

# 波浪能装置运动分析及装置优化

## 摘要

对于问题一: 我们对浮子和振子在两种不同的线性阻尼器模型的情况下进行一维受力分析, 并以它们的初始位置为坐标原点进行运动分析, 在浮子圆锥部分不会露出水面也不会完全沉入水中的假设下进行数值求解, 解出结果为浮子的最大振幅小于 1m, 符合假设。浮子和振子在一段时间后运动均趋于周期振动, 且周期均等于垂荡激励力周期, 垂荡位移和速度见图 (2) (3) 和表 (2) (3) (4) (5)。

对于问题二: 我们使用瞬时功率对时间的积分来求解一段时间内的总功, 然后除以总时间得到平均功率。我们考虑了系统受迫振动达到稳定状态时的频率与激励力频率相等, 计算稳定后一个周期内的平均功率作为总平均功率。做数值积分时, 我们对时间段进行细分, 对每一段使用辛普森公式估计积分值。最后, 考虑到数值解难以使用传统优化方法, 我们使用模拟退火算法进行优化。只改变直线阻尼器阻尼系数  $\gamma$  时, 当  $\gamma = 29249$  时, 最大平均功率为 **434.183W**; 同时改变  $\gamma$  和幂指数  $\tau$  时, 当  $\gamma = 27643$ ,  $\tau = 0.987$  时, 最大平均功率为 **434.198W**。

对于问题三: 此问变为二维问题, 经分析可以确定系统有 5 个自由度, 我们首先使用质心公式和转动惯量公式对系统的物理参数进行计算, 然后对浮子和振子进行二维受力和运动学分析, 并使用 Runge-Kutta 法进行数值求解, 在进行计算时, 我们将中轴对振子的作用力等效在振子的质心上简化了计算量, 最终得到结果依然是系统最后趋于周期性运动, 浮子与振子的垂荡位移与速度和纵摇角位移与角速度见图 (9) (10) 和表 (8) (9)。

对于问题四: 采用和之前问题一样的分析思路, 在一定范围内改变直线阻尼器阻尼系数和旋转阻尼器阻尼系数, 使用模拟退火算法进行优化, 求得当直线阻尼器阻尼系数  $\gamma = 42529$ , 旋转阻尼器阻尼系数  $\eta' = 47500$  时, 最大平均功率为 **918.111W**。

同时, 我们对模型进行了灵敏度检验, 在一定范围内改变参数模型依然稳定, 证明了模型具有很好的鲁棒性。

**关键字: 受迫振动 等效法 辛普森公式 模拟退火 Runge-Kutta 法**

## 一、问题重述

### 1.1 问题背景

为了应对气候变暖和能源需要求，人们开始逐渐将视线转移到开发可再生清洁能源，波浪能则是其中一种优质能源，它是海洋表面波浪运动所产生的能量，可转为能源作不同用途，例如发电、海水淡化或推动抽水机等。波浪能发电则是通过波浪能装置将海洋波浪运动所传输的能量转化为电能，提高波浪能装置的能量转换效率能加快波浪能的开发利用，实现规模化使用，具有巨大的经济效益和社会意义。

### 1.2 问题要求

**问题一：**在只考虑浮子在波浪中做垂荡运动的情况下，根据给出的数据求解当直线阻尼器阻尼系数为  $10000\text{N}\cdot\text{m}$  时以及直线阻尼系数与浮子和振子的相对速度的绝对值的平方根成正比时，前 40 个周期内浮子和振子的垂荡位移和速度。

**问题二：**基于第一问，仍只考虑浮子在波浪中只做垂荡运动，求解阻尼系数可取自一定范围内，以及幂指数和阻尼系数均取自一定范围内时的最大输出功率以及此时的最优阻尼系数。

**问题三：**考虑浮子只做垂荡运动和摇荡运动，建立浮子和振子的运动模型，同时直线阻尼器和旋转阻尼器的阻尼系数均为确定值，求解此时浮子和振子在波浪激励力和波浪激励力矩的作用下的垂荡速度，纵摇角位移与角速度。

**问题四：**在第三问的基础上，此时的直线阻尼器和旋转阻尼器的阻尼系数都是取自一定的范围内，利用给出的数据计算输出功率最大的情况下的输出功率以及最优阻尼系数。

## 二、模型假设

1. 忽略中轴、底座、隔层及 PTO 的质量和各种摩擦
2. 直线阻尼器和旋转阻尼器输出功率无损耗
3. 忽略水中各种湍流对浮子的影响
4. 海水是无粘及无旋的
5. 弹簧和旋转弹簧在运动过程中均处在弹性限度内
6. 浮子的质量均匀分布，即其面质量密度为常数

### 三、符号说明

表 (1) 列出了本文所需要的符号

表 1 符号说明

符号	符号描述	单位
$M$	浮子质量	kg
$R$	浮子低半径	m
$h_1$	浮子圆柱部分高度	m
$h_2$	浮子圆锥部分	m
$m$	振子质量	kg
$r$	振子半径	m
$h_3$	振子高度	m
$\rho$	海水密度	kg/m <sup>3</sup>
$g$	重力加速度	m/s <sup>2</sup>
$k$	弹簧刚度	N/m
$z_0$	弹簧原长	m
$\eta$	扭转弹簧刚度	N·m
$\eta_1$	静水恢复力矩系数	N·n
$\omega$	入射波浪频率	s <sup>-1</sup>
$M_1$	垂荡附加质量	kg
$I_1$	纵摇附加转动惯量	kg·m <sup>2</sup>
$\gamma_1$	垂荡兴波阻尼系数	N·s/m
$\gamma$	阻尼器阻尼系数	N·s/m
$f$	垂荡激励力振幅	N
$L$	纵摇激励力矩振幅	N·m

## 四、问题分析

### 4.1 问题一分析

问题一需要结合波浪能装置的各项参数，分别对浮子和振子建立一维牛顿运动定律方程，在两种不同的线性阻尼器模型的情况下，使用 Runge-Kutta 法数值求解浮子和振子的运动，为了简化方程组，先可以假设浮子的圆锥部分不露出水面，浮子和振子之间也没有碰撞，然后再根据后续求解中得到的具体数值验证假设是否合理，如果不合理则采用分段函数进行求解。

### 4.2 问题二分析

对于问题二，仍然考虑只做垂荡作用，在问题一的模型基础上改变参数，在一定参数范围内求得最大平均功率。首先，我们考虑求解平均功率的方式，先用瞬时功率对时间的积分求出总共，在除以时间长度得到平均功率。此外，我们考虑到系统稳定时，浮子的位移和速度应该呈周期性变化，且周期和激励力周期相等，我们使用稳定时一个周期内的平均功率代替总平均功率。最后，考虑到数值解不便于直接使用求导等技巧求最优值，我们选用模拟退火算法进行求解。

### 4.3 问题三分析

问题三首先需要根据几何参数和物理参数对浮子和振子的质心和转动惯量等参数进行求解，然后建立二维平面上的牛顿运动定律方程和刚体转动方程，本质上也可以转化成六元二阶微分方程的数值解问题，和问题一类似，同样假设浮子的圆锥部分不露出水面，浮子和振子之间也没有碰撞，并用结果进行检验，由于此问变量较多，在实际计算处理时可以对等效作用力臂等变量进行常数化来简化最终的表达式。

### 4.4 问题四分析

对于问题四，可以结合前几问的思路进行分析，仍然是在一定范围内改变参数取值，通过数值求解得到对应的平均功率，然后使用模拟退火算法进行求解。

## 五、模型建立与求解

### 5.1 问题一模型建立与求解

#### 5.1.1 问题一模型建立

##### (1) 运动分析

首先，我们分析浮子的运动过程，受到波浪激励力的作用，浮子有向上运动的趋势，和振子之间产生相对运动，从而带动振子运动。而浮子和振子之间用弹簧和阻尼器相连，二者的作用会相互影响，发生垂荡运动。

初始状态时，系统处于平衡状态，我们记此时水面位于浮子  $x_0$  处，开始运动时浮子相对初始位置移动了  $x$ ，振子相对于初始位置移动了  $y$ ，假设浮子的圆锥部分不露出水面，浮子和振子之间也没有碰撞，如图 (1) 所示：

