

國防科技學術合作協調小組研究計畫成果報告

錯誤診斷與可靠度控制

Fault Detection and Reliable Control Design

計畫編號：NSC 91-2623-7009-010

執行期間：91 年 1 月 1 日 至 91 年 12 月 31 日

計畫主持人：梁 耀 文

共同主持人：廖 德 誠

執行單位：國 立 交 通 大 學

中華民國 九 十 一 年 十 二 月 三 十 一 日

目 錄

目錄	I
圖例目錄	III
摘要	V
第一章 前言	1
1.1 研究緣起與背景	1
1.2 研究目的與研究動機	3
1.3 研究方法與步驟	4
1.4 研究成果	5
第二章 飛彈系統動態數學模式	7
2.1 座標軸系統	7
2.1.1 慣性座標系	7
2.1.2 體座標系(參考座標系)	7
2.2 飛彈系統	7
第三章 錯誤偵測與診斷觀察器之設計	16
3.1 問題描述	16
3.2 解藕型式的轉換	17
3.3 觀察器的設計	17
第四章 可靠度控制器設計	23
4.1 問題描述	23
4.2 被動式可靠度控制設計	25
4.3 主動式可靠度控制設計	28
第五章 數值模擬結果與討論	31
5.1 飛彈參數設定	32

5.2 觀察器設計	32
5.3 控制器設計	35
5.4 控制器參數及初始值設定	39
5.5 模擬結果與討論	40
第六章 結論	63
參考文獻	65
附錄 A	67

圖例目錄

圖 3.1	觀察器系統架構圖	17
圖 5.1	正規設計與針對致動器 1 卡死所設計之被動式控制器(模擬全部致動器正常)之 (a)狀態比較圖 (b)輸出角度比較圖	43
圖 5.2	正規設計與針對致動器 2 卡死所設計之被動式控制器(模擬全部致動器正常)之 (a)狀態比較圖 (b)輸出角度比較圖	44
圖 5.3	正規設計與針對致動器 3 卡死所設計之被動式控制器(模擬全部致動器正常)之 (a)狀態比較圖 (b)輸出角度比較圖	45
圖 5.4	正規設計與針對致動器 4 卡死所設計之被動式控制器(模擬全部致動器正常)之 (a)狀態比較圖 (b)輸出角度比較圖	46
圖 5.5	針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第一個致動器 δ_1 發生故障: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三 個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	47
圖 5.6	針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第二個致動器 δ_2 發生故障: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三 個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	48
圖 5.7	針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第三個致動器 δ_3 發生故障: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三 個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	49
圖 5.8	針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第四個致動器 δ_4 發生故障: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三 個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	50
圖 5.9	主動式控制針對第一個致動器 δ_1 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	51
圖 5.10	主動式控制針對第二個致動器 δ_2 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	52
圖 5.11	主動式控制針對第三個致動器 δ_3 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip	

angle	53
圖 5.12 主動式控制針對第四個致動器 δ_4 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	54
圖 5.13 針對第一個致動器 δ_1 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	55
圖 5.14 針對第二個致動器 δ_2 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	56
圖 5.15 針對第三個致動器 δ_3 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	57
圖 5.16 針對第四個致動器 δ_4 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle	58
圖 5.17 主動式及被動式可靠度控制器在第一個致動器 δ_1 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle	59
圖 5.18 主動式及被動式可靠度控制器在第二個致動器 δ_2 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle	60
圖 5.19 主動式及被動式可靠度控制器在第三個致動器 δ_3 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle	61
圖 5.20 主動式及被動式可靠度控制器在第四個致動器 δ_4 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle	62

摘要

近年來，已有許多國內外專家學者投注於可靠度控制的研究，其設計的目的是希望當系統某些特定的控制元件（如致動器 actuator 與偵測器 sensor）發生不正常狀況（包括老化及故障現象）時，系統仍能維持正常穩定的運作且達到基本的性能要求。在此計劃中，我們針對飛彈系統研究可靠度之設計法則，除了完成被動式可靠度控制器設計外，也提出利用錯誤偵測與診斷機制以達成主動式可靠度控制之目標。本計畫已將此兩種設計運用在飛彈系統，並經數值模擬驗證該設計確實可行。

1 前言

1.1 研究緣起與背景

隨著科技的進步，應用於各個領域的系統之設備規模、投資金額及複雜程度已明顯的大幅提昇。人們面對如此複雜之系統時，對於系統之安全性、可靠性及有效性的要求也愈來愈殷切。尤其是應用於國防武器如飛彈、飛機、衛星及具有高危險性之系統，如核電廠、煉油廠等等，更有可能因系統之不穩定而產生重大之災難。因此，如何有效的提昇系統之安全、可靠及有效性是一重要且迫切的研究課題。而藉由發展錯誤偵測與診斷技術及設計可靠度控制器為一直接且有效之重要途徑。

近年來，已有許多國內外的學者專家投注在可靠度控制的研究上，並已提出許多研究成果[1, 2, 4-6]，然而這些研究成果大多集中在理論上的分析，應用到實際系統的例子並不多。此外，這些研究成果多半偏重於處理線性系統的可靠度控制議題。例如，Vidyasagar 及 Viswanadham 在頻域 (frequency domain) 上藉由穩定互質技巧 (stable coprime factorization technique) 提出如何設計額外的

一組控制器，使得擴充系統之兩組控制器中任何一組失效而另一組正常時系統仍能維持穩定運作[3]；Veillette 使用修正 LQ 迴授控制 (modified LQ regulator) 及代數 Riccati 方程式的處理手法發展出一套可靠度控制器的設計準則，在此設計下的控制系統能容忍某些特定的致動器產生不正常的情況而仍能維持穩定運作且達到預估的二次性能表現[4]；在非線性可靠控制方面，Liang[1] 等人將 Veillette[4] 在線性系統所得到的結果推廣到非線性系統，Yang 等人則利用 H_{∞} 最佳化的方法探討非線性系統某些特定元件失效時如何設計可靠度控制器使系統仍能保證穩定及維持 H_{∞} 性能表現的條件 [5, 6]。

雖然可靠度控制可應用的層面非常廣，但現階段應用到實際系統的例子並不多，尤其是應用在飛彈系統轉向控制的問題上。由於飛彈系統的發展在國防科技工業上扮演著很重要的角色。因此，如何有效的讓飛彈能快速追蹤到設定之軌跡且穩定的飛行是極實際且重要的研究課題。本計畫把可靠度控制理論應用於飛彈系統的控制上，研究使飛彈系統在部分彈翼故障的情況下，仍能達到軌跡追蹤的性能。

