



飞机纵向动态特性分析及飞行品质 改善

院（系）名称	航空科学与工程学院
专 业 名 称	飞行器设计
学 生 姓 名	董鑫
学 号	SY1705105
任 课 教 师	孙立国 董朝阳

2018 年 4 月

目录

第一章 飞机本体动态特性计算分析	3
1.1 飞机本体运动方程	3
1.2 传递函数	3
1.3 阶跃输入相应	4
1.4 频率特性分析	5
第二章 改善飞行品质的控制设计	7
2.1 纵向 SAS 控制设计	7
2.2 俯仰角速率 CAS 控制系统设计	13
第三章 纵向自动驾驶仪设计	17
3.1 自动驾驶仪简介	17
3.2 算例验证与分析	18
附录： 程序源码	21
1. 飞机本体运动特性计算	21
2. SAS 设计	23
3. CAS 设计	26
4. 自动驾驶仪	29

第一章 飞机本体动态特性计算分析

1.1 飞机本体运动方程

对于飞机纵向特征方程，向量形式为：

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\quad (1.1)$$

其中，状态向量 $x = [v_T \quad \alpha \quad \theta \quad q]^T$ ，控制向量 $u = \delta_e$ ，输出向量 $y = [\alpha \quad q]^T$

以海平面飞行高度，飞行速度 153m/s 的定常直线平飞状态作为基准状态，可以得到基于纵向运动状态下飞机小扰动运动方程为：

$$\begin{aligned}\begin{bmatrix} \dot{v}_T \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{q} \end{bmatrix} &= \begin{bmatrix} -0.01931 & 8.8158 & -32.17 & -0.575 \\ -0.00025 & -1.0189 & 0 & -0.9051 \\ 0 & 0 & 0 & 1 \\ 0 & 0.822 & 0 & -1.0774 \end{bmatrix} \begin{bmatrix} v \\ \alpha \\ \theta \\ q \end{bmatrix} + \begin{bmatrix} 0.17370 \\ -0.00215 \\ 0 \\ -0.17555 \end{bmatrix} \Delta\delta_e \\ y &= \begin{bmatrix} 0 & 57.29578 & 0 & 0 \\ 0 & 0 & 0 & 57.29578 \end{bmatrix} x\end{aligned}\quad (1.2)$$

同时得到纵向小扰动方程的特征根：

$$\begin{bmatrix} -1.9118 & & & \\ & -0.1507-0.1153i & & \\ & & -0.1507+0.1153i & \\ & & & 0.0976 \end{bmatrix}\quad (1.3)$$

可以看出飞机纵向模态有三个，其中一个模态实部为正。

1.2 传递函数

由：

$$G(s) = \frac{Y(s)}{U(s)} = C(sE - A)^{-1}B\quad (1.4)$$

可解得未加控制器时，飞机迎角和俯仰角速率对升降舵输入的传递函数为：

$$\begin{aligned}\frac{\alpha}{\delta_e} &= \frac{-0.1232s^3 - 9.2410s^2 - 0.1825s - 0.0822}{s^4 + 2.1156s^3 + 0.3963s^2 + 0.0091s - 0.0067} \\ \frac{q}{\delta_e} &= \frac{-10.0583s^3 - 10.5439s^2 - 0.2245s - 5.255 \times 10^{-14}}{s^4 + 2.1156s^3 + 0.3963s^2 + 0.0091s - 0.0067}\end{aligned}\quad (1.5)$$