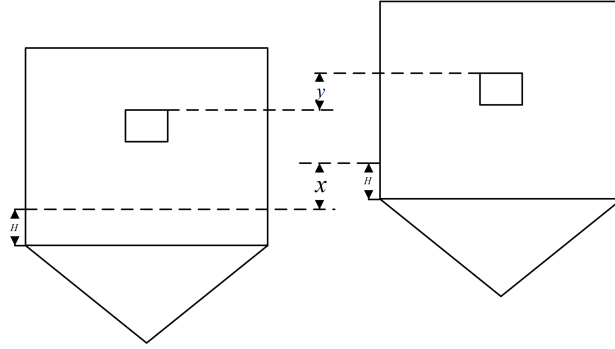


图 1 浮子运动示意图

##### (2) 受力分析

对振子分析，振子与浮子之间的相对运动距离表示为二者移动的差值  $y - x$ ，振子受到弹簧的弹力和阻尼器的阻尼力，前者与弹簧伸长量成正比，后者与速度成正比，根据牛顿第二定律可以列出如下方程：

$$m \frac{d^2 y}{dt^2} = -\gamma \left( \frac{dy}{dt} - \frac{dx}{dt} \right) - k(y - x) \quad (1)$$

其中  $m$  是振子质量， $\gamma$  是阻尼器的阻尼系数， $k$  是弹簧刚度。

对浮子分析，其质量由浮子本身质量和垂荡附加质量组成，同时受到浮力、弹簧弹力、阻尼器阻尼力和兴波阻尼力作用，根据牛顿第二定律列出方程：

$$(M + M_1) \frac{d^2 x}{dt^2} = -\rho g S x + \gamma \left( \frac{dy}{dt} - \frac{dx}{dt} \right) + k(y - x) - \gamma_1 \frac{dx}{dt} + f \cos \omega t \quad (2)$$

其中  $M$  是浮子质量， $M_1$  是垂荡附加质量。

当直线阻尼器的阻尼系数与浮子和振子的相对速度的绝对值的幂成正比时，按同样的受力分析方法可以得到方程：

$$\begin{cases} (M + M_1) \frac{d^2 x}{dt^2} = -\rho g S x + \frac{\frac{dy}{dt} - \frac{dx}{dt}}{|\frac{dy}{dt} - \frac{dx}{dt}|} \gamma |\frac{dy}{dt} - \frac{dx}{dt}|^\tau + k(z - x) - \gamma_1 \frac{dx}{dt} + f \cos \omega t \\ m \frac{d^2 y}{dt^2} = -\frac{\frac{dy}{dt} - \frac{dx}{dt}}{|\frac{dy}{dt} - \frac{dx}{dt}|} \gamma |\frac{dy}{dt} - \frac{dx}{dt}|^\tau - k(y - x) \end{cases} \quad (3)$$

### 5.1.1.2 问题一求解

#### (1) 直线阻尼器阻尼系数为固定值

根据浮力与重力平衡的关系可以得到关于初始时浮子没入水中的深度  $H$  的关系式

$$\rho g \left( \frac{1}{3} \pi R^2 h_2 + \pi R^2 H \right) = (m + M)g$$

解得

$$H = \frac{(m + M)}{\rho \pi R^2} - \frac{1}{3} h_2 = 2(\text{m})$$

因此只要浮子的运动幅度小于 1m，则浮子既不会露出圆锥部分也不会全部沉入水中。

将式 (1)(2) 联立，在 matlab 中进行数值求解，得到结果如图 (2) 所示。

图 (2) 中的浮子运动幅度始终小于 1m，满足浮子的圆锥部分不露出水面，浮子和振子之间也没有碰撞的假设。

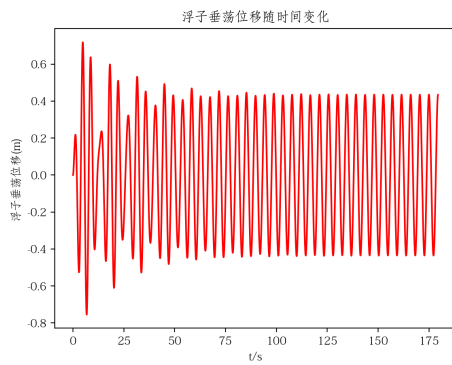
浮子和振子垂荡位移和速度随时间变化如表 (2) 所示。最终得到的 10s, 20s, 40s, 60s, 100s 的数据如表 (2), (3) 所示

**表 2 浮子垂荡位移和速度**

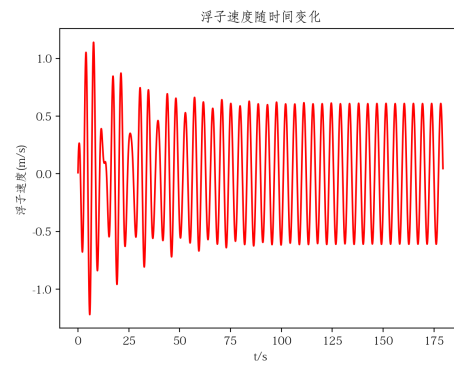
时间 (t)	垂荡位移 (m)	速度 (m/s)
10	-0.196958	-0.632366
20	-0.592868	-0.225976
40	0.288569	0.309005
60	-0.319193	-0.472396
100	-0.089623	-0.602284

**表 3 振子垂荡位移和速度**

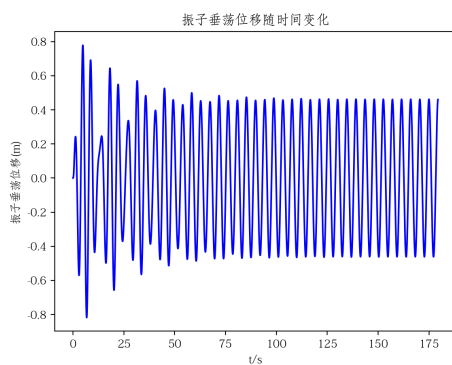
时间 (t)	垂荡位移 (m)	速度 (m/s)
10	-0.218443	-0.684446
20	-0.636709	-0.256679
40	0.299919	0.328743
60	-0.336475	-0.508440
100	-0.090458	-0.641235



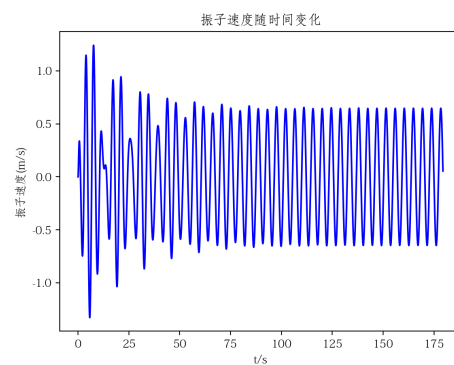
**(a) 浮子垂荡位移随时间变化**



**(b) 浮子速度随时间变化**



**(c) 振子振荡位移随时间变化**



**(d) 振子速度随时间变化**

**图 2 浮子, 振子垂荡位移, 速度随时间变化**

## (2) 直线阻尼器的阻尼系数与浮子和振子的相对速度的绝对值的幂成正比

直线阻尼器的阻尼系数与浮子和振子的相对速度的绝对值的幂成正比时, 将式 (3) 代入 MATLAB 中进行求解, 得到浮子和振子垂荡位移和速度随时间变化如表 (4),(5) 所