1.2 研究目的與研究動機

可靠度控制的目的是希望當系統某些特定的控制元件(如致動器 actuator 與偵測器 sensor)發生不正常狀況(包括老化及故障現象)時,系統仍能維持正常穩定的運作且達到基本的性能要求。此議題肇因於控制元件無法避免老化及故障現象的發生,這些現象可能導致系統發生不穩定而無法順利完成任務,甚至可能因而造成嚴重的傷害。例如,飛彈執行攔截任務時,部分彈翼可能無法順利操控或導引雷達及無線電波無法發揮正常功能而導致任務失敗;人造衛星執行任務時,也可能因噴嘴或機械故障無法順利定位或執行其他任務;高危險區域如核電廠、煉鋼廠、煉油廠之系統也有可能因機件故障而產生大災難。解決此問題的一個可行方法是當不正常現象發生時,適時的切換控制法則以避免不穩定現象的發生。然而,這麼一來不但需要增加偵測器使系統的複雜度不斷提高,更需要考慮偵測器的可靠度問題,也就是要考慮偵測器也有可能發生故障的問題。因此,本計劃的目標就是希望在不增加系統複雜度的情況下建立錯誤偵測與診斷機制及設計可靠度控制法則,使得系統在某些特定的控制元件發生不正常現象時,能即時且有效偵測到錯誤的訊息並即時切換控制法則,使系統仍能維持正常穩定的運作且達到基本的性能表現。

在應用方面，由於飛彈進行攔截任務時，一旦其彈翼發生故障，是不可能即時修復的且會因無法達成任務而造成龐大的損失。因此，如何使飛彈在飛行期間有部份致動器發生故障的情況下仍能穩定飛行是極重要的課題。在本計劃中，我們針對飛彈系統之尾翼控制進行錯誤偵測與診斷及可靠度控制理論的分析與模擬驗證。

1.3 研究方法與步驟

可靠度控制一般分為主動式可靠度控制(Active reliable control)和被動式可靠度控制(Passive reliable control)兩種。主動式可靠度控制為透過設計錯誤偵測與診斷機制隨系統當時的狀況組織控制法則，其優點為能充分發揮系統在各種狀態下之性能表現，而缺點則是必須增加監測器、使成本提升及提高系統複雜度，同時必需考慮監測器之可靠度問題及診斷機制可能產生誤報及漏報之狀況。不適用於當錯誤發生時容許反應時間極短之系統。另一方面，被動式可靠度控制則強調設計固定之控制器使閉迴路系統能容忍特定異常現象之發生。其優點為不需額外花費時間及成本作錯誤偵測、診斷及控制器重組之工作，對於當異常現象發生時容許反應時間極短之系統特別重要，然而其缺點則為必需犧牲正常狀況之部分性能表現。

在本計劃中，我們除了進行被動式主動式可靠度控制之研究外，也探討相關錯誤診斷機制來達成主動式可靠度控制的目標。

為了避免 LQR 及 H_{∞} 方法必須求解 Hamilton-Jacobi 方程式之困難及計算負擔，本研究計畫利用可變結構控制(Variable Structure Control)的概念提出設計飛彈系統之可靠度控制器的方法，所設計的控制器不但能使系統在正常情況下達到穩定及追蹤到所想要的軌跡，也能使故障系統達到同樣的性能表現。首先是設計被動式可靠度控制器，針對系統某些致動器故障或損壞的狀況下分別設計主動式及被動式可靠度控制器。此外，也設計一錯誤診斷機制使其能有效且正確地偵測到飛彈尾翼的狀況，並即時切換至主動式可靠度控制器，使系統達到良好的性能要求。最後，將這些理論結果應用於飛彈系統尾翼之可靠度控制分析上。

1.4 研究成果

本計劃的目標是希望在不增加系統的複雜度及負擔的情況下，針對飛彈系統研究可靠度之設計法則，使得系統在某些特定的元件發生不正常現象時，仍能維持正常穩定的運作且達到基本的性能表現，本計畫除了完成被動式可靠度控制設計，也提出錯誤診斷機制以達成主

動式可靠度控制之目標。此兩種設計已成功運用在飛彈系統，並經數值模擬驗證該設計確實可行。

2 飛彈系統動態數學模式

2.1 座標軸系統

2.1.1 慣性座標系

一般而言，如要表示一質點之物理量時，需考慮該質點所參考的座標軸系統，若座標軸在空間上不會隨時間變化，一般定義為以宇宙恆星星體為指標建立，稱此座標系統為慣性座標系統。

2.1.2 體座標系(參考座標系)

體座標系為以物體本身為一座標系，原點定義在物體的質心上，又稱參考座標系，通常在描述物體運動與慣性座標系統的相對關係。

2.2 飛彈系統

在本章節中，我們將介紹 HAVE DASH II 飛彈系統的動態數學模式。此動態模式將以如下的一般式來表示：

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

$$y(t) = h(x(t)) \quad (2.1)$$

根據上述之型式，我們將對HAVE DASH II飛彈模式做詳細的說明。首先介紹飛彈相對於飛彈本體座標之剛體運動方程式，描述如下：

$$\begin{aligned} \dot{P} &= -\frac{I_{zz} - I_{yy}}{I_{xx}}QR + \frac{L}{I_{xx}} \\ \dot{Q} &= -\frac{I_{xx} - I_{zz}}{I_{yy}}PR + \frac{M}{I_{yy}} \\ \dot{R} &= -\frac{I_{yy} - I_{xx}}{I_{zz}}PQ + \frac{N}{I_{zz}} \\ \dot{\Psi} &= \sin \Phi \sec \Theta Q + \cos \Phi \sec \Theta R \\ \dot{\Theta} &= \cos \Phi Q - \sin \Phi R \\ \dot{\Phi} &= P + \sin \Phi \tan \Theta Q + \cos \Phi \tan \Theta R \\ \dot{V}_m &= \frac{1}{m} \{ \cos \alpha \cos \beta (F_x + g_x) + \sin \beta (F_y + g_y) \} \\ \dot{\alpha} &= Q - (P \cos \alpha + R \sin \alpha) \tan \beta \\ &\quad + \frac{1}{mV_m \cos \beta} \{ -\sin \alpha (F_x + g_x) + \cos \alpha (F_z + g_z) \} \\ \dot{\beta} &= P \sin \alpha - R \cos \alpha \\ &\quad + \frac{1}{mV_m} \{ -\cos \alpha \sin \beta (F_x + g_x) + \cos \beta (F_y + g_y) - \sin \alpha \sin \beta (F_z + g_z) \} \end{aligned} \quad (2.2)$$

其中受重力場影響部份為

$$g_x = -mg \sin \Theta$$

$$\begin{aligned}
g_y &= mg \cos \Theta \sin \Phi \\
g_z &= mg \cos \Theta \cos \Phi
\end{aligned} \tag{2.3}$$

P、Q、R 分別代表飛彈本體座標 x、y、z 軸三個方向的角速度

V_m 、 α 和 β 分別代表相對於體座標的飛彈速度、攻角及側滑角

Θ 、 Φ 、 Ψ 分別代表飛彈之傾斜角、轉動角、偏離角

I_{xx} 、 I_{yy} 、 I_{zz} 分別代表體座標 x、y、z 軸三個方向的轉動慣量

m 表飛彈質量，g 表重力常數

(F_x, F_y, F_z) 和 (L, M, N) 分別代表相對於體座標的氣動力和轉動力

矩、傾斜力矩和偏離力矩

它們分別滿足下列的式子：

$$\begin{aligned}
F_x &= k_F \rho V_m^2 C_x & L &= k_M \rho V_m^2 C_l \\
F_y &= k_F \rho V_m^2 C_y & M &= k_M \rho V_m^2 C_m \\
F_z &= k_F \rho V_m^2 C_z & N &= k_M \rho V_m^2 C_n
\end{aligned} \tag{2.4}$$