1.3 阶跃输入相应

迎角对升降舵下偏一度的阶跃响应：

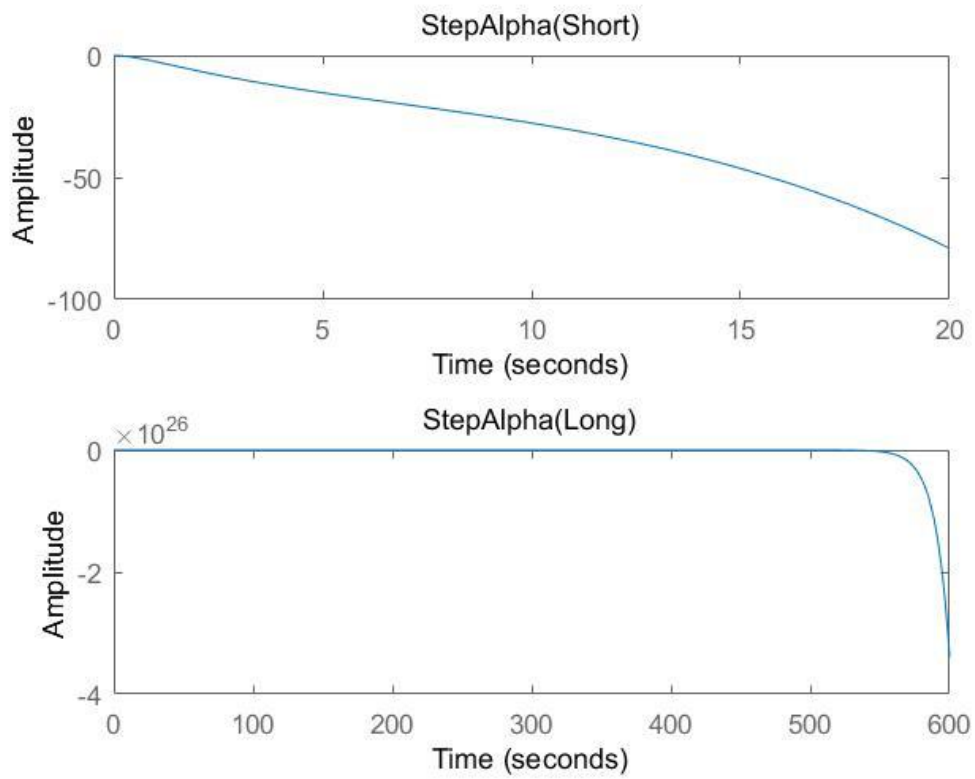


图 1.1 迎角单位阶跃响应

俯仰角速度对升降舵下偏一度的阶跃响应：

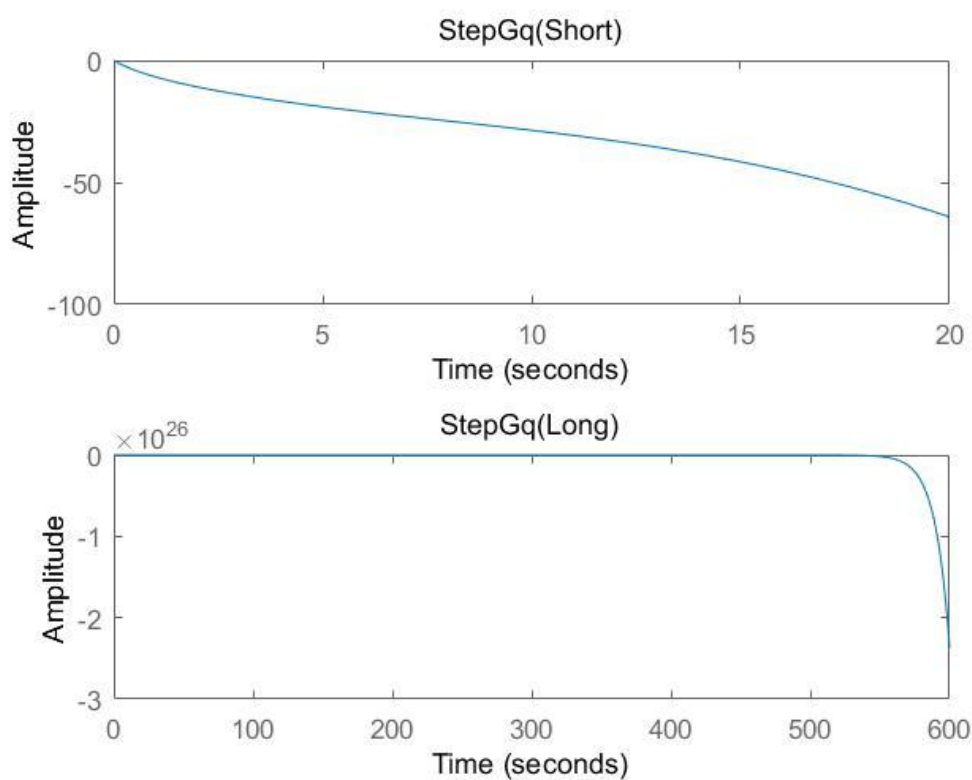


图 1.2 俯仰角速度单位阶跃响应

可以看出，飞机在无反馈控制的情况下，迎角和俯仰角速度对单位升降舵输入响应均是发散的，因此需要引入增稳系统对系统进行控制。

1.4 频率特性分析

对于升降舵偏角为输入，迎角和俯仰角速度作为输出，系统 Bode 图如下

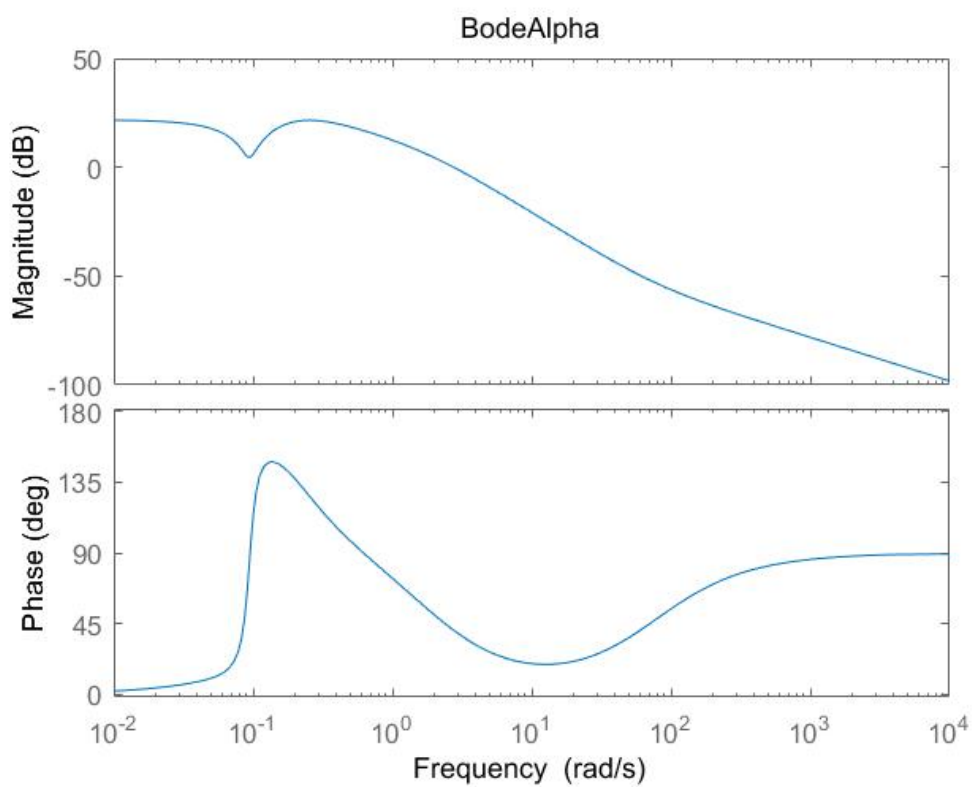


图 1.3 迎角 Bode 图

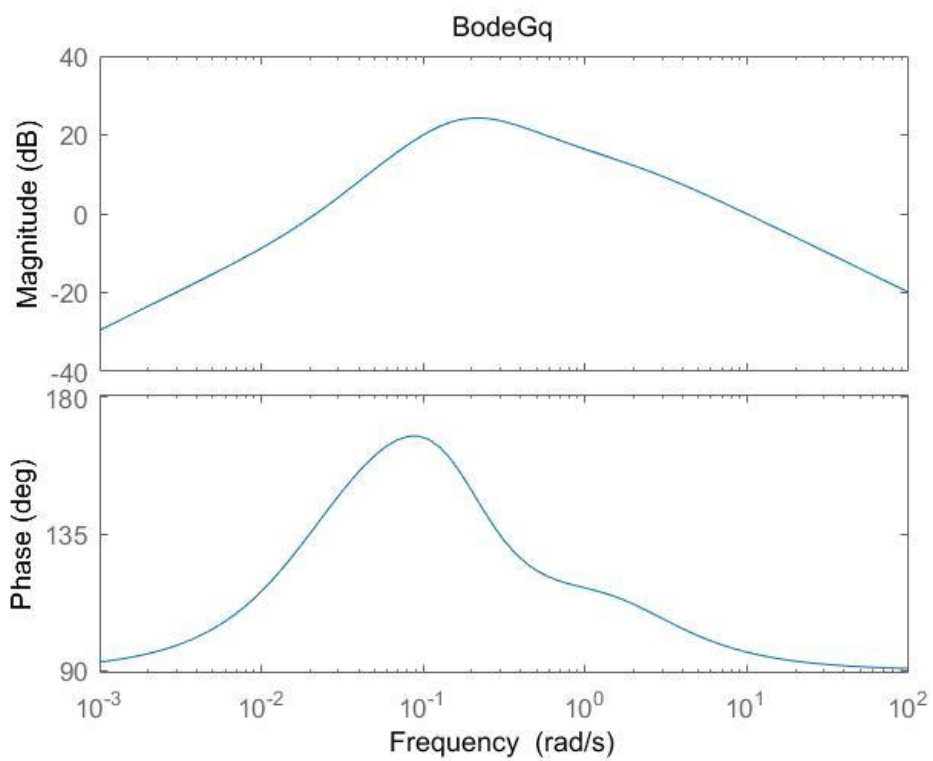


图 1.4 俯仰角速率 Bode 图

根据对数幅频特性稳定判据：一个反馈控制系统，其闭环特征方程正实部根个数为 Z ，可以根据开环传递函数右半 s 平面极点数 P 和开环对数幅频特性为正值的频率范围内，对数相频曲线对 -180° 线的正负穿越之差 $N = N_+ - N_-$ 决定： $Z = P - 2N = 0$ 闭环系统稳定；否则，闭环系统不稳定。由上述 Bode 图可以判断出系统是不稳定的。

第二章. 改善飞行品质的控制设计

2.1 纵向 SAS 控制设计

纵向 SAS 控制可为短周期模态提供合适的自然频率和阻尼。将迎角和俯仰角速率反馈到升降舵控制器可以调节系统的频率和阻尼，俯仰增稳系统对长周期模态影响较小，常用作外环反馈，其结构如图 2.1。

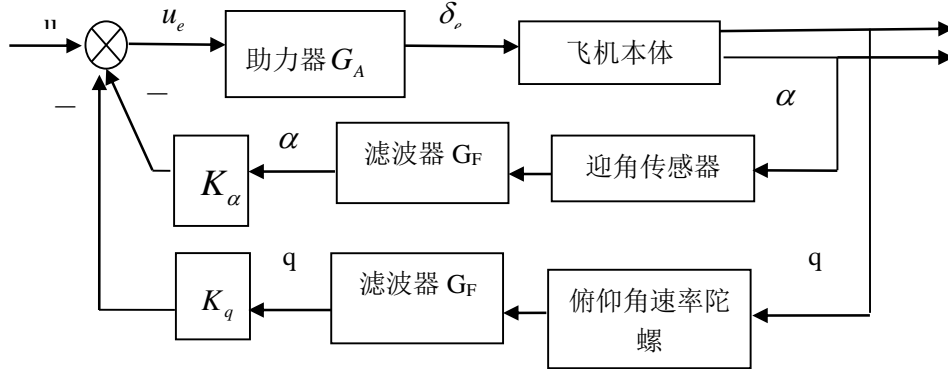


图 2.1 SAS 控制结构图

为了简化，忽略迎角传感器、俯仰角速率陀螺和俯仰角速度反馈通道的滤波器环节，将俯仰助力器和迎角滤波器简化为时间常数分别为 $\tau_\alpha = 1/20.2s$ 和 $\tau_F = 0.1s$ 的惯性环节，即

$$G_\alpha(s) = \frac{10}{s+10}, \quad G_\delta(s) = \frac{20.2}{s+20.2} \quad (1.6)$$

首先确定仅有迎角反馈时的迎角反馈增益 K_α ，由于增加了助力器环节和滤波器环节，此时系统状态方程状态变量取为 $x = [v_T \quad \alpha \quad \theta \quad q \quad x_a \quad x_F]^T$ ，由此可以得到该模型的飞机状态方程为：

$$\dot{x} = \begin{bmatrix} & & & & 0 \\ & A & & -B & 0 \\ & & & & 0 \\ & & & & 0 \\ 0 & 0 & 0 & 0 & -20.2 & 0 \\ 0 & 10.0 & 0 & 0 & 0 & -10.0 \end{bmatrix} \begin{bmatrix} v_T \\ \alpha \\ \theta \\ q \\ x_a \\ x_F \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 20.2 \\ 0 \end{bmatrix} \delta_e \quad (1.7)$$

$$y = \begin{bmatrix} \alpha \\ q \\ \alpha_F \end{bmatrix} = \begin{bmatrix} C & & 0 & 0 \\ & & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 57.29578 \end{bmatrix} x$$

