

基于自适应 Backstepping 的飞机重心 变化稳定控制律设计

明瑞晨, 刘小雄, 李煜, 张悦航

(西北工业大学 自动化学院, 西安 710072)

摘要: 飞机重心变化会影响飞机气动力从而产生较强的扰动误差, 可以采用自适应 Backstepping 控制方法消除这种扰动误差对飞行控制的影响; 首先建立飞机重心位置和重量变化的非线性模型, 然后设计自适应 Backstepping 控制律, 对模型误差进行自适应估计, 针对 F16 飞机纵向操纵力矩不是仿射非线性的形式, 采用牛顿迭代法进行求解, 最后进行正常情况下、重心突变和重心渐变情况下的仿真验证, 仿真结果表明设计的自适应 Backstepping 控制律具有很好的动态和稳态性能。

关键词: 自适应 Backstepping; 重心变化; 控制律设计; 牛顿迭代法

Adaptive Backstepping Control for Gravity Change of Aircraft Center

Ming Ruichen, Liu Xiaoxiong, Li Yu, Zhang Yuehang

(College of Automation, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: The Gravity Change of Aircraft Center has influence on the aerodynamic force of the aircraft and produce strong disturbance errors. An adaptive backstepping control method is used to eliminate the influence of the disturbance errors on flight control. Firstly, a non-linear model of aircraft gravity center position and weight change is established. Then an adaptive Backstepping control law is designed to estimate the model error adaptively. Aiming at the fact that the longitudinal control moment of F16 aircraft is not an affine non-linear form, the Newton's method is used to solve the problem. Finally, the simulation verification under normal conditions, sudden change of gravity center and gradual change of gravity center is carried out. The simulation results show that the adaptive backstepping control law designed has good dynamic and steady-state performance.

Keywords: adaptive backstepping; center of gravity change; control law design; Newton's method

0 引言

飞机在飞行过程中, 由于燃油消耗、武器投掷等因素影响, 机体的重心和重量都会随之发生变化。其变化程度直接影响到所设计控制律的控制性能, 这一变化轻则使得飞机飞行品质下降, 重则使得飞机坠毁。例如, 2013 年美军波音 747 货机因军车没有固定好, 使得重心改变, 最终导致飞机坠毁; 2010 年, 刚果飞机一架 A330 飞机因客舱出现鳄鱼导致人员恐慌, 最终飞机因失去平衡而坠毁。从以上空难实例能看出, 飞机重心的变化对飞机飞行影响极为严重, 因此要求控制工作者在设计飞机飞行控制律的过程中必须要考虑重心变化对飞机的影响。飞机在飞行时很难实时测量出飞机重心的变化, 因此, 飞机重心变化带给飞机的不利影响通常情况下是由所设计控制律的鲁棒性和自适应能力来克服, 可以通过提高控制律的鲁棒性来进一

步减小飞机在飞行过程中重心对飞机造成的影响。

Backstepping 是近年来发展起来的处理飞机非线性控制律设计的方法, 可以很好进行飞机参数不确定条件下的自适应控制。其控制思想是采用代数运算选择反馈函数的方法来期望的动态去消除系统中不期望的动态。由于 Backstepping 方法采用代数求逆的方式来消除模型中不期望动态, 因此需要提前获取被控对象精确的数学模型, 针对飞机重心变化对模型参数的影响, 需要原有 Backstepping 控制方法具有鲁棒性或者自适应能力, 使得所设计的控制器达到期望性能的同时降低控制器对飞机模型的依赖。文献 [1-3] 介绍了 Backstepping 在无人机上的应用; 文献 [4] 介绍了带神经网络 Backstepping 的控制方法在编队飞行中的应用; 文献 [5] 介绍了自适应 Backstepping 在带推力矢量发动机的 F16 模型上的应用; 文献 [6] 利用自适应增量 Backstepping 通过消除对模型的依赖, 从而解决高性能飞机上的不确定性问题; 文献 [7] 介绍了基于 F18 飞机的 Backstepping 控制方法的鲁棒控制器设计, 主要为使用自适应 Backstepping 设计控制律, 然后使用鲁棒控制分配方法进行解算, 并对比了鲁棒控制分配与最小二乘控制分配。文献 [8] 介绍了自适应 Backstepping 在纵向通道的应用。

