_i^n \left( L^2_i - \frac{H^2(1 - \sin^2 h_i)}{\sin^2 h_i} \right)^2$$

其中  $\sinh = \sin \alpha \sin \delta + \cos \alpha \cos \delta \cos T$

$$E_Q = 0.028 - 1.9857 \sin \theta + 9.9059 \sin 2\theta - 7.0924 \cos \theta - 0.6882 \cos 2\theta$$

$$TT = \left( t - \frac{(120 - \lambda) \times 4}{60} + E_Q \right)$$

$$= \left( t - \frac{(120 - \lambda) \times 4}{60} + E_Q - 12 \right) \times 15^0$$

$$= 15t - 300 + \lambda - 15E_Q$$

将  $G(L_i, t_i, \lambda, \varphi) = \sum_i^n \left( L^2_i - \frac{H^2(1 - \sin^2 h_i)}{\sin^2 h_i} \right)^2$  作为目标函数，采用二进制编码、多点交叉，

利用 Matlab 计算平台自带的 GA 工具箱进行求解。算法参数如下：种群规模 200，最大代数 500，交叉概率 0.95，变异概率 0.05。最终求得结果为：经纬度为（109.4642°，18.2554°）地点：三亚；经纬度为（109.8613°，18.8127°）地点：海南

## 六、模型二的建立与求解

### 6.1 问题三模型的建立

对于问题三，将经纬度、日期和杆高均作为未知量，利用模型二中的  $G(L_i, t_i, \lambda, \varphi)$  作为对应目标函数，采用模型二求解参数，解的结果为：

附件二数据对应的地点：

- (1) 新疆（经度 79.75°，纬度 39.53°）日期为：2015 年 1 月 13 日
- (2) 青海省海东市（经度 101.9°，纬度 36.9°）日期为：2015 年 8 月 29 日。

附件三数据对应的地点：

(1) 内蒙古（经度 101.9°，纬度 36.9°），日期为：2014 年 4 月 18 日。

## 七、问题四模型的建立

### 7.1 模型的建立

用数字图像处理技术，并通过 MATLAB 语言编程，实现对视频中影子位置参数的自动测量。一幅图片被保存到计算机当中都是被划分成好多个小的区域被称为图像单元，图像元素被称为像素。每一个图像单元的像素值得多少就是图片对应的亮度的直接反应。

首先对视频进行处理，把视频读取进入 MATLAB，知道其大小为 61020。视频时间跨度为 2015 年 7 月 13 日 8 时 54 分 6 秒到 2015 年 7 月 13 日 9 时 34 分 46 秒。我们取  $t_0 = 8h54\text{min}06ma$  这一秒的图片为第一帧，将时间的分钟与秒都转换成小时，即  $t_0 = 8.9167$ 。

取出第一帧的图片建立  $x, y$  轴，以左上角为原点  $o$ ，向右的方向为  $x$  轴，以向下的方向为  $y$  轴。（见图 4-1）



图 4-1

自然情况下计算机不可能直接在一张图片上找到某一物体的具体参数数据，进而将一幅图像在计算机进行处理前一般都必须转化成数字形式。向  $x, y$  轴方向数值依次增大。

由于图片容量较大，但是对于结题我们只需要的几点：

- (1) 影子的端点  $A_1$
- (2) 杆子的顶点  $A_2$
- (3) 杆子与影子的焦点  $A_3$  (也就是底座的中心点)

7.1.1 对  $A_1$  (影子端点) 进行数据实量化:

步骤如下:

(1) 截图: 由于只需要知道影子的端点的位置, 不用整张图片, 所以只对图 4-1 中的影子部分通过 MATLAB 图片中在上面找到影子大概所处图片中的区间 (见图 4-2、4-3)



