
武汉理工大学

数学建模暑期培训论文

第 1 题

基于非线性优化的龙门吊运动规划模型

第 21 组

姓名	方向
肖善	建模
易雨谦	建模
林荣武	编程

2016 年 7 月 22 日

基于非线性优化的龙门吊运动规划模型

摘要

本文针对不同目标要求下的龙门吊吊运运动规划问题，建立了不同背景下的基于动力学分析的非线性优化模型，求解微分方程组将其转化为线性优化问题，并结合评价函数法在 matlab 中用遗传算法进行求解。

对于问题一，考虑 BC 段（即第四时间段）吊运过程只要求摆动最小的目标，建立了以四段时间为决策变量的优化模型。首先，以吊车为研究对象，地面为参考系，确定吊车与货物位置关系，根据动力学分析对系统用拉格朗日法建立关于摆角的二阶常系数线性非齐次微分方程组，分别求解得到四段时间摆角与时间的函数关系，以第四阶段最大摆角最小为目标，并根据货物到达终点的最大速度限制，总位移一定等运动学公式建立相应的约束条件，首先用 Lingo 求出初步结果，再用遗传算法求解进行结果对比分析，最终得出总时间约为 52.39 秒，最大摆角的最小值为 0.0154rad 即 0.88 度，在 0 到 3 度范围之内，因此，我们认为结果是合理的且符合实际。

对于问题二，考虑 BC 段要求兼顾效率最高（即经过同样的行程用时最短），且摆动最小，在第一问的基础上，增加一个目标函数变为双目标规划问题，首先找到两个理想点，建立评价函数用遗传算法求解得到距离两个理想点距离最小的综合最优解：全程所用总时间 51.373 秒最大摆角最小值为 0.0174rad 即 0.996 度，相对于第一问总时间减少约 1.94%，角度增加约 12.9%，我们认为结果是合理的且符合实际。

对于问题三，考虑最大承载力，同时兼顾时间最短、最大摆角最小、能吊运的货物重量最大建立了三目标规划模型，同样建立评价函数找到三个理想点，用遗传算法得出综合最优结果为：总用时 94.1 秒，加速度 $0.697 m/s^2$ 。

对于问题四，结合实际情况从三个方面考虑更多因素：三维摆动、绳长变化、阻力影响建立了模型并用 simulink 进行仿真，在 matlab 中创建了龙门吊运动规划问题的 GUI 界面。

关键词：非线性优化 拉格朗日法 评价函数法 遗传算法 吊运规划

一、问题重述

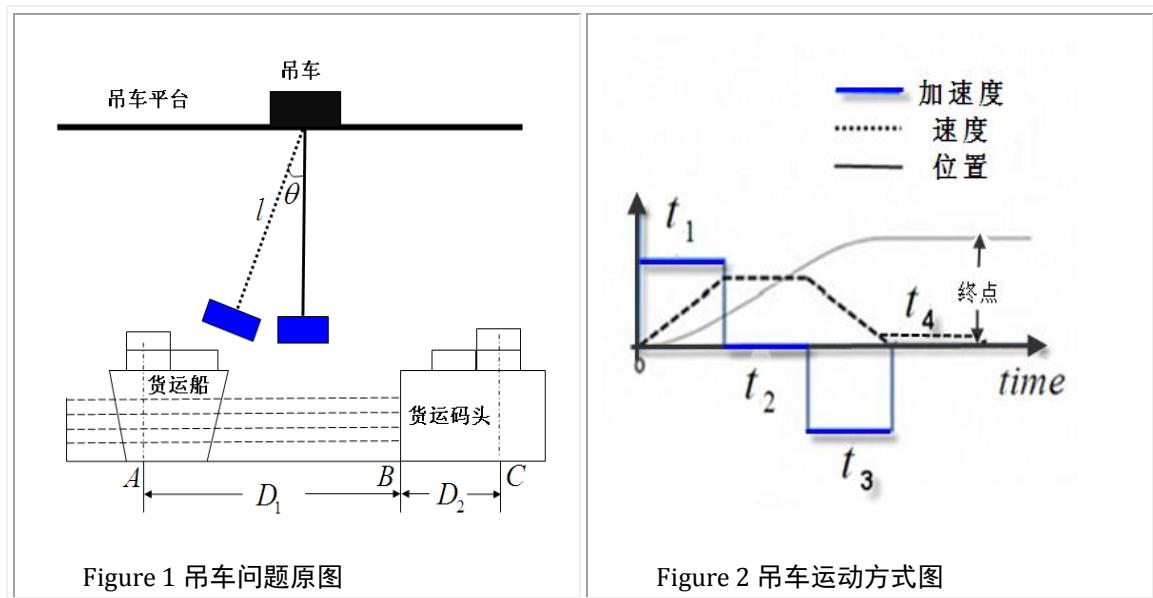
一带有动力驱动的货运吊车,需将货物从货运船的起吊点 A 吊运到货运码头的终点 B。已知吊车运动方式: AB 段先以加速度 a 匀加速运行,达到一定速度后,匀速运行,然后以加速度 a 匀减速运行; BC 段匀速运行,四段运动时间依次为 t_1 、 t_2 、 t_3 、 t_4 ,AB 间的距离为 D_1 米,BC 间的距离为 D_2 米,缆绳长度为 1 米,缆绳最大拉力 $T_{\max} = 20000\text{kg}$,吊车运行最大加速度 $a_{\max} = 1\text{m/s}^2$,要求货物到达终点时 $v_x \leq 0.5\text{m/s}$ 。

问题一: 已知加速度及货物质量,在不考虑吊车缆绳承载能力的情况下,建立模型确定吊车的运行方式,即确定时间 t_i , $i=1,2,3,4$, 使货物在 BC 段摆动最小。

问题二: 在第一问的假设下,建立模型确定时间,使货物在 BC 段不仅摆动幅度小而且整体效率高。

问题三: 考虑绳的最大承载力,并兼顾摆动大小和效率,建立模型确定运动方式(确定 a 、时间 t_i) 及最大起吊货物重量 m_{\max} 。

问题四: 对于实际问题,考虑更多因素且相对完善的数学模型,并为该公司开发设计一套用户界面,直观演示龙门吊吊运问题。



二、问题分析

2.1 问题一的分析

问题一属于二维定绳长龙门吊防摆动非线性规划问题,给出了加速度 a 需要确定吊车运动的四个时段,要使得货物在 BC 段运动过程最大摆角最小,属于非线性优化问题。首先,建立坐标系,通过摆角建立吊车与货物位置坐标的联系,然后对位移求一阶倒数得到速度分量关系从而得出系统动能;其次,对系统用 Lagrange 建模,即“分析动力学”建模,优点在于利用“分析动力学”建模不需要分析作用于系统的各个质点或刚体上的力、力矩或加速度等物理量,而是将力学的基本定律表示为数学的形式,并通过位形空间描述力学系统的运动问题,它适合于研究受约束质点系的运动。然后得到运动系统模型非线性微分方程组,通过 θ 较小这一条件进行模型简化,把它转化为二阶常系数线性非齐次微分方程组的求解问题,给定边界条件,从而用遗传算法求出解析解得到摆角与时间的关系。最后,问题转化为以最大摆角最小值为目标函数、各运动参数为约束条件、 t_i 为决策变量的单目标线性规划的最优解问题。

2.2 问题二的分析

问题二在问题一的基础上增加了一个目标,要求确定四个时间段满足摆动最小的同时整体效率最高,属于双目标非线性优化问题。首先,确定两个目标函数即:龙门吊车经过相同的位移($D_1 + D_2$)所用时间之和最短(吊运效率最高),同时满足摆角的最大值最小。接着,考虑到对于多目标规划问题有约束法、分层序列法、功效系数法、评价函数法等,但大多对于权重的赋值具有主观性,我们选择评价函数法中的理想点法,从而避开用主观方式确定主目标或者确定权重,极小化评价函数,以距离两个理想点最近的点作为最优解。

2.3 问题三的分析

问题三属于多目标规划模型,考虑绳索最大承载力,并要求满足摆动最小且效率最高的条件,要求解时间 t_i 、加速度 a 以及最大起吊货物重量,我们在问题一动力学方程的基础上引入牛顿第二定律方程,并建立吊车驱动力与摆角的二阶导数之间的方程,再通过简单的受力分析,从而将绳索的最大承载力与最大起吊重量联系起来。同样用问题二中的评价函数法,首先分别只考虑时间最短、只考虑摆角最大值最小、只考虑质量最大求出三个理想点,然后建立评价函数并极小化,最后以距离三个点最近的点作为最优解。

2.4 问题四的分析

问题四要求结合实际考虑更多因素：在实际问题中，绳索往往并不只限于二维平面而是在三维空间内发生摆动，并且由于重物质量的变化，或者小车运输过程中的加速、匀速、减速运动中，重物由于惯性作用产生很大的摆动；钢丝伸长收缩的变化，轨道的摩擦与磨损；大风等外界干扰都会增加三维桥式吊车的不稳定性。因此，在前三问的基础上我们从以下三个方面考虑：第一，将模型从二维平面问题扩展到三维空间问题；第二，考虑绳长的变化；第三，考虑摩擦阻力、空气阻力等影响建立模型。首先，建立空间位置坐标关系；然后进行动力学分析，用拉格朗日方程推出最终模型；最后，由于时间不足在理论上进行 simulink PID 控制仿真并创建 GUI 界面对龙门吊问题进行直观的演示。