考虑到飞机在飞行过程中燃油消耗、武器投掷等因素对飞机重心的影响, 本文主要在 Backstepping 控制的基础上, 结合自适应控制方法的思想, 针对重心变化对飞机产

收稿日期: 2019-05-09; 修回日期: 2019-06-14。

基金项目: 国家自然科学基金(61374032); 国防科工局基础科研项目(JCKY2016205A004); 陕西省飞行控制与仿真技术重点实验室。

作者简介: 明瑞晨(1996-), 男, 湖北黄石人, 硕士研究生, 主要从事飞行控制方向的研究。

刘小雄(1973-), 男, 陕西周至人, 副教授, 博士, 主要从事飞行控制、容错飞行控制、无人机导航算法方向的研究。

生的建模误差和干扰, 完成自适应 Backstepping 控制律的设计, 从而提高飞机飞行控制律的自适应能力。

1 考虑重心变化的 F16 飞机模型

飞机角速率回路动力学模型如下所示:

$$\begin{cases} \dot{p} = (c_1 r + c_2 p)q + c_3 \bar{L} + c_4(N + h_{E}q) \\ \dot{q} = c_5 pr - c_6(p - r) + c_7(M + F_T - h_{E}r) \\ \dot{r} = (c_8 p - c_2 r)q + c_4 \bar{L} + c_9(N + h_{E}q) \end{cases} \quad (1)$$

$$\text{其中: } c_1 = \frac{(I_y - I_z)I_z - I_{xz}^2}{\sum}, c_2 = \frac{(I_x - I_y + I_z)I_{xz}}{\sum}, c_3 =$$

$$= \frac{I_z}{\sum}, c_4 = \frac{I_{xz}}{\sum}, c_5 = \frac{I_z - I_x}{I_y}, c_6 = \frac{I_{xz}}{I_y}, c_7 = \frac{1}{I_y}, c_8 =$$

$$\frac{I_x(I_x - I_y) + I_{xz}^2}{\sum}, c_9 = \frac{I_x}{\sum}, \sum = I_x I_z - I_{xz}^2, I_x, I_y, I_z, I_{xz} \text{ 为飞机}$$

三轴的转动惯量和惯性积。 pqr 分别为三轴的角速率。 $\bar{L}MN$ 分别为滚转力矩、俯仰力矩和偏航力矩, 按照飞机的特点, 可将其分为:

$$\begin{aligned} \bar{L} &= \bar{L}_a + \bar{L}_c \\ M &= M_a + M_c \\ N &= N_a + N_c \end{aligned} \quad (2)$$

式中, $\bar{L}_a M_a N_a$ 表示与飞机飞行状态相关的力矩, $\bar{L}_c M_c N_c$ 则表示为与飞机舵面相关的控制力矩。

飞机重心位置移动 (x_{cg}, y_{cg}, z_{cg}) 后, 修正后的俯仰力矩系数 C_m^* 、滚转力矩系数 C_l^* 、偏航力矩系数 C_n^* 利用以下公式计算:

$$\begin{cases} C_m^* = C_m - \frac{C_N x_{cg}}{c_A} + \frac{C_T z_{cg}}{c_A} \\ C_l^* = C_l - \frac{C_C z_{cg}}{b} + \frac{C_N y_{cg}}{b} \\ C_n^* = C_n + \frac{C_C x_{cg}}{b} - \frac{C_T y_{cg}}{b} \end{cases} \quad (3)$$

其中:

$$\begin{bmatrix} C_T \\ C_C \\ C_N \end{bmatrix} = \begin{bmatrix} \cos\alpha \cos\beta & -\cos\alpha \sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha \cos\beta & -\sin\alpha \sin\beta & \cos\alpha \end{bmatrix} \cdot \begin{bmatrix} C_D \\ C_Y \\ C_L \end{bmatrix} \quad (4)$$

式中, α, β 分别为迎角、侧滑角, C_D, C_Y, C_L 分别为阻力系数、侧力系数和升力系数, C_T, C_C, C_N 分别为气动力在机体坐标系下沿 x, y, z 轴上的分力系数。