图 4-1

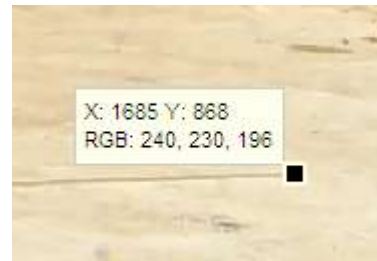


图 4-3

(2) 灰度处理: 由上面的预估计值确定截取  $x \in (,1800), y \in (850,900)$  截取出来的图片, 并且进行灰度处理。图片见图4-4



图 4-4

(3) 阈值预估计

每一幅图片中每一个图像单元的像素值得多少就是图片对应的亮度的直接反应。进行灰度处理后寻找阈值。阈值与寻找  $x, y$  坐标的区间方法一样。见图 4-5、4-6

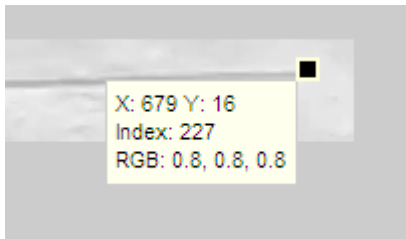


图 4-5

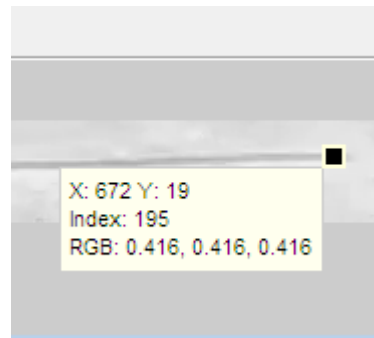
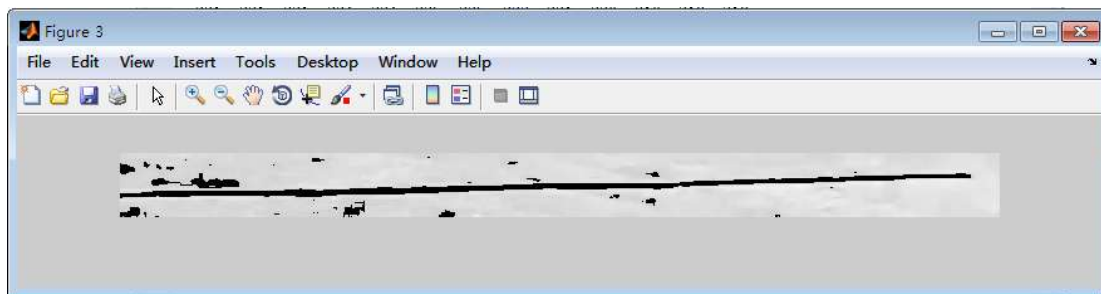


图 4-6

#### (4) 阈值分化

阈值经过估计大概为 210 左右，就当阈值小于 210 时取 1，在 MATLAB 中进行运行，得出处理后的图片，见图 4-7

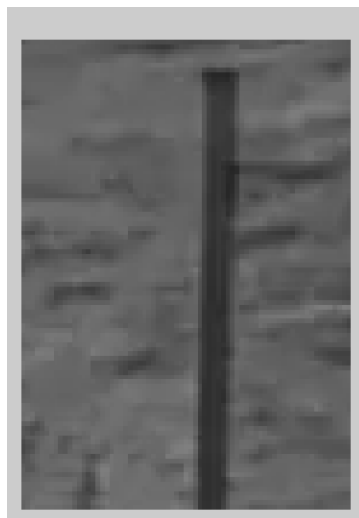


#### (5) 计算第一帧影子端点的坐标 图 4-7

使用 MATLAB 编程在上图中找到最远点的纵坐标  $y$ ，再找到这  $y$  值所对应得  $x$  值，多组  $x$  取其平均数。输出坐标  $(x_1, y_1) = (869, 1678)$

#### 7.1.2 对 $A_2$ (杆子的端点) 进行数据实量化:

##### (1) 截取 (图 4-8)



12  
图 4-8

(2) 灰度处理 (图 4-8)

