

高压油管压力控制的数学模型

摘要 对高压油管中的燃油注入和喷出的数学模型的建立，是研究和改善燃油发动机工作状况的必要前提。在本问题中，我们考察了一个由高压油泵、高压油管、喷油嘴和减压阀等部件组成的简单高压油路系统。我们将复杂的高压油路系统建立一个形式简单而统一的数学形式

$$\frac{dp}{Edt} = \frac{Q_{in} - Q_{out}}{V}$$

并从这个简洁优美的式子，推导出关于这个高压油路的控制理论依据。

关键词 高压油管 压力控制 数学模型

1 问题重述与分析

对高压油管中的燃油注入和喷出的数学模型的建立，是研究和改善燃油发动机工作状况的必要前提。在本问题中，我们考察了一个由高压油泵、高压油管、喷油嘴和减压阀等部件组成的简单高压油路系统。

我们首先考虑了这个高压油路系统最简单的情况，即油泵可以稳定注入恒压的高压油，喷嘴以简单函数方式喷出油料；随后，我们把高压油泵的实际工作情况考虑进来，同时考虑了喷油嘴工作的本质，模型变得复杂起来；最后，我们考虑了实践中常见的多喷嘴油路的工作情况，并引入减压阀来控制高压油管内部的压力变化。问题不断复杂化，并不断接近高压油路的真实工作状态，实用性不断提高。

2 模型设计与计算

2.1 基本模型设计

我们的目标是解出高压油管内部压力 p 与时间 t 的关系, 记作

$$p = p(t) \quad (1)$$

由条件可见，高压油管的内部体积 V 不变，由 $\rho = \frac{m}{V}$ 可见，对高压油管内燃油密度造成影响的仅有高压油管内的燃油质量，用微分方程来表达就是

$$d\rho = d\left(\frac{m}{V}\right) = \frac{dm}{V} \quad (2)$$

首先我们考虑 $d\rho$ 和 p 的关系。

根据注 1 给出的条件，燃油的压力与其密度具有正相关关系，其关系可以用微分方程

$$\begin{cases} dp = \frac{E}{\rho} d\rho \\ p_0 = 100 \text{ MPa}, \rho_0 = 0.850 \text{ mg mm}^{-3} \\ E = E(p) \end{cases} \quad (3)$$

描述，其中 $E(p)$ 由附件 3 的数据给出，对附件 3 的数据用 MATLAB 作图并拟合得到方程，如图 Figure 1

$$E(p) = 0.0001p^3 - 0.001082p^2 + 5.474p + 1532 \quad (r^2 = 1.0000) \quad (4)$$

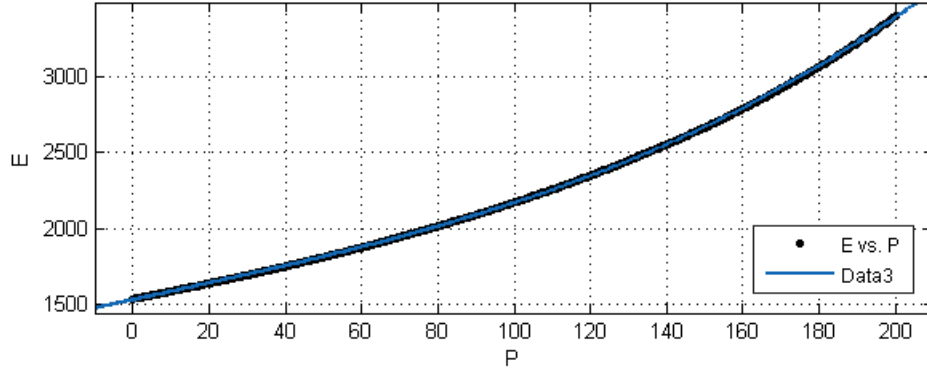


Figure 1. 燃油弹性模量 E 与压力 p 的非线性拟合结果

由以上的方程我们可以解出关系

$$\rho(p) = 0.850e^{\int_{100}^p \frac{dp}{E(p)}} \quad (5)$$

其次，我们再来考虑 dm 和 p 与 t 的关系。

dm 由进入和喷出两部分组成。喷出部分对 dm 的贡献是时间的函数。单位时间喷出的油量（体积）用函数 $Q_{out}(t)$ 表示，它的解析式在各个小题中有所不同，并且分段可微，所以喷出端造成的 dm 可以表述成

$$dm_{out} = -\rho Q_{out}(t)dt \quad (6)$$

进入部分对 dm 的贡献也是时间的函数。这个函数含有参数 T ，在上面提及过，它描述了单向阀的开启时间。单位时间进入的油量（体积）用函数 $Q_{in}(t)$ 表示，它的解析式在各个小题中也有所不同，并且分段可微，所以喷出端造成的 dm 可以表述成

$$dm_{in} = \rho Q_{in}(t)dt \quad (7)$$

综上，联立方程 2、5、6、7，我们可以列出方程

$$\frac{dp}{Edt} = \frac{Q_{in} - Q_{out}}{V} \quad (8)$$

方程 8 就是我们对本题建立的基本数学模型。接下来的部分中，我们将根据 $Q_{in}(t)$ 和 $Q_{out}(t)$ 的具体形式，对整个体系最佳 T 值的选择进行讨论。