2.5 总体思路图

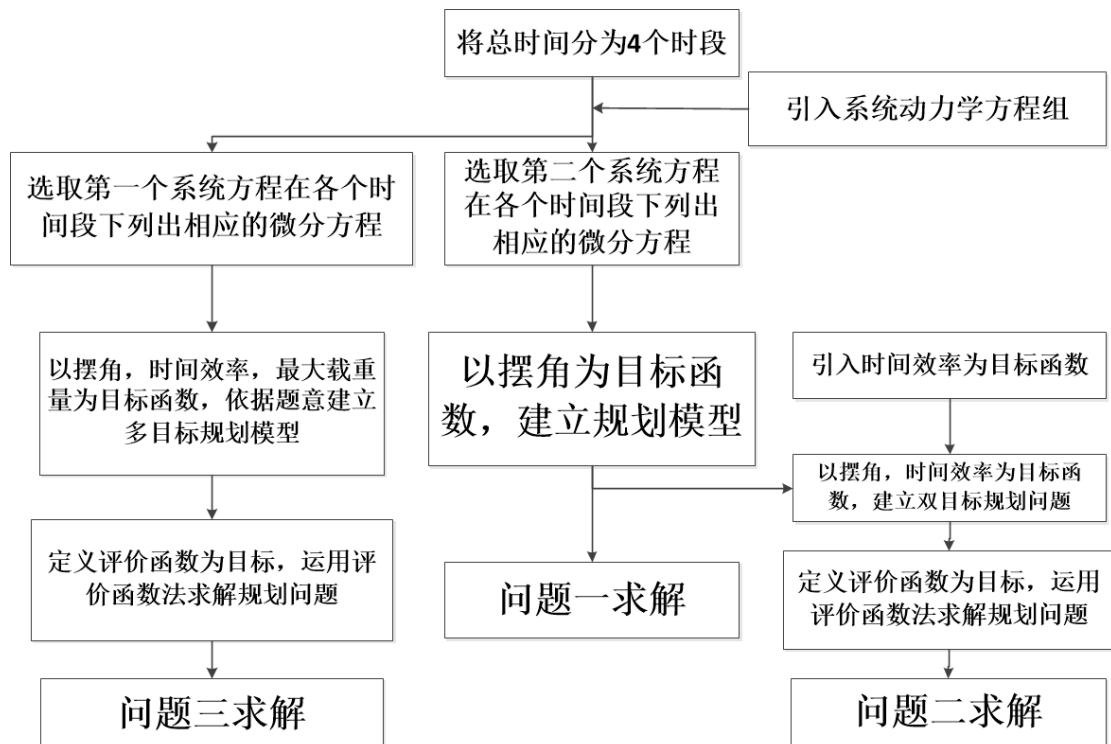


Figure 3 前三问总体思路图

三、模型假设

- 1、问题一、二、三假设吊索只在垂直于水平面的平面内运动；
- 2、吊索的质量相对货物的质量可以忽略不计；
- 3、问题一二三假设吊索的刚度足够大，其长度变化可忽略不计；
- 4、问题一、二、三中假设不计风力、空气阻力和传动机构之间的摩擦；

5、问题一、二、三假设摆角 θ 非常小；

四、符号说明

符号	含义
θ_{\max}	货物运动过程中的最大摆角
$\theta_i(t)$	第 i 时间段的摆角随时间变化函数, $i=1, 2, 3, 4$
Q_{θ_x}	重力在 x 轴上的投影
Q_{θ_y}	重力在 y 轴上的投影
F	水平驱动力
t_i	第时间段 i
μ	重物摆动的阻尼系数
D	小车与水平轨道的摩擦阻尼系数为

五、模型的建立和求解

5.1 问题一模型的建立与求解

5.1.1 模型的简化

将系统简化为如图所示的模型。重物通过绳索与吊车相连，吊车在行走电机的水平拉力 F (N) 的作用下在水平轨道运动，小车质量为 m (kg)，重物的质量为 M (kg)，绳索的长度为 l (m)；绳索的弹性，质量，运动的阻尼系数可忽略；小车与水平轨道的摩擦阻尼系数为 D (kg/s)；重物摆动时的阻尼系数为 μ (kg.m²/s),其它扰动可忽略。

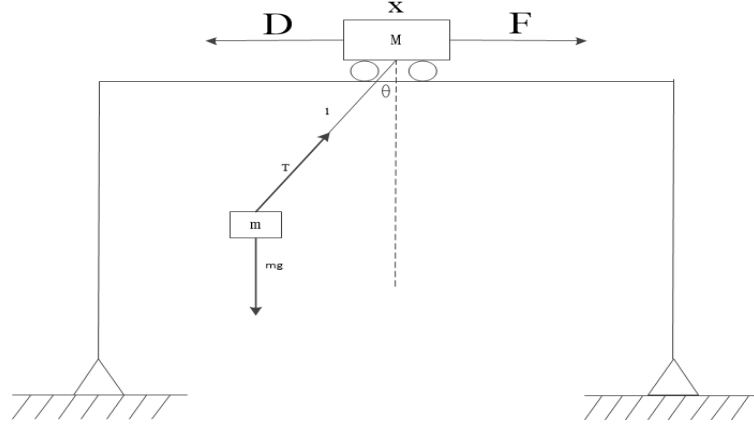


Figure 4 二维简图

5.1.2 建模的准备：分析动力学

系统的动能：

$$E = \frac{1}{2}mv_m^2 + \frac{1}{2}Mv_M^2 = \frac{1}{2}m(\dot{x}_m^2 + \dot{y}_m^2) + \frac{1}{2}M(\dot{x}_M^2 + \dot{y}_M^2)$$

此系统的拉格朗日方程组为

$$\begin{cases} \frac{d}{dt} \left(\frac{\partial E}{\partial \dot{x}_2} \right) - \frac{\partial E}{\partial x_2} = F_2 - mg \cos x_3 \\ \frac{d}{dt} \left(\frac{\partial E}{\partial \dot{x}_3} \right) - \frac{\partial E}{\partial x_3} = -mgx_2 \sin x_3 - \mu \dot{x}_3 \\ \frac{d}{dt} \left(\frac{\partial E}{\partial \dot{x}_1} \right) - \frac{\partial E}{\partial x_1} = F - D \dot{x}_1 \end{cases}$$

忽略阻尼系数，摩擦力，由此得到的吊车模型如下：

$$\begin{cases} (M + m)\ddot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = F \\ ml^2\ddot{\theta} + ml\ddot{x} \cos \theta + mgl \sin \theta = 0 \end{cases}$$

考虑到实际吊车运动过程中摆动角较小（不超过 10° ），且平衡位置为 $\theta = 0$ ，可将上述模型在 $\theta = 0$ 处进行线性化。此时有如下近似结果： $\sin \theta \approx \theta$ ， $\cos \theta \approx 1$ ， $\theta \sin \theta \approx 0$ ，考虑到摆动的阻尼系数 μ 很小，可认为 $\mu = 0$ ，所以上式可简化成(线性化)：

$$(M + m)\ddot{x} + ml\ddot{\theta} = F \quad (1)$$

$$l^2\ddot{\theta} + \ddot{x} + g\theta = 0 \quad (2)$$

5.1.3 微分方程组建立

问题一忽略掉了最大牵引力所以引用(2)的求解方法我们首先利用式子(2)建立相应的二阶常系数线性非齐次微分方程组,求解出在每一个阶段的摆动幅度与时间的符号解:

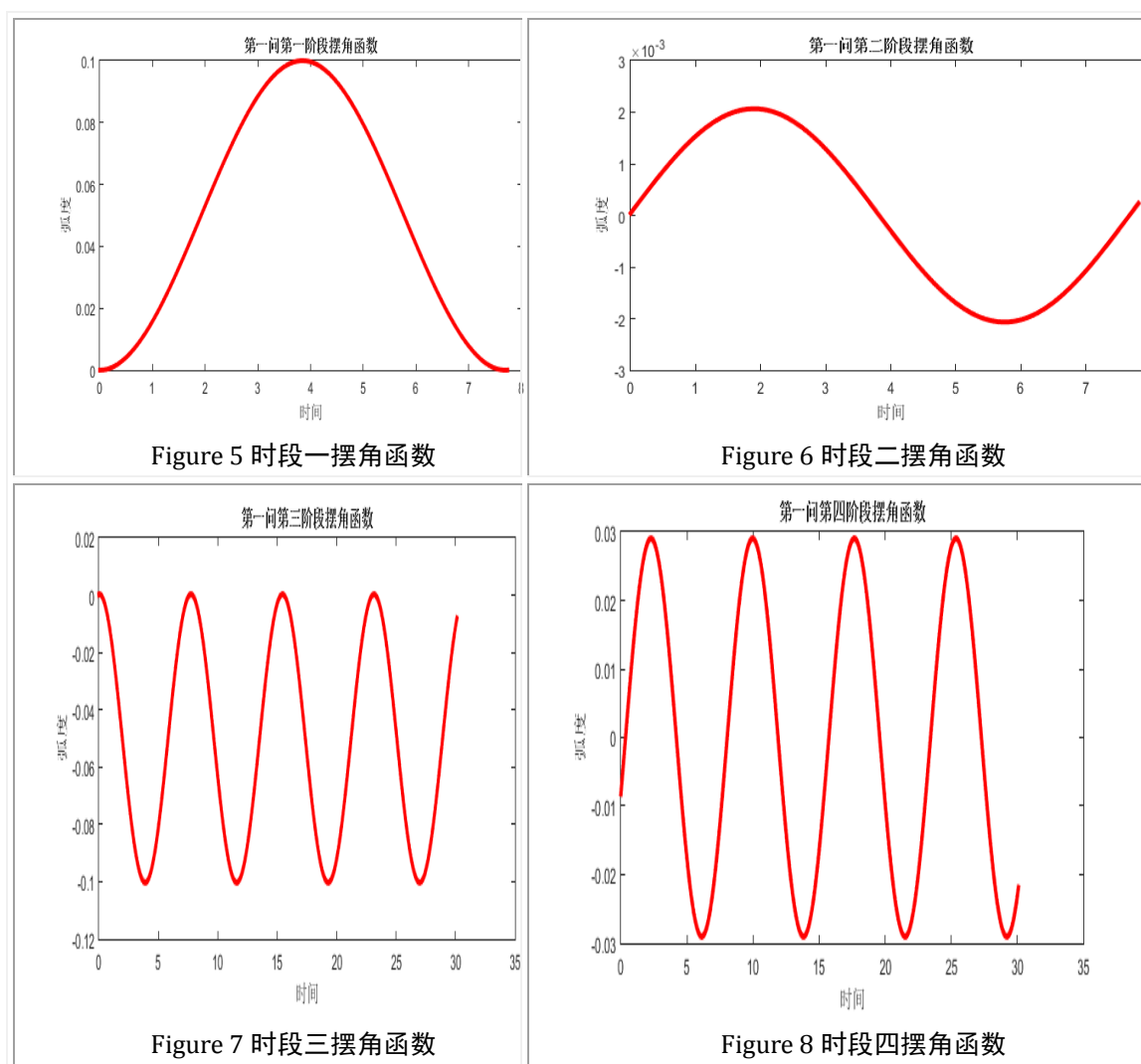
$$g\theta_i + l \frac{d^2\theta_i}{dt^2} + a_i = 0 (i = 1, 2, 3, 4)$$

$$\begin{cases} \theta_1(0) = 0, \theta_1'(0) = 0 \\ \theta_i(t_i) = \theta_{i+1}(t_i) (i = 1, 2, 3) \\ \theta_4(t_4) = 0, \theta_4'(t_4) = \frac{v_x}{l} \end{cases}$$

求出微分方程 $\theta_4(t)$ 的解析解(其他三个时段见附录):

$$\begin{aligned} \theta_4(t) = & \cos((6^{\frac{1}{2}} \times t) / 3) \times (\cos(6^{\frac{1}{2}} \times (t_2 / 3 + t_3 / 3)) / 20 + \\ & \cos((6^{\frac{1}{2}} \times t_3) / 3) / 20 - \cos(6^{\frac{1}{2}} \times (t_1 / 3 + t_2 / 3 + t_3 / 3)) \\ & / 20 - 1 / 20) - \sin((6^{\frac{1}{2}} \times t) / 3) \times (\sin((6^{\frac{1}{2}} \times t_2) / 3 + (6^{\frac{1}{2}} \times t_3) \\ & / 3) / 20 - \sin((6^{\frac{1}{2}} \times t_1) / 3 + 6^{\frac{1}{2}} \times t_2 / 3 + (6^{(1/2)} \times t_3) / 3) / 20 \\ & + \sin((6^{\frac{1}{2}} \times t_3) / 3) / 20) \end{aligned}$$

作出四个阶段摆角随时间的变化图像及其对应的函数表达式如下图,其中纵轴 y 表示摆角,横轴 x 表示时间。



5.1.4 优化模型的建立

第一问的优化模型如下：选取第四阶段摆角函数的极大值为目标函数，使之最小，约束条件依次是：首先，货物到达终点的速度不能超过 $v_{\max} = 0.5m/s$ ；其次，由运动学定律有 $v_x = v_1 - at_3$ ，其中 $v_1 = at_1$ ，根据速度图像对时间积分等于总位移即速度时间图像与坐标轴围成的面积等于总位移 $D_1 + D_2$ 可以得出另外两个约束条件。

$$\begin{aligned} & \min_{t_1, t_2, t_3, t_4} \max_t \theta_4(t) \\ \text{s.t.} & \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \end{cases} \end{aligned}$$

5.1.4 模型的求解：遗传算法

1、首先用 Lingo 非线性规划求解时间及摆角全局初步最优解结果如下：

Table 1 问题一 Lingo 初步求解结果

t_1	t_2	t_3	t_4	t_{\max}	$\min \max \theta_4$ (弧度)	$\min \max \theta_4$ (角度)
4.4067	22.8646	3.8110	50.3654	81.3	0.197369	10.88

2、在 matlab 中用遗传算法求解：

根据建立的模型，发现约束条件很多，直接求解很困难。所以我们采用改进后的遗传算法来求解出目标规划的最优值，遗传算法的步骤如下：

(1) 个体编码

遗传算法的运算对象是表示个体的符号串，所以把变量编码为一种符号串。

(2) 初始群体的产生

遗传算法是对群体进行的进化操作，需要给其准备一些表示起始搜索点的初始群体数据。

(3) 适应度计算

遗传算法中以个体适应度的大小来评定各个个体的优劣程度，从而决定其遗传机会的大小。

(4) 选择运算

选择运算(或称为复制运算)把当前群体中适应度较高的个体按某种规则或模型遗传到下一代群体中。一般要求适应度较高的个体将有更多的机会遗传到下一代群体中。

(5) 交叉运算

交叉运算是遗传算法中产生新个体的主要操作过程，它以某一概率相互交换某两个个体之间的部分染色体。

比如采用单点交叉的方法，其具体操作过程是：

- 先对群体进行随机配对；
- 其次随机设置交叉点位置；
- 最后再相互交换配对染色体之间的部分基因。

(6) 变异运算

变异运算是针对个体的某一个或某一些基因座上的基因值按某一较小的概率

进行改变，它也是产生新个体的一种操作方法。本例中，我们采用基本位变异的方法来进行变异运算，其具体操作过程是：

- 首先确定出各个个体的基因变异位置，下表所示为随机产生的变异点位置，其中的数字表示变异点设置在该基因座处；
- 然后依照某一概率将变异点的原有基因值取反，对群体 $P(t)$ 进行一轮选择、交叉、变异运算之后可得到新一代的群体 $p(t+1)$ 。

遗传算法实施的具体流程如图 3：

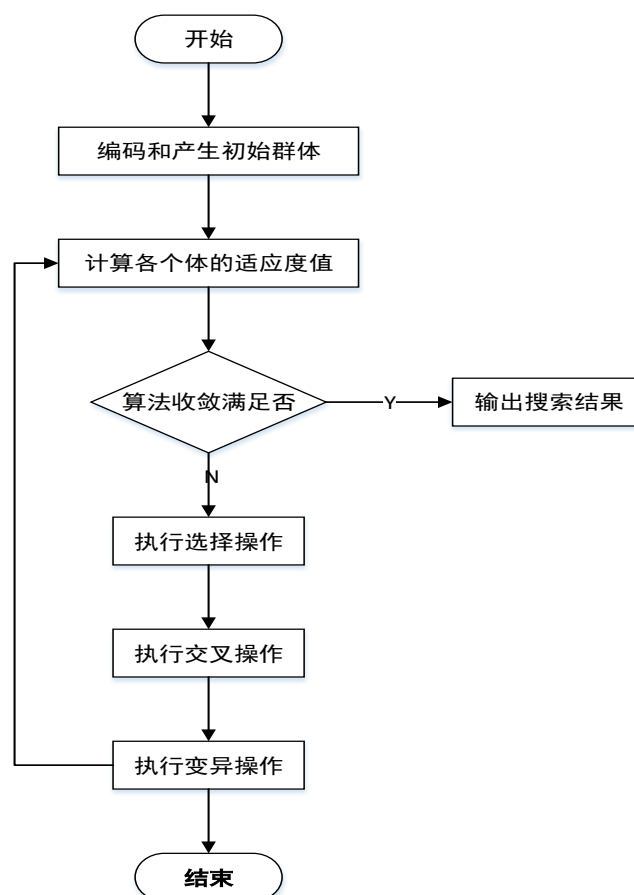


Figure 9 遗传算法（SGA）流程图

1. 编码和产生初始群体

根据问题选择相应的编码方法，并随机产生一个确定长度的 N 个染色体组成的初、始群体：

$$pop_i(t), t=1, i=1, 2, 3, \dots, N \quad (5)$$

2. 计算适应度值

对群体 $pop(t)$ 中的每一个染色体 $pop_i(t)$ 计算它的适应度：

$$f_i = fitness(pop_i(t)) \quad (6)$$

3. 判断算法收敛准则是否满足。若满足输出搜索结果，否则继续执行以下步骤；
 4. 选择操作
 根据各个个体的适应度值计算选择概率：

$$P_i = \frac{f_i}{\sum_{i=1}^N f_i}, \quad i=1,2,3, \dots, N \quad (7)$$

并以此式中的概率分布从当前一代群体 $pop_i(t)$ 中随机选择一些染色体遗传到下一代群体中构成一个新种群

$$newpop(t+1) = \{ pop_i(t) | j=1,2, \dots, N \} \quad (8)$$

5. 交叉操作

以概率 P_c 交配，得到一个有 N 个染色体组成的群体 $crosspop(t+1)$ ；

6. 变异操作

用某一较小的概率 P_i 使染色体的基因发生变异，形成新的群 $mutpop(t+1)$ ；
 该新的群体即为完成一次遗传操作后的子代记为 $pop(t) = mutpop(t+1)$ ，同时它又作为下一次遗传操作的父代，返回 2。