(3) 阈值预估计 (图 4-9、4-10)

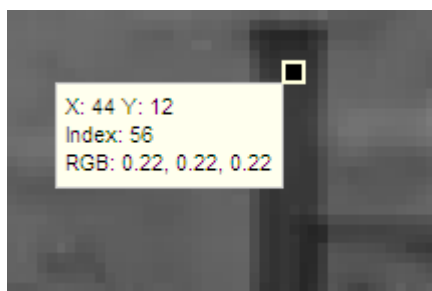


图 4-9

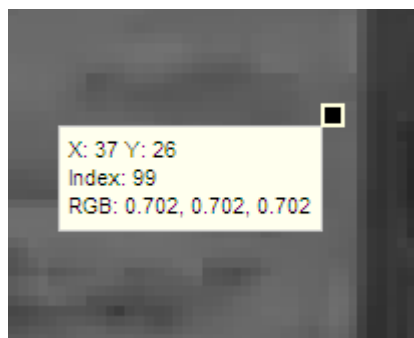


图 4-10

(4) 阈值分化

阈值经过估计大概为左右, 就当阈值小于 50 时取 1, 在 MATLAB 中进行运行, 得出处理后的图片 (图 4-11)

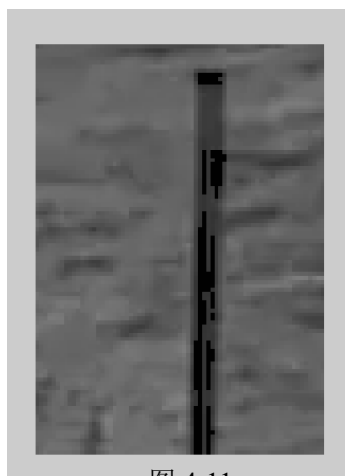


图 4-11

(5) 计算第一帧杆子端点的坐标

使用 MATLAB 编程在上图中找到横坐标  $gx$  纵坐标  $gy$ , 这里的  $gx, gy$  分别取最大最小值之间的中位数。输出坐标  $(gx, gy) = (208, 893)$

7.1.3 对  $A_3$  (杆子与影子的焦点) 进行数据实量化:

(1) 截取 (图 4-12)

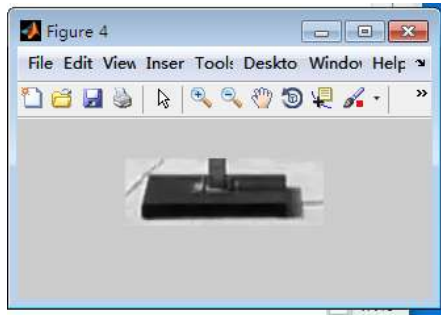


图 4-12

(2) 灰度处理 (图 4-12)

(3) 阈值预估计 (图 4-13、4-14、4-15、4-16)