2.2 问题 1

在问题 1 中 $Q_{out}(t)$ 的形式比较简单，它是由题目中图 Figure 2 给出的

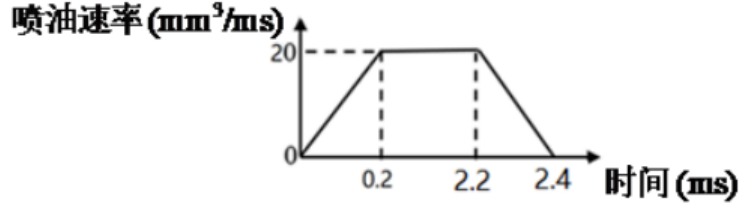


Figure 2. 问题 1 中的喷油速率

写成函数为

$$Q_{out}(t) = \begin{cases} 100(t - 100k) & 100k \leq t \leq 100k + 0.2, k \in \mathbb{N} \\ 20 & 100k + 0.2 \leq t \leq 100k + 2.2, k \in \mathbb{N} \\ -100(t - 100k) + 240 & 100k + 2.2 \leq t \leq 100k + 2.4, k \in \mathbb{N} \end{cases} \quad (9)$$

在问题 1 中 $Q_{in}(t)$ 的形式也比较简单，它包括一个参数 T ，描述力单向阀每次开启的时长。我们定义 $0-1$ 变量 $\lambda = \lambda(t)$ 来描述供油处入口的截面积，它的形式为

$$\lambda = \begin{cases} 1 & k(T + 10) \leq t \leq k(T + 10) + T, k \in \mathbb{N} \\ 0 & k(T + 10) + T \leq t \leq (k + 1)(T + 10), k \in \mathbb{N} \end{cases} \quad (10)$$

代入题目中给出的流量公式可得

$$Q_{in}(t) = \lambda C A \sqrt{\frac{2(p_h - p)}{\rho_h}} \quad (11)$$

将 $Q_{in}(t)$ 和 $Q_{out}(t)$ 的具体形式代入方程 8，发现所得方程是一阶常微分方程，所以尝试用差分法通过计算机进行数值解微分方程，取 $\Delta t = 0.01$ ms，用 C++ 程序进行计算并用 Python 绘制出对应于不同 T 取值 $p-t$ 变化图，以及差分过程中选用不同 Δt 对结果的影响，发现两条曲线非常接近，可以认为结果收敛可靠，如图 Figure 3

$$\frac{p_{i+1} - p_i}{E(p_i)\Delta t} = \frac{1}{V_0} \left(\lambda C A \sqrt{\frac{2(p_h - p_i)}{\rho_h}} - Q_{out}(t_i) \right) \quad (12)$$

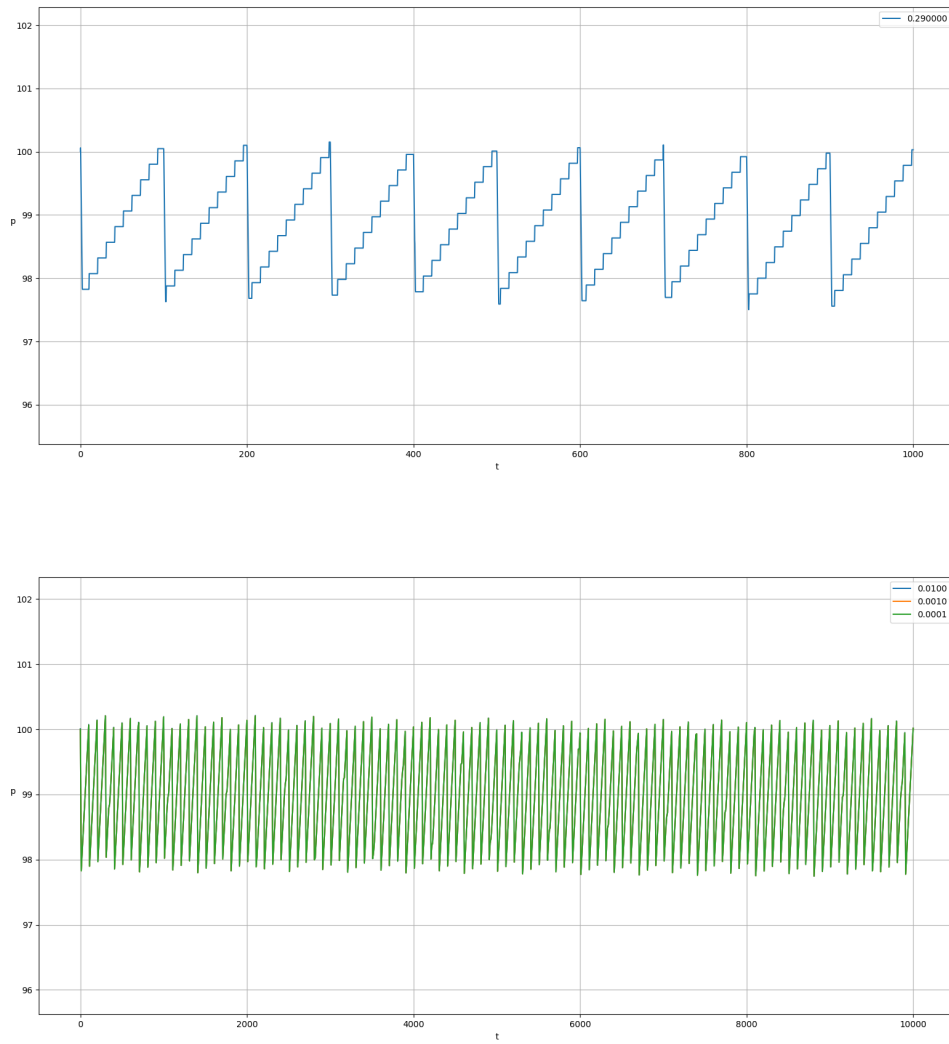


Figure 3. 差分方程数值模拟结果

我们使用二分法对一定范围内的 T 进行搜索，用经过一定时间段后高压油管内油压力的变化 Δp 来评价“压力稳定在100 MPa”的好坏，搜索出使压力变化最小时的单向阀开启时长 $T = 0.29 \text{ ms}$ 。