示。

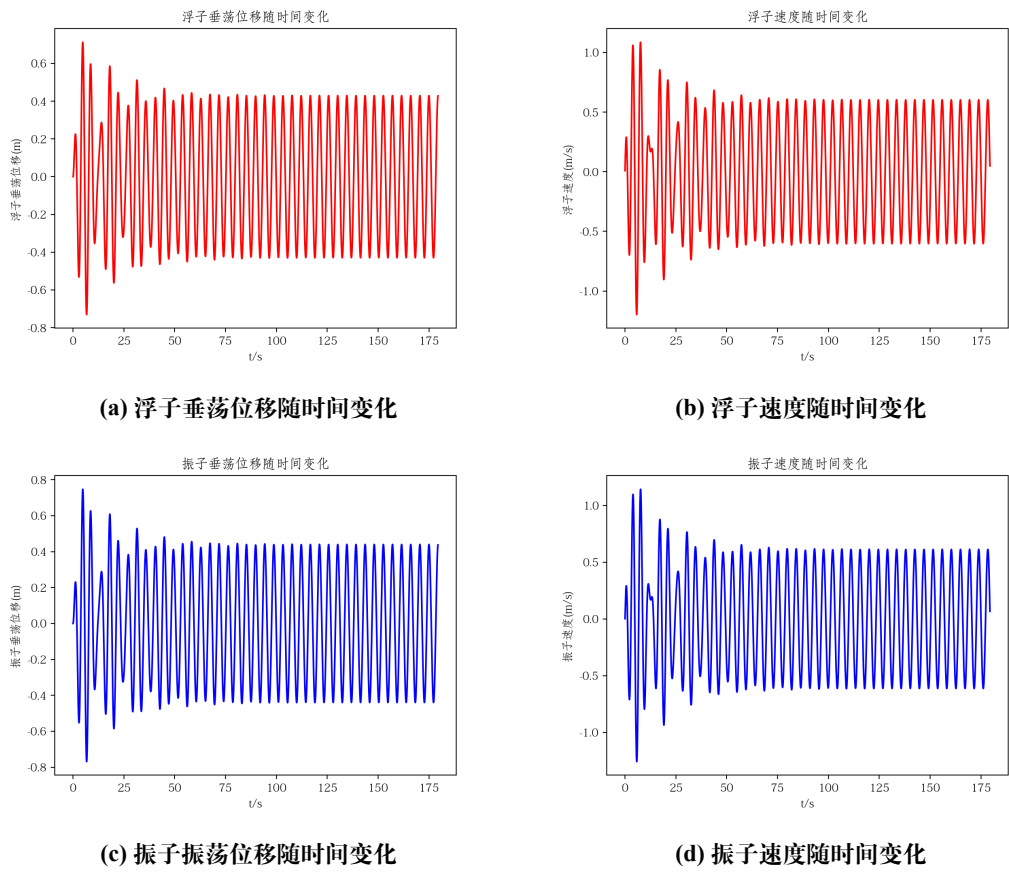


图 3 浮子，振子垂荡位移, 速度随时间变化

表 4 浮子垂荡位移和速度

时间 (t)	垂荡位移 (m)	速度 (m/s)
10	-0.172138	-0.552781
20	-0.555771	-0.132000
40	0.310834	0.379306
60	-0.296186	-0.427597
100	-0.082290	-0.587252



表 5 振子垂荡位移和速度

时间 (t)	垂荡位移 (m)	速度 (m/s)
10	-0.169112	-0.599592
20	-0.575928	-0.164235
40	0.308729	0.400184
60	-0.293954	-0.454573
100	-0.070764	-0.602468

## 5.2 问题二模型建立与求解

### 5.2.1 问题二模型建立

#### (1) 分析功率表达式

对于 PTO 系统，弹簧有时做正功有时做负功，始终在相互转化，在整个过程中其实没有做功；而阻尼器的阻尼力总是与运动方向相反，始终在做负功，因此整个 PTO 系统实际上是阻尼器在做功。

在一段时间内，阻尼器做功可由瞬时功率在时间段上的积分求得，平均功率可表示为总功除以时间如下式所示：

$$\bar{P} = \frac{1}{T} \int_0^T f(t)v(t) dt$$

同时由假设可知， $f(t)$  与速度的幂指数成正比，且由问题一的分析知速度可表示为  $\frac{dy}{dt} - \frac{dx}{dt}$ ，因此平均功率可以转换为下式：

$$\bar{P} = \frac{1}{T} \int_0^T \gamma \left( \frac{dy}{dt} - \frac{dx}{dt} \right)^{1+\tau} dt$$

#### (2) 模型优化

结合问题一的求解，对于确定的  $\gamma$  和  $\tau$ ，我们可以通过式 (4) 的积分求得平均功率。

对于受迫振动的系统，在经过一段时间的波动后最终会趋于稳定，并且振荡的频率与激励力频率相同 [2]，因此我们选取稳定后的一个周期，求出该周期内的平均功率作为总的平均功率。

于是优化模型可归纳如下：

$$\max \bar{P} = \frac{1}{T} \int_{nT}^{(n+1)T} \gamma \left( \frac{dy}{dt} - \frac{dx}{dt} \right)^{1+\tau} dt$$

$$s.t. \begin{cases} (M + M_1) \frac{d^2 x}{dt^2} = -\rho S x + \frac{\frac{dy}{dt} - \frac{dx}{dt}}{|\frac{dy}{dt} - \frac{dx}{dt}|} \gamma |\frac{dy}{dt} - \frac{dx}{dt}|^\tau + k(z - x) - \gamma_1 \frac{dx}{dt} + f \cos \omega t \\ m \frac{d^2 y}{dt^2} = -\frac{\frac{dy}{dt} - \frac{dx}{dt}}{|\frac{dy}{dt} - \frac{dx}{dt}|} \gamma |\frac{dy}{dt} - \frac{dx}{dt}|^\tau - k(y - x) \end{cases} \quad (4)$$

式中  $n$  表示第  $n$  个稳定的周期。

我们采用数值积分的方法进行计算，对于一个时间段，将其细分为若干个小时的时间段，利用辛普森公式计算小时间段内的积分最后加和，辛普森公式如下：

$$\int_t^{t+2h} \approx \frac{h}{3} (f(t) + 4f(t+h) + f(t+2h)) dt \quad (5)$$

式中  $h$  表示极小时间间隔， $t$  表示起始时间。

### (3) 模拟退火求解

由于我们使用数值方法求解模型，每当  $\gamma$  和  $\tau$  变化时，我们需要重新对微分方程组进行求解。而模型相当于是一个黑箱，将  $\gamma$  和  $\tau$  作为输入，模型输出平均功率，中间的过程很复杂，我们无法通过简单的求导进行优化，因此我们使用模拟退火算法进行优化求解，算法流程图如下：

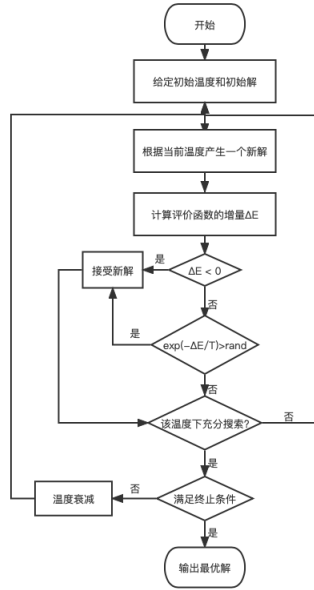


图4 模拟退火算法流程图

模拟退火算法可以有效地避免陷入局部最优解，并且模拟退火算法具有的马尔可夫链可以扩大搜索范围，适合对该问题进行求解。

#### 5.2.2 问题二模型求解

设置参数  $\gamma \in [0, 100000]$ ， $\tau = 1$ ，使用模拟退火算法求解，适应度函数取最大功率的倒数，得到结果如下所示：

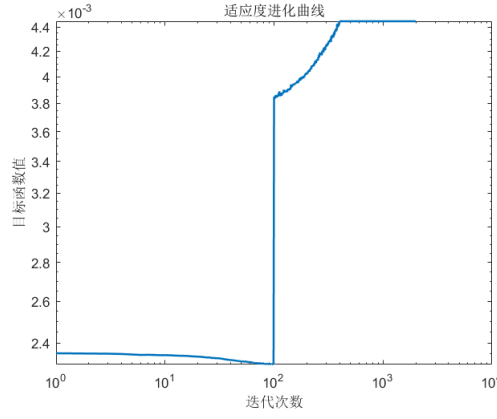


图 5 第 (1) 小问模拟退火适应度曲线

当  $\gamma = 29249$ ，功率最大达到 434.183W，在搜索过程中适应度曲线变化较大。如果幂指数也变化，重新设定参数  $\gamma \in [0, 100000]$ ， $\tau \in [0, 1]$ ，得到结果如下所示：

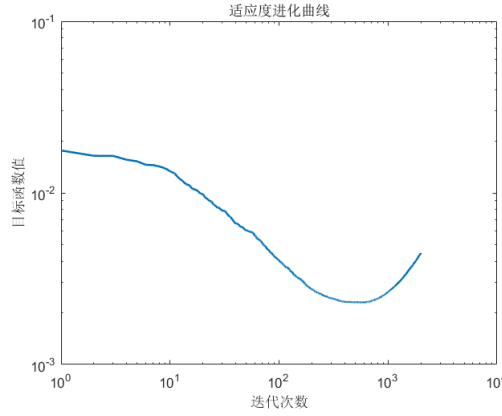


图 6 第 (2) 小问模拟退火适应度曲线

当  $\gamma = 27643$ ， $\tau = 0.987$  时，功率最大达到 434.198W。在搜索过程中，适应度函数先递减再递增，最优解在最低处取到。