可以得到输入量 u_e 与输出 α_F 间的开环传递函数：

$$\frac{2.4882(s+75)(s+10)(s^2+0.01964s+0.008892)}{(s+10)(s+20.2)(s+1.912)(s-0.09755)(s^2+0.3014s+0.03601)} \quad (1.8)$$

由此可以得到迎角反馈的根轨迹如图：

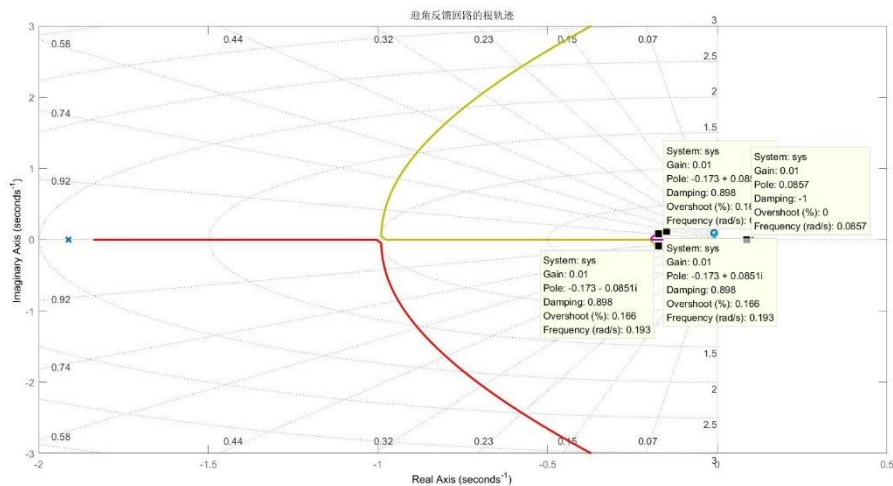


图 2.2 迎角根轨迹

在原点附近的放大如图：

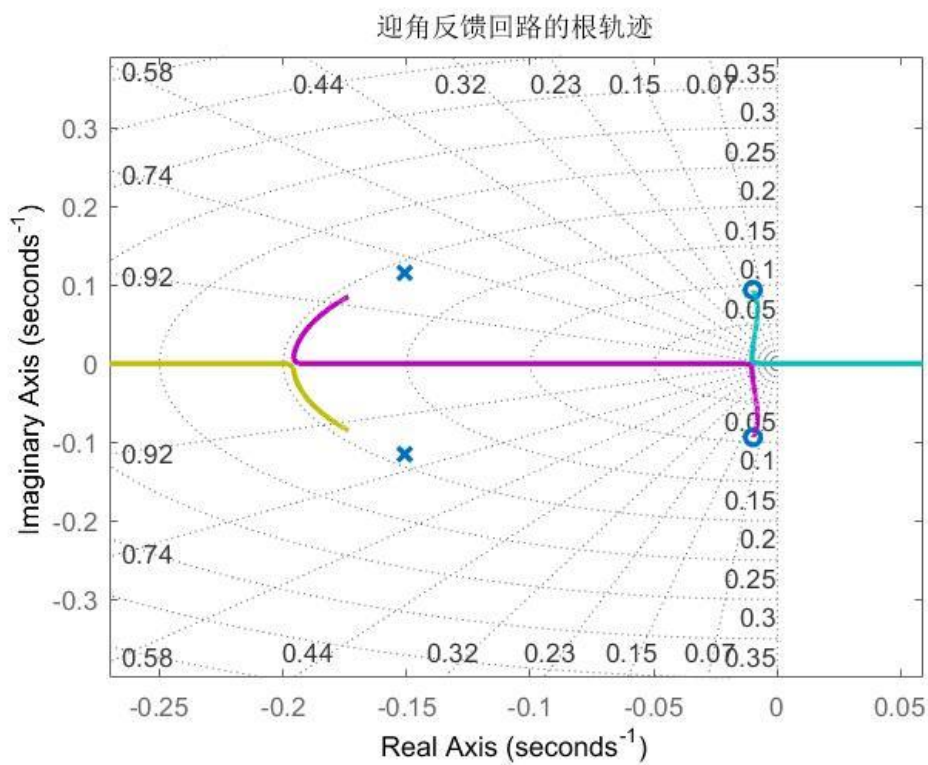
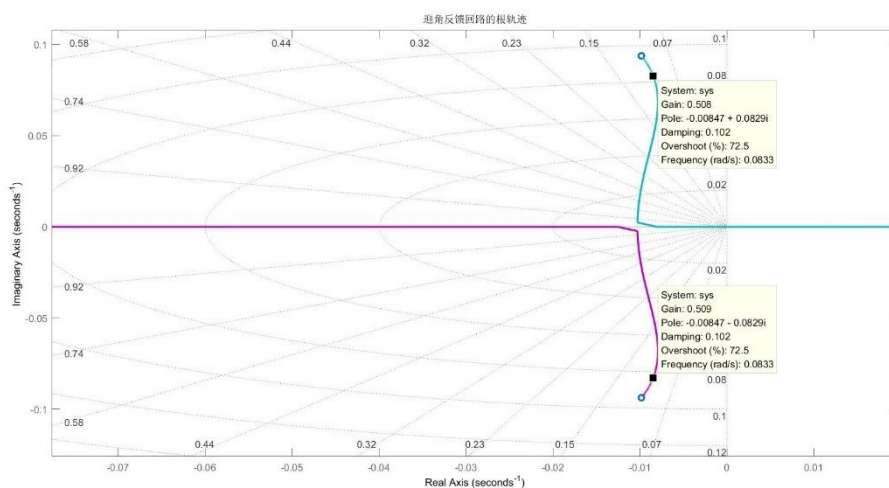


图 2.2 迎角根轨迹细节

由上图可以看到迎角传感器和助力器极点在左半平面，离原点距离较远，并向负无穷远和 $s = -75$ 的零点发展，故其对于其他模态的影响较小。同时迎角反馈使得长周期振荡模态逐渐变成了两个实根模态，其中一支向原点附近的复零点靠近，另一点与负实根的一支相会合。取 $K=0.5$ ，如图：



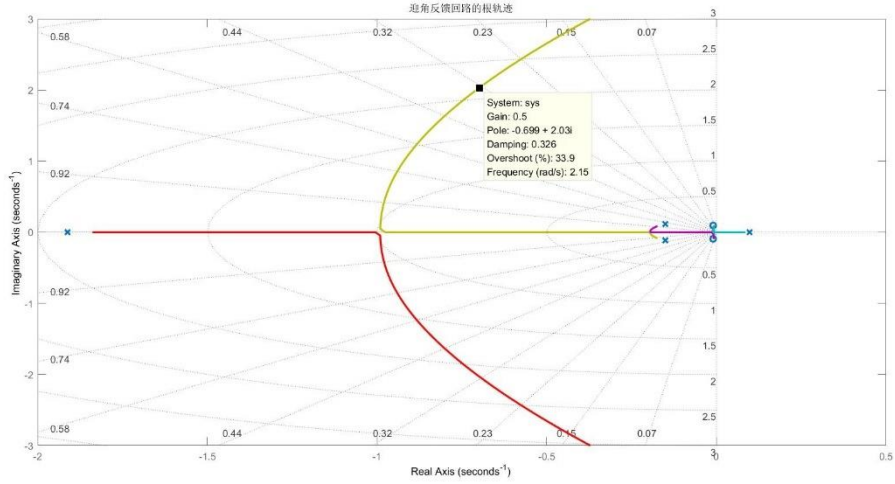


图 2.3 增益为 0.5 时的特征根位置

可以看出，采用 $K_\alpha = 0.5$ 的迎角反馈，使飞机闭环特征根位于实轴左侧，得到了稳定的纵向模态。此时求得方向舵偏角输入对俯仰角速度 q 的传递函数为：

$$\frac{q}{\delta_e} = \frac{203.18s(s+10.0)(s+1.027)(s+0.02174)}{(s+20.01)(s+10.89)(s^2+0.01691s+0.006911)(s^2+1.398s+4.608)} \quad (1.9)$$

引入迎角反馈后，系统状态方程为：

$$\dot{x} = \begin{bmatrix} A & -B \\ 0 & 0 & 0 & 0 & -20.2 & -578.6874 \\ 0 & 10.0 & 0 & 0 & 0 & -10.0 \end{bmatrix} \begin{bmatrix} v_T \\ \alpha \\ \theta \\ q \\ x_a \\ x_F \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 20.2 \\ 0 \end{bmatrix} \delta_e \quad (1.10)$$

$$y = \begin{bmatrix} \alpha \\ q \\ \alpha_F \end{bmatrix} = \begin{bmatrix} C & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 57.29578 \end{bmatrix} x$$