其中 ρ 為空氣密度； k_F 和 k_M 是由飛彈幾何尺寸所決定的常數；

C_x, C_y, C_z, C_l, C_m 和 C_n 為氣動力參數並表示成氣動力表的形式。

氣動力參數 $C_i, i = x, y, z, l, m, n$ 的函數形式可以表示如下：

$$\begin{aligned}
C_i &= C_{i0}(\alpha, \beta, M_m) + C_{i\delta_e}(\alpha, \delta_e, M_m) \\
&\quad + C_{i\delta_a}(\alpha, \delta_a, M_m) + C_{i\delta_r}(\alpha, \delta_r, M_m)
\end{aligned}$$

其中 $M_m = V_m/c$ 為馬赫數 (c : 聲速)。 δ_e, δ_a 和 δ_r 為相對的有效傾斜、滾動、偏離尾翼控制偏角。它們隨著致動器和尾翼調結器所下的偏角 $\delta_{ec}, \delta_{ac}, \delta_{rc}$ 命令而改變。

HAVE DASH II 飛彈系統有四個分布不對稱的尾翼。根據下列式子控制偏角產生四個尾翼偏角 δ_{ic} 如下：

$$\begin{aligned}\delta_{1c} &= \delta_{ec} - \delta_{ac} + \delta_{rc} \\ \delta_{2c} &= -\delta_{ec} - \delta_{ac} + \delta_{rc} \\ \delta_{3c} &= -\delta_{ec} - \delta_{ac} - \delta_{rc} \\ \delta_{4c} &= \delta_{ec} - \delta_{ac} - \delta_{rc}\end{aligned}\tag{2.5}$$

根據下列式子，可以得到結合實際尾翼偏角所產生的等效傾斜角、滾動角及偏離角：

$$\begin{aligned}\delta_e &= \frac{1}{4}(\delta_1 - \delta_2 - \delta_3 + \delta_4) \\ \delta_a &= -\frac{1}{4}(\delta_1 + \delta_2 + \delta_3 + \delta_4) \\ \delta_r &= \frac{1}{4}(\delta_1 + \delta_2 - \delta_3 - \delta_4)\end{aligned}\tag{2.6}$$

為了把飛彈模型描述成如式 (2.1) 的形式，我們需把氣動力係數

根據下列函數做近似：

$$\begin{aligned} \begin{pmatrix} C_x \\ C_y \\ C_z \end{pmatrix} &= \begin{pmatrix} C_{x0} \\ C_{y0} \\ C_{z0} \end{pmatrix} + \begin{pmatrix} C_{xe} & C_{xa} & C_{xr} \\ C_{ye} & C_{ya} & C_{yr} \\ C_{ze} & C_{za} & C_{zr} \end{pmatrix} \begin{pmatrix} \delta_e \\ \delta_a \\ \delta_r \end{pmatrix} \\ &= C_{F0} + C_{Fu}u \end{aligned} \quad (2.7)$$

$$\begin{aligned} \begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix} &= \begin{pmatrix} C_{l0} \\ C_{m0} \\ C_{n0} \end{pmatrix} + \begin{pmatrix} C_{le} & C_{la} & C_{lr} \\ C_{me} & C_{ma} & C_{mr} \\ C_{ne} & C_{na} & C_{nr} \end{pmatrix} \begin{pmatrix} \delta_e \\ \delta_a \\ \delta_r \end{pmatrix} \\ &= C_{M0} + C_{Mu}u \end{aligned} \quad (2.8)$$

矩陣 $C_{F0}, C_{Fu}, C_{M0}, C_{Mu}$ 為 α, β, M_m 的函數。根據最小平方誤差

準則，利用簡單的曲線近似系統可得：

$$\begin{aligned} C_{F0} &= \begin{pmatrix} -0.57 + 0.0083\alpha \\ -0.21\beta \\ C_{z0}(\beta, M_m) \end{pmatrix} & C_{Fu} &= \begin{pmatrix} 0.004 & 0 & 0 \\ 0 & 0 & 0.08 \\ -0.09 & 0 & 0 \end{pmatrix} \\ C_{M0} &= \begin{pmatrix} 0.116\beta \\ C_{m0}(\alpha, M_m) \\ 0.08\beta \end{pmatrix} & C_{Mu} &= \begin{pmatrix} 0 & -0.127 & 0 \\ -0.675 & 0 & 0 \\ 0 & 0 & -0.584 \end{pmatrix} \end{aligned} \quad (2.9)$$

其中 $C_{z0}(\beta, M_m)$ 和 $C_{m0}(\alpha, M_m)$ 如下所示：

$$\begin{aligned} C_{z0}(\alpha, M_m) &= C_{z1}(\alpha) + C_{z2}(\alpha)M_m \\ C_{m0}(\alpha, M_m) &= C_{m1}(\alpha) + C_{m2}(\alpha)M_m \end{aligned} \quad (2.10)$$

且

$$\begin{aligned} C_{z1}(\alpha) &= -0.0015\alpha^3 + 0.0125\alpha^2 - 0.5052\alpha + 0.0429 \\ C_{z2}(\alpha) &= 0.0006\alpha^3 - 0.0138\alpha^2 + 0.123\alpha - 0.0191 \\ C_{m1}(\alpha) &= -0.0055\alpha^3 + 0.2131\alpha^2 - 2.7419\alpha - 0.0381 \\ C_{m2}(\alpha) &= 0.0014\alpha^3 - 0.0623\alpha^2 + 0.8715\alpha - 0.4041 \end{aligned} \quad (2.11)$$

為使讓設計較簡單，我們做了以下的假設

- 忽略致動器的動態，即假設 $\delta_e = \delta_{ec}, \delta_a = \delta_{ac}, \delta_r = \delta_{rc}$
- 忽略 V_m 的變動，假設飛彈定速飛行，即 $\dot{V}_m = 0$
- 忽略氣動力和尾翼控制偏角的耦合現象，即假設 $C_{Fu} = 0$

前兩項假設主要在便於設計，其通常實現在自動駕駛設計上。第三項假設是基於觀察氣動力和控制偏角耦合現象相對於轉動力矩和控制偏角的耦合現象小很多，故做此假設，因為副翼、方向舵、升降舵是主要力矩產生裝置。