同时，在数值模拟的过程中我们发现，只要 T 处于一定的范围内，在经过较长时间后，高压油管内的压力 p 都可以稳定地收敛到一个区间内，如图 Figure 4，它描述了取不同的 T 值之后200 000 ms内高压油管内的压力的变化趋势和范围。利用数值模拟可以求出这个区间大致为 $[0.292, 0.298] \text{ ms}$ 。

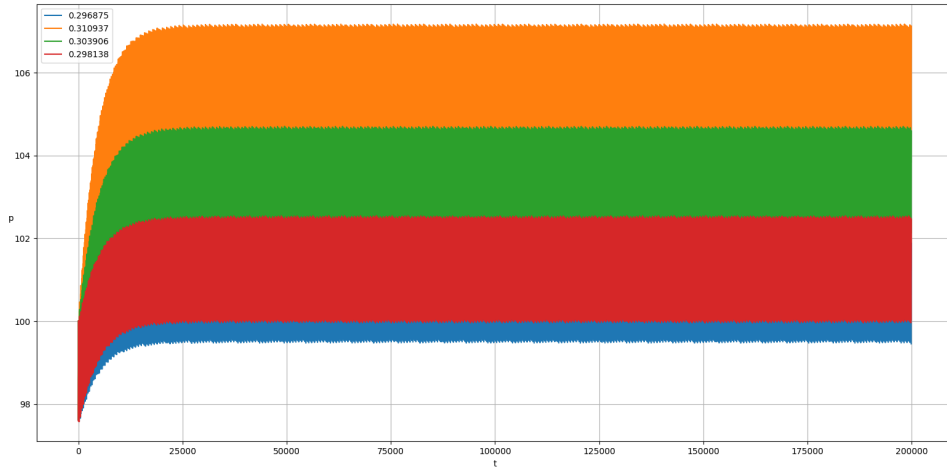


Figure 4. 200 000 ms内不同喷油间隔值下高压油管压力与时间的关系

同时，在前面的计算中我们只考虑了 $t = 0$ 时刻同时开始喷油和开始供油的情况，实际问题中，供油周期和喷油周期可能存在着相位差，但是通过数值模拟发现相位差造成的影响很小，在较长时间段内可以忽略，如图 Figure 5。

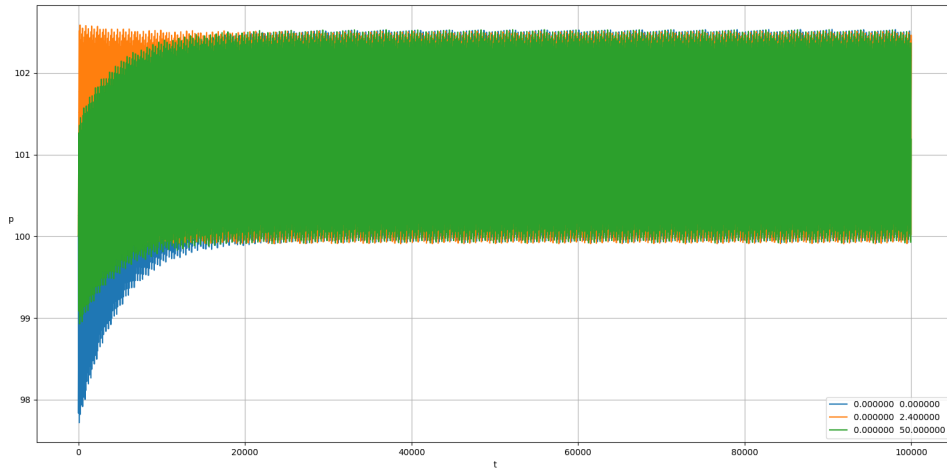


Figure 5. 相位差对 $p - t$ 关系的影响

对于将油管内的压力从100 MPa提高到150 MPa的情况，我们可以用同样的差分方程，分别模拟起始压力与2 s、5 s和10 s后的压力差别为50 MPa时的情况，同样通过二分法可以找出最佳的精确到千分位的单向阀开启时长 $T_{2s} = 0.89 \text{ ms}$ 、 $T_{5s} = 0.71 \text{ ms}$ 和 $T_{10s} = 0.70 \text{ ms}$ 。