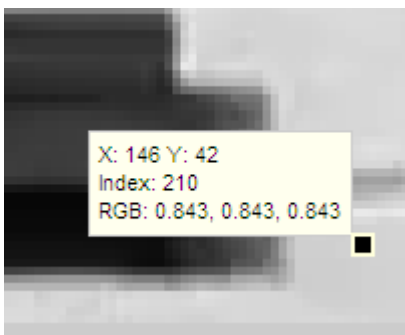


图 4-13

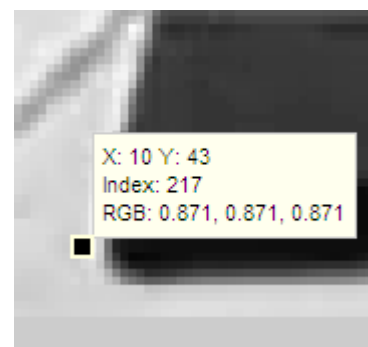


图 4-14

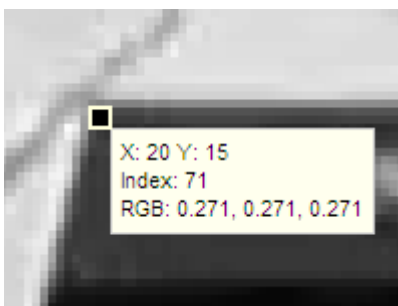


图 4-15

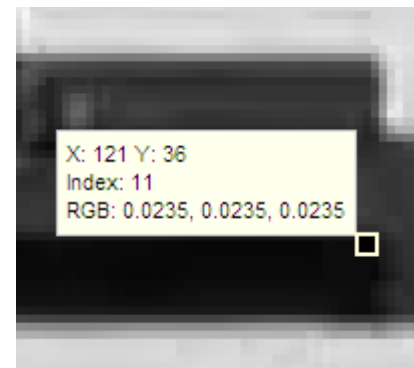


图 4-16

(4) 阈值分化

阈值经过估计大概为左右，就当阈值小于 50 时取 1，在 MATLAB 中进行运行，得出处理后的图片 (图 4-17)

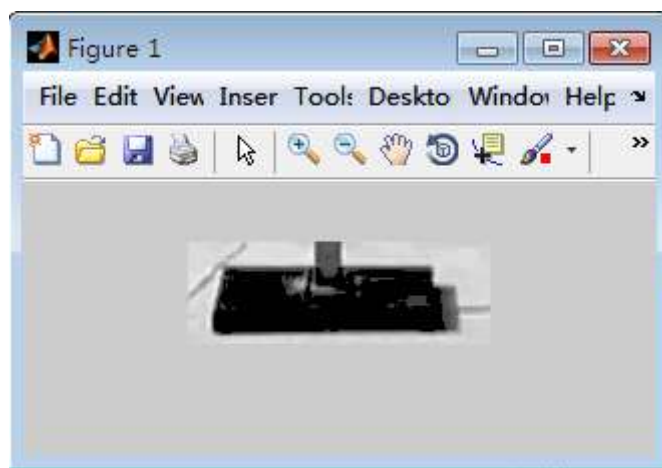


图 4-17

(5) 计算第一帧杆子与影子的焦点的坐标

使用 MATLAB 编程在上图中找到横坐标  $x_0$  纵坐标  $y_0$ ，这里的  $x_0, y_0$  分别取最大最小值之间的中位数。输出坐标  $(x_0, y_0) = (880, 893)$

最终代入模型二遗传算法中进行运算得到最终结果

## 7.2 问题四模型的求解

根据模型中对第一帧的处理，当每隔三分钟的时候在取出其他帧。

运行 MATLAB 的最终 14 组结果见下表：

t	x	y	xx	yy
8.9167	869	1678	-0.0327	2.3363
8.9667	870	1664	-0.0298	2.2946
9.0167	872	1649	-0.0238	2.25
9.0667	873	1633	-0.0208	2.2024
9.1167	882	1693	0.006	2.381
9.2167	876	1605	-0.0119	2.119
9.3167	878	1592	-0.006	2.0804
9.3367	880	1578	0	2.0387
9.4167	880	1566	0	2.003
9.4667	898	1562	0.0536	1.9911
9.5167	884	1537	0.0119	1.9167
9.5667	901	152	0.0625	1.878
	882	1565	0.006	2
	874	1566	-0.0178	2.003

表 1

带入遗传算法中得：

经纬度：（75.7841654387737°， 24.2226298782621°）

经纬度：（80.5384150823715°， 23.1132786439848°）

经纬度：（77.1055842206714°， 21.1596507006832°）

经纬度：（76.2738924663349°， 20.1905857054977°）

经纬度：（72.1995134383368°， 22.6161028834597°）

最后一组拟合程度最好，地址为：印度

## 八、模型的优缺点

### 8.1 优点

- 1、问题一中对经度、时差、日期进行了修正，减小了误差。
- 2、遗传算法即可以处理线性连续函数，又可以处理非线性不连续的函数。
- 3、遗传算法在运行过程中很难出现局部最优，很大程度上可以找到全局最优解
- 4、遗传算法中不管变量变成多少个该模型皆可推广