算法终止条件中，最简单的终止条件是规定最大遗传（迭代）代数，当其运行到最大的进化代数后就停止运行，并将当前群体中的最佳个体作为所求问题的最优解输出；另一种判断的方法，利用某种判定准则判定出群体已经进化成熟，且不再有进化趋势时终止算法运行，如群体中所有个体适应度的方差或连续几代个体平均适应度的差异小于某一极小的阈值或最优值满足所需的精度等来终止算法运行。

5.1.5 结果分析

Table 2 问题一最终结果

t_1	t_2	t_3	t_4	$t_{\text{总}}$	min max θ_4 (弧度)	min max θ_4 (角度)
7.7459	7.8100	6.7496	30.1120	52.39	0.0154	0.882

比较后发现用遗传算法得出的最大摆角最小弧度为 0.0154rad 结果比用 Lingo 求解的结果 0.1973 更小，更符合题目要求，且在 0 到 1 度，经查阅文献认为符合实际。

5.2 问题二模型的建立与求解

5.2.1 模型的建立：双目标规划

问题二要求总效率最高的同时摆动最小，以时间之和最小及第四阶段摆角函数的极大值为目标函数，约束条件同第一问：

$$\begin{aligned} \min \quad & \sum_{i=1}^4 t_i \\ \min \quad & \max_t \theta_4(t) \\ \text{s.t.} \quad & \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \end{cases} \end{aligned}$$

5.2.2 模型的求解：遗传算法

对于问题二的双目标规划问题，我们采用评价函数法来求解这个双目标规划，即用一个评价函数来集中反映不同目标的重要性等因素，并极小化此评价函数，据理想点最近的点作为最优解。

我们首先在两个单目标相同约束条件的情况下分别得到这两个单目标的最优值设为 f_1^*, f_2^* ，由于第一问已经给出了以最大摆角最小为目标的最优值，只需要再计算一次时间最短的最优值即可。

$$\begin{aligned} \min \quad & \sum_{i=1}^4 t_i \\ \text{s.t.} \quad & \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \end{cases} \end{aligned}$$

而在双目标规划的情况下，设所求得的两个目标函数的值分别为 f_1, f_2 由评

价函数法，我们希望能够找到一对 f_1, f_2 ，使之与 f_1^*, f_2^* 的距离达到最小，即 $V - \min_{x \in D} \{f_1(x), f_2(x)\} \Rightarrow f_j^* = \min_{x \in D} f_j(x) (j=1, 2)$ ，我们定义评价函数 $h(F(x)) = h(f_1, f_2) = \sum_{j=1}^2 (f_j(x) - f_j^*)^2$ ，并以此求解此目标规划问题，由此所建立的最终模型如下：

$$\begin{aligned} & \min_{x \in D} h(F(x)) \\ & \text{s.t.} \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \\ f_1 = \sum_{i=1}^4 t_i \\ f_2 = \max \theta_4(t) \end{cases} \end{aligned}$$

5.2.3 结果分析

用遗传算法得出结果：

(1) 仅考虑总时间最短：

Table 3 问题二时间理想结果

t_1	t_2	t_3	t_4	$t_{\text{总}}$	$\min \max \theta_4(\text{弧度})$	$\min \max \theta_4(\text{角度})$
8.979	4.440	37.978	29.975	51.373	0.0487	2.790

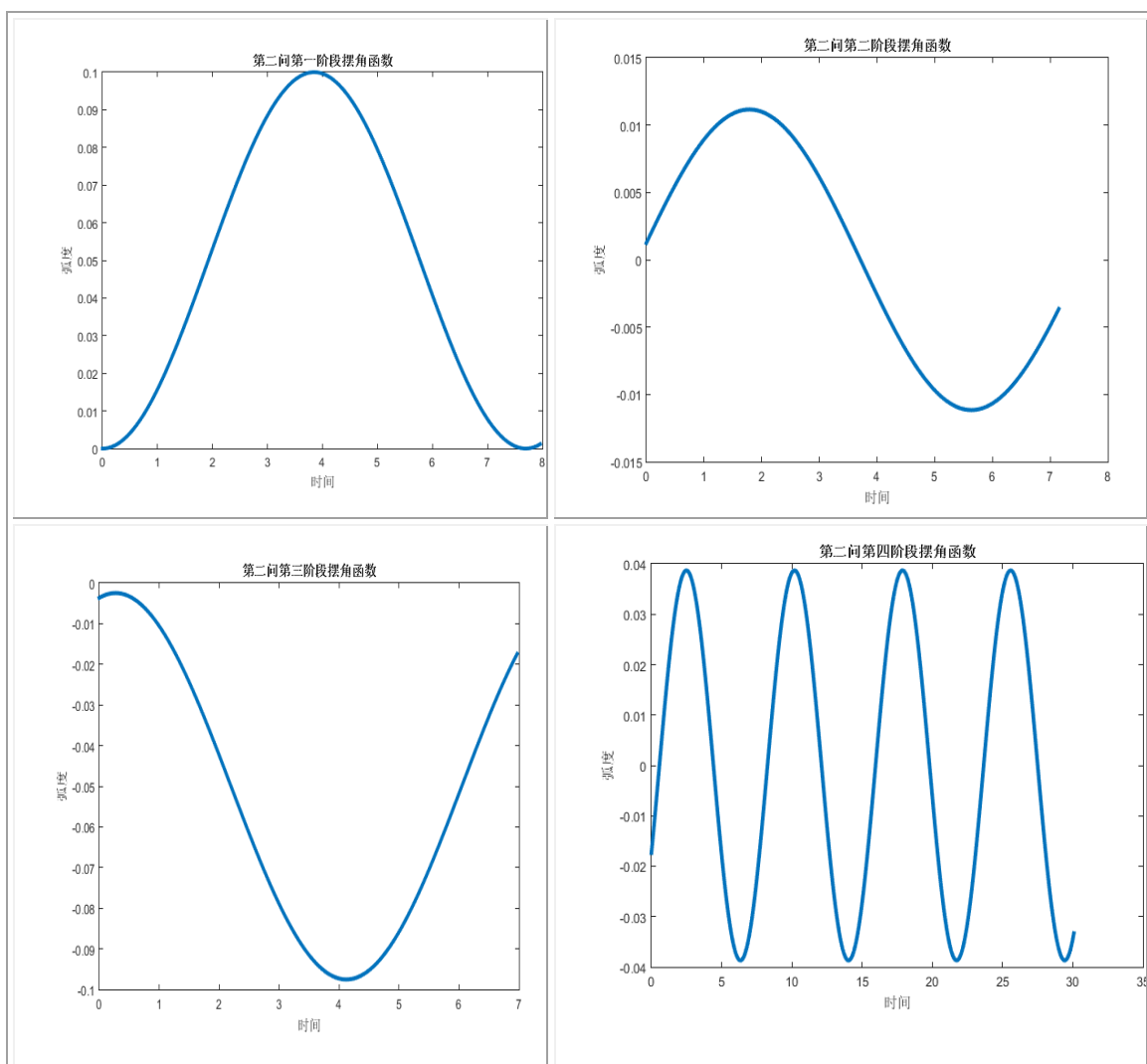
(2) 综合两个理想点得到的最终结果：

Table 4 问题二综合最终结果

t_1	t_2	t_3	t_4	$t_{\text{总}}$	$\min \max \theta_4(\text{弧度})$	$\min \max \theta_4(\text{角度})$
7.962	7.151	6.972	30.08	52	0.0174	0.996

全程所用总时间 51.373 秒最大摆角最小值为 0.0174rad 即 0.996 度，相对于第一问总时间减少约 1.94%，角度增加约 12.9%，我们认为结果是合理的且符合实际。

问题二四段时间摆角函数如下图：



5.3 问题三模型的建立和求解

5.3.1 模型的建立

第三问，利用式子（1）进行微分方程组的建立与求出其解析解

$$Ml \frac{d^2 \theta_i}{dt^2} = \frac{(M+m)g}{\theta} - F (i=1,2,3,4)$$

其中 $F = T\theta + ma_i (i=1,2,3,4)$

$$\begin{cases} \theta_1(0) = 0, \theta_1'(0) = 0 \\ \theta_i(t_i) = \theta_{i+1}(t_i) (i=1,2,3) \\ \theta_4(t_4) = 0, \theta_4'(t_4) = \frac{v_x}{l} \end{cases}$$