根據以上的假設，飛彈系統可化成如 (2.1) 的標準形式，其中

$$x = [P \ Q \ R \ V_m \ \alpha \ \beta \ \Psi \ \Phi \ \Theta]^T,$$

$$u = [\delta_e \quad \delta_a \quad \delta_r]^T,$$

$$y = [P_s \quad \alpha \quad \beta]^T,$$

$$P_s = P \cos \alpha + \sin \alpha \tag{2.12}$$

$$\begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \\ f_7(x) \\ f_8(x) \\ f_9(x) \end{pmatrix} = \begin{pmatrix} \sin \Phi \sec \Theta Q + \cos \Phi \sec \Theta R \\ \cos \Phi Q - \sin \Phi R \\ P + \sin \Phi \tan \Theta Q + \cos \Phi \tan \Theta R \\ \frac{1}{m} \{ \cos \alpha \cos \beta (F_x + g_x) + \sin \beta (F_y + g_y) + \sin \alpha \cos \beta (F_z + g_z) \} \\ Q - (P \cos \alpha + R \sin \alpha) \tan \beta + \frac{1}{m V_m \cos \beta} \{ -\sin \alpha (F_x + g_x) + \cos \alpha (F_z + g_z) \} \\ P \sin \alpha - R \cos \alpha + \frac{1}{m V_m} \{ -\cos \alpha \sin \beta (F_x + g_x) + \cos \beta (F_y + g_y) - \sin \alpha \sin \beta (F_z + g_z) \} \\ -\frac{I_{zz} - I_{yy}}{I_{xx}} QR + \frac{L}{I_{xx}} \\ -\frac{I_{xx} - I_{zz}}{I_{yy}} PR + \frac{M}{I_{yy}} \\ -\frac{I_{yy} - I_{zz}}{I_{xx}} PQ + \frac{N}{I_{xx}} \end{pmatrix}$$

$$\tag{2.13}$$

$$g(x) = 0.25K_m\rho V_m^2 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{C_{le}-C_{la}+C_{lr}}{I_{xx}} & \frac{-C_{le}-C_{la}+C_{lr}}{I_{xx}} & \frac{-C_{le}-C_{la}-C_{lr}}{I_{xx}} & \frac{C_{le}-C_{la}-C_{lr}}{I_{xx}} \\ \frac{C_{le}-C_{la}+C_{lr}}{I_{yy}} & \frac{-C_{le}-C_{la}+C_{lr}}{I_{yy}} & \frac{-C_{le}-C_{la}-C_{lr}}{I_{yy}} & \frac{C_{le}-C_{la}-C_{lr}}{I_{yy}} \\ \frac{C_{le}-C_{la}+C_{lr}}{I_{zz}} & \frac{-C_{le}-C_{la}+C_{lr}}{I_{zz}} & \frac{-C_{le}-C_{la}-C_{lr}}{I_{zz}} & \frac{C_{le}-C_{la}-C_{lr}}{I_{zz}} \end{pmatrix} \quad (2.14)$$

P_s 代表針對穩定軸之滾動速率 (roll rate with respect to stability axis)。

經過簡單的推導我們可證明此系統的 relative degree 為 (1,2,2)。

計算後可推得下列的輸入-輸出關係式：

$$\begin{pmatrix} \dot{P}_s \\ \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} = \begin{pmatrix} B_1 \\ B_2 \\ B_3 \end{pmatrix} + \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} \delta_e \\ \delta_a \\ \delta_r \end{pmatrix} \\ = B(x) + A(x)u \quad (2.15)$$

其中 $A(x)$ 為

$$A(x) = k_{M\rho} V_m^2 \begin{pmatrix} \frac{\cos \alpha}{I_{xx}} & 0 & \frac{\sin \alpha}{I_{yy}} \\ -\frac{\cos \alpha \tan \beta}{I_{xx}} & \frac{1}{I_{yy}} & -\frac{\sin \alpha \tan \beta}{I_{xx}} \\ \frac{\sin \alpha}{I_{xx}} & 0 & -\frac{\cos \alpha}{I_{xx}} \end{pmatrix} C_{Mu} \quad (2.16)$$

我們可以檢查矩陣 $A(x)$ 永遠是可逆的 (invertible)，此特性有利於往後的控制器設計。

3 錯誤偵測與診斷觀察器之設計

3.1 問題描述

在此計劃中我們考慮如下之非線性系統

$$\dot{x} = f(x) + g(x)u, \quad f(0) = 0 \quad (3.1)$$

其中 $x \in R^n$ 代表狀態變數 (State Variable)， $u \in R^m$ 代表控制輸入 (Control input)， $g(x) = (g_1(x), \dots, g_m(x)) \in R^{n \times m}$ ，且 $f(x)$ 及 $g_i(x)$ 都是平滑向量場 (Smooth Vector Field)。本計劃之錯誤偵測與診斷的主要目標為設計一觀察器使得當致動器發生故障時能夠即時的偵測出來，且能成功的診斷出哪一個致動器損壞以提供主動式可靠度控制切換控制律的依據。

在本計畫中，我們將透過設計觀察器來進行錯誤偵測與診斷的工作，並由觀察器所產生的 residual 來判斷系統故障的產生及故障的來源。整個錯誤偵測與診斷系統描述如下圖：

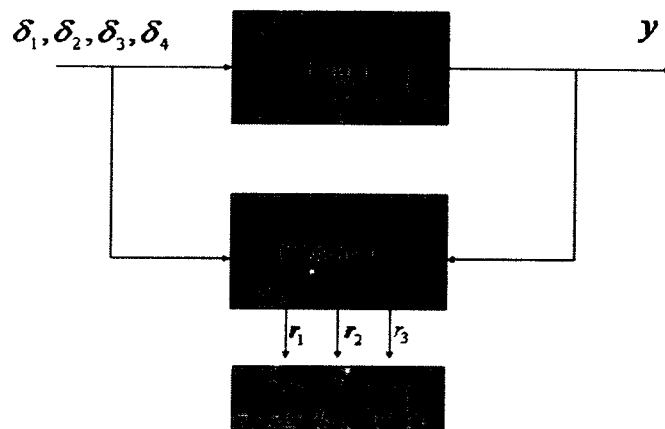


圖 3.1

在圖 3.1 中， $\delta_1, \delta_2, \delta_3, \delta_4$ 分別為四個致動器，而 Plant 為所要被觀測的系統，在此為飛彈系統。藉由輸出 y 以及四個致動器的輸入來設計觀察器，我們要求此觀察器所產生的 residual signal 可以偵測出致動器的故障並診斷出故障的來源。

3.2 解藕型式的轉換

由第二章飛彈系統模型的描述，可以得知飛彈系統為九個狀態 $x = (\psi, \Theta, \Phi, V_m, \alpha, \beta, P, Q, R)$ 所組成的系統。假設其中 α, β, P, Q, R 五個狀態是可以量測得到的訊號，將飛彈的模型化為(3.1)式的型式，化成的結果如下：

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1 \cdots x_9) \\
&\vdots \\
\dot{x}_6 &= f_6(x_1 \cdots x_9) \\
\begin{pmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \end{pmatrix} &= \begin{pmatrix} f_7(x_5 \cdots x_9) \\ f_8(x_5 \cdots x_9) \\ f_9(x_5 \cdots x_9) \end{pmatrix} + \begin{pmatrix} g_{11} \cdots g_{14} \\ g_{21} \cdots g_{24} \\ g_{31} \cdots g_{34} \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{pmatrix} \quad (3.2)
\end{aligned}$$