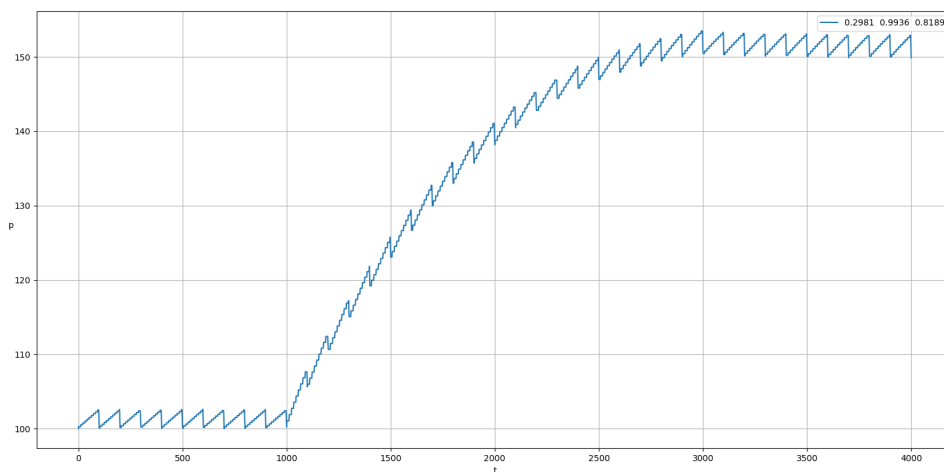


Figure 6. 调整过程中的压力周期变化

图 Figure 7描述了相位差对调整过程的影响，在调节完成后依旧可以收敛。

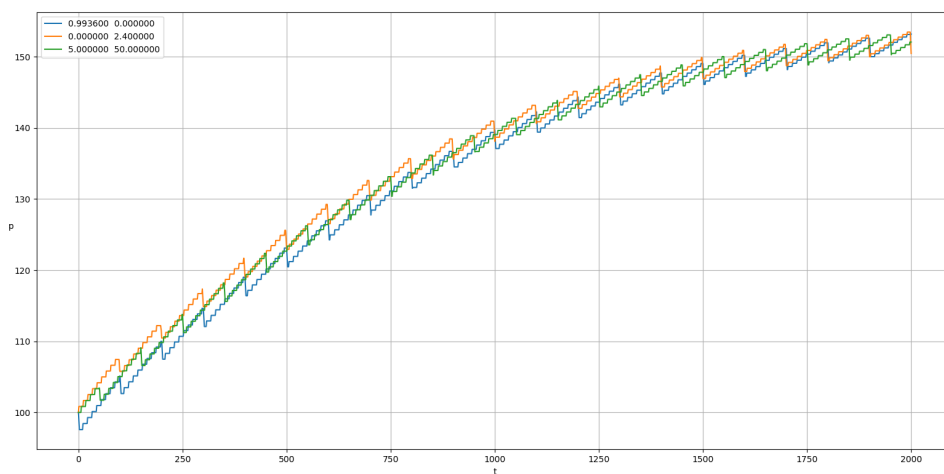


Figure 7. 相位差对调节过程的影响

而进过该段时间后，维持150 MPa所需要的单向阀开启时长为 $T_{150} = 0.70 \text{ ms}$ 。图 Figure 6展示了其中一个调整过程中的压力对时间的变化图像。

问题1 结论 综上所述，将高压油管内的压力尽可能稳定在100 MPa左右，应该设置单向阀每次开启0.29 ms。如果要将高压油管内的压力从100 MPa增加到150 MPa, 且分别经过约2 s、5 s和10 s的调整之后稳定在150 MPa时的

情况,对应时长内单向阀应分别先调整为开启 $T_{2s} = 0.89 \text{ ms}$ 、 $T_{5s} = 0.71 \text{ ms}$ 和 $T_{10s} = 0.70 \text{ ms}$, 随后维持开启时长为 $T_{150} = 0.70 \text{ ms}$, 以维持 150 MPa 。

3 问题 2

问题 2 中 $Q_{out}(t)$ 的形式比较复杂。

首先我们定义等效面积函数 $B(t)$, 用来描述针阀运动过程中喷油嘴喷孔的等效面积。由简单的几何计算可知, 它的形式如下。

$$B(t) = \pi \left(\left(\frac{d_B}{2} + h(t) \tan 9^\circ \right)^2 - \left(\frac{d_B}{2} \right)^2 \right) \quad (13)$$

但是, 当针阀上升到一定高度时, 等效面积函数值会超过喷嘴的面积, 此时喷油能力转而由喷嘴的面积限制, 为了不改变 $B(t)$ 的表达形式, 我们定义了等效升程函数 $h(t)$, 它表示针阀的等效升程, 由题目附件 2 给出的数据拟合, 并按照前述进行修正得到, 其图形和拟合结果如下