此时，系统的特征根为：

模态	特征根	自然频率	阻尼比
短周期	$-0.699 \pm 2.0297i$	2.147	0.33
长周期	$-0.0085 \pm 0.0827i$	0.083	0.102

通过迎角反馈，得到了传统意义上的短周期模态和长周期模态，但是，在仅有迎角反馈的情况下系统的短周期阻尼比偏低，需要通过引入俯仰角速度来增加

系统阻尼比。

加入俯仰角速度反馈，得到系统根轨迹如图：

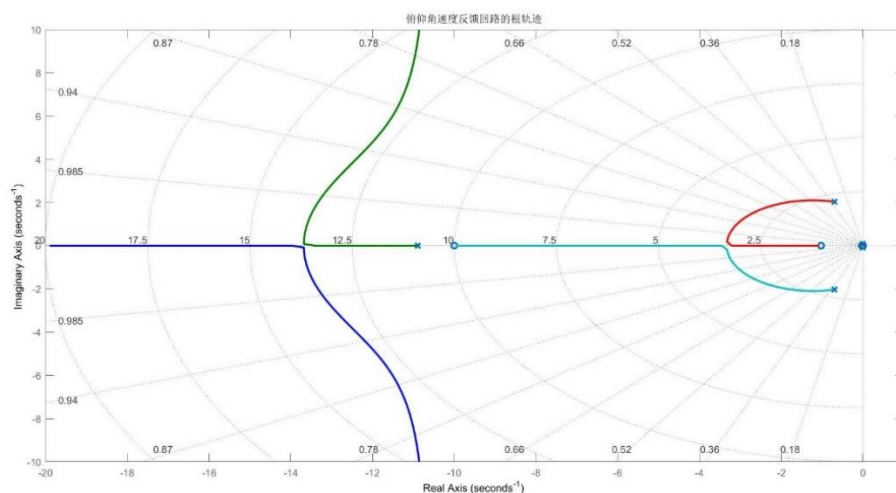


图 2.4 俯仰角速度反馈根轨迹

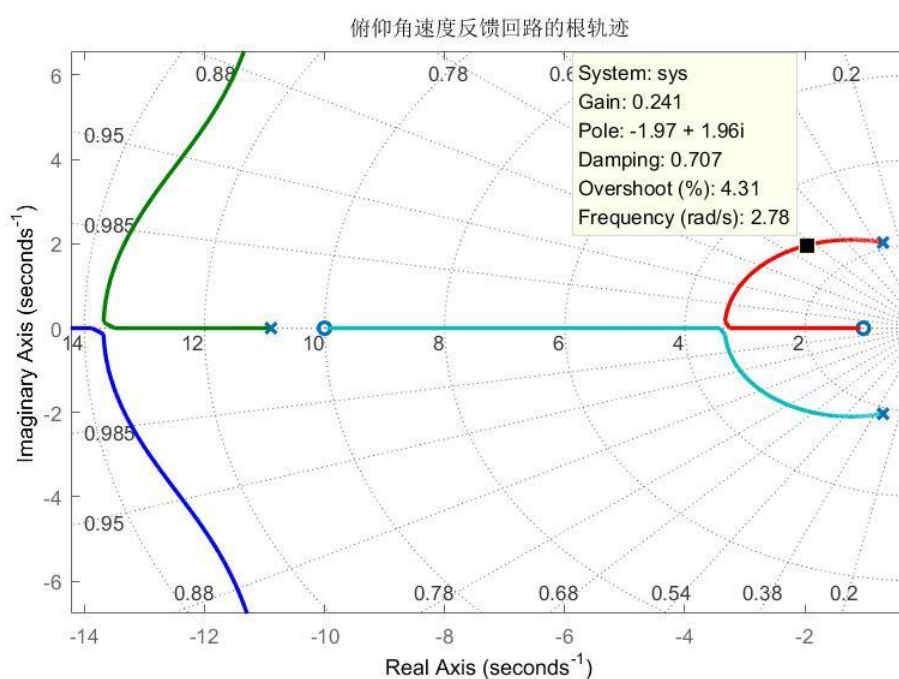


图 2.5 角速率增益选择

可以看出当选择俯仰角速率增益 $K_q = 0.241$ 时短周期模态具有最佳阻尼比

0.707. 计算得到此时系统的传递函数为

$$\frac{\alpha}{\delta_e} = \frac{2.4882(s+75)(s+10)(s^2+0.01964s+0.008892)}{(s+16.57)(s+11.8)(s^2+0.01753s+0.004599)(s^2+3.929s+7.72)} \quad (1.11)$$

$$\frac{q}{\delta_e} = \frac{203.18 s (s+10) (s+1.027) (s+0.02174)}{(s+16.57)(s+11.8)(s^2+0.01753s+0.004599)(s^2+3.929s+7.72)}$$

同时，在 $K_\alpha = 0.5, K_q = 0.241$ 时，系统状态方程为：

$$\dot{x} = \begin{bmatrix} & & & & & 0 \\ & & & & & 0 \\ & A & & -B & & 0 \\ & & & & & 0 \\ 0 & 0 & 0 & -278.9273 & -20.2 & -578.6874 \\ 0 & 10.0 & 0 & 0 & 0 & -10.0 \end{bmatrix} \begin{bmatrix} v_T \\ \alpha \\ \theta \\ q \\ x_a \\ x_F \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 20.2 \\ 0 \end{bmatrix} \delta_e \quad (1.12)$$

$$y = \begin{bmatrix} \alpha \\ q \\ \alpha_F \end{bmatrix} = \begin{bmatrix} & & & & 0 & 0 \\ & C & & & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 57.29578 \end{bmatrix} x$$