將原來的模型化為(3.2)式的型式後，可以清楚的看到只有四個致動器輸入只出現在 x_7, x_8, x_9 三個狀態方程式中。為了達到我們所希望的解藕現象，我們針對原系統作變數變換

$$Z = \bar{M} X \quad (3.3)$$

其中

$$\bar{M} = \begin{pmatrix} I_{6 \times 6} & \vdots & 0 \\ \dots & \vdots & \dots \\ 0 & \vdots & M_{3 \times 3} \end{pmatrix} \quad (3.4)$$

$$M_{3 \times 3} = \begin{pmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{pmatrix}^{-1} \quad (3.5)$$

新的狀態方程為

$$\begin{aligned}
\dot{Z} &= \bar{M} \dot{X} = \bar{M} f(x) + \bar{M} g(x)u \\
&= \bar{M} f(\bar{M}^{-1} Z) + \bar{M} g(x) \\
&= f_{new}(z) + \bar{M} g(x)
\end{aligned} \quad (3.6)$$

其中 $g(x)$, $f_{new}(z)$ 與 $\bar{M}g(x)$ 分別為

$$g(x) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \end{pmatrix} \quad (3.7)$$

$$f_{new}(z) = \begin{pmatrix} f_1(z) \\ f_2(z) \\ f_3(z) \\ f_4(z) \\ f_5(z) \\ f_6(z) \\ f_{7\ new}(z) \\ f_{8\ new}(z) \\ f_{9\ new}(z) \end{pmatrix} \quad (3.8)$$

$$\bar{M}g(x) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & l_1 \\ 0 & 1 & 0 & l_2 \\ 0 & 0 & 1 & l_3 \end{pmatrix} \quad (3.9)$$

在此設計下，任何一個單一致動器故障都可經由所產生的 residual 信號偵測及診斷出來。實際上，當第一個致動器 δ_1 故障時，residual 信號將會出現 $|r_1| \neq 0$ 但 $|r_2| = |r_3| = 0$ ；第二個致動器 δ_2 故障時， $|r_2| \neq 0$ 但 $|r_1| = |r_3| = 0$ ；第三個致動器 δ_3 故障時， $|r_3| \neq 0$ 但 $|r_1| = |r_2| = 0$ ；第四個致動器 δ_4 故障時， $|r_1| \neq 0$ 、 $|r_2| \neq 0$ 且 $|r_3| \neq 0$ 。如此一來，錯誤訊號就可以被解耦並且成功的被偵測診斷出來。詳細情況說明如下：

考慮第一個致動器 δ_1 出現卡死的情形時，假設實際卡死的角度為 δ_1^* ，我們將錯誤訊號以如下之 m_1 來表示

$$m_1 = \delta_1^* - \delta_1 \quad (3.13)$$

在此情況下， Z_7 的狀態方程式變成

$$\dot{Z}_7 = f_{7_{new}} + \delta_1 + l_1 \delta_4 + m_1 \quad (3.14)$$

由(3.10)、(3.11)、(3.12)式可以得到誤差訊號

$$\begin{aligned} e_1 &= Z_7 - \xi_1 \\ \dot{e}_1 &= -k_1 e_1 + m_1 \end{aligned} \quad (3.15)$$

根據前面所提到 $k_1 > 0$ ，所以只要適當的選取 $k_1 > 0$ ，則經過一段短暫態時間後 e_1 會趨近至 $-\frac{m_1}{k_1}$ 。從(3.12)及(3.15)式可以看出 residual

signal r_1 只受到錯誤訊號 m_1 的影響，並且

$$m_1 \neq 0 \Leftrightarrow r_1 \neq 0 \quad (3.16)$$

同樣地，定義

$$m_i = \delta_i^* - \delta_i, i=1,2,3,4 \quad (3.17)$$

$$e_2 = z_8 - \xi_2 \quad (3.18)$$

$$e_3 = z_9 - \xi_3 \quad (3.19)$$

則當第 i 個致動器， $i=1, 2, 3$ ，發生卡死情況時，

$$\dot{e}_i = -k_i e_i + m_i \quad (3.20)$$

而當第 4 個致動器發生卡死時，

$$\dot{e}_i = -k_i e_i + m_4, i=1,2,3 \quad (3.21)$$

由(3.20)及(3.21)可以清楚的看到當 δ_i ， $i=1, 2, 3$ ，發生卡死現象時只有 $m_i \neq 0$ 及 $r_i \neq 0$ 其餘 $r_j = 0, j \neq i$ 。但若 δ_4 發生故障時表示 $m_4 \neq 0$ ，由(3.21)知對所有 $i=1,2,3$ ， $e_i \neq 0$ ，也就是說所有 residual 信號 r_i 都會反映出 δ_4 的故障。因此，透過監測 residual 信號可有效的偵測並診斷單一尾翼發生故障的狀況。

4 可靠度控制器設計

4.1 問題描述

首先考慮一個非線性控制系統如下：

$$\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + G_0(\mathbf{x})\mathbf{u}, \quad (4.1)$$

$$\text{且 } \mathbf{y} = \mathbf{h}_0(\mathbf{x}), \quad (4.2)$$

其中 $\mathbf{x} \in \mathbb{R}^n$ ， $\mathbf{u} \in \mathbb{R}^m$ 與 $\mathbf{y} \in \mathbb{R}^p$ 分別表示系統狀態變數，控制輸入與系統輸出， $\mathbf{f}_0(\cdot)$ ， $G_0(\cdot)$ 與 $\mathbf{h}_0(\cdot)$ 假設為平滑向量場。除此之外，我們假設控制系統擁有下列特性：

假設 1: 系統 (4.1)-(4.2) 有相對維度 (k_1, \dots, k_n) (定義見 [7]) 且 $m > p$ 。

在假設 1 中表示系統式 (4.1)-(4.2) 具有以下輸入-輸出的數學關

係式：

$$\begin{pmatrix} y_1^{(k_1)} \\ y_2^{(k_2)} \\ \vdots \\ y_p^{(k_p)} \end{pmatrix} = \mathbf{f}(\mathbf{x}) + G(\mathbf{x})\mathbf{u} + \mathbf{d}, \quad (4.3)$$

其中 $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^p$ ， $\mathbf{d} \in \mathbb{R}^p$ ， $G : \mathbb{R}^n \rightarrow \mathbb{R}^{p \times m}$ 且 $G(\mathbf{x})$ 為全秩 (full rank)。值得注意的是，我們已經介紹過在式 (4.3) 中的 \mathbf{d} 用來表現可能出現的不確定模式，量測雜訊跟外部干擾。除此之外，在假設 1 中 $m > p$ 這種情況表示控制系統在輸出追蹤任務中將有多餘的控制元件。在不失任何一般性的情況下，我們假設 $(\mathbf{g}_1(\mathbf{x}) \cdots \mathbf{g}_p(\mathbf{x}))$ 是一個非奇異矩陣且容許 u_{p+1}, \dots, u_m 是可以發生錯誤的，在此定義 $\mathbf{g}_i(\mathbf{x})$ 為矩陣 $G(\mathbf{x})$ 的第 i 個行。