### 5.3 问题三模型建立与求解

#### 5.3.1 问题三模型建立

##### (1) 运动分析

第三问由于增加了纵摇的情况，完整地描述体系一共需要 5 个自由度，分别是描述浮子的  $(x, y, \theta)$  和描述振子的  $(z, \phi)$ ，其中  $x, y$  是浮子的转轴相对于初始位置的横向位移和纵向位移， $\theta$  是浮子由初始状态旋转的角度，取垂直纸面向外方向为正， $z$  是弹簧相对初始位置的伸长量， $\phi$  是中轴取垂直纸面向里方向为正时由初始状态旋转的角度。

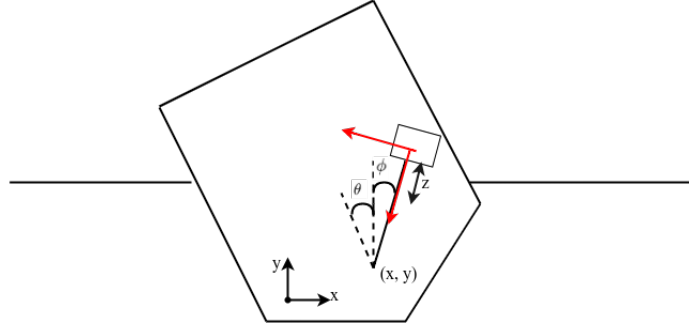


图7 运动参量示意图

## (2) 受力分析

对浮子在  $x$ ,  $y$  方向分别进行受力分析可以得到式 (6), 式 (7)。

$$(M + M_1) \frac{d^2 x}{dt^2} = [\gamma \frac{dz}{dt} + kz] \sin \phi - \gamma_1 \frac{dx}{dt} + \frac{\eta(\phi + \theta) + \eta'(\frac{d\phi}{dt} + \frac{d\theta}{dt})}{l} \cos \phi - mg \sin \phi \quad (6)$$

$$(M + M_1) \frac{d^2 y}{dt^2} = [\gamma \frac{dz}{dt} + kz] \cos \phi - \gamma_1 \frac{dy}{dt} - \frac{\eta(\phi + \theta) + \eta'(\frac{d\phi}{dt} + \frac{d\theta}{dt})}{l} \sin \phi + mg(1 - \cos \phi) + f \cos(\omega t) - \rho g S y \quad (7)$$

式中  $\gamma_1$  是垂荡兴波阻尼系数,  $\eta$  是扭转弹簧刚度,  $\eta'$  是扭转弹簧阻尼系数,  $\phi$  是振子纵摇角位移,  $\theta$  是浮子纵摇角位移。

对于式 (6), (7), 等号右边第一项是弹簧和直线阻尼器对浮子的作用, 第二项是垂荡兴波阻尼作用, 第三项是扭转弹簧和旋转阻尼器的作用, 第四项是由于初始时弹簧弹力与振子重力平衡而中轴转过一定角度后即使弹簧长度不变弹力和重力方向也不再相反所产生的受力修正项。

对振子的位置进行几何计算可以得到振子的位置为  $(x + (z'_0 + z + \frac{1}{2}h_3) \sin \phi, y + (z'_0 + z + \frac{1}{2}h_3) \cos \phi)$ , 使用牛顿第二定律对振子进行分析可得式 (8), (9)。

$$m \frac{d^2 [x + (z'_0 + z + \frac{1}{2}h_3) \sin \phi]}{dt^2} = - \frac{\eta(\phi + \theta) + \eta'(\frac{d\phi}{dt} + \frac{d\theta}{dt})}{l} \cos \phi - [\gamma \frac{dz}{dt} + kz] \sin \phi + mg \sin \phi \quad (8)$$

$$m \frac{d^2 [y + (z'_0 + z + \frac{1}{2}h_3) \cos \phi]}{dt^2} = \frac{\eta(\phi + \theta) + \eta'(\frac{d\phi}{dt} + \frac{d\theta}{dt})}{l} \sin \phi - [\gamma \frac{dz}{dt} + kz] \cos \phi - (1 - \cos \phi) mg \quad (9)$$

对式 (8),(9) 左边进行展开可以得到式 (10),(11)

$$m \frac{d^2[x + (z'_0 + z) \sin \phi]}{dt^2} = m \left[ \frac{d^2x}{dt^2} + \frac{d^2z}{dt^2} \sin \phi + \frac{dz}{dt} \cos \phi \frac{d\phi}{dt} + \left[ \frac{dz}{dt} \sin \phi - (z + z'_0) \sin \phi \cdot \frac{d\phi}{dt} \right] \frac{d\phi}{dt} + (z + z'_0) \cos \phi \frac{d^2\phi}{dt^2} \right] \quad (10)$$

$$m \frac{d^2[y + (z'_0 + z) \cos \phi]}{dt^2} = m \left[ \frac{d^2y}{dt^2} + \frac{d^2z}{dt^2} \cos \phi - \frac{dz}{dt} \sin \phi - \left[ \frac{dz}{dt} \sin \phi - (z'_0 + z) \cos \phi \frac{d\phi}{dt} \right] - (z'_0 + z) \sin \phi \frac{d^2\phi}{dt^2} \right] \quad (11)$$

式中  $z'_0$  是弹簧初始长度,  $h_3$  是振子高度。

因为不考虑中轴质量, 所以各项与浮子受力公式中的类似, 除垂荡兴波阻尼作用外均取相反作用力。

对浮子在水中的转动进行分析可以得到式 (12)。

$$\begin{aligned} (I_M + I_1) \frac{d^2\theta}{dt^2} = & Mgd \sin \theta - (\eta(\phi + \theta) + \eta' \left( \frac{d\phi}{dt} + \frac{d\theta}{dt} \right)) - \eta_1 \theta - \eta_2 \frac{d\theta}{dt} \\ & + L \cos \omega t + M \frac{d^2x}{dt^2} d \cos \theta + M \frac{d^2y}{dt^2} d \sin \theta \\ & + mg \sin \phi \cdot l \cos \phi + mg(1 - \cos \phi) l \sin \phi \end{aligned} \quad (12)$$

式中  $I_M$  是浮子绕转轴转动惯量,  $I_1$  是纵摇附加转动惯量,  $L$  是纵摇激励力矩振幅,  $l$  是中轴等效作用力臂。

对于式 (12), 转轴选取过一条底座直径, 等号右边第一项为浮子重力的力矩, 第二项为扭转弹簧和旋转阻尼器产生的力矩, 第三项为静水恢复力矩, 第四项为纵摇兴波阻尼力矩, 第五项为纵摇激励力矩, 第六, 七项为在非惯性系中浮子质心的惯性力产生的力矩, 第八, 九项为角度改变后弹力和重力合力修正项产生的力矩。

对振子在水中的转动进行分析, 设置  $l$  为中轴对振子作用力的等效力臂, 可以得到式 (13)。

$$-\frac{\eta(\phi + \theta)}{l} [l - (z + z'_0)] = I_m \frac{d^2\phi}{dt^2} \quad (13)$$

结合上面的式 (6)(7)(8)(9)(12)(13)(10)(11) 建立模型即可得到浮子与振子的垂荡位移与速度和纵摇角位移与角速度。

### 5.3.2 问题三问题求解

在进行计算时, 需要进行计算的数据如表6 所示

表 6 需要计算的参数

符号	含义
$I_M$	浮子相对于底座直径的转动惯量
$I_m$	振子相对于经过质心的一条直径的转动惯量
$d$	浮子的质心距离底座的距离
$z'_0$	初始弹簧的长度

转动惯量公式为  $I_M = \sum_i m_i \cdot r_i^2$ , 结合平行轴定理以及图 (8), 可以得到各部位相对于圆盘隔层直径的转动惯量公式如表 (7) 所示