对于 F16 飞机, 其纵向通道俯仰力矩 M 无法写成仿射形式, 例如 $C_m = f(\alpha, \beta, \delta e)$, 无法写成 $C_m = f(\alpha, \beta) \delta e$, 针对这类问题, 本文使用牛顿迭代法来避开原本需要的仿射形式。

给出 \bar{L}, M, N 与气动导数和舵面输入量的关系:

$$\begin{cases} \bar{L} = \bar{q}Sb \left(C_l + \frac{b}{2V}(C_{lr}r + C_{lp}p) \right) + C_{\bar{L}a}\delta a + C_{\bar{L}r}\delta r \\ M = \bar{q}Sc \left(C_m + \frac{c}{2V}C_{mq}q \right) + \bar{q}ScC_{ZT}(x_{cgr} - x_{cg}) \\ N = \bar{q}Sb \left(C_n + \frac{b}{2V}(C_{nr}r + C_{np}p) \right) + C_{\bar{N}a}\delta a + C_{\bar{N}r}\delta r \\ \bar{q}ScC_{YT}(x_{cgr} - x_{cg}) \end{cases} \quad (5)$$

而对于纵向通道, 用下式表示:

$$\begin{cases} C_m = C_m(\alpha, \beta, \delta e) \\ C_{ZT} = C_{ZT}(\alpha, \beta, \delta e) \end{cases} \quad (6)$$

因此我们设计一个关于 δe 的函数 $g_{\delta e}(\delta e)$, 控制律求出 $g_{\delta e}(\delta e)$ 的值, 再利用牛顿迭代法反解出 $g_{\delta e}(\delta e) = y_0, \delta e = g_{\delta e}^{-1}(y_0)$ 。

设 $g_{\delta e}(\delta e)$ 为:

$$g_{\delta e}(\delta e) = C_m + C_z(x_{cgr} - x_{cg}) \quad (7)$$

$$u = [g_{\delta e}(\delta e), \delta a, \delta r]^T \quad (8)$$

可以将 (1) 改成下面形式:

$$\dot{x} = f(x) + g(x)u \quad (9)$$

其中:

$$x = [p, q, r]^T \quad (10)$$

$$u = [g_{\delta e}(\delta e), \delta a, \delta r]^T \quad (11)$$

$$f_2(x) =$$

$$\begin{bmatrix} (c_1 r + c_2 p)q + \bar{q}Sbc_3 \left(C_l + \frac{b}{2V}(C_{lr}r + C_{lp}p) \right) - \\ \bar{q}ScC_4(x_{cgr} - x_{cg})C_{\beta\beta}\beta + \\ \bar{q}Sbc_4 \left(C_n + \frac{b}{2V}(C_{nr}r + C_{np}p) - \frac{c(x_{cgr} - x_{cg})}{2V}(C_{yr}r + C_{yp}p) \right) \\ c_5 pr - c_6(p^2 - r^2) + \bar{q}Sc_7 \frac{c^2}{2V}C_{mq}q + \bar{q}Sc_7 \frac{c^2}{2V}C_{\omega}q(x_{cgr} - x_{cg}) \\ (c_8 p - c_2 r)q + \bar{q}Sbc_4 \left(C_l + \frac{b}{2V}(C_{lr}r + C_{lp}p) \right) - \\ \bar{q}ScC_9(x_{cgr} - x_{cg})C_{\beta\beta}\beta + \\ \bar{q}Sbc_9 \left(C_n + \frac{b}{2V}(C_{nr}r + C_{np}p) - \frac{c(x_{cgr} - x_{cg})}{2V}(C_{yr}r + C_{yp}p) \right) \end{bmatrix} \quad (12)$$

$$g_2(x) = \bar{q}Sb \begin{bmatrix} \frac{c}{b} & 0 \\ 0 & \frac{c}{b} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} c_3 C_{\bar{L}a} + c_4 \left(C_{\bar{L}a} - \frac{c}{b}(x_{cgr} - x_{cg})C_{\beta\beta a} \right) \\ c_4 C_{\bar{L}a} + c_9 \left(C_{\bar{N}a} - \frac{c}{b}(x_{cgr} - x_{cg})C_{\beta\beta a} \right) \\ c_3 C_{\bar{L}r} + c_4 \left(C_{\bar{N}r} - \frac{c}{b}(x_{cgr} - x_{cg})C_{\beta\beta r} \right) \\ c_4 C_{\bar{L}r} + c_9 \left(C_{\bar{N}r} - \frac{c}{b}(x_{cgr} - x_{cg})C_{\beta\beta r} \right) \end{bmatrix} \quad (13)$$