此研究的主要目的是綜合一種控制定律能使得即使當系統致動器 u_{p+1}, \dots, u_m 發生錯誤時，輸出追蹤性能仍然可以被達成。也就是說，當 $t \rightarrow \infty$ 時可達成 $y_i(t) \rightarrow y_{id}(t)$ 對所有 $i = 1, \dots, p$ 而言，其中 y_{id} 是指參考輸出軌跡。

4.2 被動式可靠度控制設計

首先假設致動器 u_{p+1}, \dots, u_m 發生錯誤。則式 (4.3) 將變為

$$\begin{pmatrix} y_1^{(k_1)} \\ y_2^{(k_2)} \\ \vdots \\ y_p^{(k_p)} \end{pmatrix} = \mathbf{f}(\mathbf{x}) + G(\mathbf{x})\mathbf{u} + \mathbf{d} + \sum_{i=p+1}^m \mathbf{g}_i(\mathbf{x})[u_j^* - u_j], \quad (4.4)$$

其中 u_j^* 為第 j 個控制輸入的實際值且 u_j 為我們設計想要達到之控制。接下來定義輸出追縱誤差為

$$e_i(t) = y_i(t) - y_{id}(t) \quad (4.5)$$

與順滑曲面 (sliding surface) 為

$$s_i(t) = e_i^{(k_i-1)}(t) + a_{i(k_i-1)}e_i^{(k_i-2)}(t) + \dots + a_{i2}\dot{e}_i(t) + a_{i1}e_i(t) \quad (4.6)$$

其中 $i = 1, \dots, p$ 。此處， a_{ij} , $i = 1, \dots, p$ 與 $j = 1, \dots, k_i - 1$, 為選擇的常數能使得下列式子為 Hurwitz 多項式 (也就是說系統所有的特徵值都在 s 平面的左半平面)

$$\lambda^{(k_i-1)} + a_{i(k_i-1)}\lambda^{(k_i-2)} + \dots + a_{i2}\lambda + a_{i1}. \quad (4.7)$$

顯然的，如果系統狀態保持在順滑曲面 (sliding surface) 上，則輸出追縱性能將可以被達成。此外，對於所有 $i = 1, \dots, p$ 而言，利

用式(4.4)，(4.5)及(4.6)推導，我們可以得到下列式子

$$\begin{aligned} \dot{s}_i(t) = & f_i(\mathbf{x}) + \sum_{j=1}^p g_{ij}(\mathbf{x})u_j + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j^* + d_i - y_{id}^{(k_i)} \\ & + a_{i(k_i-1)}e_i^{(k_i-1)}(t) + \cdots + a_{i2}\ddot{e}_i(t) + a_{i1}\dot{e}_i(t), \end{aligned} \quad (4.8)$$

此時定義 $g_{ij}(\mathbf{x})$ 為矩陣 $G(\mathbf{x})$ 的第 (i, j) -項。因此完整的控制器可以被設計成以下形式

$$u_i = u_i^{eq} + u_i^{re} \quad \text{for } i = 1, \dots, n, \quad (4.9)$$

此時 u_i^{re} 使系統狀態達到順滑曲面 (sliding surface)，同時 u_i^{eq} 使得順滑曲面 (sliding surface) 成為一個不變的曲面 (invariant manifold)。為了滿足要求，我們假設

$$\begin{pmatrix} u_1^{eq} \\ \vdots \\ u_p^{eq} \end{pmatrix} = -[\mathbf{g}_1, \dots, \mathbf{g}_p]^{-1} \begin{pmatrix} f_1(\mathbf{x}) - y_{1d}^{(k_1)} + a_{1(k_1-1)}e_1^{(k_1-1)}(t) + \cdots + a_{11}\dot{e}_1(t) \\ \vdots \\ f_n(\mathbf{x}) - y_{nd}^{(k_n)} + a_{n(k_n-1)}e_n^{(k_n-1)}(t) + \cdots + a_{n1}\dot{e}_n(t) \end{pmatrix}. \quad (4.10)$$

這也就是說

$$\dot{s}_i(t) = \sum_{j=1}^p g_{ij}(\mathbf{x})u_j^{re} + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j^* + d_i. \quad (4.11)$$

為了保證滿足追蹤到達的條件，我們增加下一個假設：

假設 2: 存在一個非負數的函數 $\rho_i(\mathbf{x}, t)$, $i = 1, \dots, p$ ，使得

$$|d_i + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j^*| \leq \rho_i(\mathbf{x}, t). \quad (4.12)$$

從 (4.11)，我們可以得到

$$s_i(t)\dot{s}_i(t) = s_i(t) \left\{ \sum_{j=1}^p g_{ij}(\mathbf{x})u_j^{re} + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j^* + d_i \right\}. \quad (4.13)$$

接下來選擇

$$\begin{pmatrix} u_1^{re} \\ \vdots \\ u_p^{re} \end{pmatrix} = -[g_1(\mathbf{x}) \cdots g_p(\mathbf{x})]^{-1} \begin{pmatrix} (\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ \vdots \\ (\rho_p + \eta_p) \cdot \text{sgn}(s_p) \end{pmatrix}, \quad (4.14)$$

此時 η_1, \dots, η_p 為所選的正常數。接下來我們獲得以下結果

$$s_i(t)\dot{s}_i(t) \leq -\eta_i \cdot |s_i| \quad (4.15)$$

對所有的 $i = 1, \dots, p$ 而言。式 (4.15) 表示能保證系統狀態會在有限時間內到達所選擇的順滑曲面 (sliding surface)。

上述為考慮控制輸入如果控制輸入 u_{p+1}, \dots, u_m 發生錯誤的情況，接下來我們考慮 u_{p+1}, \dots, u_m 操作在正常情況，在相同的 e_i, s_i 與 u_i 設定下，對 $i = 1, \dots, p$ 而言，我們得到

$$\dot{s}_i(t) = \sum_{j=1}^p g_{ij}(\mathbf{x})u_j^{re} + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j + d_i. \quad (4.16)$$

與

$$s_i(t)\dot{s}_i(t) = s_i(t) \left\{ \sum_{j=1}^p g_{ij}(\mathbf{x})u_j^{re} + \sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j + d_i \right\}. \quad (4.17)$$

我們可以選取

$$(u_{p+1}^{eq}, \dots, u_m^{eq})^T \in N(\mathbf{g}_{p+1}(\mathbf{x}) \cdots \mathbf{g}_m(\mathbf{x})) \quad (4.18)$$

與

$$u_i^{re} = -\eta_i \cdot \text{sgn}(\mathbf{s}^T \mathbf{g}_i(\mathbf{x})) \text{ for } j = p+1, \dots, m \quad (4.19)$$