表 7 波浪能各部位转动惯量

符号	部位	几何形状	转动惯量
$I_{M_1}$	浮子圆柱壳体侧面	圆柱形圆环	$\int_0^{h_1} (\frac{1}{2}R^2 + h^2)\sigma 2\pi R dh$
$I_{M_2}$	浮子圆柱顶面顶面	圆形	$\frac{1}{4}\sigma\pi R^4 + \sigma\pi R^2 h_1^2$
$I_{M_3}$	浮子圆锥面	圆锥面	$\int_0^R [\frac{1}{2}r^2 + (h_2 - r \tan \alpha)^2]\sigma 2\pi \frac{dr}{\cos \alpha}$
$I_m$	振子	圆柱体	$2 \int_0^{\frac{h_3}{2}} m \frac{dh}{h_3} (\frac{1}{4}r^2 + h^2)$

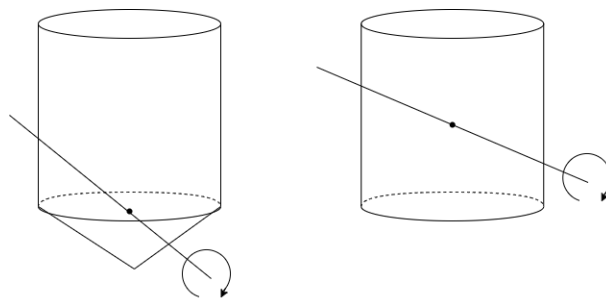


图 8 浮子转动惯量 (左), 振子转动惯量 (右边)

代入实际数值进行计算得到的结果如图 (9)

$$z'_0 = z_0 - \frac{mg}{k} = 0.2019575(m)$$

$$\alpha = \arctan \frac{h_2}{R} = 38.6598(^{\circ})$$

$$d = \frac{\int_0^R \frac{2\pi r^2 \tan \alpha}{\cos \alpha} dr + \int_{h_2}^{h_1+h_2} 2\pi R h dh + \pi R^2 (h_1 + h_2)}{\int_0^R \frac{2\pi r dr}{\cos \alpha} + \int_{h_2}^{h_1+h_2} 2\pi R dh + \pi R^2} - h_2$$

$$= 1.407925(m)$$

$$I_m = 202.75(\text{kg} \cdot \text{m}^2)$$

$$I_M = I_{M_1} + I_{M_2} + I_{M_3} = 18044.4197(\text{kg} \cdot \text{m}^2)$$

在对所需的数值进行求解完成后，结合上面的式 (6)(7)(8)(9)(12)(13)(10)(11) 建立起的约束关系，代入到 MATLAB 中进行求解，可以得到图 (9), (10) 所示结果。在特定时间节点上的数据如表 (8) (9) 所示。

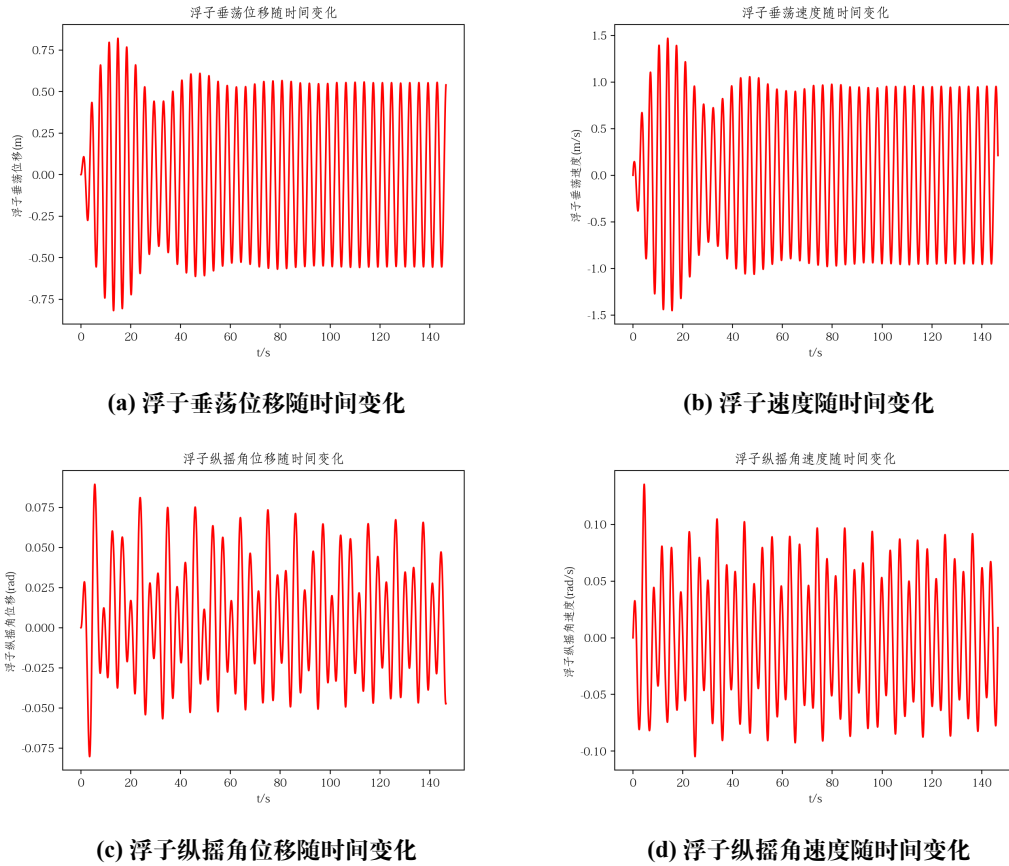


图 9 浮子垂荡位移, 速度, 纵摇角位移, 纵摇角速度随时间变化

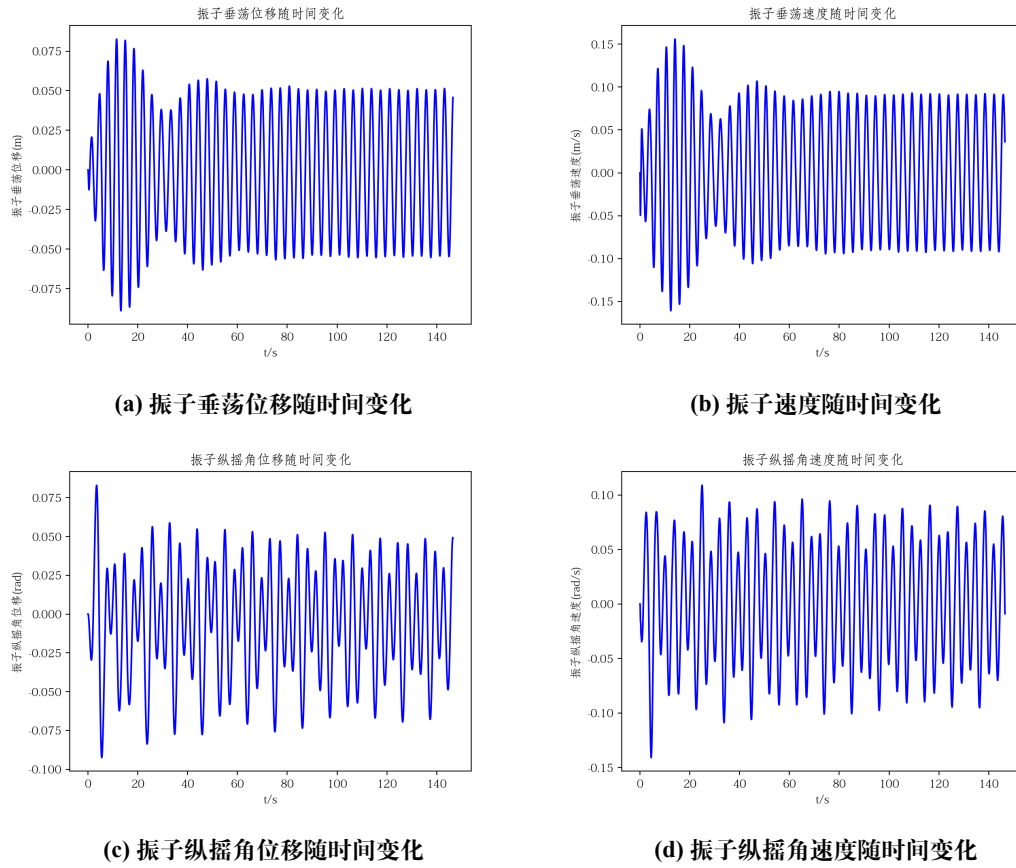


图 10 振子垂荡位移, 速度, 纵摇角位移, 纵摇角速度随时间变化

表 8 浮子垂荡位移, 速度, 纵摇角位移, 纵摇角速度

时间 (t)	垂荡位移 (m)	垂荡速度 (m/s)	纵摇角位移	纵摇角速度
10	-0.5277278	0.956116	-0.009925	-0.042429
20	-0.703615	-0.292289	0.016931	0.000150
40	0.364231	0.750058	-0.019996	-0.018699
60	-0.322955	-0.721478	0.011832	0.027803
100	-0.052414	-0.944013	-0.001555	0.051139