当飞机重量或者重心发生变化的时候, 影响到飞机的气动力矩系数, 可以在控制律设计中通过自适应估计完成对参数扰动的消除。

2 自适应 Backstepping 控制律设计

Backstepping 方法是一种通过 Lyapunov 稳定性理论反推出控制律的一种方法。该方法在非线性系统中有着独特的优势, 能够保证系统的稳定性, 同时系统的性能也十分优秀。但缺点是, 第一, backstepping 方法需要知道被控对象的精确的运动模型; 第二 backstepping 在通常情况下需要系统可以写成仿射形式。本文通过设计了系统建模误差的自适应律, 同时使用牛顿迭代法求解控制律来规避了上

述的两个缺点。

常规 Backstepping 控制律对模型的精度有比较高的要求。然而实际情况中, 重量或者重心位置变化时会产生偏差, 这些偏差累积起来可能会对飞机模型产生较大的偏差, 进而会影响到系统的动态性能, 甚至导致系统发散。

自适应 Backstepping 控制律在设计过程中, 就已经考虑到了系统模型的偏差, 通过自适应律对这些偏差进行估计, 并修正控制律, 以达到减小这些建模误差带来的影响。

2.1 控制律设计

首先由重心位置变化引起的模型的不确定误差对原模型 (9) 进行修正得:

$$\bar{x} = f(x) + g(x)u + \delta(t) \quad (14)$$

其中: $\delta(t)$ 为各个干扰和建模误差的总和, 包含重心变化导致的模型不确定引起的误差和外界干扰等。

首先可设:

$$z = x - x_{ref} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{ref} \quad (15)$$

其中: x_{ref} 为欧拉角速率的指令信号, 则可以设 Lyapunov 函数为:

$$V = \frac{1}{2}z^T z \quad (16)$$

则可以看出该 Lyapunov 函数满足 $V \geq 0$, 两边求导得:

$$\dot{V} = \frac{1}{2}(\dot{z}^T z + z^T \dot{z}) = \dot{z}^T z = (\bar{x} - \bar{x}_{ref})^T \dot{z} \quad (17)$$

将 (14) 带入上式得:

$$\dot{V} = (f(x) + g(x)u + \delta(t) - \bar{x}_{ref})^T \dot{z} \quad (18)$$

要想系统稳定, 则需要使 Lyapunov 函数的导数小于 0, 则可以设 K 为对角阵, 且对角线上的每个元素都大于 0, 使得下式成立:

$$f(x) + g(x)u + \delta(t) - \bar{x}_{ref} = -Kz \quad (19)$$

因为 $\delta(t)$ 为模型不确定性以及外界干扰导致的误差, 是未知的, 因此我们用自适应律对其估计, 即用 $\hat{\delta}(t)$ 代替 $\delta(t)$ 。因此可由 (19) 解得控制律为:

$$u = [g(x)]^{-1}(-Kz + \bar{x}_{ref} - f(x) - \hat{\delta}(t)) \quad (20)$$

由该控制律可以知道, 在 $\hat{\delta}(t)$ 与 $\delta(t)$ 相同时, 式 (16) 恒大于等于 0, 式 (17) 恒小于等于 0, 因此 z 满足 Lyapunov 稳定的。但 $\hat{\delta}(t)$ 与 $\delta(t)$ 不会总相等, 因此我们需要在此基础上, 设计关于 $\delta(t)$ 的自适应律, 使得系统稳定。