因此，式(4.17)及(4.19)將會保持順滑曲面(sliding surface)為一個不變的曲面和增加系統狀態收斂至順滑表面(sliding surface)的速度。此處定義 $N(\cdot)$ 為矩陣的零空間(null space)，對 $i = p+1, \dots, m$ 而言。另外， η_i 為所選擇的正常數，對所有的 $i = p+1, \dots, m$ 而言。而 $u_i^{eq} = 0$ 為滿足(4.18)的一個明顯解。

4.3 主動式可靠度控制設計

上述所討論的設計為被動式控制，此種設計不需要錯誤偵測與診斷的訊息。雖然被動式控制也可以達到追蹤性能，被動式控制是

一種比較保守的方法，一般來說，它會過度評估錯誤的程度，然而，過度預估將可能會導致設計出控制力大於系統的物理最大轉矩限制，並且導致不良的性能。接下來我們將討論主動式控制設計方法。首先我們假設當錯誤發生時，主動式控制的值會被成功的偵測與診斷如下式

$$u_j^* = u_j^0 + \Delta u_j \quad (4.20)$$

對 $j = p + 1, \dots, m$ 而言，其中 u_j^0 與 Δu_j 分別定義為估測控制值與估算誤差。因此在式 (4.8) 中的 $\dot{s}_i(t)$ 與在式 (4.10) 中的 u_i^{eq} 被修正如下

$$\begin{aligned} \dot{s}_i(t) = & f_i(\mathbf{x}) + \sum_{j=1}^p g_{ij}(\mathbf{x})u_j + \sum_{j=p+1}^m g_{ij}(\mathbf{x})[u_j^0 + \Delta u_j] + d_i - y_{id}^{(k_i)} \\ & + a_{i(k_i-1)}e_i^{(k_i-1)}(t) + \dots + a_{i2}\ddot{e}_i(t) + a_{i1}\dot{e}_i(t), \end{aligned} \quad (4.21)$$

$$\begin{pmatrix} u_1^{eq} \\ \vdots \\ u_p^{eq} \end{pmatrix} = -[\mathbf{g}_1, \dots, \mathbf{g}_p]^{-1} \cdot \begin{pmatrix} f_1(\mathbf{x}) + \sum_{j=p+1}^m g_{1j}(\mathbf{x})u_j^0 - y_{1d}^{(k_1)} + a_{1(k_1-1)}e_1^{(k_1-1)}(t) + \dots + a_{11}\dot{e}_1(t) \\ \vdots \\ f_n(\mathbf{x}) + \sum_{j=p+1}^m g_{nj}(\mathbf{x})u_j^0 - y_{nd}^{(k_n)} + a_{n(k_n-1)}e_n^{(k_n-1)}(t) + \dots + a_{n1}\dot{e}_n(t) \end{pmatrix}. \quad (4.22)$$

值得注意的是，在(4.22)中的 u_i^{eq} 多了一項包含診斷的訊息

$\sum_{j=p+1}^m g_{ij}(\mathbf{x})u_j^0$ ，且得到以下的結果，

$$s_i(t)\dot{s}_i(t) = s_i(t) \left\{ \sum_{j=1}^p g_{ij}(\mathbf{x})u_j^{re} + \sum_{j=p+1}^m g_{ij}(\mathbf{x})\Delta u_j + d_i \right\}. \quad (4.23)$$

在這一個問題中，我們也使用 $\rho_i(\mathbf{x}, t)$ 去表示對系統模型不確定性與干擾的估測上界，如下所示

$$|d_i + \sum_{j=p+1}^m g_{ij}(\mathbf{x})\Delta u_j| \leq \rho_i(\mathbf{x}, t). \quad (4.24)$$

一般來說，如果 $|\Delta u_j|$ 夠小，估測上界 $\rho_i(\mathbf{x}, t)$ 將會比在(4.12)式中
 得到之值還來得更為小。的確，錯誤診斷愈精確，則估測上界
 則愈小。最後，將式(4.24)所得到之 $\rho_i(\mathbf{x}, t)$ 代入式(4.14)中可求得
 u_i^{re} 。

5 數值模擬結果與討論

本章節將針對 HAVE DASH II 飛彈系統之非線性動態模式進行飛彈尾翼錯誤偵測與診斷及可靠度控制器設計之模擬。本次模擬是利用 MATLAB 及 SIMNON 兩種套裝軟體來進行系統模擬。模擬包含下列幾種結果：

- 用傳統可變結構控制律設計方式(正規設計)模擬全部致動器皆正常以及任一致動器卡死的狀況
- 針對允許 δ_1 、 δ_2 、 δ_3 及 δ_4 任一致動器故障所設計之被動式可靠度控制律來模擬全部致動器皆正常及致動器 δ_1 、 δ_2 、 δ_3 及 δ_4 分別故障的情況
- 針對 δ_1 、 δ_2 、 δ_3 及 δ_4 任一致動器故障之情況設計主動式可靠度控制器並包含致動器發生故障時之偵測結果

最後，我們將上述的數種情況利用模擬比較圖來展現並分析系統在正規設計、主動式設計及被動式設計情況下的性能表現。

5.1 飛彈參數設定

在此模擬中，我們採用的飛彈參數設定如下：

(1) 飛彈之轉動慣量：

$$I_{xx} = 1.1913 \text{ slug-ft}^2, \quad I_{yy} = 100.51 \text{ slug-ft}^2, \quad I_{zz} = 10057 \text{ slug-ft}^2$$

(2) 重力加速度： $g = 32.174 \text{ ft/sec}^2$

(3) 飛彈質量： $c = 968 \text{ ft/sec}$

(4) $k_F = 0.1534 \text{ ft}^2$ ， $k_M = 0.0959 \text{ ft}^3$ (由飛彈幾何模型決定)

(5) 聲速： $c = 968 \text{ ft/sec}$

(6) 大氣密度： $\rho = 5.124 \times 10^{-4} \text{ slug/ft}^3$

(其中聲速及大氣密度皆為高度 40000 英尺時所量測之數據)

5.2 觀察器設計

為了方便觀察器的設計，首先，將第二章所介紹的飛彈系統模型化成(3.1)式的型式，

$$\dot{x} = \begin{pmatrix} \dot{\psi} \\ \dot{\Theta} \\ \dot{\Phi} \\ \dot{Vm} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{P} \\ \dot{Q} \\ \dot{R} \end{pmatrix} = \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \\ f_7(x) \\ f_8(x) \\ f_9(x) \end{pmatrix} + g(x) \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{pmatrix}$$

其中 $(f_1(x) \dots \dots f_9(x))^T$ 與 $g(x)$ 分別如(2.13)與(2.14)。

將實際的參數

$$Km = 0.0959 ft^3,$$

$$\rho = 5.124 \times 10^{-4} slugs / ft^3,$$

$$I_{xx} = 1.1913 slug ft^2,$$

$$I_{yy} = 100.51 slug ft^2,$$

$$\text{and } I_{zz} = 100.57 slug ft^2$$

帶入 $g(x)$ 得到

$$\begin{pmatrix} \dot{\psi} \\ \dot{\Theta} \\ \dot{\Phi} \\ \dot{Vm} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{P} \\ \dot{Q} \\ \dot{R} \end{pmatrix} = \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \\ f_7(x) \\ f_8(x) \\ f_9(x) \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 9.2804 & 9.2804 & 9.2804 & 9.2804 \\ -0.5846 & 0.5846 & 0.5846 & -0.5846 \\ -0.5055 & -0.5055 & 0.5055 & 0.5055 \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{pmatrix}$$