### 8.1 缺点

- 1、问题四中阈值只是手动找出，没有使用阈值寻找函数可能不准确；
- 2、不考虑到拍摄角度的不同产生的影响。

## 九、参考文献

- [1]林根石，利用太阳视坐标的计算进行物高测量与定位[J]，1991年9月15日，南京林业大学学报第51卷第3期
- [2]于贺军，气象用太阳赤纬和时差计算方法研究[J]，2006年9月3日文章编号：1006-009X-(2006)03-0050—04，(北京交通大学，北京 100044)
- [3]金芬，遗传算法在函数优化中的应用与研究[D]，2008年，苏州大学
- [4]谢金星，数学建模[D]，第三版，高等教育出版社，2003年08月
- [5]丁雪荣，基于MATLAB图像处理技术的研究[J]，2006年，天津大学
- [6]高尚，基于MATLAB遗传算法优化工具箱的优化计算[D]，2002年18(8)
- [7]王炳忠，MATLAB绘图，太阳能中天文参数的计算[J]，1999年8月10日



**Main:**

```
%UNTITLED3 此处显示有关此函数的摘要
% 此处显示详细说明
clear all
sprintf('请输入算法参数:')
fun=input('请输入优化函数名(字符串):','s');
fun_max=input('请估计函数最大值:');
n=input('请输入优化函数参数个数:');
bounds=[];
for i=1:n
    fprintf('请输入第%d个参数范围(共%d个,格式为:[下限,上限]):',i,n)
    temp=input('');
    bounds=[bounds;temp];
end
eps=input('请输入精度(精度越高(编码的长度越长):');
max_pop=input('请输入种群规模:');
max_gen=input('请输入最大代数:');
pc=input('请输入交叉概率:');
pm=input('请输入变异概率:');
iteration=input('请输入运算次数:');

junyun_iter_bestv=[];
junyun_iter_bestx=[];
%junyun_gen=[];
for i=1:iteration

[junyun_bestx,junyun_bestv,junyun_end_gen,junyun_popmax,junyun_popave]=my_f
un_junyun_sga(max_pop,fun,bounds,eps,max_gen,pc,pm,fun_max,i);

    junyun_iter_bestv=[junyun_iter_bestv,junyun_bestv];
    junyun_iter_bestx=[junyun_iter_bestx;junyun_bestx];
    %junyun_gen=[junyun_gen,junyun_end_gen];
end
```