系统升降舵单位阶跃响应如下：

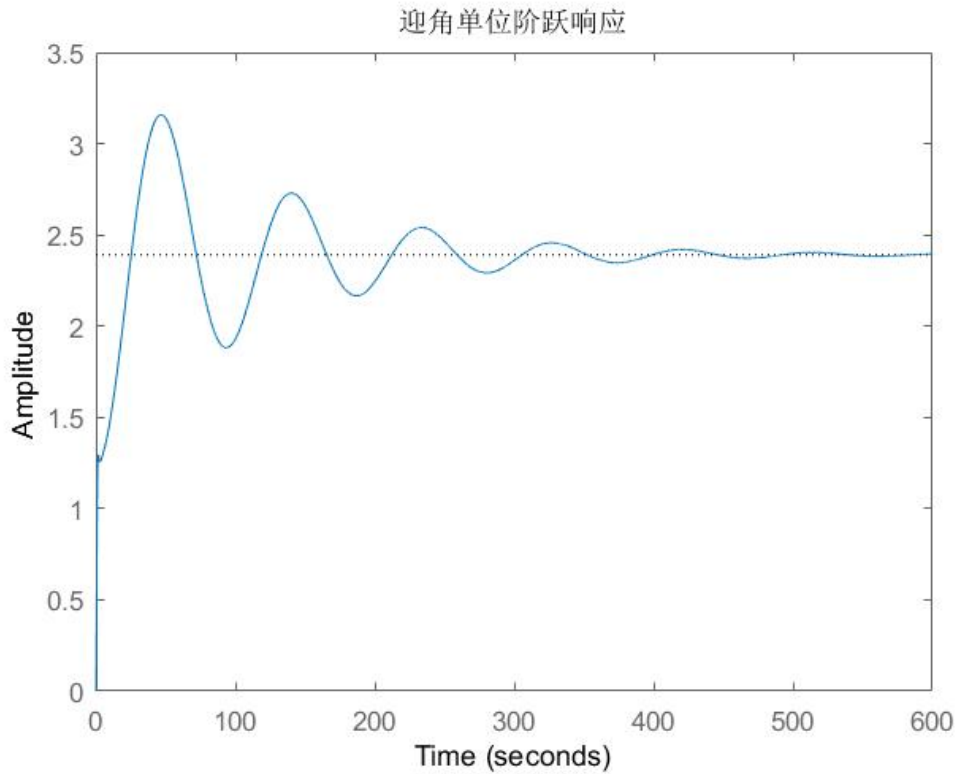


图 2.6 迎角升降舵单位阶跃响应

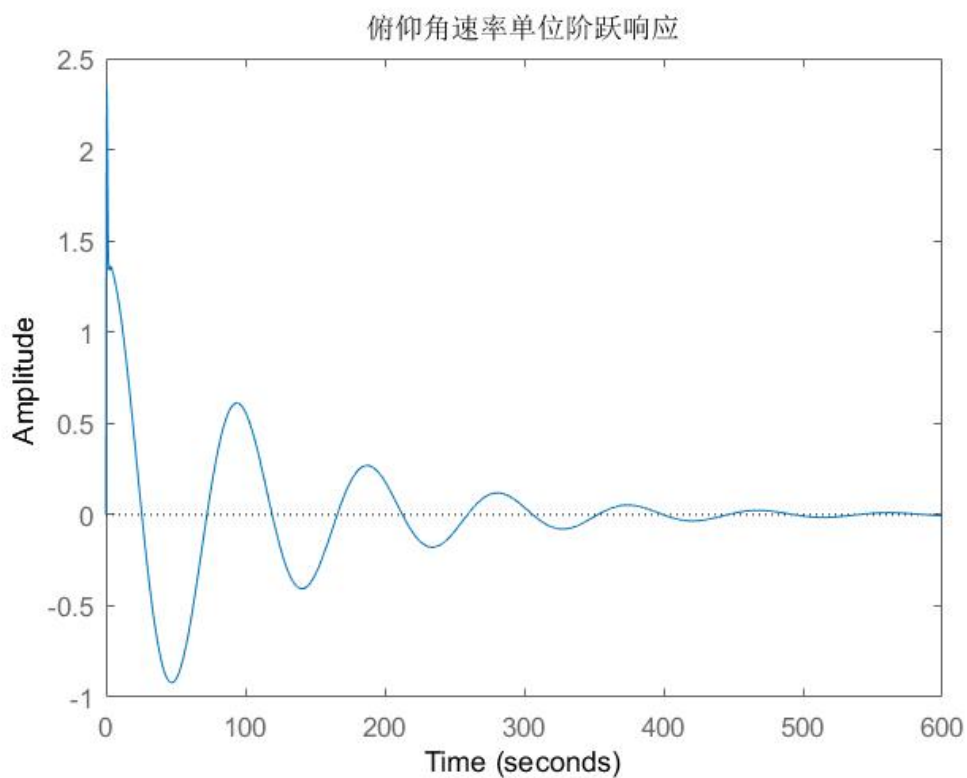


图 2.7 俯仰角速率升降舵单位阶跃响应

可以看到加入 SAS 控制后迎角和俯仰角速率均能达到稳定值，系统短周期特性得到了显著改善。

2.2 俯仰角速率 CAS 控制系统设计

控制增稳系统能够提高飞机稳定性又能改善操纵系统的飞行控制系统。飞机的稳定性是由气动布局决定的。现代高速飞机的稳定性与操纵性相互矛盾。应用控制系统的机理，引入飞机的角速度和过载信号形成闭环控制，以提高飞机的稳定性；引入前馈指令复合控制，以改善飞机的操纵性能。控制增稳系统是电传操纵系统的基础，已被广泛地应用到现代飞行控制系统中。其系统框图如下：

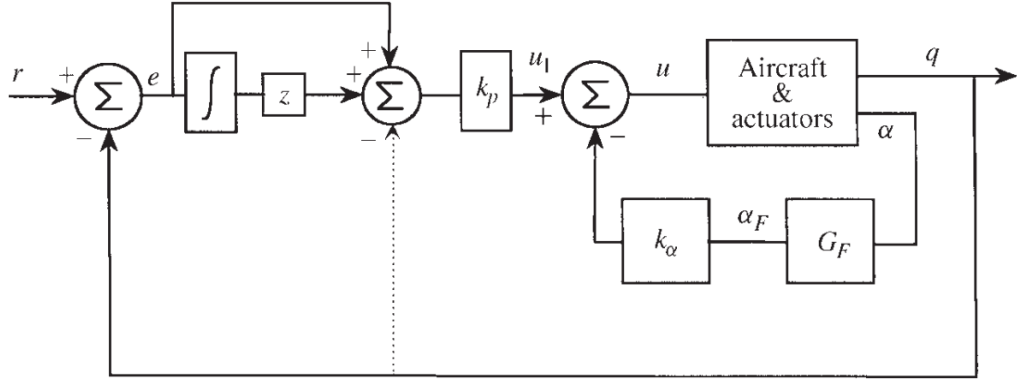


图 2.8 俯仰角速率 CAS 控制结构图

从飞机本体状态方程可以看出，并没有明显的短周期模态，迎角和俯仰角速率受 v_T 和 θ 影响较小。因此可以通过近似的短周期模态对 CAS 系统进行设计，再通过完整的动力学方程进行验证。

飞机短周期模态近似矩阵：

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -1.0189 & 0.90506 \\ 0.82225 & -1.0774 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} -0.00215 \\ -0.17555 \end{bmatrix} \delta_e \quad (1.13)$$

$$y = \begin{bmatrix} 57.296 & 0 \\ 0 & 57.296 \end{bmatrix} x$$

执行机构和迎角滤波器仍采用前述传递函数：

$$G_F(s) = \frac{10}{s+10}, \quad G_\delta(s) = \frac{20.2}{s+20.2} \quad (1.14)$$

因此得到迎角反馈的开环传递函数为：

$$\frac{\alpha_F}{\delta_e} = \frac{24.8824 (s+74.98)}{(s+20.2) (s+10) (s+1.911) (s+0.185)} \quad (1.15)$$

取 $K_\alpha = 0.2$ 得到 $\frac{q}{\delta_e}$ 传递函数：

$$\frac{q}{\delta_e} = \frac{203.18 (s+1.029) (s+10)}{(s+20.13)(s+10.38)(s^2+1.791s+2.128)} \quad (1.16)$$

此时系统短周期根为 $-0.889 \pm 1.15i$ ，阻尼比 0.611。固有频率 $\omega_n = 1.45 \text{ rad/s}$

在外环(俯仰角速率反馈)设计中，引入比例积分环节

$$PI = \frac{s+3}{s} \quad (1.17)$$

得到外环设计状态方程为：

$$\dot{x} = \begin{bmatrix} -10 & 572.9578 & 0 & 0 & 0 \\ 0 & -1.0189 & 0.9051 & 0.0021 & 0 \\ 0 & 0.8225 & -1.0774 & 0.1756 & 0 \\ -10.1 & 0 & 0 & -20.2 & 20.2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 20.2 \\ 3 \end{bmatrix} \delta_e \quad (1.18)$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 57.2958 & 0 & 0 \end{bmatrix} x$$

因此可得到角速度反馈根轨迹如下图所示：

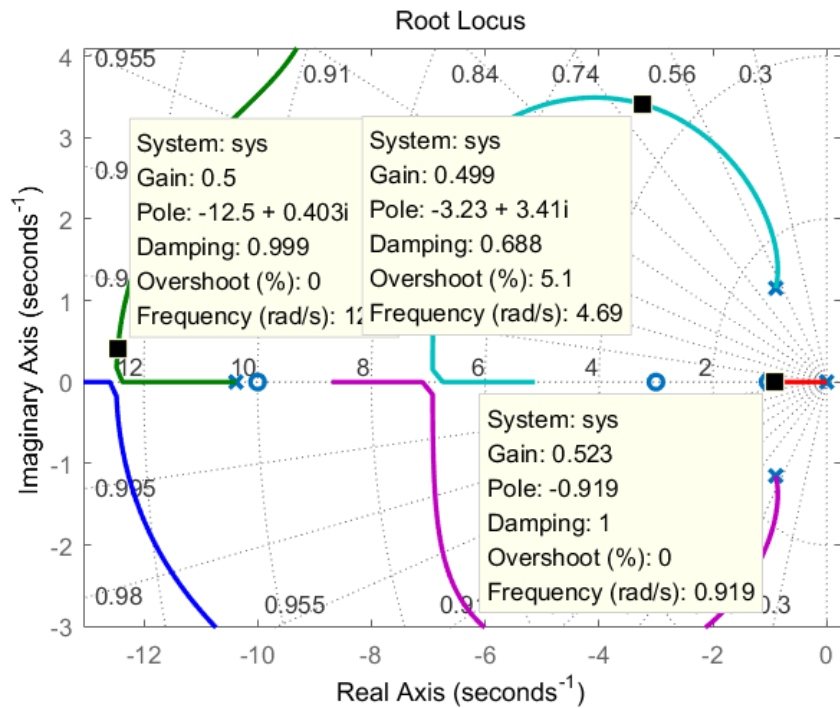


图 2.9 角速度反馈根轨迹

可以看到当 K_p 增大到 0.5 左右时，系统的滤波器及执行机构环节会由两个负实根相会变成一对共轭复根，这不符合 CAS 的要求，并且可以观察到，此时的短周期根为： $-3.23 \pm 3.41i$ ，阻尼比为 0.688，固有自然频率为 4.69，尚未达到最优。