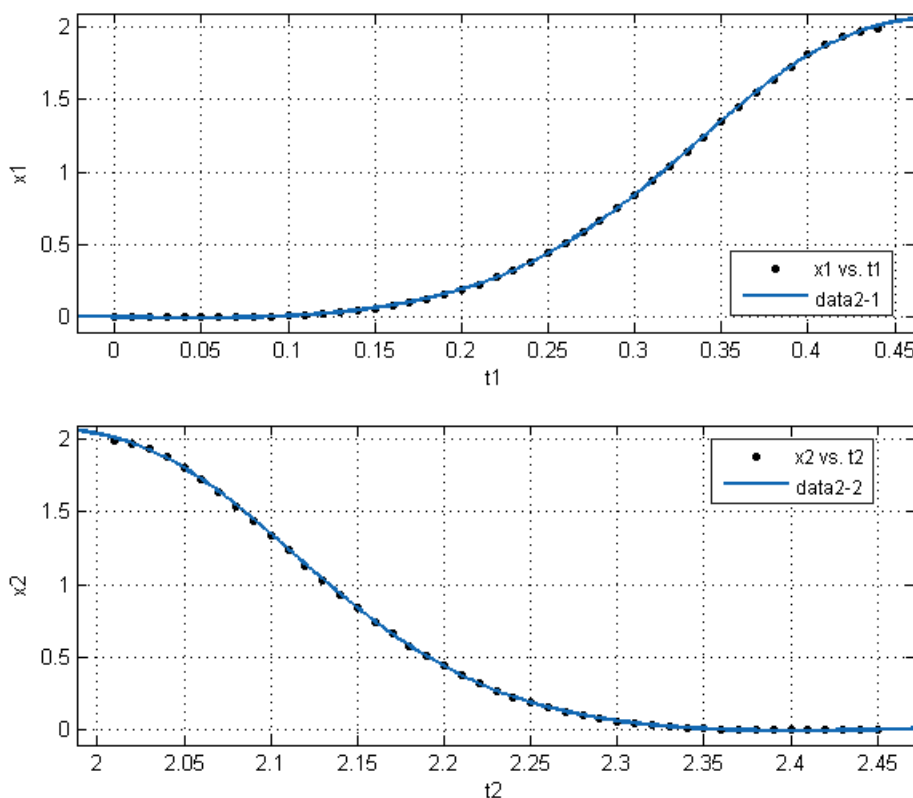


Figure 8. 针阀升程

$$h(t) = \begin{cases} \frac{0.5342(t-100k)^2 - 0.04835(t-100k) + 0.000726}{(t-100k)^2 - 0.7362(t-100k) + 0.1716} & 100k \leq t \leq 100k + 0.3309, k \in \mathbb{N} \\ 1.1532 & 100k \leq t + 0.3309 \leq 100k + 2.1213, k \in \mathbb{N} \\ \frac{0.5358(t-100k)^2 - 2.576(t-100k) + 3.096}{(t-100k)^2 - 4.163(t-100k) + 4.368} & 100k + 2.1213 \leq t \leq 100k + 2.45, k \in \mathbb{N} \\ 0 & 100k + 2.45 \leq t \leq 100(k+1), k \in \mathbb{N} \end{cases} \quad (14)$$

由于高压油管内的油压远高于大气压，所以我们可以忽略喷油嘴外的压强，即 $\Delta p \approx p$ ，由此 $Q_{out}(t)$ 就可以被写成

$$Q_{out}(t) = CB\sqrt{\frac{2p}{\rho}} \quad (15)$$

再来看 Q_{in} 的表达形式，它形式上与上一题类似，如方程 16（以下标为 h 的项均为油泵油缸的性质）

$$Q_{in}(t) = \delta CA\sqrt{\frac{2(p_h - p)}{\rho_h}} \quad (16)$$

其中 0-1 变量 δ 定义如下

$$\delta = \begin{cases} 1 & p_h \geq p \\ 0 & p_h < p \end{cases} \quad (17)$$

下面我们来求油缸内压力 p_h 的表达式。油缸的体积可以表示为时间 t 的周期函数

$$V_h(t) = 20 + (R_{up} - R(\theta(t)))\pi \left(\frac{5}{2}\right)^2 \quad (18)$$

其中 R_{up} 为凸轮位于上止点时的极径，由题目给出，此时缸内压力为 0.5 MPa， $R(\theta)$ 的表达式由题目附件 1 给出， $\theta(t)$ 的表达式如下，为了计算方便，我们把时刻 $t = 0$ 的相位设置为下止点，也就是 $\theta = \pi$ 。

$$\theta(t) = \omega t + \pi - 2k\pi, k \in \mathbb{N}, \theta(t) \geq 0 \quad (19)$$

当油泵活塞运行到下止点时，油泵吸慢了油，并不再把油返回低压油路，所以我们可以给油泵位于下止点时直接给定一个初值 $p_0 = 0.5 \text{ MPa}$ ，对每一个周期都是这样。

考虑油泵油缸的质量守恒，可以得到下面的方程。

$$\frac{dm_h}{dt} = -Q_{in}\rho \quad (20)$$

又由密度的定义 $m_h = \rho_h V_h$ 微分得到

$$dm_h = \rho_h dV_h + V_h d\rho_h \quad (21)$$

将方程 3、20、21 联立，加入初值条件，可以整理得到

$$\begin{aligned} \frac{dp_h}{E(p_h)dt} &= \frac{1}{V_h} \left(-Q_{in} \frac{\rho}{\rho_h} - \frac{dV_h}{dt} \right) \\ p_h\left(\frac{2k\pi}{\omega}\right) &= 0.5, k \in \mathbb{N} \end{aligned} \quad (22)$$

在高压油泵供油的过程中我们可以用方程 22 去描述它的情况。这个方程的形式和刚才对高压油管的建模模型——方程 8 形式很相似，也就是我们可以把高压油泵的油缸视作另一个高压油管，活塞对它的压缩可以视作 Q_{in} ，而向高压油管供油可以视作 Q_{out} ，当然此时 V 是变量，所以不可以直接代入。