在第二问的双目标规划模型基础上，增加最大承载力这一约束条件以及货物质量最大这一目标，第三问的多目标规划模型如下：

$$\begin{aligned}
 & \max \quad m \\
 & \min \quad \sum_{i=1}^4 t_i \\
 & \min_{t_1, t_2, t_3, t_4} \quad \max_t \theta_4(t) \\
 & \text{s.t.} \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \\ T \leq T_{\max} \end{cases}
 \end{aligned}$$

5.3.2 模型的求解

我们同样采用评价函数法来求解这个双目标规划，即用一个评价函数来集中反映不同目标的重要性等因素，并极小化此评价函数，据理想点最近的点作为最优解。我们首先在两个单目标相同约束条件的情况下分别求出在 3 个单目标下的最优目标值作为理想点，同样设为： f_1^*, f_2^*, f_3^*

$$\begin{aligned}
 & \max \quad m \\
 & \text{s.t.} \begin{cases} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \\ T \leq T_{\max} \end{cases}
 \end{aligned}$$

在三目标规划的情况下，设所求得的两个目标函数的值分别为 f_1, f_2, f_3 由理想解法，我们希望能够找到一对 f_1, f_2, f_3 ，使之与 f_1^*, f_2^*, f_3^* 的距离达到最小，即

$V - \min_{X \in D} \{f_1(x), f_2(x), f_3(x)\} \Rightarrow f_j^* = \min_{X \in D} f_j(x) (j=1, 2, 3)$ ，我们定义评价函数

$$h(F(x)) = h(f_1, f_2, f_3) = \sqrt{\sum_{j=1}^3 (f_j(x) - f_j^*)^2}$$
，并以此求解此目标规划问题，由此所建

立的最终模型如下：

由此所建立的最终模型如下：

$$\begin{aligned}
& \min \left(f_1 - f_1^* \right)^2 + \left(f_2 - f_2^* \right)^2 + \left(f_3 - f_3^* \right)^2 \\
& \text{s.t.} \left\{ \begin{array}{l} v_x \leq v_{\max} \\ v_x = a(t_1 - t_3 + t_2) \\ D_1 = \frac{1}{2}at_1^2 + at_1(t_2 - t_1) + \frac{1}{2}(at_1 + v_x)(t_3 - t_2) \\ D_2 = v_x(t_4 - t_3) \\ T \leq T_{\max} \\ f_1 = \sum_{i=1}^4 t_i \\ f_2 = \theta_4(t) \\ f_3 = m \end{array} \right.
\end{aligned}$$

5.3.1 结果分析

问题三用遗传算法得出综合最优结果为：总用时 94.1 秒，加速度 0.697 m/s^2 。

在 matlab 中用遗传算法仅考虑重物时得出的第一个理想点结果如下：

Table 5 问题三重量理想结果

t_1	t_2	t_3	t_4	$m(\text{kg})$	$T(\text{kg})$	$a(\text{m/s}^2)$
3.823	27.67	2.826	30.068	23908.21	19757.16	0.54

仅考虑时间最短时第二个理想点结果如下：

Table 6 问题三时间理想结果

t_1	t_2	t_3	t_4	$m(\text{kg})$	$T(\text{kg})$	$a(\text{m/s}^2)$
3.08	36.03	2.08	30.12	4182.94	947.95	0.626

仅考虑摆角最大值最小时得出第三个理想点结果如下：

Table 7 问题三摆角理想结果

t_1	t_2	t_3	t_4	$m(\text{kg})$	$T(\text{kg})$	$a(\text{m/s}^2)$
0.240	499.78	0.030	142.86	4205.75	292.82	0.57

综合考虑到三个理想点距离最近的结果如下：

Table 8 问题三综合最终结果

t_1	t_2	t_3	t_4	$m(\text{kg})$	$T(\text{kg})$	$a(\text{m/s}^2)$
1.92	60.63	0.97	31.55	4000.01	2127.76	0.697

5.4 问题四模型建立求解

第四问，我们在前面 3 问的基础上，考虑到摩擦力即阻尼系数的问题，同时由于在实际生活中，龙门吊是处于一个三维的运动状态，其抽象图如下图所示，我们通过查阅相应的文献，运用三维状况下系统动力学方程进行建模。

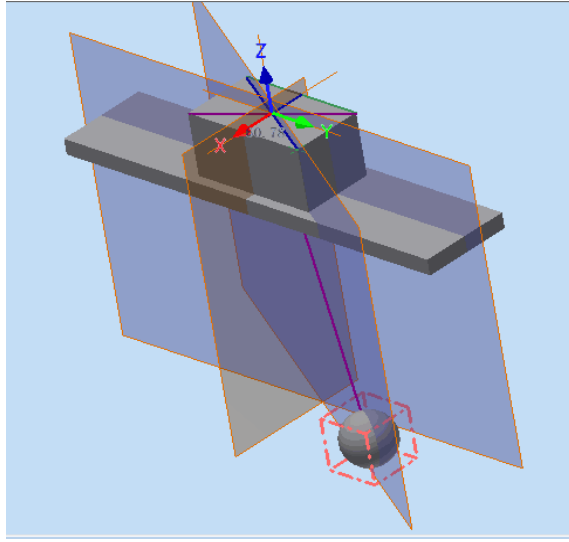


Figure 10 三维调运问题原图

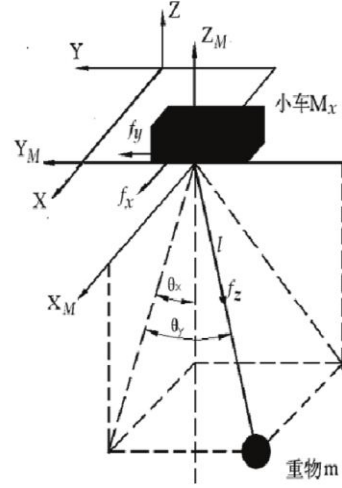


Figure 11 三维调运问题简化图

如图所示为桥式吊装系统坐标系图。小车和大车的质量分别为 x_M ， y_M ，重物质量为 m ，绳长为 l ，负载通过绳索与小车相连，小车在作用力 f_x 作沿向运动，在作用力 f_y 作用下沿 y 方向运动，在作用力 f_z 的作用下进行升降运动。负载在空间的摆动的角度为 θ ，包含 θ_x 和 θ_y 两个元素。

由图建立吊车的位置坐标如下：

$$\begin{cases} x_M = x \\ y_M = y \\ z_M = 0 \end{cases}$$

由图建立货物的位置坐标如下：

$$\begin{cases} x_m = x + l \cos \theta_y \sin \theta_x \\ y_m = y - l \sin \theta_y \\ z_m = -l \cos \theta_y \cos \theta_x \end{cases}$$

由此可以建立吊车与货物的速度如下：

$$\begin{cases} \dot{x}_M = \dot{x} \\ \dot{y}_M = \dot{y} \\ \dot{z}_M = 0 \\ \dot{x}_m = \dot{x} + \dot{l} \cos \theta_y \sin \theta_x - l \sin \theta_y \sin \theta_x \dot{\theta}_y + l \cos \theta_y \cos \theta_x \dot{\theta}_x \\ \dot{y}_m = \dot{y} - \dot{l} \sin \theta_y - l \cos \theta_y \dot{\theta}_y \\ \dot{z}_m = -\dot{l} \cos \theta_y \cos \theta_x + l \sin \theta_y \cos \theta_x \dot{\theta}_x + l \cos \theta_y \sin \theta_x \dot{\theta}_x \end{cases}$$

所以可以得到系统的动能表达式为：

$$\begin{aligned} T &= \frac{1}{2} M V_M^2 + \frac{1}{2} m V_m^2 \\ &= \frac{1}{2} M (\dot{x}_M^2 + \dot{y}_M^2 + \dot{z}_M^2) + \frac{1}{2} m (\dot{x}_m^2 + \dot{y}_m^2 + \dot{z}_m^2) \end{aligned}$$

接着由拉格朗日方程可得：

$$\begin{cases} \frac{d}{dt} \left(\frac{\delta T}{\delta \dot{x}} \right) - \frac{\delta T}{\delta x} = f_x \\ \frac{d}{dt} \left(\frac{\delta T}{\delta \dot{\theta}_x} \right) - \frac{\delta T}{\delta \theta_x} = Q_{\theta_x} \\ \frac{d}{dt} \left(\frac{\delta T}{\delta \dot{y}} \right) - \frac{\delta T}{\delta y} = f_y \\ \frac{d}{dt} \left(\frac{\delta T}{\delta \dot{\theta}_y} \right) - \frac{\delta T}{\delta \theta_y} = Q_{\theta_y} \\ \frac{d}{dt} \left(\frac{\delta T}{\delta \dot{l}} \right) - \frac{\delta T}{\delta l} = f_z + mgl \cos \theta_x \cos \theta_y \end{cases}$$

其中， $Q_{\theta_x} = -mgl \sin \theta_x \cos \theta_y$ ， $Q_{\theta_y} = -mgl \cos \theta_x \cos \theta_y$ ，分别表示重力向 θ_x ， θ_y 这两个方向的投影，称为在 θ_x ， θ_y 两个方向上的广义力。