解决此问题的一个方法是增大 K_p ，这会导致振荡环节阻尼迅速减小，因此采用另一种方式，减小 K_α ，取 $K_\alpha=0.08$ ，得到此时角速率反馈随 K_p 变化的根轨迹图：

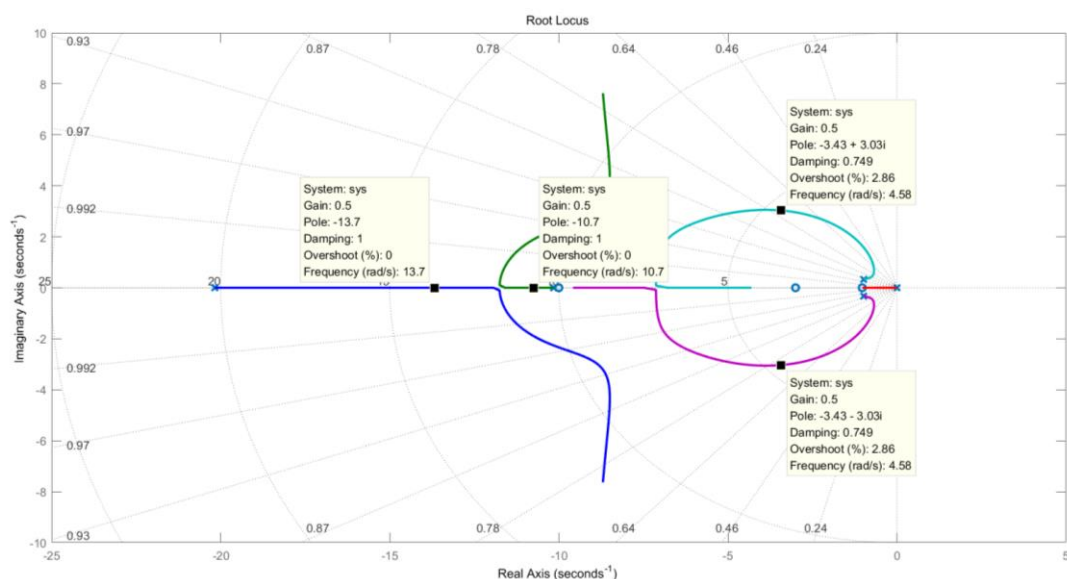


图 2.10 $K_\alpha = 0.08$ 时俯仰角速率反馈根轨迹

可以看出，在 $K_\alpha = 0.08$ ， $K_p = 0.5$ 时，系统滤波器和机构环节特征根未发展为共轭复根，此时短周期共轭复根为 $-3.43 \pm 3.03i$ ，阻尼比 0.749 固有频率 4.58rad/s

由此得到此时的系统闭环函数：

$$\frac{q}{\delta_e} = \frac{101.59 (s+10) (s+3) (s+1.029)}{(s+13.67)(s+10.75)(s+1.018)(s^2 + 6.86s + 20.96)} \quad (1.19)$$

同时可以得到阶跃操纵时系统的动态响应如图：

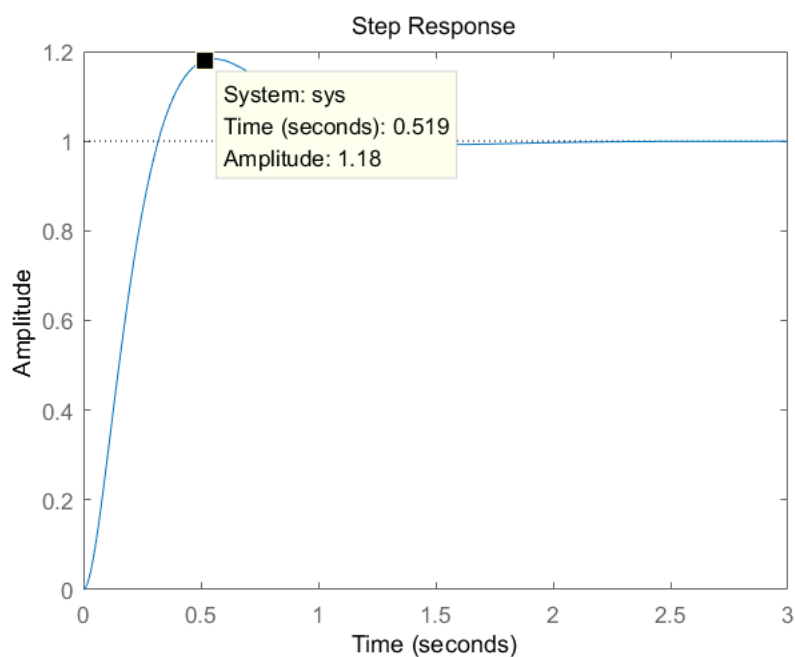


图 2.11 俯仰角速率升降舵单位阶跃响应

可以看出，系统存在 18% 的超调。如果移除补偿器的零点，即采用系统结构图中虚线所示的线路对系统进行修改，消除零点 $s=-3$ ，俯仰角速度传递函数：

$$\frac{q}{\delta_e} = \frac{304.77 (s+10) (s+1.029)}{(s+13.67)(s+10.75)(s+1.018)(s^2 + 6.86s + 20.96)} \quad (1.20)$$

消除零点后系统超调得到了明显改善，如图：

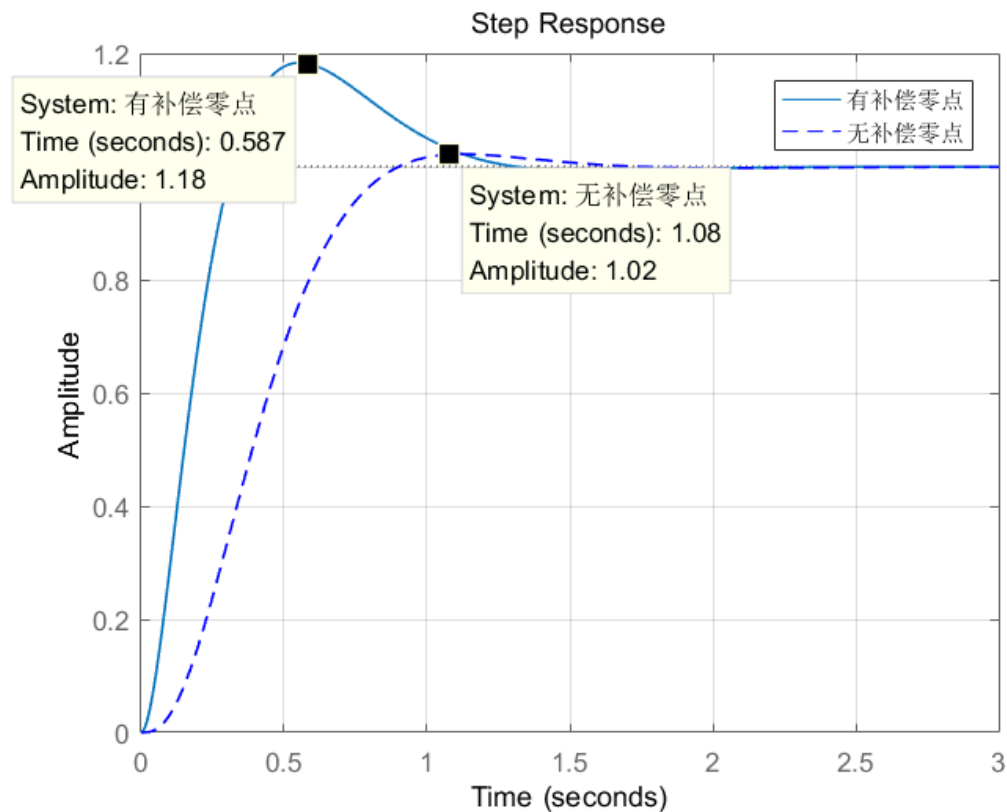


图 2.12 无补偿零点时俯仰角速率单位阶跃响应

第三章 纵向自动驾驶仪设计

3.1 自动驾驶仪简介

自动驾驶用于减轻飞行员操纵负担，自动驾驶仪的设计必须满足特定的稳态误差和干扰抑制指标，但不太强调动态响应。另外必须特别考虑自动驾驶参与操纵与脱离操纵瞬间的响应问题，避免产生不舒适或危险的瞬态运动。