2.2 自适应律设计

在控制律 (20) 中可以知道 $\hat{\delta}(t)$ 是 $\delta(t)$ 的估计, 因此这里可以设 $\tilde{\delta}(t) = \delta(t) - \hat{\delta}(t)$, 表示估计值与真实值之间的偏差。下面将推导得到干扰估计的自适应律。

如果此时的 Lyapunov 函数是 $V = \frac{1}{2}z^T z$, 则对其求导后, 将控制律 (20) 代入后得:

$$\dot{V} = -Kz^T z + \tilde{\delta}(t)^T z \quad (21)$$

由上式可知, 当 $\tilde{\delta}(t)$ 为零时即偏差的估计值等于真实

值时, 才能保证式 (21) 小于零, 即系统满足李雅普诺夫稳定。当估计误差过大的时候, 会导致系统不稳定。因此这里重新定义新的 Lyapunov 函数:

$$V_2 = V + \frac{1}{2}\Gamma^{-1}\tilde{\delta}^T\tilde{\delta} = \frac{1}{2}z^T z + \frac{1}{2}\Gamma^{-1}\tilde{\delta}^T\tilde{\delta} \quad (22)$$

其中: Γ 为 3×3 对角阵, 且各元素大于零, 为自适应律增益系数, Γ 每个元素越大, 相应通道的自适应律跟踪得越快。

将 (22) 对时间求导得:

$$\begin{aligned} \dot{V}_2 = \dot{V} - \Gamma^{-1}\tilde{\delta}^T\dot{\tilde{\delta}} = & -Kz^T z + \tilde{\delta}^T \dot{z} - \Gamma^{-1}\tilde{\delta}^T \dot{\tilde{\delta}} = \\ & -Kz^T z + \tilde{\delta}^T (z_1 - \Gamma^{-1}\tilde{\delta}) \end{aligned} \quad (23)$$

要想使 $\dot{V}_2 \leq 0$ 成立, 自适应律需满足下式:

$$\dot{\tilde{\delta}} = \Gamma z \quad (24)$$

通过前面控制律和自适应律设计过程, 可以总结成下面框图。

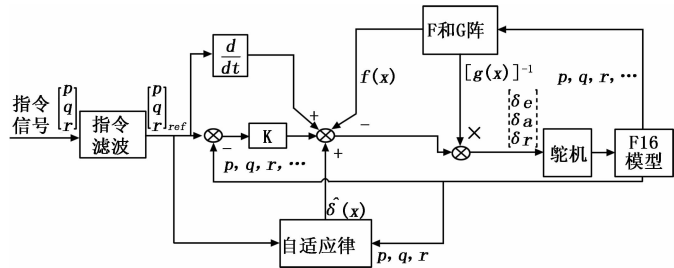


图 1 角速率自适应 Backstepping 控制结构框图

2.3 指令滤波器

由 (20) 可知控制律中有指令信号的导数这一项。在指令的导数在某时刻很大时, 比如阶跃信号, 会导致此时的控制律解算溢出, 进而影响系统的稳定性。因此通常在指令信号后加一个临界阻尼的二阶滤波器。这里我们可以取阻尼比为 1, 无阻尼自然频率为 5 rad/s, 滤波器传递函数为:

$$G(s) = \frac{25}{s^2 + 10 + 25} \quad (25)$$

3 基于牛顿迭代法的纵向力矩解算

对于解一元函数的数值解的方法有牛顿迭代法, 割线法^[9], 二分法等, 其中这些方法有着各自的优势和局限性。对于割线法, 在一定程度上可视为牛顿迭代的弱化版, 收敛条件和牛顿法相同, 但收敛速度会比牛顿法要弱; 牛顿法在收敛速度上比其他方法更快, 在精度上, 比重新对气动插值的精度更高。牛顿法在解算过程中基本上一到两次迭代就能够达到精度要求, 而且牛顿法中导数可以用差分近似代替, 因此运算量较小。

由式 (8) 和式 (20) 可以看出, 在滚转和偏航通道上, 控制律解算出的结果是舵面输入; 在俯仰通道上, 控制律解算的结果还不是最终的舵面输入, 而是关于升降舵输入 δ_e 的一个函数 $g_{\delta}(\delta_e)$, 其定义由 (7) 给出。因此需要解一元方程, 求出 δ_e 。下面给出牛顿迭代法的算法过程。