表 9 振子垂荡位移, 速度, 纵摇角位移, 纵摇角速度

时间 (t)	垂荡位移 (m)	垂荡速度 (m/s)	纵摇角位移	纵摇角速度
10	-0.069996	0.069355	0.010223	0.043682
20	-0.0677253615	-0.054090	-0.017469	-0.000058
40	0.022742	0.084078	0.020821	0.019454
60	-0.020903	-0.078669	-0.012227	-0.028880
100	0.007242	-0.087404	0.001633	-0.052695

## 5.4 问题四模型建立与求解

### 5.4.1 问题四模型建立

和问题二中功率求解方法类似, PTO 系统做功来自直线阻尼器和旋转阻尼器, 直线阻尼器的做功同式 (4), 旋转阻尼器做功表达式如式 (14)

$$w = M(d\theta + d\phi) \quad (14)$$

旋转阻尼器做功的平均功率为式 (15)

$$\overline{P_M} = \frac{1}{T} \int_0^T \eta' \left( \frac{d\phi}{dt} + \frac{d\theta}{dt} \right)^2 dt \quad (15)$$

因此系统的总平均功率为式 (16)

$$P_{Total} = \frac{1}{T} \left[ \int_0^T \left( \gamma \left( \frac{dz}{dt} \right)^2 dt \right) + \int_0^T \eta' \left( \frac{d\phi}{dt} + \frac{d\theta}{dt} \right)^2 dt \right] \quad (16)$$

### 5.4.2 问题四模型求解

设置参数  $\gamma \in [0, 100000]$ ,  $\eta' \in [0, 100000]$ , 将适应度函数设为最大平均功率的倒数, 求得结果如图 (11) 所示。

当直线阻尼器阻尼系数  $\gamma = 42529$ , 旋转阻尼器阻尼系数  $\eta' = 47500$  时, 最大平均功率达到 918.111W。

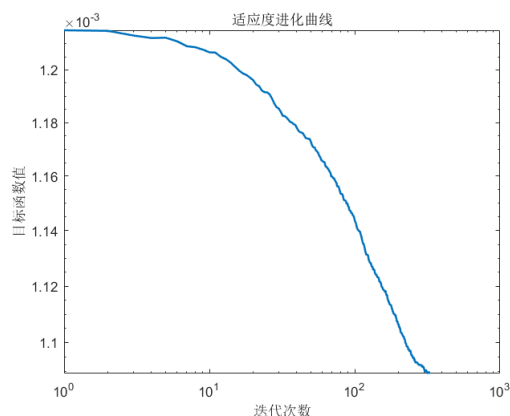


图 11 问题四模拟退火适应度曲线

## 六、模型评价

### 6.1 灵敏度分析

针对问题一中的模型，弹簧刚度、阻尼系数和激励力振幅是模型中的重要参数，而阻尼系数在之前的问题中已经讨论过了，因此我们讨论参数弹簧刚度和激励力振幅增加或减少 5% 对浮子垂荡位移的影响，结果如图 (12) 所示。

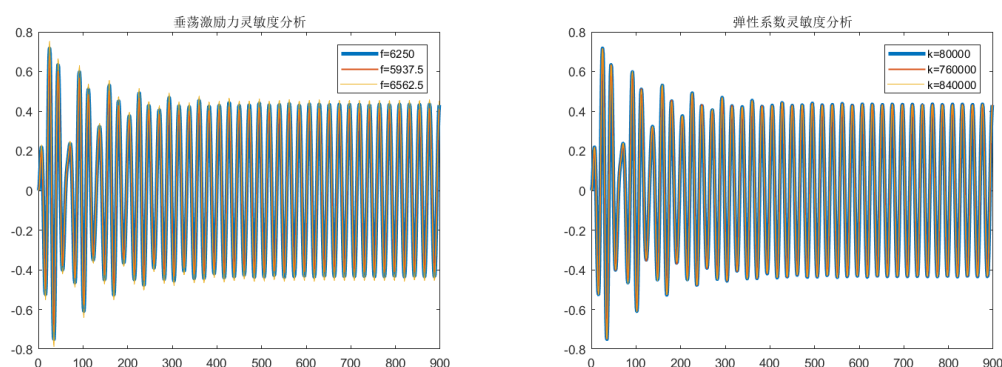


图 12 问题 1 中弹簧刚度和激励力振幅增加 5%

针对问题三中的模型，仍然改变 5% 的弹簧刚度和激励力振幅，结果如图 (13) 所示。可以发现，改变参数前后浮子垂荡位移并没有发生明显变化，说明模型对参数不敏感，模型很稳定。

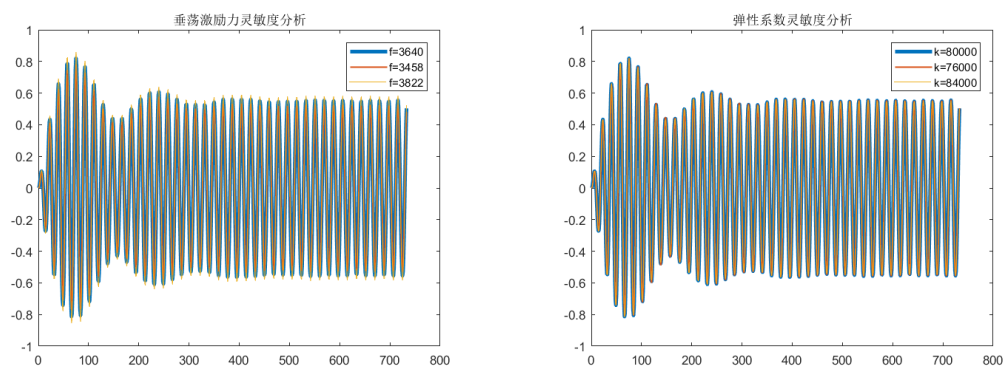


图 13 问题 3 中弹簧刚度和激励力振幅增加 5%

## 6.2 模型优点

1. 选择合适参考坐标系与变量简化方程组，使得系统具有较强的可解性。
2. 考虑了多种系统可能出现的情况，在假设的前提下求解并使用结果验证假设，使系统更具有严谨性。
3. 采用等效法使用一个力代替多个力或者复杂的积分表达式，降低复杂度。

## 6.3 模型缺点

1. 模型使用了数值计算，在求解过程中，为了减小时间复杂度而降低了精度。
2. 没有考虑水湍流等复杂的流体运动和浮子形状给受力分析带来的影响。

## 参考文献

- [1] 史峰. MATLAB 智能算法 30 个案例分析 [M]. 北京航空航天大学出版社, 2011.
- [2] 王铁平. 受迫振动及共振频率的计算 [J]. 技术物理教学, 1997(2):19-21.