本次设计的纵向自动驾驶仪为飞行高度控制器，以升降舵偏角进行飞行高度控制，使飞机在预定高度飞行。

自动驾驶仪设计主要分为三部分

1. 阻尼器设计：用于满足飞行品质的要求
2. 理想自驾仪反馈增益设计：满足动态响应要求
3. 补偿器设计：使自驾仪满足稳态精度、飞行品质以及动态响应要求。

纵向自动驾驶仪原理图如下：

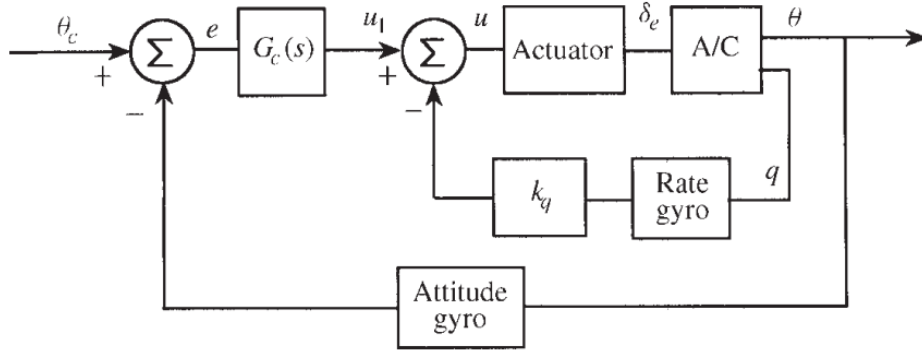


图 3.1 纵向自动驾驶仪结构图

3.2 算例验证与分析

在飞行高度 7620m, 飞行速度 152.4m/s 的平飞状态下，飞机状态方程：

$$\begin{bmatrix} \dot{v}_T \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{q} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} -0.0082354 & 18.938 & -32.17 & 0 & 5.902e-05 \\ -2.56e-04 & -0.5676 & 0 & 1 & 2.26e-06 \\ 0 & 0 & 0 & 1 & 0 \\ 1.31e-05 & -1.4847 & 0 & -0.47599 & -1.495e-07 \\ 0 & -500 & 500 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_T \\ \alpha \\ \theta \\ q \\ h \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.01978 \\ 0 \end{bmatrix} \delta_e$$

$$\begin{bmatrix} \theta \\ q \end{bmatrix} = \begin{bmatrix} 0 & 0 & 57.296 & 0 & 0 \\ 0 & 0 & 0 & 57.296 & 0 \end{bmatrix} x$$

(1. 21)

考虑执行器 0.1s 的延迟，调整符号使正俯仰角速率对应正升降舵偏角

因此 θ 对 δ_e 的传递函数为：

$$\frac{\theta}{\delta_e} = \frac{11.3337 (s+0.5567) (s+0.01897) (s+0.0001666)}{(s+10)(s+0.0001892)(s^2+0.004943s+0.008084)(s^2+1.047s+1.756)} \quad (1. 22)$$

可以看到所有特征根均具有复实部，但是震荡模态的阻尼过小 ($\zeta_{sp} = 0.395$ ，

$\zeta_p = 0.027$). 高度极点被零点抵消, 但是忽略高度会造成长周期模态的较大的误差。为了减小高度误差, 可以通过提高补偿器增益, 固定 k_θ , 然后利用根轨迹来调节 k_q , 实现最佳的短周期阻尼。

当 $k_\theta = 4.0$ 时, k_q 的根轨迹如图所示, 除了短周期极点, 其它极点都在实轴上。根轨迹的上半枝将会随着 k_θ 向右移动, 这样减小了可实现的最大阻尼。

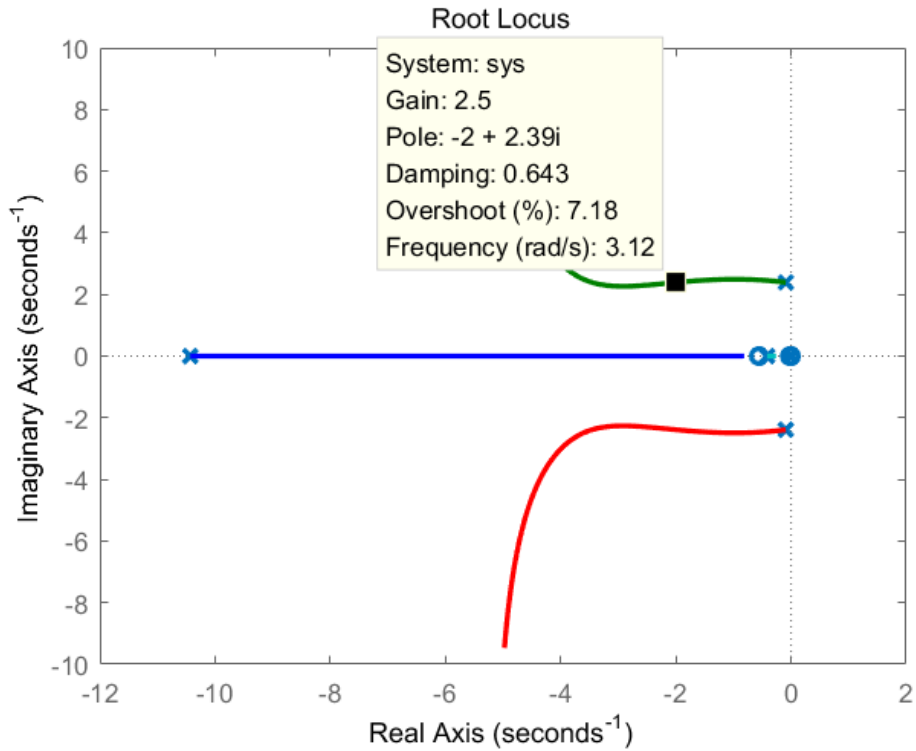


图 3.2 俯仰角速度反馈增益选择

当选择对应小阻尼低自然频率的增益时, 即 $k_q = 2.5$, 可以得到俯仰角的传递函数:

$$\frac{\theta}{\theta_c} = \frac{45.3349 (s+0.5567) (s+0.01897) (s+0.0001666)}{(s+6.646)(s+0.3815)(s+0.02522)(s+0.0001718)(s^2+3.999s+9.704)} \quad (1.23)$$

此时飞机姿态阶跃响应如图:

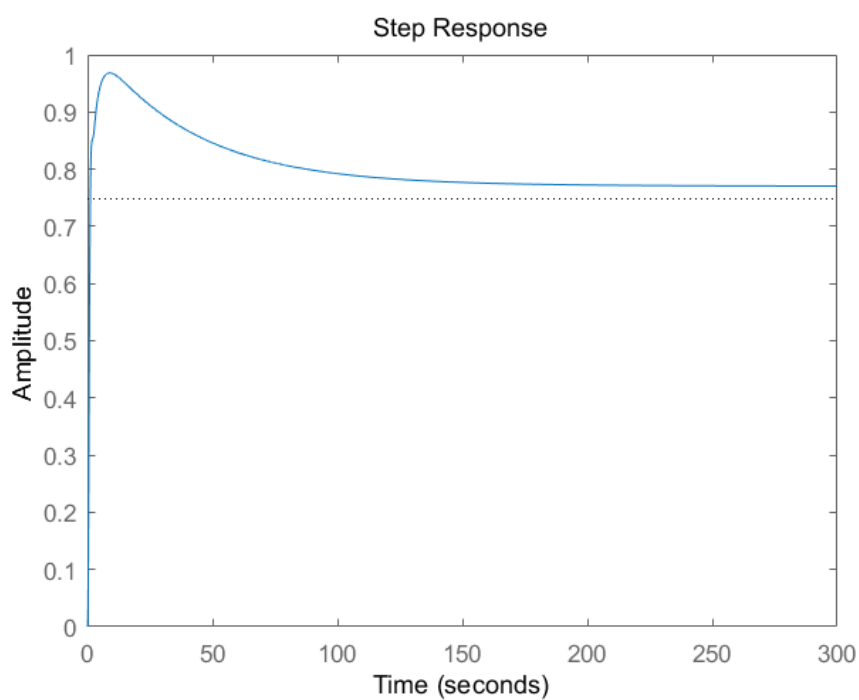


图 3.3 姿态角单位阶跃响应

可以看出响应速度较快，但存在较大的稳态误差。

附录： 程序源码

1. 飞机本体运动特性计算

```
close all

clear

clc

syms 's'

a = [-0.019311 8.8157 -32.17 -0.57499;
     -0.00025389 -1.0189 0 0.90506;
     0 0 0 1;
     2.9465*10^(-12) 0.82225 0 -1.0774];

b = [0.1737 -0.0021499 0 -0.17555]';

c = [0 57.29578 0 0;
     0 0 0 57.29578];

d = [0 ; 0];

[eigm eigv] = eig(a);

%% 传递函数

[num den] = ss2tf(a,b,c,d);

Galpha = tf(num(1,:),den);

Gq = tf(num(2,:),den);

%% 伯德图
```

```

figure(1)

bode(Galpha)

title('BodeAlpha')

grid on

figure(2)

bode(Gq)

title('BodeGq')

grid on

%% 那奎斯特图

figure(3)

nyquist(Galpha)

title('NyquistAlpha')


figure(4)

nyquist(Gq)

title('NyquistGq')


%% 阶跃响应

figure(5)

subplot(2,1,1)

step(Galpha,20);

title('StepAlpha(Short)')