同时，根据方程 8 我们可以写出高压油管的方程

$$\frac{dp}{Edt} = \frac{Q_{in} - Q_{out}}{V} \quad (23)$$

联立方程 23、22 并使用和问题 1 中相似的差分方程计算

$$\begin{aligned} \frac{p_{h\ i+1} - p_{h\ i}}{E(p_{h\ i})\Delta t} &= \frac{1}{V_h(t_i)} \left(-\delta C A \sqrt{\frac{2(p_h - p_i)}{\rho_h}} \frac{\rho}{\rho_h} - \frac{dV_h}{dt} \right) \\ \frac{p_{i+1} - p_i}{E(p_i)\Delta t} &= \frac{1}{V_0} \left(\delta C A \sqrt{\frac{2(p_h - p_i)}{\rho_h}} - Q_{out}(t_i) \right) \\ p_h\left(\frac{2k\pi}{\omega}\right) &= 0.5, k \in \mathbb{N} \end{aligned} \quad (24)$$

我们使用二分法对一定范围内的 ω 进行搜索，用经过一定时间段后高压油管内油压力的变化 Δp 来评价“压力稳定在 100 MPa”的好坏，搜索出使压力变化最小时的 $\omega = 0.0152 \text{ rad ms}^{-1}$ 。对应的 $p - t$ 变化曲线如图 Figure 9。

下面的曲线则描述了二分搜索中不同 ω 的选取造成的 $p - t$ 变化曲线。

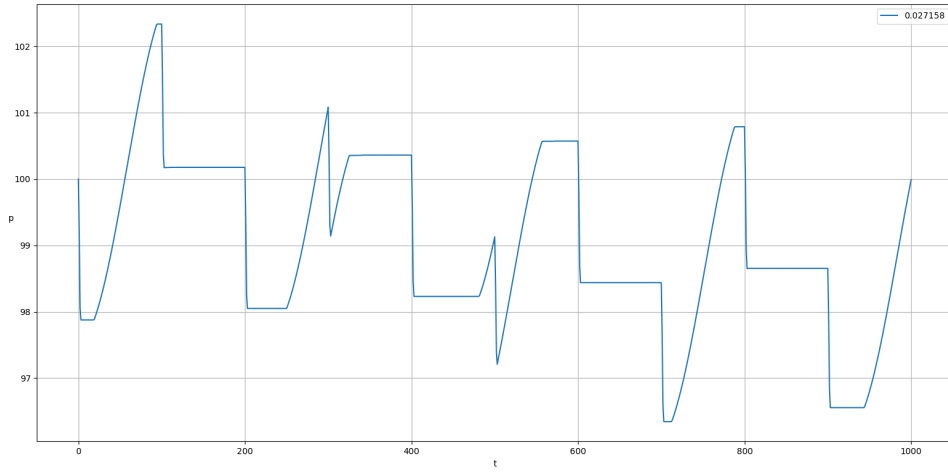


Figure 9. 问题 2 中高压油管内压力随着时间的变化图

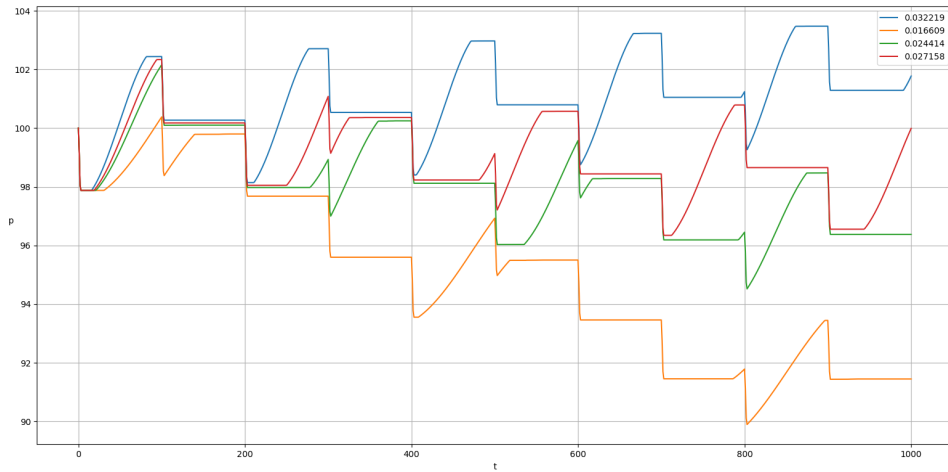


Figure 10. 二分搜索中不同 ω 的选取造成的 $p-t$ 变化

问题 2 结论 通过数值模拟和二分搜索，我们给出凸轮的角速度 $\omega = 0.028 \text{ rad ms}^{-1}$ 时，高压油管内的压力尽量稳定在 100 MPa 左右。

4 问题 3

从问题 2 给出的图 Figure 9 可以看到，造成压力波动的主要因素是喷油和注油过程。相比两个喷油嘴同时喷油，交替喷油可以减少高压油管中的压力波动。我们可以通过数值模拟验证这一点。

增加一个喷油嘴只改变了问题 2 中 $Q_{out}(t)$ 的具体形式，对于交替喷油模型， $Q_{out}(t)$ 的具体形式如下。

$$Q_{out}(t) = CB\sqrt{\frac{2p}{\rho}} \quad (25)$$

$$B(t) = \pi \left(\left(\frac{d_B}{2} + h(t) \tan 9^\circ \right)^2 - \left(\frac{d_B}{2} \right)^2 \right) \quad (26)$$

$$h(t) = \begin{cases} \frac{0.5342(t-50k)^2 - 0.04835(t-50k) + 0.000726}{(t-50k)^2 - 0.7362(t-50k) + 0.1716} & 50k \leq t \leq 50k + 0.3309, k \in \mathbb{N} \\ 1.1532 & 50k \leq t + 0.3309 \leq 50k + 2.1213, k \in \mathbb{N} \\ \frac{0.5358(t-50k)^2 - 2.576(t-50k) + 3.096}{(t-50k)^2 - 4.163(t-50k) + 4.368} & 50k + 2.1213 \leq t \leq 50k + 2.45, k \in \mathbb{N} \\ 0 & 50k + 2.45 \leq t \leq 50(k+1), k \in \mathbb{N} \end{cases} \quad (27)$$