## 附录 A 程序中所用到的初始数据

```
M=4866;  
R=1;  
h1=3;  
h2=0.8;  
m=2433;  
r=0.5;  
h3=0.5;  
rho=1025;  
g=9.8;  
k=80000;  
z0=0.5;  
eta=250000;  
eta1=8890.7
```

```
%问题一
```

```
gamma=10000;
```

```
omiga=1.4005;  
M1=1335.535;  
I1=6779.315;  
gamma1=656.3616;  
eta2=151.4388;  
f=6250;  
L=1230;
```

```
% 问题二
```

```
% omiga=2.2143;  
% M1=1165.992;  
% I1=7131.29;  
% gamma1=167.8395;  
% eta2=2992.724;  
% f=4890;  
% L=2560;  
%
```

```
% 问题三
```

```
% omiga=1.7152;  
% M1=1028.876;  
% I1=7001.914;  
% gamma1=683.4558;  
% eta2=654.3383;  
% f=3640;  
% L=1690;  
%
```

```
% 问题四
```

```

% omiga=1.9806;
% M1=1091.099;
% I1=7142.493;
% gamma1=528.5018;
% eta2=1655.909;
% f=1760;
% L=2140;

```

## 附录 B 第一问

```

omiga=1.4005;
delta_t = 0.01;

%微分方程求解
T = 2*pi/omiga;
tic
[t,X]=ode45(@root1,[0:delta_t:40*T],[0,0,0,0]);
toc

function Dx = root1(t,x) % dy是标准化后左边的式子，x是式子右边的自变量，y是因变量

%问题一题目中数据
gamma=10000;

%表格中数据
M=4866;
R=1;
h1=3;
h2=0.8;
m=2433;
r=0.5;
h3=0.5;
rho=1025;
g=9.8;
k=80000;
z0=0.5;
eta=250000;
eta1=8890.7

omiga=1.4005;
M1=1335.535;
I1=6779.315;
gamma1=656.3616;
eta2=151.4388;
f=6250;

```

```

L=1230;

%中间变量计算
S=pi*R^2;
M_T=M+M1;

    Dx = zeros(4,1);
    Dx(1) = x(3);
    Dx(2) = x(4);
    Dx(3) = (-rho*g*S*x(1)+gamma*(x(4)-x(3))+k*(x(2)-x(1))-gamma1*x(3)+f*cos(omiga*t))/M_T;
    Dx(4) = -(gamma*(x(4)-x(3))+k*(x(2)-x(1)))/m;

end

```

## 附录 C 第二问

```

    clc,clear
global delta_t T gamma tau rho g S k gamma1 f omiga M_T m
delta_t = 0.1;
M=4866;
R=1;
h1=3;
h2=0.8;
m=2433;
r=0.5;
h3=0.5;
rho=1025;
g=9.8;
k=80000;
z0=0.5;
eta=250000;
omiga=2.2143;
M1=1165.992;
I1=7131.29;
gamma1=167.8395;
eta2=2992.724;
f=4890;
L=2560;
S=pi*R^2;
M_T=M+M1;

T = 2*pi/omiga;
tau=1;
gamma=10000;

```

```

tic
%%
X=[0,100000;0,1];
Temperature=1e-1;
decay=0.0005;
Len=10;
step=0.001;
P=0;
detf=1e-8; %误差阈值
size1=size(X,1);

pre=rand(size1,1).*(X(:,2)-X(:,1))+X(:,1);
prebest=pre;
pre=rand(size1,1).*(X(:,2)-X(:,1))+X(:,1);
best=pre;
det=abs(func(best(1),best(2))-func(prebest(1),prebest(2)));
while (det>detf) && (Temperature>1e-4)
    Temperature=Temperature-decay;
    for i=1:Len
        next=pre+step.*(rand(size1,1).*(X(:,2)-X(:,1))+X(:,1));
        if next(1)>X(1,2)
            next(1)=X(1,2);
        end
        if next(1)<X(1,1)
            next(1)=X(1,1);
        end
        if next(2)>X(2,2)
            next(2)=X(2,2);
        end
        if next(2)<X(2,1)
            next(2)=X(2,1);
        end
        pre_func=func(pre(1),pre(2));
        next_func=func(next(1),next(2));
        best_func=func(best(1),best(2));
        if (best_func>next_func)
            prebest=best;
            best=next;
        end
        if (pre_func-next_func>0)
            pre=next;
        else
            changer=-1*(next_func-pre_func)/Temperature;
            p1=exp(changer);
            if p1>rand()
                pre=next;
            end
        end
    end
end

```



```

        end
    end
    trace(P+1)=pre_func;
    P=P+1;
end
det=abs(func(best(1),best(2))-func(prebest(1),prebest(2)));
end
disp('最大值在点')
disp(best(1))
disp(best(2))
disp('最大值为: ')
disp(1./func(best(1),best(2)))
figure
loglog(trace(2:end),'linewidth',1.5)
xlabel('迭代次数')
ylabel('目标函数值')
title('适应度进化曲线')
y=best;
toc
%%
%函数定义
function y=solve_func()
    global delta_t T gamma tau rho g S k gamma1 f omiga M_T m
    [t,X]=ode45(@root,0:delta_t:40*T,[0,0,0,0]);
    index=find(t>=38*T & t<=39*T);
    absfunc=gamma*(abs(X(index,4)-X(index,3))).^(1+tau);
    len=length(index);
    avg_p=zeros(len-2,1);
    for i=1:len-2
        avg_p(i)=integral(absfunc(i),absfunc(i+1),absfunc(i+2),delta_t);
    end
    avg_p=sum(avg_p)/T;
    y=avg_p;
    function Dx = root(t,x) % dy是标准化后左边的式子, x是式子右边的自变量, y是因变量
        %x1是x, x2是y, x3是x', x4是y'
        Dx = zeros(4,1);
        Dx(1) = x(3);
        Dx(2) = x(4);
        Dx(3) = (-rho*g*S*x(1)+gamma*sign(x(4)-x(3))*(abs(x(4)-x(3)))
            ^tau+k*(x(2)-x(1))-gamma1*x(3)+f*cos(omiga*t))/M_T;
        Dx(4) = -(gamma*sign(x(4)-x(3))*(abs(x(4)-x(3)))^tau+k*(x(2)-x(1)))/m;
    end
end

function y=integral(x1,x2,x3,delta_t)
    y=delta_t*(x1+4*x2+x3)/3;
end

```

```

function y=func(x1,x2)
    global gamma tau
    gamma=x1;
    tau=x2;
    y=1./solve_func();
end

```

## 附录 D 第三问

```

    omiga=1.7152;
delta_t = 0.01;

%微分方程求解
T = 2*pi/omiga;
[t,X]=ode45(@root3,[0:delta_t:40*T],[0,0,0,0,0,0.45,0,0,0,0,0,0]);

function Dx = root3(t,x)

%新加的常数，具体数值计算见论文
    Im=202.75;
    IM=18044.4197;
    z1=0.2019575;
    d=1.407925;

%题目中数据
    gamma=10000;
    eta3=1000;%旋转阻尼器的阻尼系数

%表格中数据
    M=4866;
    R=1;
    h1=3;
    h2=0.8;
    m=2433;
    r=0.5;
    h3=0.5;
    rho=1025;
    g=9.8;
    k=80000;
    z0=0.5;
    eta=250000;
    eta1=8890.7;

    omiga=1.7152;

```

```

M1=1028.876;
I1=7001.914;
gamma1=683.4558;
eta2=654.3383;
f=3640;
L=1690;

%中间变量计算
S=pi*R^2;
M_T=M+M1;

Dx = zeros(12,1);
Dx(1) = x(7); %x, 浮子横向位移
Dx(2) = x(8); %y, 浮子纵向位移
Dx(3) = x(9); %z, 振子相对初始状态伸长量
Dx(4) = x(10); %phi, 中轴转动角度
Dx(5) = x(11); %theta, 浮子转动角度
Dx(7) = ((gamma*(x(9))+k*(x(3)))*sin(x(4))-gamma1*x(7)+
(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4))-m*g*sin(x(4)))/M_T;
Dx(8) = ((gamma*(x(9))+k*(x(3)))*cos(x(4))-gamma1*x(8)-
(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))+f*cos(omega*t)
-rho*g*S*x(2)+m*g*(1-cos(x(4))))/M_T;

Dx(9)= (((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))
-(gamma*(x(9))+k*(x(3)))*cos(x(4))-m*g*(1-cos(x(4))))/m-(((gamma*(x(9))
+k*(x(3)))*cos(x(4))-gamma1*x(8)...
-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))+
f*cos(omega*t)-rho*g*S*x(2))/M_T)+cos(x(4))*x(9)*x(10)...
+(x(9)*sin(x(4))-(x(3)+z1+0.25)*cos(x(4))*x(10))*x(10)+
(x(3)+z1+0.25)*cos(x(4))*((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))
*(0.45-(x(3)+z1+0.25))/0.45/Im))...
/cos(x(4)));

Dx(10)= ((-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4))-
(gamma*(x(9))+k*(x(3)))*sin(x(4))+m*g*sin(x(4)))/m-(((gamma*(x(9))
+k*(x(3)))*sin(x(4))-gamma1*x(7))...
+(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4)))/M_T)-(
(((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))...
-(gamma*(x(9))+k*(x(3)))*cos(x(4)))/m-(((gamma*(x(9))+k*(x(3))
*cos(x(4))...
-gamma1*x(8)-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))
/M_T)+cos(x(4))*x(9)*x(10)...
+(x(9)*sin(x(4))-(x(3)+z1+0.25)*cos(x(4))*x(10))*x(10)+(x(3)+z1+0.25)
*cos(x(4))*((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))*(0.45-(x(3)+z1+0.25))/0.45/Im))...
/cos(x(4)))*sin(x(4))-cos(x(4))*x(9)*x(10)...
-(x(9)*cos(x(4))-(x(3)+z1+0.25)*sin(x(4))*x(10))*x(10))/((x(3)+z1+0.25)*cos(x(4)));