```

```

subplot(2,1,2)

step(Galpha)

title('StepAlpha(Long)')

figure(6)

subplot(2,1,1)

step(Gq,20)

title('StepGq(Short)')


subplot(2,1,2)

step(Gq)

title('StepGq(Long)')

```

2. SAS 设计

```

%% Sas

test = [0;0;0;0];

test2 = [0 0 0 0 -20.2 0;

         0 10 0 0 0 -10];

saa = [a,-1*b,test;test2];

sab = [0;0;0;0;20.2;0];

sac = [c,[0 0 ;0 0 ];0 0 0   0 0 57.29578];

```

```
sad = [0; 0;0];
```

```
[sanum saden] = ss2tf(saa,sab,sac,sad);
```

```
saGalpha = tf(sanum(1,:),saden);
```

```
[z1,p1,k1]=ss2zp(saa,sab,sac(1,:),0);
```

```
G_alphamy=zpk(z1,p1,k1)%求出迎角闭环后的传递函数\
```

```
saGq = tf(sanum(2,:),saden);
```

```
saGalphaf = tf(sanum(3,:),saden);
```

```
rlocus(saGalphaf)
```

```
title('Root Locus of alpha')
```

```
%反馈增益选择
```

```
figure(2)
```

```
w3=logspace(-2,1,4000);
```

```
rlocus(saa,sab,sac(3,:),0,w3)%引入反馈回路的俯仰角速度回路根轨迹
```

```
grid on
```

```
title('迎角反馈回路的根轨迹')
```

```
axis([-2,0.5,-3,3])
```



```

set(findobj(get(gca,'Children'),'LineWidth',0.5),'LineWidth',2);

%% ka = 0.5

%saGa_close = feedback(saGalpha,-0.5);

k1=0.5;

A3=saa-sab*k1*sac(3,:);%迎角形成闭环

[z,p,k]=ss2zp(A3,sab,sac(2,:),0);

G_alpha1=zpk(z,p,k)%求出迎角闭环后的传递函数

w=logspace(-2,1,3000);

rlocus(A3,sab,sac(2,:),0,w)%引入俯仰角速度反馈回路的根轨迹

grid on

title('俯仰角速度反馈回路的根轨迹')

axis([-20,1,-10,10])

set(findobj(get(gca,'Children'),'LineWidth',0.5),'LineWidth',2);

k2 = 0.241

A4=A3-sab*k2*sac(2,:);

[z,p,k]=ss2zp(A4,sab,sac(2,:),0);

G_q2=zpk(z,p,k)%求出迎角闭环后的传递函数

[z,p,k]=ss2zp(A4,sab,sac(1,:),0);

G_alpha2=zpk(z,p,k)%求出迎角闭环后的传递函数

figure(3)

```

```

step(G_alpha2)

title('迎角单位阶跃响应');

figure(4)

step(G_q2)

title('俯仰角速率单位阶跃响应')

```

3. CAS 设计

```

%% CAS

Ashort = [-1.0189 0.90506;0.82225 -1.0774];

Bshort = [-0.0021499;-0.17555];

Cshort=[57.296 0;0 57.296];

Dshort = [0;0];

% [z,p,k]=ss2zp(Ashort,Bshort,Cshort(1,:),0);

% G_alpha3=zpk(z,p,k)%求出迎角闭环后的传递函数

% w=logspace(-2,1,3000);

%

% ka = 0.2;

% acl= Ashort- Bshort*[ka 0]*Cshort; % Close Alpha-loop

% [z,p,k] = ss2zp(Ashort,Bshort,Cshort(2,:),0) % q/u1 transf. fn.

% gte = zpk(z,p,k)

```

```

sysp= ss(Ashort,Bshort,Cshort,Dshort); % Plant

sysa= ss(-20.2,20.2,-1,0); % Actuator & SIGN CHANGE

[sys1]= series(sysa,sysp); % Actuator then Plant

sysf= ss(-10,[10 0],[1; 0],[0 0; 0 1]); % Alpha Filter

[sys2]= series(sys1,sysf); % Actuator+Plant+Filter


[casa casb casc casd]= ssdata(sys2); % Extract a,b,c,d

[z,p,k]= ss2zp(casa,casb,casc(1,:),0) % q/u1 transf. fn.

gte0 = zpk(z,p,k)


ka = 0.2;

casac1= casa- casb*[ka 0]*casc; % Close Alpha-loop

[z,p,k]= ss2zp(casac1,casb,casc(2,:),0) % q/u1 transf. fn.

gte = zpk(z,p,k)


sys3= ss(casac1,casb,casc,[0;0]);

sysi= ss(0,3,1,1);

sys4= series(sysi,sys3);

[aa,bb,cc,dd]= ssdata(sys4);

figure(5)

k= linspace(0,.9,1000);

rlocus(aa,bb,cc(2,:),0,k);

```

```

axis=([-16,0,-8,8])

grid on

set(findobj(get(gca,'Children'),'LineWidth',0.5),'LineWidth',2);

%% new CAS

ka = 0.08;

casac12= casa- casb*[ka 0]*casc; % Close Alpha-loop

[z,p,k]= ss2zp(casac12,casb,casc(2,:),0) % q/u1 transf. fn.

gte = zpk(z,p,k)


sys3= ss(casac12,casb,casc,[0;0]);

sysi= ss(0,3,1,1);

sys4= series(sysi,sys3);

[aa,bb,cc,dd]= ssdata(sys4);

figure(6)

k= linspace(0,.9,1000);

rlocus(aa,bb,cc(2,:),0,k);

axis=([-16,0,-8,8])

grid on

set(findobj(get(gca,'Children'),'LineWidth',0.5),'LineWidth',2);

kq=0.5

```

```

casacl3 = aa-bb*kq*cc(2,:);

sys = ss(casacl3,0.5*bb,cc(2,:),0);

[z,p,k]= ss2zp(casacl3,0.5*bb,cc(2,:),0) % q/u1 transf. fn.

gte = zpk(z,p,k)

step(sys,3)

grid on

hold on

sys5 = ss(-3,3,1,0);

sys6 = series(sys,sys5);

step(sys6,'--',3)

h=findobj(gcf,'type','line');

set(h,'LineWidth',1);

legend('有补偿零点','无补偿零点');

```

4. 自动驾驶仪

```
clear
```

```

close all

clc

ap = [-0.0082354 18.938 -32.170 0.0 5.9022E-05;
-0.00025617 -0.56761 0.0 1.0 2.2633E-06;
0.0 0.0 0.0 1.0 0.0;
1.3114E-05 -1.4847 0.0 -0.47599 -1.4947E-07;
0.0 -500.00 500.00 0.0 0.0];

bp = [0;0;0;-0.019781;0];

cp = [0 0 57.296 0 0 ;0 0 0 57.296 0];

dp = [0;0];

plant= ss(ap,bp,cp,dp);

aa= [-10]; ba= [10]; % Actuator

ca= [-1]; da= [0]; % sign change for plant

actua= ss(aa,ba,ca,da);

sys1 = series(actua,plant);

[a,b,c,d]= ssdata(sys1);

[z,p,k]=ss2zp(a,b,c(1,:),0);

G_alpha=zpk(z,p,k)%外环闭环，迎角传递函数求解

[z,p,k]=ss2zp(a,b,c(2,:),0);

G_q=zpk(z,p,k)%外环闭环，迎角传递函数求解

%% Ktheta

```

```

ktheta = 4;

acl= a - b*[ktheta 0]*c; % Choose k
k= linspace(0,10,1000);

rlocus(acl,b,c(2,:),0,k); % Root locus for k

set(findobj(get(gca,'Children'),'LineWidth',0.5),'LineWidth',2);

figure(2)

acl= a - b*[4 2.5]*c; % Close both loops

[z,p,k]= ss2zp(acl,4*b,c(1,:),0) % Closed-loop, Unity fb

Gtheta = zpk(z,p,k);

step(Gtheta)

```