## GenerateCoordinates:

```
function [t,x,y,x0,y0,gx,gy,xx,yy] =GenerateCoordinates ( )
%UNTITLED 此处显示有关此函数的摘要
% 读取视频，产生杆子影子端点坐标
clear all;
[name, pathname, ~] =
uigetfile({'*.avi;*.rnvb;*.rm;*.asf;*.divx;*.mpg;*.mpeg;*.mpe;*.wmv;*.mp4;*.
.mkv;*.vob','file (*.avi, *.rmvb,*.rm,*.asf,*.mp4,*.mkv,*.vob)'},'请选择输入数
据的文件名');
obj=VideoReader([pathname,name]);
numFrames = obj.NumberOfFrames;
%H=fix(obj.Height/2);
%W=fix(obj.Width/3);
%每隔 3 分钟读取一帧图片，求出影子端点像素坐标
i=1;
k=1;
t0=8.9167;%计算初始时间为：8 点 54 分 06 秒
while k<=numFrames% 读取数据
    frame = read(obj,k);
    temp{i}=rgb2gray(frame(850:900,1000:1700,:)); %利用手工方法取图像的部分
    A=temp{i};
    ind= A<210;%利用手工方法估计出阈值 210
    A(ind)=1;
    [~,col]=find(A==1);
    maxcol=max(col);
    ind=find(A(:,maxcol)==1);
    maxrow=fix(mean(ind));
    x(i)=850+maxrow;
    y(i)=1000+maxcol;
    t(i)=t0+(i-1)*(3/60);
    i=i+1;
    k=k+obj.frameRate*180;
end

%求杆子上端点像素坐标
A=rgb2gray(frame(200:300,850:920,:));
ind= A<=50;
A(ind)=1;
[row,~]=find(A==1);
gx=min(row);
ind=find(A(gx,:)==1);
gy=fix(mean(ind));
gx=gx+200;
```

```

gy=gy+850;

%求原点像素坐标
frame = read(obj,1);
A=rgb2gray(frame(850:900,800:950,:));
ind= A<=60;
A(ind)=1;
[row,col]=find(A==1);
x0=fix((max(row)-min(row))/2)+min(row)+850;
%y0=fix((max(col)-min(col))/2)+min(col)+800;
y0=gy;

%计算像素和长度(米)间的比例关系 beta
beta=2/abs(gx-x0);

%以(x0,y0)为原点,影子端点的真实坐标
m=length(x);
for i=1:m
    xx(i)=beta*(x(i)-x0);
    yy(i)=beta*(y(i)-y0);
end
end

```

Funvall:

```
function [ F ] = Funvall( x )
%UNTITLED2 此处显示有关此函数的摘要
% 此处显示详细说明
arf=x(1);
fri=x(2);
h=x(3);
%N=x(4);
N=108;
%Y=x(5);
Y=2015;
n=79.6764+0.2422*(Y-1985)-7;
m=N-n;
Q=2*pi*m/365.2422;
%EQ=0.5523;
EQ=0.0028-1.9857*sin(Q)+9.9059*sin(2*Q)-7.0924*cos(Q)-0.6882*cos(2*Q);
%delta=10.6305;
delta=0.3723+23.2567*sin(Q)+0.1149*sin(2*Q)-0.1712*sin(3*Q)-0.758*cos(Q)+0.3656*cos(2*Q)+0.0201*cos(3*Q);
L=[1.1496 1.1822 1.2153 1.2491 1.2832 1.3180 1.3534 1.3894 1.4262 1.4634 1.5015
1.5402 1.5799 1.6201 1.6613 1.7033 1.7462 1.7901 1.8350 1.8809 1.9279];
t=[14.7 14.75 14.8 14.85 14.9 14.95 15 15.05 15.1 15.15 15.2 15.25 15.3 15.35
15.4 15.45 15.5 15.55 15.6 15.65 15.7];
n=length(t);
F=0;
for i=1:n
    tao=15*t(i)-300+arf+15*EQ;

temp=1/((sin(fri*pi/180)*sin(delta*pi/180)+cos(fri*pi/180)*cos(delta*pi/180)
)*cos(tao*pi/180))^2);
    F=F+(L(i)^2-h^2*(temp-1))^2;
end
end
```