将这个结果代回方程 23 就可以进行与问题 1 相同的数值模拟，解出此时最合适的凸轮角速度为 $\omega = 0.053 \text{ rad ms}^{-1}$ 。图 Figure 11 中的红色线描述了此时的 $p-t$ 变化情况，其他色线描述了二分逼近的结果。

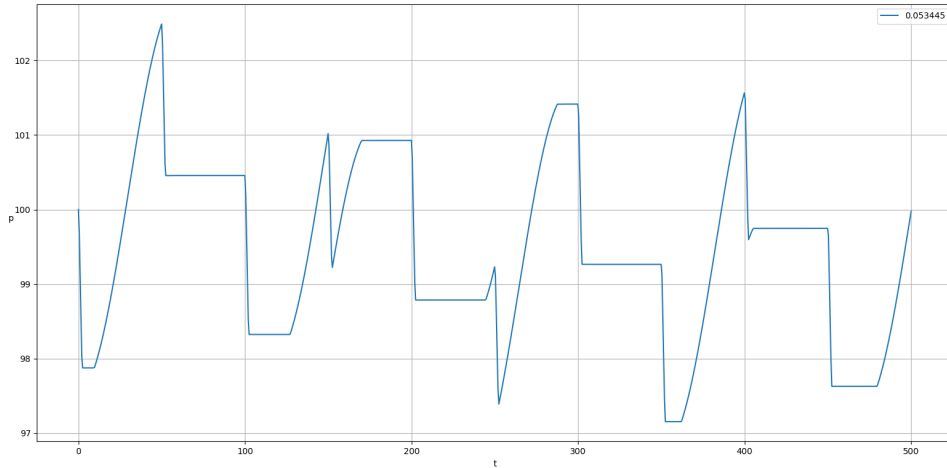


Figure 11. 增加一个喷油嘴之后的高压油管内压力与时间变化

减压阀可以控制体系内部压力不高于某一定值，该定值即为减压阀的开放阈值，记作 p_0 ，通过设定 p_0 即可控制减压阀。

5 总结

在本题中，我们从简单到复杂考虑了高压油路系统的三种模型，并把他们统一到

$$\frac{dp}{Edt} = \frac{Q_{in} - Q_{out}}{V}$$

这一简洁的形式，利用具体条件下不同

6 附录

我们使用的软件与工具包括 MATLAB、Python、C++、 \LaTeX 、git、github 和 cnki.net。

6.1 MATLAB 代码

我们使用 MATLAB 进行拟合运算，没有输入交互代码，只用鼠标操作。

6.2 Python 代码

我们使用 Python 绘制本文中几乎大多数插图，使用的程序代码如下

```
1 import matplotlib.pyplot as plt
2 f=open('data.out')
3 r=f.read()
4 r=r.split('\n\n')
5 r.pop()
6 x=[];y=[]
7 for i in range(len(r)):
8     x.append([]);y.append([])
9     r[i]=r[i].split('\n')
10    for j in range(len(r[i])-1):
11        r[i][j]=r[i][j].split(' ')
12        x[i].append(float(r[i][j][0]))
13        y[i].append(float(r[i][j][1]))
14    plt.plot(x[i],y[i],label=str(r[i][len(r[i])-1]))
15    print(str(r[i][len(r[i])-1]))
16 plt.xlabel('t');plt.ylabel('p',rotation=0)
17 plt.legend()
18 plt.grid()
19 plt.show()
```

6.3 C++ 代码

本题中所有的二分法优化都是通过不断改动以下程序的参数实现的

```
1 #include <cstdio>
2 #include <cmath>
3 #define TimeLimit 1000
4 #define GAP 0.01
```

```

5      #define INIT 100
6      #define V (3.1416*500*25)
7      #define C 0.85
8      #define Phigh 160
9      #define Precision 0.001
10     #define DIF 0
11     #define phinit 0.5
12     double omega = 20;
13     // #define INF
14     /*//this part for question 1
15     double T = 0.3;
16     double A(double t)
17     {
18         double temp = t-(double)((int)(t/(T+10)))*(T+10);
19         if(temp <= T){
20             return (0.49*3.1416);
21         }
22         else return 0;
23     }
24     double Qout(double t)
25     {
26         double temp = t-(double)((int)(t/(100)))*(100);
27         // printf("%f",temp);
28         if(temp <= 0.2)
29         {
30             return 100*temp;
31         }
32         else if(temp >0.2 && temp < 2.2)
33         {
34             return 20;
35         }
36         else if(temp >= 2.2 && temp <= 2.4)
37         {
38             return temp*(-100)+240;
39         }
40         else return 0;
41     }
42     double p()
43     {