首先直接给出牛顿迭代法的基本公式:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad (26)$$

因为牛顿迭代法是求解 $f(x)$ 的零解, 因此设 $f(x) = -u(1) + g_{\delta e}(\delta e)$, 自变量 x 为 δe 。

1) 给定迭代初值 δe_0 , 即飞机上一时刻升降舵偏转角度。根据(7)中对 $g_{\delta e}(\delta e)$ 的定义, 设 $f(\delta e) = -u(1) + C_m + C_z(x_{cg_r} - x_{cg_0})$, 并计算出 $f(\delta e_0)$ 。

2) 判断 $f(\delta e_i) < \epsilon, i = 0, 1, \dots$ 是否成立, 其中 ϵ 表示精度, 比如 0.000 000 1。如果不等式成立, 则 u_i 为所得解, 如果不成立, 进行 3)。

3) 更新 $u_i, \delta e_{i+1} = \delta e_i - \frac{f(\delta e_i)}{f'(\delta e_i)}$, 然后执行 2)。

4 仿真验证

仿真验证中, 飞机高度为 3 000 m, 速度为 150 m/s, 给到的每个角速率指令信号具体如下:

$$x_{cmd} = \begin{cases} 0 & (0 \leq t < 8) \\ x_h & (8 \leq t < 15) \\ x_l & (15 \leq t < 22) \\ 0 & (t \geq 22) \end{cases} \quad (27)$$

其中: $p_h = 3, p_l = -3, q_h = 3, q_l = -3, r_h = 1, r_l = -1$, 单位 $^\circ/s$, 并且指令信号经过一个二阶的指令滤波器。取控制器增益 $K = \text{diag}(10, 10, 10)$, 自适应律增益为 $\Gamma = \text{diag}(20, 20, 20)$, 仿真结果如下。

4.1 正常条件下仿真结果

在正常条件下仿真的重心位置保持不变, 仿真结果如图 2~5。

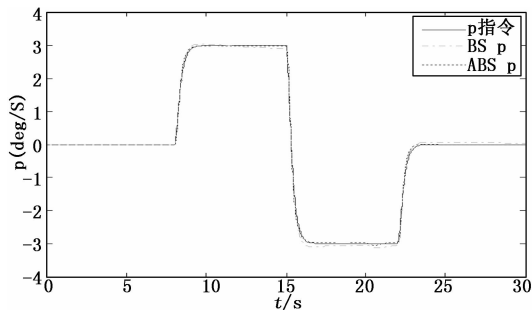


图 2 正常情况下常规 Backstepping 与自适应 Backstepping 滚转角速率比较

从仿真结果可以看出, 在正常情况下, 常规 Backstepping 与自适应 Backstepping 的动态性能性能和稳态性能都十分良好。两种控制律的舵面输出量也基本一致。

4.2 重心位置偏移仿真结果

4.2.1 重心突变时

对于特定的飞机, 在飞行过程中会投掷弹药或空投等负载, 这样会导致重心发生突变, 在这一条件下, 我们在 $t \geq 3$ 时, 将模型重心位置加一个阶跃信号, 仿真结果如图 6~9。