```

```

Dx(11)=(0*M*g*d*sin(x(5))-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))-eta1*x(5)
-eta2*x(11)+L*cos(omiga*t)+...
    M*d*cos(x(5))*(((gamma*(x(9))+k*(x(3)))*sin(x(4))-gamma1*x(7)
    +(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))...
    /0.45*cos(x(4)))/M_T)+M*d*sin(x(5))*(((gamma*(x(9))+k*(x(3))
    *cos(x(4))-gamma1*x(8)-eta*(x(4)...
    +x(5))/0.45*sin(x(4))+f*cos(omiga*t)-rho*g*S*x(2))/M_T)+
    m*g*(x(3)+z1+0.25)*sin(x(4)))/(IM+I1);
end

```

## 附录 E 第四问

```

clc,clear
global gamma k gamma1 eta eta3 m g M_T f rho S omiga Im z1 d eta1 eta2 L M IM I1 delta_t T
delta_t = 0.1;
M=4866;
R=1;
h1=3;
h2=0.8;
m=2433;
r=0.5;
h3=0.5;
rho=1025;
g=9.8;
k=80000;
z0=0.5;
eta=250000;
eta1=8890.7;
omiga=1.9806;
T = 2*pi/omiga;
M1=1091.099;
I1=7142.493;
gamma1=528.5018;
eta2=1655.909;
f=1760;
L=2140;
Im=202.75;
IM=18044.4197;
z1=0.2019575;
d=1.407925;
S=pi*R^2;
M_T=M+M1;
gamma=50000;
eta3=10000;

```

```

tic
%%
X=[0,100000;0,100000];
Temperature=0.001;
decay=0.000005;
Len=10;
step=0.001;
P=0;
detf=1e-8; %误差阈值
size1=size(X,1);

pre=rand(size1,1).*(X(:,2)-X(:,1))+X(:,1);
prebest=pre;
pre=rand(size1,1).*(X(:,2)-X(:,1))+X(:,1);
best=pre;
det=abs(func(best(1),best(2))-func(prebest(1),prebest(2)));
while (det>detf) && (Temperature>0.000001)
    Temperature=Temperature-decay;
    for i=1:Len
        next=pre+step.*(rand(size1,1).*(X(:,2)-X(:,1))+X(:,1));
        if next(1)>X(1,2)
            next(1)=X(1,2);
        end
        if next(1)<X(1,1)
            next(1)=X(1,1);
        end
        if next(2)>X(2,2)
            next(2)=X(2,2);
        end
        if next(2)<X(2,1)
            next(2)=X(2,1);
        end
        pre_func=func(pre(1),pre(2));
        next_func=func(next(1),next(2));
        best_func=func(best(1),best(2));
        if (best_func>next_func)
            prebest=best;
            best=next;
        end
        if (pre_func-next_func>0)
            pre=next;
        else
            changer=-1*(next_func-pre_func)/Temperature;
            p1=exp(changer);
            disp(p1)
            if p1>rand()
                pre=next;
            end
        end
    end
end

```

```

        end
    end
    trace(P+1)=pre_func;
    P=P+1;
end
det=abs(func(best(1),best(2))-func(prebest(1),prebest(2)));
end
disp('最大值在点')
disp(best(1))
disp(best(2))
disp('最大值为: ')
disp(1/func(best(1),best(2)))
figure
loglog(trace(2:end),'linewidth',1.5)
xlabel('迭代次数')
ylabel('目标函数值')
title('适应度进化曲线')
y=best;
toc

%%
function y=solve_func()
    global gamma k gamma1 eta eta3 m g M_T f rho S omiga Im z1 d eta1 eta2 L M IM I1 delta_t T

    [t,X]=ode45(@root3,0:delta_t:40*T,[0,0,0,0,0,0.45,0,0,0,0,0,0]);
    index=find(t>=38*T & t<=39*T);
    absfunc=gamma*(X(index,9)).^2+eta3*(X(index,10)+X(index,11)).^2;
    len=length(index);
    avg_p=zeros(len-2,1);
    for i=1:len-2
        avg_p(i)=integral(absfunc(i),absfunc(i+1),absfunc(i+2),delta_t);
    end
    avg_p=sum(avg_p)/T;
    y=avg_p;

    function Dx = root3(t,x)
        Dx = zeros(12,1);
        Dx(1) = x(7); %x, 浮子横向位移
        Dx(2) = x(8); %y, 浮子纵向位移
        Dx(3) = x(9); %z, 振子相对初始状态伸长量
        Dx(4) = x(10); %phi, 中轴转动角度
        Dx(5) = x(11); %theta, 浮子转动角度
        Dx(7) = ((gamma*(x(9))+k*(x(3)))*sin(x(4))-gamma1*x(7)+
            (eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4))-m*g*sin(x(4)))/M_T;
        Dx(8) = ((gamma*(x(9))+k*(x(3)))*cos(x(4))-gamma1*x(8)-
            (eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))+f*cos(omiga*t)
            -rho*g*S*x(2)+m*g*(1-cos(x(4))))/M_T;
    end
end

```

```

Dx(9)= (((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))-
(gamma*(x(9))+k*(x(3)))*cos(x(4))-m*g*(1-cos(x(4))))/m-
(((gamma*(x(9))+k*(x(3)))*cos(x(4))-gamma1*x(8)...
-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))
+f*cos(omiga*t)-rho*g*S*x(2))/M_T)+cos(x(4))*x(9)*x(10)...
+(x(9)*sin(x(4))-(x(3)+z1+0.25)*cos(x(4))*x(10))*x(10)+(x(3)
+z1+0.25)*cos(x(4))*((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))
*(0.45-(x(3)+z1+0.25))/0.45/Im))...
/cos(x(4));

Dx(10)= ((-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4))
-(gamma*(x(9))+k*(x(3)))*sin(x(4))+m*g*sin(x(4)))/m-(((gamma*(x(9))
+k*(x(3)))*sin(x(4))-gamma1*x(7)...
+(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*cos(x(4)))/M_T)-
(((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))/0.45*sin(x(4))...
-(gamma*(x(9))+k*(x(3)))*cos(x(4)))/m-(((gamma*(x(9))+k*(x(3)))
*cos(x(4))...
-gamma1*x(8)-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))
/0.45*sin(x(4)))/M_T)+cos(x(4))*x(9)*x(10)...
+(x(9)*sin(x(4))-(x(3)+z1+0.25)*cos(x(4))*x(10))*x(10)+
(x(3)+z1+0.25)*cos(x(4))*((eta*(x(4)+x(5))+eta3*(x(10)+x(11)))
*(0.45-(x(3)+z1+0.25))/0.45/Im))...
/cos(x(4))*sin(x(4))-cos(x(4))*x(9)*x(10)...
-(x(9)*cos(x(4))-(x(3)+z1+0.25)*sin(x(4))*x(10))*x(10))/((x(3)+z1+0.25)*cos(x(4)));

Dx(11)=(-M*g*d*sin(x(5))-(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))
-eta1*x(5)-eta2*x(11)+L*cos(omiga*t)+...
M*d*cos(x(5))*(((gamma*(x(9))+k*(x(3)))*sin(x(4))
-gamma1*x(7)+(eta*(x(4)+x(5))+eta3*(x(10)+x(11)))...
/0.45*cos(x(4)))/M_T)+M*d*sin(x(5))*(((gamma*(x(9))
+k*(x(3)))*cos(x(4))-gamma1*x(8)-eta*(x(4)...
+x(5))/0.45*sin(x(4))+f*cos(omiga*t)-rho*g*S*x(2))/M_T)
+m*g*(x(3)+z1+0.25)*sin(x(4)))/(IM+I1);

end
end

function y=integral(x1,x2,x3,delta_t)
y=delta_t*(x1+4*x2+x3)/3;
end

function y=func(x1,x2)
global gamma eta3
gamma=x1;
eta3=x2;
y=1./solve_func();
end

```

---