Funval2:

```
function [ F ] = Funval2( x )
```

```
%UNTITLED2 此处显示有关此函数的摘要
```

```
% 此处显示详细说明
```

```
arf=x(1);  
fri=x(2);  
h=x(3);  
N=x(4);  
%N=108;  
Y=x(5);  
%Y=2015;  
n=79.6764+0.2422*(fix(Y)-1985)-7;  
m=fix(N)-n;  
Q=2*pi*m/365.2422;  
%EQ=0.5523;  
EQ=0.0028-1.9857*sin(Q)+9.9059*sin(2*Q)-7.0924*cos(Q)-0.6882*cos(2*Q);  
%delta=10.6305;  
delta=0.3723+23.2567*sin(Q)+0.1149*sin(2*Q)-0.1712*sin(3*Q)-0.758*cos(Q)+0.  
3656*cos(2*Q)+0.0201*cos(3*Q);  
L=[1.2473 1.2228 1.1989 1.1754 1.1524 1.1299 1.1078 1.0863 1.0651 1.0444 1.0243  
1.0046 0.9855 0.9668 0.9486 0.9309 0.9138 0.8971 0.8810 0.8655 0.8505];  
t=[12.6833 12.7333 12.7833 12.8333 12.8833 12.9333 12.9833 13.0333 13.0833  
13.1333 13.1833 13.2333 13.2833 13.3333 13.3833 13.4333 13.4833 13.5333 13.5833  
13.6333 13.6833];  
%L=[1.1496 1.1822 1.2153 1.2491 1.2832 1.3180 1.3534 1.3894 1.4262 1.4634 1.5015  
1.5402 1.5799 1.6201 1.6613 1.7033 1.7462 1.7901 1.8350 1.8809 1.9279];  
%t=[14.7 14.75 14.8 14.85 14.9 14.95 15 15.05 15.1 15.15 15.2 15.25 15.3 15.35  
15.4 15.45 15.5 15.55 15.6 15.65 15.7];  
n=length(t);  
F=0;  
for i=1:n  
    tao=15*t(i)-300+arf+15*EQ;  
  
    temp=1/((sin(fri*pi/180)*sin(delta*pi/180)+cos(fri*pi/180)*cos(delta*pi/180  
) *cos(tao*pi/180))^2);  
    F=F+(L(i)^2-h^2*(temp-1))^2;  
end  
end
```

### Funval3:

```
function [ F ] = Funval3( x )
%UNTITLED2 此处显示有关此函数的摘要
% 此处显示详细说明
arf=x(1);
fri=x(2);
h=x(3);
N=x(4);
%N=108;
Y=x(5);
%Y=2015;
n=79.6764+0.2422*(fix(Y)-1985)-7;
m=fix(N)-n;
Q=2*pi*m/365.2422;
%EQ=0.5523;
EQ=0.0028-1.9857*sin(Q)+9.9059*sin(2*Q)-7.0924*cos(Q)-0.6882*cos(2*Q);
%delta=10.6305;
delta=0.3723+23.2567*sin(Q)+0.1149*sin(2*Q)-0.1712*sin(3*Q)-0.758*cos(Q)+0.3656*cos(2*Q)+0.0201*cos(3*Q);
L=[3.5331 3.5468 3.5618 3.5781 3.5958 3.6149 3.6354 3.6572 3.6805 3.7052 3.7313
3.7589 3.7881 3.8187 3.8508 3.8846 3.9199 3.9569 3.9955 4.0358 4.0779];
t=[13.15 13.20 13.25 13.30 13.35 13.40 13.45 13.50 13.55 13.60 13.65 13.70 13.75
13.80 13.85 13.90 13.95 14.00 14.05 14.10 14.15];
%L=[1.2473 1.2228 1.1989 1.1754 1.1524 1.1299 1.1078 1.0863 1.0651 1.0444 1.0243
1.0046 0.9855 0.9668 0.9486 0.9309 0.9138 0.8971 0.8810 0.8655 0.8505];
%t=[12.6833 12.7333 12.7833 12.8333 12.8833 12.9333 12.9833 13.0333 13.0833
13.1333 13.1833 13.2333 13.2833 13.3333 13.3833 13.4333 13.4833 13.5333 13.5833
13.6333 13.6833];
%L=[1.1496 1.1822 1.2153 1.2491 1.2832 1.3180 1.3534 1.3894 1.4262 1.4634 1.5015
1.5402 1.5799 1.6201 1.6613 1.7033 1.7462 1.7901 1.8350 1.8809 1.9279];
%t=[14.7 14.75 14.8 14.85 14.9 14.95 15 15.05 15.1 15.15 15.2 15.25 15.3 15.35
15.4 15.45 15.5 15.55 15.6 15.65 15.7];
n=length(t);
F=0;
for i=1:n
    tao=15*t(i)-300+arf+15*EQ;

    temp=1/((sin(fri*pi/180)*sin(delta*pi/180)+cos(fri*pi/180)*cos(delta*pi/180)
)*cos(tao*pi/180))^2);
    F=F+(L(i)^2-h^2*(temp-1))^2;
end
end
```