在重心突变时, 仿真这里只改变重心纵向的位置, 因此滚转通道影响较小, 偏航通道有一定的影响, 俯仰通道

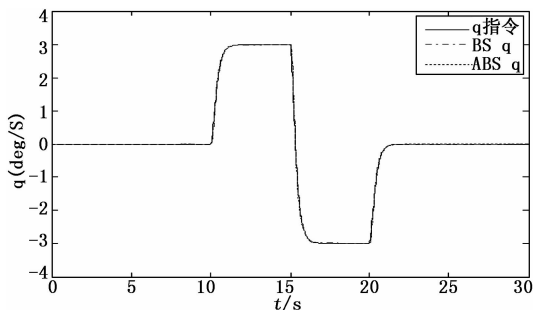


图 3 正常情况下常规 Backstepping 与自适应 Backstepping 俯仰角速率比较

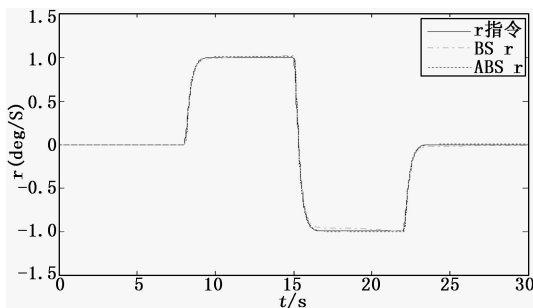


图 4 正常情况下常规 Backstepping 与自适应 Backstepping 偏航角速率比较

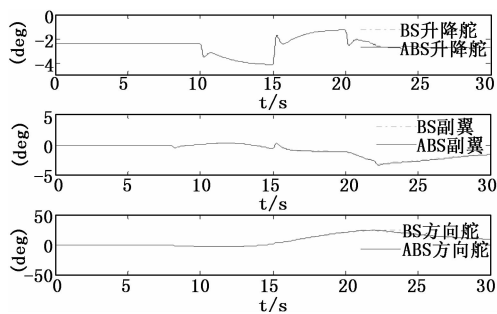


图 5 正常情况下常规 Backstepping 与自适应 Backstepping 舵面输出比较

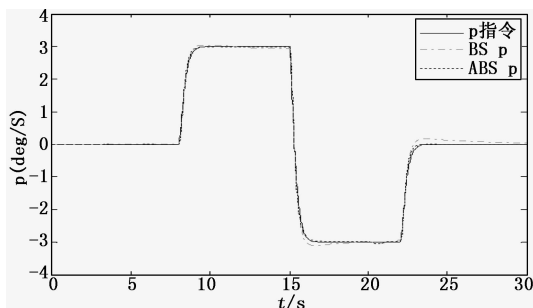


图 6 重心突变时常规 Backstepping 与自适应 Backstepping 滚转角速率比较

影响最大。

从仿真结果可看出, 对于常规 Backstepping, 滚转角速率表现良好, 而俯仰和偏航角速率已经不能稳定了; 对于自适应 Backstepping, 滚转角速率性能更好, 而且俯仰角速率