所以，最终建立的数学模型如下所示：

$$\begin{cases} (M_x + m)\ddot{x} + ml \cos \theta_x \cos \theta_y \ddot{\theta}_x - ml \sin \theta_x \sin \theta_y \ddot{\theta}_y + m \sin \theta_x \cos \theta_y \ddot{l} + 2m \cos \theta_x \cos \theta_y \dot{l} \dot{\theta}_x \\ - 2m \sin \theta_x \sin \theta_y \dot{l} \dot{\theta}_y - ml \sin \theta_x \cos \theta_y \dot{\theta}_x^2 - 2ml \cos \theta_x \sin \theta_y \dot{\theta}_x \dot{\theta}_y - ml \sin \theta_x \cos \theta_y \dot{\theta}_y^2 = f_x \\ m \cos^2 \theta_y \ddot{\theta}_x + \cos \theta_x \cos \theta_y \ddot{x} + g \sin \theta_x \cos \theta_y - 2l \sin \theta_y \cos \theta_y \dot{\theta}_x \dot{\theta}_y + 2 \cos^2 \theta_y \dot{l} \dot{\theta}_x = 0 \\ (M_y + m)\ddot{y} + ml \cos \theta_y \ddot{\theta}_y - m \sin \theta_y \ddot{l} + 2m \cos \theta_y \dot{l} \dot{\theta}_y - ml \sin \theta_y \dot{\theta}_y^2 = f_y \\ l \ddot{\theta}_y + \cos \theta_y \ddot{y} + g \sin \theta_y \cos \theta_x + 2l \dot{\theta}_y + l \cos \theta_y \sin \theta_y \dot{\theta}_x^2 - \sin \theta_y \ddot{x} = 0 \\ m \sin \theta_x \cos \theta_y \ddot{x} + m \sin \theta_y \ddot{y} + m \ddot{l} - ml \cos^2 \theta_y \dot{\theta}_x^2 - ml \dot{\theta}_y^2 - mg \cos \theta_x \cos \theta_y = f_z \end{cases}$$

为了方便进行仿真实验，我们对此模型进行仿真实验，得到的最终模型下：

$$\begin{cases} (M_x + m)\ddot{x} + ml \ddot{\theta}_x + D_x \dot{x} = f_x \\ l \ddot{\theta}_x + \ddot{x} + g \theta_x = 0 \\ (M_y + m)\ddot{y} + ml \ddot{\theta}_y + D_y \dot{y} = f_y \\ l \ddot{\theta}_y + \ddot{y} + g \theta_y = 0 \\ m \ddot{l} + D_l \dot{l} - mg = f_l \end{cases}$$

由于时间问题，我们仅做出了吊车货物系统在 x 轴方向上的运动变化模型，并且在理论上运用双闭环 PID 控制器设计原理进行仿真，并未得到最后的结果。我们在 Matlab 软件上做 Simulink 仿真时建立的仿真框图在附录中显示，下面是其原理图：

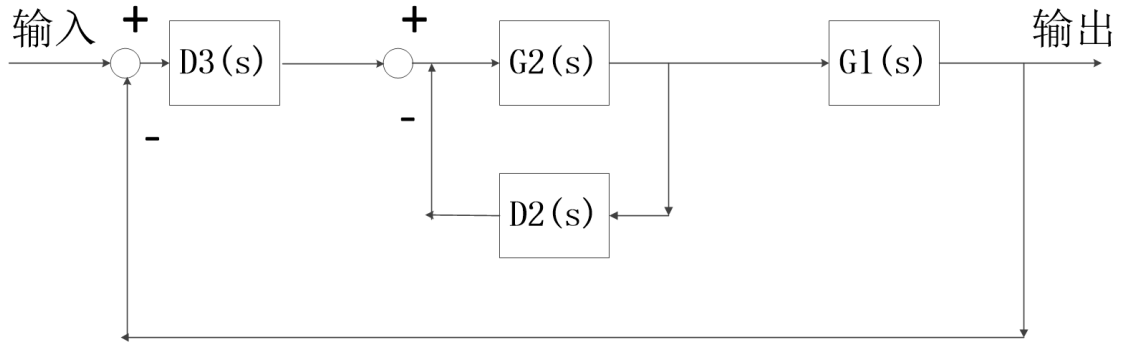


Figure 12 PID 控制流程图

其中传递函数模型为：

$$G_2(s) = \frac{\theta(s)}{F_x(s)} = K_s \frac{s}{Mls^3 + D_x ls^2 + (M + m)gs + D_x g}$$

$$G_1(s) = \frac{X(s)}{\theta(s)} = \frac{ls^2 + g}{s^2}$$

其中， $M = 50, m = 10, l = 1, K_s = 30, D_x = 0.1$

这是在仿真时输入的参数值

由于龙门吊车系统核心是在重物摆动幅度不大的情况下，使重物的位置可控。因此，依据反馈闭环控制原理，将摆角作为内环来进行控制，外环来用于控制位置，以实现双闭环控制。

内环采用反馈校正，校正装置的结构为控制器，其传递函数为 $D_2(s) = K_{p2} + K_{D2}s$ ，其中， $K_{p2}=88.7$ $K_{D2}=26.7$