## Funval41:

```
function [ F ] = Funval41( x )
%UNTITLED2 此处显示有关此函数的摘要
% 此处显示详细说明
arf=x(1);
fri=x(2);
h=2;
N=194;
%N=108;
Y=2015;
%Y=2015;
n=79.6764+0.2422*(fix(Y)-1985)-7;
m=fix(N)-n;
Q=2*pi*m/365.2422;
%EQ=0.5523;
EQ=0.0028-1.9857*sin(Q)+9.9059*sin(2*Q)-7.0924*cos(Q)-0.6882*cos(2*Q);
%delta=10.6305;
delta=0.3723+23.2567*sin(Q)+0.1149*sin(2*Q)-0.1712*sin(3*Q)-0.758*cos(Q)+0.3656*cos(2*Q)+0.0201*cos(3*Q);
L=[2.4020 2.3871 2.3424 2.3156 2.2918 2.2530 2.4464 2.1994
2.1726 2.1458 2.1131 2.0923 2.0685 2.0596 2.0060 1.9822
2.0596 2.0656 2.0655 2.0626 2.1728];
t=[8.9167 8.9667 9.0167 9.0667 9.1167 9.1667 9.2167 9.2667
9.3167 9.3667 9.4167 9.4667 9.5167 9.5667 9.6167 9.6667
9.7167 9.7667 9.8167 9.8667 9.9167];
n=length(t);
F=0;
for i=1:n
    tao=15*t(i)-300+arf+15*EQ;

temp=1/((sin(fri*pi/180)*sin(delta*pi/180)+cos(fri*pi/180)*cos(delta*pi/180)
)*cos(tao*pi/180))^2);
    F=F+(L(i)^2-h^2*(temp-1))^2;
end
end
```

## Funval42:

```
function [ F ] = Funval42( x )
%UNTITLED2 此处显示有关此函数的摘要
% 此处显示详细说明
arf=x(1);
fri=x(2);
N=x(3);
%N=108;
Y=x(4);
%Y=2015;
h=2;
n=79.6764+0.2422*(fix(Y)-1985)-7;
m=fix(N)-n;
Q=2*pi*m/365.2422;
%EQ=0.5523;
EQ=0.0028-1.9857*sin(Q)+9.9059*sin(2*Q)-7.0924*cos(Q)-0.6882*cos(2*Q);
%delta=10.6305;
delta=0.3723+23.2567*sin(Q)+0.1149*sin(2*Q)-0.1712*sin(3*Q)-0.758*cos(Q)+0.3656*cos(2*Q)+0.0201*cos(3*Q);
L=[2.4020  2.3871  2.3424  2.3156  2.2918  2.2530  2.4464  2.1994
 2.1726  2.1458  2.1131  2.0923  2.0685  2.0596  2.0060  1.9822
 2.0596  2.0656  2.0655  2.0626  2.1728];
t=[8.9167  8.9667  9.0167  9.0667  9.1167  9.1667  9.2167  9.2667
 9.3167  9.3667  9.4167  9.4667  9.5167  9.5667  9.6167  9.6667
 9.7167  9.7667  9.8167  9.8667  9.9167];
n=length(t);
F=0;
for i=1:n
    tao=15*t(i)-300+arf+15*EQ;

    temp=1/((sin(fri*pi/180)*sin(delta*pi/180)+cos(fri*pi/180)*cos(delta*pi/180)
    )*cos(tao*pi/180))^2);
    F=F+(L(i)^2-h^2*(temp-1))^2;
end
end
```