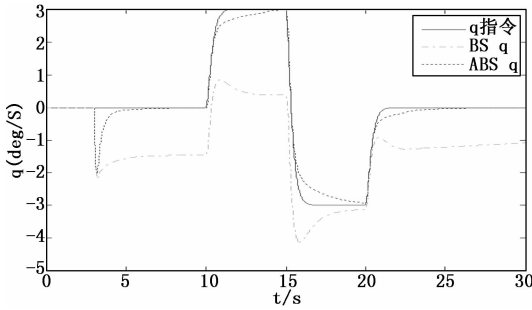


图 7 重心突变时常规 Backstepping 与自适应 Backstepping 俯仰角速率比较

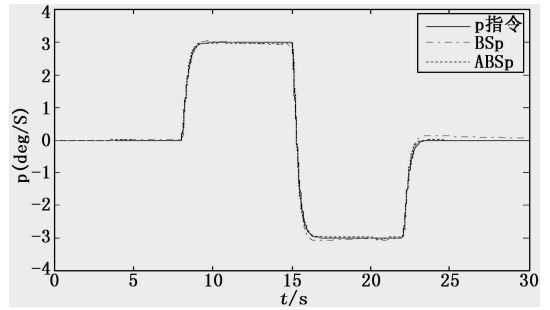


图 10 重心渐变时常规 Backstepping 与自适应 Backstepping 滚转角速率比较

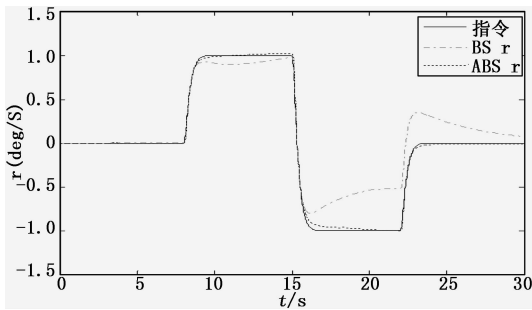


图 8 重心突变时常规 Backstepping 与自适应 Backstepping 偏航角速率比较

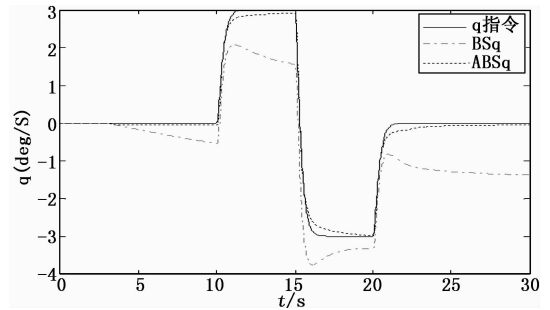


图 11 重心渐变时常规 Backstepping 与自适应 Backstepping 俯仰角速率比较

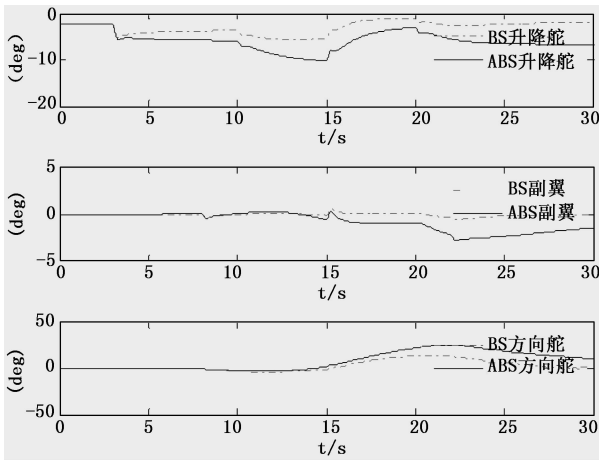


图 9 重心突变时常规 Backstepping 与自适应 Backstepping 舵面输出比较

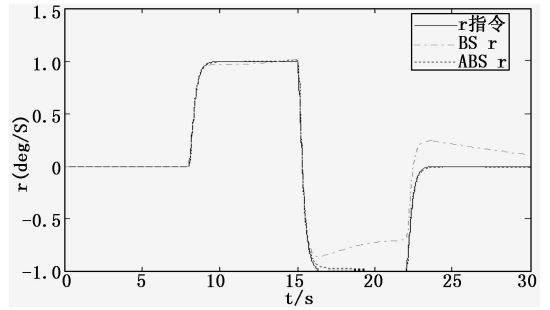


图 12 重心渐变时常规 Backstepping 与自适应 Backstepping 偏航角速率比较

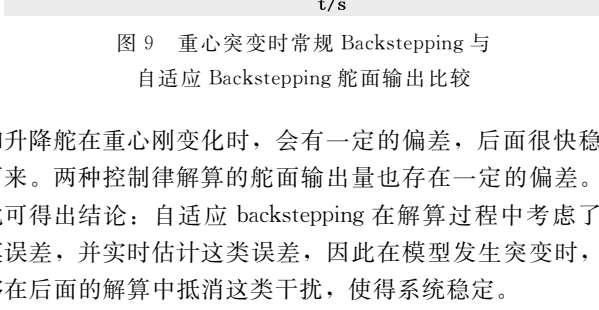


图 13 重心渐变时常规 Backstepping 与自适应 Backstepping 舵面输出比较

和升降舵在重心刚变化时，会有一定的偏差，后面很快稳定下来。两种控制律解算的舵面输出量也存在一定的偏差。由此得出结论：自适应 backstepping 在解算过程中考虑了建模误差，并实时估计这类误差，因此在模型发生突变时，能够在后面的解算中抵消这类干扰，使得系统稳定。

4.2.2 重心渐变时

在飞行过程中，油耗会导致重心有一个缓慢的变化过程^[10]，在这一过程中，我们在 $t \geq 3$ 时，将模型重心位置加一个斜坡信号，仿真结果如图 10~13。

在重心渐变时，仿真这里只改变重心纵向的位置，因此滚转通道影响较小，偏航通道有一定的影响，俯仰通道

影响最大。

从仿真结果可看出，即使是重心变化是渐变过程，常规
(下转第 105 页)