```

```

44     double t = 0;
45     double press = INIT;
46     // printf("%f\n%f\n", t, press);
47     while (t < TimeLimit)
48     {
49         // printf("press=%f\t", press);
50         if (press > Phigh) press = Phigh;
51         press = press + ((C*A(t)*sqrt(2*(Phigh-
52         press)/0.87113) - Qout(t))/V)*GAP
53         *(0.0001*press*press*press - 0.001082
54         *press*press + 5.474*press + 1532);
55         t += GAP;
56         // printf("%f\n%f\n", t, press);
57     }
58     return (press - INIT);
59 }
60 */
61 double rho(double p){
62     double sum=0;
63     if (p>100){
64         for (double i=100; i<=p; i+=0.001){
65             sum+=0.001/(0.0001*i*i*i - 0.001082*i
66             *i + 5.474*i + 1532);
67         }
68         return 0.85*pow(2.718281828459, sum);
69     }
70     else if (p<100){
71         for (double i=100; i>=p; i-=0.001){
72             sum-=0.001/(0.0001*i*i*i - 0.001082*i
73             *i + 5.474*i + 1532);
74         }
75         return 0.85*pow(2.718281828459, sum);
76     }
77     return 0.85;
78 }
79 inline double E(double p)
80 {
81     return (0.0001*p*p*p - 0.001082*p*p + 5.474*p
82     + 1532);

```

```

83     }
84     inline int delta(double ph, double p)
85     {
86         return (ph >= p);
87     }
88     inline double Theta(double t)
89     {
90         return (omege*t+3.14-6.28*((int)((omege*t
91         +3.14)/6.28)));
92     }
93     inline double R(double theta)
94     {
95         return (4.826+2.413*sin(theta+1.5708));
96     }
97     double Vh(double t)
98     {
99         return (20+(7.239-R(Theta(t)))*3.1416*2.5
100         *2.5);
101     }
102     double h(double t)
103     {
104         double temp = t - ((int)(t/50))*50;
105         if(temp < 0.3309)
106         {
107             return (0.5342*temp*temp-0.04835*temp+
108             0.000726)/(temp*temp-0.7362*temp+0.1716);
109         }
110         else if(temp >= 0.3309 && temp <= 2.1213)
111         {
112             return 1.1532;
113         }
114         else if(temp > 2.1213 && temp < 2.45)
115         {
116             return (0.5358*temp*temp-2.576*temp+
117             3.096)/(temp*temp-4.163*temp+4.368);
118         }
119         else
120         {
121             return 0;

```



```

122     }
123 }
124 double B(double t)
125 {
126     return 3.1416*((1.25+h(t)*0.1584)*(1.25
127     +h(t)*0.1584) - 1.25*1.25);
128 }
129 double Qout(double t, double p)
130 {
131     return (0.85*B(t)*sqrt(2*p/rho(p)));
132 }
133 double p() //for question 2
134 {
135     double t = 0;
136     double press = INIT;
137     double pressh = phinit;
138     // printf("%f\n%f\n", t, press);
139     while (t < TimeLimit)
140     {
141         // printf("press=%f\t", press);
142         t += GAP;
143         if(omege*t/6.2832 - (int)(omege*t
144         /6.2832) < 0.001 && omege*t/6.2832
145         -(int)(omege*t/6.2832) > -0.001)
146         {
147             pressh = phinit;
148         }
149         // printf("Qout = %f\t",
150         Qout(t, press)); // 喷油
151         if(delta(pressh, press))
152         {
153             // printf("inittime = %f\tpressh
154             = %f\tpress = %f\n", t, pressh, press);
155             pressh = pressh + (GAP*E(pressh)
156             /Vh(t))*(-0.85*3.1416*0.49*sqrt(2
157             *(pressh - press)/rho(pressh))
158             *(rho(press)/rho(pressh)) - (Vh(t)
159             - Vh(t - GAP))/GAP);
160             if(delta(pressh, press))

```

```

161         {
162             press = press + (GAP*E(press)/V)
163             *(0.85*3.1416*0.49*sqrt(2*(pressh
164             - press)/rho(pressh)) - Qout(t
165             ,press));
166         }
167         else
168         {
169             press = press + (GAP*E(press)/V)
170             *( - Qout(t,press));
171         }
172     }
173     else
174     {
175         pressh = pressh + (GAP*E(pressh)
176         /Vh(t))* ( - (Vh(t) - Vh(t - GAP))/GAP);
177         press = press + (GAP*E(press)/V)
178         *( - Qout(t,press));
179     }
180     // printf("time = %f\tpressh = %f\tpress
181     = %f\n",t,pressh,press);
182 }
183 return (press - INIT);
184 }
185 void binarysearch(double start , double end)
186 {
187     printf("search[%f,%f]\t",start ,end);
188     if (end - start < Precision)
189     {
190         printf("found the solution : %f",end);
191         return;
192     }
193     omege = start + (end - start)/2;
194     double mid = p();
195     printf("mid p=%f\n",mid);
196     if(mid - DIF > 0){
197         binarysearch(start , start + (end - start)/2);
198     }
199     else

```

```

200         {
201             binarysearch(start + (end - start)/2, end);
202         }
203     }
204     int main()
205     {
206
207         // freopen("dataof1.out", "w", stdout);
208         // printf("%f", (GAP*E(102)/V)*( - Qout(57,102)) );
209         binarysearch(0.03, 0.1);
210
211         return 0;
212     }

```

6.4 L^AT_EX 代码

本文用 L^AT_EX 排版，L^AT_EX 代码置于支持文件中。