外环采用单位反馈控制，其评价函数为： $D_3(s) = K_{p3} + K_{D3}s$ 其中， $K_{p3}=88.7$ $K_{D3}=26.7$

我们在 Matlab 软件上做 Simulink 仿真时建立的仿真框图在附录中显示

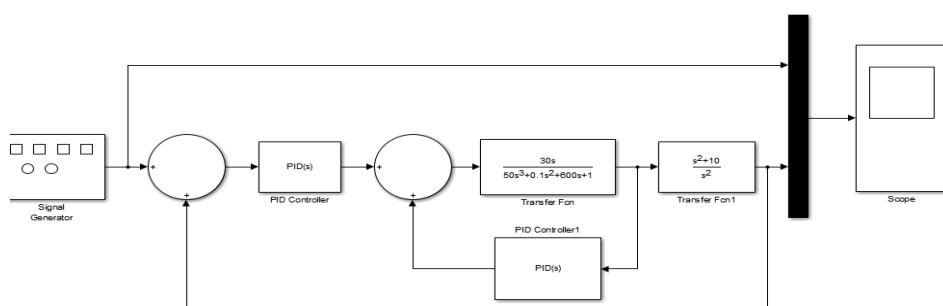


Figure 13 simulink 仿真框图

在 matlab 中创建 GUI 界面如下：输入参量为加速度 a 和绳长 l, 输出摆角函数图：

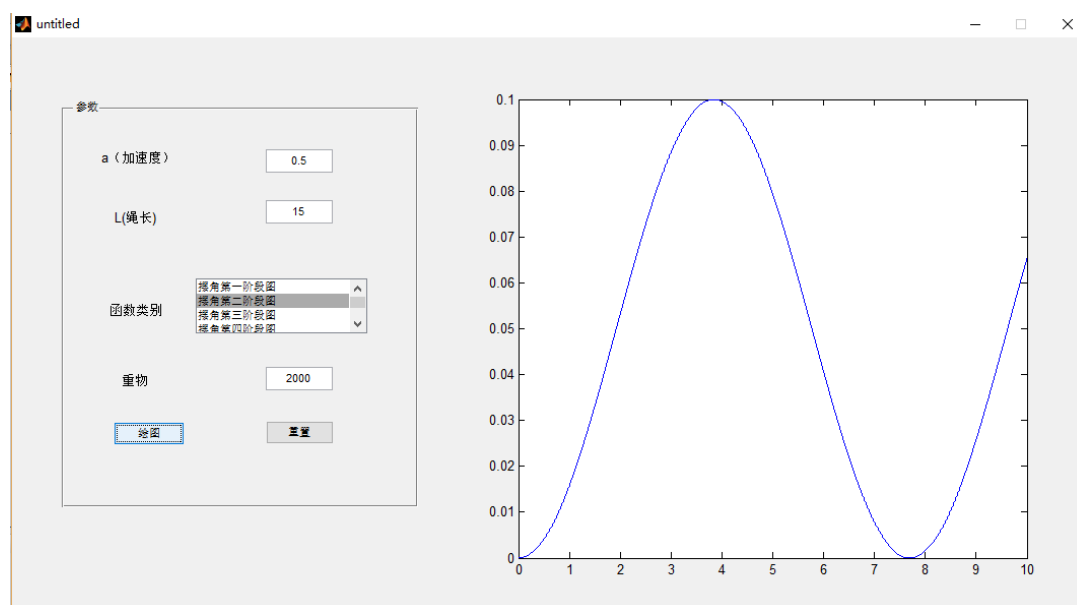


Figure 14 GUI 界面

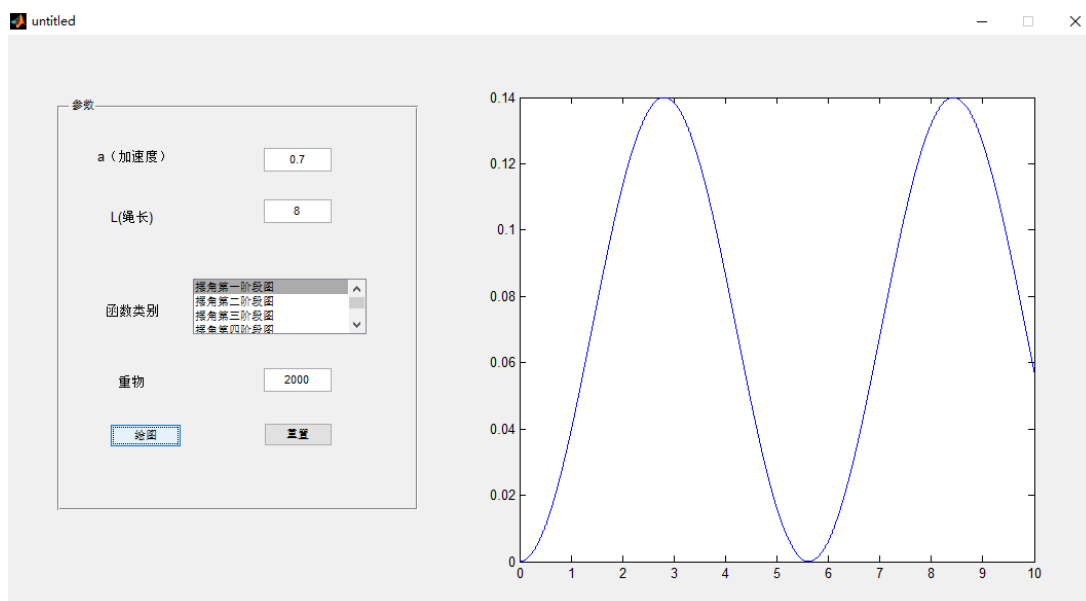


Figure 15 GUI 界面

六、 模型的评价和推广

6.1.模型优点:

- 1.问题一、二以系统动力学方程为基础，以各个阶段的摆角函数作为目标函数建立相应的非线性规划问题，能够简化模型，同时具有较强的理论意义。
- 2.问题二以各个时间段的时间总和作为效率的特征反映，既结合了实际，同时对于问题的简化求解有很大的帮助。
- 3.问题二、三对于所建立的模型统一进行评价函数法的求解，所得到的结果准确可靠。
- 4.问题四在三维空间中考虑龙门吊问题，并且考虑了绳长，牵引力等多种因素，更加贴近实际。并且在仿真过程中和自动化控制相结合，具有很强的实际意义。
5. 优点在于利用“分析动力学”

6.2.模型缺点:

- 1.问题三并未能求出目标函数解析解，运用遗传算法求解时存在一定的误差。
- 2.问题四所得到的模型较为复杂，给模型的求解带来很大的困难。

七、参考文献

- [1] Fang Y, Wang P, Sun N, Zhang Y. Dynamic analysis and nonlinear control of an offshore boom crane. *IEEE Transactions on Industrial Electronics*, 2014, 61(1):414-427
- [2] Y. Fang, X. Liu, and X. Zhang, "Adaptive active visual servoing of nonholonomic mobile robots," *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 486 - 497, Jan.

-
- 2012.
- [3] M. T. Ravichandran and A. D. Mahindrakar, “Robust stabilization of a class of underactuated mechanical systems using time scaling and Lyapunov redesign,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4299 – 4313, Sep. 2011.
- [4] A. Widyotriatmo and K.-S. Hong, “Navigation function-based control of multiple wheeled vehicles,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 1896 – 1906, May 2011.
- [5] 孙宁. 欠驱动吊车轨迹规划与非线性控制策略设计、分析及应用[D].南开大学,2014.
- [6] 刘剑锋. 绳系系统的非线性最优控制及其实验[D].南京航空航天大学,2008.
- [7] 赵伟. 基于三维空间吊车系统的神经模糊滑模控制[D].大连理工大学,2011.
- [8] 司守奎, 孙玺菁.数学建模算法与应用.北京: 国防工业出版社, 2013 年, 345-350
- [9] 马群. 基于粒子群模糊滑模控制的三维桥式吊车系统[D].大连理工大学,2014.

附录

第一问微分方程四个阶段摆角时间函数解析解结果：

$\theta_1(t)$	$y_1 = 1/20 - \cos((6^{1/2} * x_1)/3)/20$
$\theta_2(t)$	$y_2 = (\sin((6^{1/2} * x_2)/3) * \sin((6^{1/2} * t_1)/3))/20 - \cos((6^{1/2} * x_2)/3) * (\cos((6^{1/2} * t_1)/3)/20 - 1/20)$
$\theta_3(t)$	$y_3 = \sin((6^{1/2} * x_3)/3) * (\sin((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3)/20 - \sin((6^{1/2} * t_2)/3)/20) + \cos((6^{1/2} * x_3)/3) * (\cos((6^{1/2} * t_2)/3)/20 - \cos((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3)/20 + 1/20) - 1/20$
$\theta_4(t)$	$y_4 = \cos((6^{1/2} * x_4)/3) * (\cos(6^{1/2} * (t_2/3 + t_3/3))/20 + \cos((6^{1/2} * t_3)/3)/20 - \cos(6^{1/2} * (t_1/3 + t_2/3 + t_3/3))/20 - 1/20) - \sin((6^{1/2} * x_4)/3) * (\sin((6^{1/2} * t_2)/3 + (6^{1/2} * t_3)/3)/20 - \sin((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3 + (6^{1/2} * t_3)/3)/20 + \sin((6^{1/2} * t_3)/3)/20)$

第一问微分方程四个阶段摆角时间函数解析解结果：

$\theta_1(t)$	$y_1 = 1/20 - \cos((6^{1/2} * x_1)/3)/20$
$\theta_2(t)$	$y_2 = (\sin((6^{1/2} * x_2)/3) * \sin((6^{1/2} * t_1)/3))/20 - \cos((6^{1/2} * x_2)/3) * (\cos((6^{1/2} * t_1)/3)/20 - 1/20)$
$\theta_3(t)$	$y_3 = \sin((6^{1/2} * x_3)/3) * (\sin((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3)/20 - \sin((6^{1/2} * t_2)/3)/20) + \cos((6^{1/2} * x_3)/3) * (\cos((6^{1/2} * t_2)/3)/20 - \cos((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3)/20 + 1/20) - 1/20$
$\theta_4(t)$	$y_4 = \cos((6^{1/2} * x_4)/3) * (\cos(6^{1/2} * (t_2/3 + t_3/3))/20 + \cos((6^{1/2} * t_3)/3)/20 - \cos(6^{1/2} * (t_1/3 + t_2/3 + t_3/3))/20 - 1/20) - \sin((6^{1/2} * x_4)/3) * (\sin((6^{1/2} * t_2)/3 + (6^{1/2} * t_3)/3)/20 - \sin((6^{1/2} * t_1)/3 + (6^{1/2} * t_2)/3 + (6^{1/2} * t_3)/3)/20 + \sin((6^{1/2} * t_3)/3)/20)$

lingo 第二问主要代码

model:

!非线性规划有约束;

sets:

T/1..4/:to;

Tt/1..4/:tts;

Ts/1..4/:tss;

endsets

data:

tts=10.97725 0 9.977249 30.00000;

tss=4.406726 22.86463 3.811080 50.316549;

enddata

min=(@sum(T(i):to(i)-tts(i)))^2+(@sum(T(i):to(i)-tss(i)))^2;

0.25*to(1)^2+0.5*to(1)*to(2)+0.5*to(3)*(0.5*to(1)+0.5*(to(1)-to(3)))=60;

0.5*(to(1)-to(3))*to(4)=15;

0.5*(to(1)-to(3))<=0.5;

to(1)>0;

to(2)>0;

to(3)>0;

to(4)>0;

to(1)>to(3);

end

Matlab第三题主要代码

```
function [y1, y2, y3, y4] =  
weifen(t1,t2,t3,t4,M,m,g,T,a,l)
```

%微分方程求数值解

yt=0;%初始值

dyt=0;%初始一阶导数值

y1=shuzhijie(rand()*t1,M,m,g,T,a,l,t1,yt,dyt);%在0到t1
内求数值解

diff1_y1=shuzhijie(t1,M,m,g,T,a,l,t1,yt,dyt);%在t1的函
数值

```

        diff2_y1=shuzhijie(t1-0.001,M,m,g,T,a,l,t1,yt,dyt);%用
ti附近的差分值代替导数值

        %同上

        y2=shuzhijie(rand()*t2,M,m,g,T,0,l,t2,diff1_y1,(diff1_
y1-diff2_y1)/0.001);
        diff1_y2=shuzhijie(t2,M,m,g,T,0,l,t2,diff1_y1,(diff1_y
1-diff2_y1)/0.001);
        diff2_y2=shuzhijie(t2-0.001,M,m,g,T,0,l,t2,diff1_y1,(d
iff1_y1-diff2_y1)/0.001);

        %同上

        y3=shuzhijie(rand()*t3,M,m,g,T,-a,l,t3,diff1_y2,(diff1
_y2-diff2_y2)/0.001);
        diff1_y3=shuzhijie(t3,M,m,g,T,-a,l,t3,diff1_y2,(diff1_
y2-diff2_y2)/0.001);
        diff2_y3=shuzhijie(t3-0.001,M,m,g,T,-a,l,t3,diff1_y2,(
diff1_y2-diff2_y2)/0.001);

        %同上

        y4=shuzhijie(rand()*t4,M,m,g,T,0,l,t4,diff1_y3,(diff1_
y3-diff2_y3)/0.001);
        %diff1_y4=shuzhijie(t4,M,m,g,T,-a,l,t4,diff1_y3,(diff1
_y3-diff2_y3)/0.001);
        %diff2_y4=shuzhijie(t4-0.001,M,m,g,T,-a,l,t4,diff1_y3,
(diff1_y3-diff2_y3)/0.001);
        end
        function y = my_fun(pop)

        %遗传算法适应度函数

        timer=0;
        for i=1:4

            timer=timer+pop(i);%时间总和

        end

        [y1 ,y2 ,y3 ,y4] =
weifen(pop(1),pop(2),pop(3),pop(4),400000,pop(5),10,pop(6
),pop(7),15);
        y5=[y1 y2 y3 y4];

        y6=max(y5);%求各阶段角度值的最大值

```

```

m=-pop(5);%重物质量

y6=max(y5);
m=-pop(5);
timer=(timer-71.3240)^2;
m=-(23908.212318793056+m)^2;
y6=(y6-0)^2;
y=timer+y6+m;

%适应度值

end

function dy= f(t,y,~,m,g,T,a,l)

%微分方程

%sys=[M m g T a l];
sys(1)=400;

sys(2)=m;%重物质量

sys(3)=g;%重力加速度

sys(4)=T;%T拉力

sys(5)=a;%吊车加速度

sys(6)=1;%绳zhang1

dy=[y(2);((sys(1)+sys(2))*sys(3))/y(1)-(sys(4)*y(1))/(
sys(1)*sys(6))+(sys(2)*sys(5))/(sys(1)*sys(6))];%
end

function y= shuzhijie(t0,M,m,g,T,a,l,t,y0,y1)

%此函数主要是ode45函数

[t,y5]=ode45(@(t,y)f([t0
t0+0.001],[y0,y1],M,m,g,T,a,l),[t0 t0+0.001],[y0,y1]);
%[t,y5]=ode45('f',[t0 t0+0.001],[y0,y1],M,m,g,T,a,l);

y=y5(1);%求t0点的函数值

end

```

```

function [c,ceq]= fun_noliner(t)

%约束条件

dist1=0.25*t(1)^2+0.5*t(1)*t(2)+0.5*(0.5*t(1)+0.5*(t(1)
)-t(3)))*t(3)-60;%D1距离默认60

dist2=0.5*(t(1)-t(3))*t(4)-15;%D2默认15

ceq=[dist1 dist2]%等式约束

c=[];

tu1.m
t1=7.74595147889141
t2=7.810082380662074
t3=6.749673680670896
t3=30.112082357185024
x1=0:0.01:t1
x2=0:0.01:t2
x3=0:0.01:t3
x4=0:0.01:t4
y1 =1/20 - cos((6^(1/2)*x1)/3)/20
y2 =(sin((6^(1/2)*x2)/3)*sin((6^(1/2)*t1)/3))/20 -
cos((6^(1/2)*x2)/3)*(cos((6^(1/2)*t1)/3)/20 - 1/20)
y3 =sin((6^(1/2)*x3)/3)*(sin((6^(1/2)*t1)/3 +
(6^(1/2)*t2)/3)/20 - sin((6^(1/2)*t2)/3)/20) +
cos((6^(1/2)*x3)/3)*(cos((6^(1/2)*t2)/3)/20 -
cos((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3)/20 + 1/20) - 1/20
y4 =cos((6^(1/2)*x4)/3)*(cos(6^(1/2)*(t2/3 + t3/3)))/20 +
cos((6^(1/2)*t3)/3)/20 - cos(6^(1/2)*(t1/3 + t2/3 +
t3/3))/20 - 1/20) -
sin((6^(1/2)*x4)/3)*(sin((6^(1/2)*t2)/3 +
(6^(1/2)*t3)/3)/20 - sin((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3
+ (6^(1/2)*t3)/3)/20 + sin((6^(1/2)*t3)/3)/20)
figure(1)
plot(x1,y1)

title('第一问第一阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y1 =1/20 - cos((6^(1/2)*x1)/3)/20')

```

```

figure(2)
plot(x2,y2)

title('第一问第二阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y2
= (sin((6^(1/2)*x2)/3)*sin((6^(1/2)*t1)/3))/20 -
cos((6^(1/2)*x2)/3)*(cos((6^(1/2)*t1)/3)/20 - 1/20)')
figure(3)
plot(x3,y3)

title('第一问第三阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y3 = sin((6^(1/2)*x3)/3)*(sin((6^(1/2)*t1)/3 +
(6^(1/2)*t2)/3)/20 - sin((6^(1/2)*t2)/3)/20) +
cos((6^(1/2)*x3)/3)*(cos((6^(1/2)*t2)/3)/20 -
cos((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3)/20 + 1/20) - 1/20')
figure(4)
plot(x4,y4)

title('第一问第四阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y4 = cos((6^(1/2)*x4)/3)*(cos(6^(1/2)*(t2/3 +
t3/3))/20 + cos((6^(1/2)*t3)/3)/20 - cos(6^(1/2)*(t1/3 + t2/3
+ t3/3))/20 - 1/20) -
sin((6^(1/2)*x4)/3)*(sin((6^(1/2)*t2)/3 +
(6^(1/2)*t3)/3)/20 - sin((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3 +
(6^(1/2)*t3)/3)/20 + sin((6^(1/2)*t3)/3)/20)')

tu2.m
figure(1)
plot(x1,y1)

title('第二问第一阶段摆角函数')

```

```

xlabel('时间')

ylabel('弧度')

gtext('y1 =1/20 - cos((6^(1/2)*x1)/3)/20')
figure(2)
plot(x2,y2)

title('第二问第二阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y2
=(sin((6^(1/2)*x2)/3)*sin((6^(1/2)*t1)/3))/20 -
cos((6^(1/2)*x2)/3)*(cos((6^(1/2)*t1)/3)/20 - 1/20)')
figure(3)
plot(x3,y3)

title('第二问第三阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y3 =sin((6^(1/2)*x3)/3)*(sin((6^(1/2)*t1)/3 +
(6^(1/2)*t2)/3)/20 - sin((6^(1/2)*t2)/3)/20) +
cos((6^(1/2)*x3)/3)*(cos((6^(1/2)*t2)/3)/20 -
cos((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3)/20 + 1/20) - 1/20')
figure(4)
plot(x4,y4)

title('第二问第四阶段摆角函数')

xlabel('时间')

ylabel('弧度')

gtext('y4 =cos((6^(1/2)*x4)/3)*(cos(6^(1/2)*(t2/3 +
t3/3))/20 + cos((6^(1/2)*t3)/3)/20 - cos(6^(1/2)*(t1/3 + t2/3
+ t3/3))/20 - 1/20) -
sin((6^(1/2)*x4)/3)*(sin((6^(1/2)*t2)/3 +
(6^(1/2)*t3)/3)/20 - sin((6^(1/2)*t1)/3 + (6^(1/2)*t2)/3 +
(6^(1/2)*t3)/3)/20 + sin((6^(1/2)*t3)/3)/20)')

```

创建GUI界面代码:

```
function varargout = untitled(varargin)
% UNTITLED MATLAB code for untitled.fig
%     UNTITLED, by itself, creates a new UNTITLED or raises the
existing
%     singleton*.
%
%     H = UNTITLED returns the handle to a new UNTITLED or the handle
to
%     the existing singleton*.
%
%     UNTITLED('CALLBACK',hObject,eventData,handles,...) calls the
local
%     function named CALLBACK in UNTITLED.M with the given input
arguments.
%
%     UNTITLED('Property','Value',...) creates a new UNTITLED or
raises the
%     existing singleton*. Starting from the left, property value
pairs are
%     applied to the GUI before untitled_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to untitled_OpeningFcn via
varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only
one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help untitled

% Last Modified by GUIDE v2.5 24-Jul-2016 17:18:30

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @untitled_OpeningFcn, ...
                  'gui_OutputFcn',  @untitled_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
```

```

        'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before untitled is made visible.
function untitled_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)
% varargin    command line arguments to untitled (see VARARGIN)


% Choose default command line output for untitled
handles.output = hObject;


% Update handles structure
guidata(hObject, handles);


% UIWAIT makes untitled wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = untitled_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)


% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)

```

```

a=str2num(get(handles.edit1,'String'));
L=str2num(get(handles.edit2,'String'));
m=str2num(get(handles.edit3,'String'));
HuiTu(a,L,m);
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of edit1
as a double

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)

```

```

% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%         str2double(get(hObject,'String')) returns contents of edit2
as a double

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in listbox1.
function listbox1_Callback(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns listbox1
contents as cell array
%         contents{get(hObject,'Value')} returns selected item from
listbox1

% --- Executes during object creation, after setting all properties.
function listbox1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

% Hint: listbox controls usually have a white background on Windows.

```

```

%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%       str2double(get(hObject,'String')) returns contents of edit3
as a double

% --- Executes during object creation, after setting all properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes during object creation, after setting all properties.
function axes2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: place code in OpeningFcn to populate axes2

```