我們選擇一變數變換矩陣 \bar{M}

$$\bar{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0539 & -0.8552 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8552 & -0.9891 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0539 & 0 & 0.9891 \end{pmatrix}$$

得到新的狀態變數 Z

$$\dot{Z} = f_{new}(z) + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \delta_1 + \delta_4 \\ \delta_2 - \delta_4 \\ \delta_3 + \delta_4 \end{pmatrix} = \begin{pmatrix} f_{1new}(z) \\ f_{2new}(z) \\ f_{3new}(z) \\ f_{4new}(z) \\ f_{5new}(z) \\ f_{6new}(z) \\ f_{7new}(z) + \delta_1 + \delta_4 \\ f_{8new}(z) + \delta_2 - \delta_4 \\ f_{9new}(z) + \delta_3 + \delta_4 \end{pmatrix}$$

其中 $f_{new}(z)$ 為

$$f_{new}(z) = \bar{M} f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \\ 0.0539f_7(x) - 0.8552f_8(x) \\ 0.8552f_8(x) - 0.9891f_9(x) \\ 0.0539f_7(x) - 0.8552f_9(x) \end{pmatrix}$$

因此，經過座標轉換後可以得到新的系統方程式

$$\begin{aligned}
\dot{Z}_1 &= f_1(x) \\
\dot{Z}_2 &= f_2(x) \\
\dot{Z}_3 &= f_3(x) \\
\dot{Z}_4 &= f_4(x) \\
\dot{Z}_5 &= f_5(x) \\
\dot{Z}_6 &= f_6(x) \\
\dot{Z}_7 &= 0.0539f_7(x) - 0.8552f_8(x) + \delta_1 + \delta_4 \\
\dot{Z}_8 &= 0.8552f_8(x) - 0.9891f_9(x) + \delta_2 - \delta_4 \\
\dot{Z}_9 &= 0.0539f_7(x) + 0.9891f_9(x) + \delta_3 + \delta_4
\end{aligned}$$

由上式，可以設計觀察器如下：

$$\begin{aligned}
\dot{\xi}_1 &= 0.0539f_7(x) - 0.8552f_8(x) + \delta_1 + \delta_4 + k_1(Z_7 - \xi_1) \\
\dot{\xi}_2 &= 0.8552f_8(x) - 0.9891f_9(x) + \delta_2 - \delta_4 + k_2(Z_8 - \xi_2) \\
\dot{\xi}_3 &= 0.0539f_7(x) + 0.9891f_9(x) + \delta_3 + \delta_4 + k_3(Z_9 - \xi_3)
\end{aligned}$$

在 K 值得選取方面，正常型式之控制器選擇 $k_1 = k_2 = k_3 = 10$ ，而主動型式及被動型式控制律之下為了使 alarm signal 能即時的反應，我們分別選擇了 $k_1 = 3$, $k_2 = 3$, $k_3 = 1.5$ 及 $k_1 = 10$, $k_2 = 5$, $k_3 = 2$ ，並 residual signals 為

$$\begin{aligned}
r_1 &= z_7 - \xi_1 \\
r_2 &= z_8 - \xi_2 \\
r_3 &= z_9 - \xi_3
\end{aligned}$$

就完成了觀察器的設計。

5.3 控制器設計

根據可靠度控制器設計法則，我們針對 HAVE DASH II 飛彈系統進行控制器設計，對應至飛彈系統數學模式可求得 $n=4$ (四個輸入)和

$m=9$ (九個狀態)。此外，我們選取 P_s 、 α (攻角)及 β (側滑角)做為系統的輸出，因此可將系統輸出寫成以下的形式：

$$y = \begin{bmatrix} P_s & \alpha & \beta \end{bmatrix}^T,$$

此處 $P_s = P \cos \alpha + R \sin \alpha$ 定義為相對於飛彈穩定軸的滾動速率。經過直接計算可以得到飛彈系統的相對維度為 $(1, 2, 2)$ ，利用此特性可將飛彈系統描述為下列輸入-輸出之動態數學模式：

$$(\dot{P}_s, \ddot{\alpha}, \ddot{\beta})^T = f(x) + G(x)\delta + d,$$

其中矩陣 $G(x)$ 可表示如下：

$$G(x) = k_M V_m^2 \begin{bmatrix} \frac{\cos \alpha}{I_{xx}} & 0 & \frac{\sin \alpha}{I_{yy}} \\ -\frac{\cos \alpha \tan \beta}{I_{xx}} & \frac{1}{I_{yy}} & -\frac{\sin \alpha \tan \beta}{I_{zz}} \\ \frac{\sin \alpha}{I_{xx}} & 0 & -\frac{\cos \alpha}{I_{zz}} \end{bmatrix} C_{Mu} W,$$

$$W = \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix}.$$

在確定上述飛彈系統之輸出-輸入數學動態模式後，接下來要進行控制器的推導，首先選取順滑表面為下列的形式：

$$s = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} = \begin{pmatrix} e_1 \\ \dot{e}_2 + a_{21}e_2 \\ \dot{e}_3 + a_{31}e_3 \end{pmatrix},$$

定義 $e_1 = P_s - P_{sc}$ ， $e_2 = \alpha - \alpha_c$ 和 $e_3 = \beta - 0$ ，其中 P_{sc} 及 α_c 為所欲追蹤之滾動角及攻角軌跡。

根據上述順滑曲面的選取及可靠度控制器設計法則，我們以 δ_4 卡死的情況為例來設計被動式可靠度控制器。同時，根據錯誤偵測與診斷結果設計符合 δ_1 、 δ_2 、 δ_3 及 δ_4 任一致動器故障時之主動式可靠度控制器，設計結果如下：

(1) 被動式可靠度控制器(針對 δ_4 故障設計控制器，當 δ_4 正常時

$$\delta_4^{re} = -\eta_4 \cdot \text{sgn}(s^T g_4) \text{ 即發生作用)}$$

$$\begin{pmatrix} \delta_1^{eq} \\ \delta_2^{eq} \\ \delta_3^{eq} \end{pmatrix} = (g_1 \quad g_2 \quad g_3)^{-1} \begin{pmatrix} -f_1(x) + \dot{P}_{sc} - e_1 \\ -f_2(x) + \ddot{\alpha}_c - \dot{e}_2 - a_{21}e_2(t) \\ -f_3(x) + \ddot{\beta}_c - \dot{e}_3 - a_{31}e_3(t) \end{pmatrix}$$

$$\delta_4^{eq} = 0$$

$$\begin{pmatrix} \delta_1^{re} \\ \delta_2^{re} \\ \delta_3^{re} \end{pmatrix} = (g_1 \quad g_2 \quad g_3)^{-1} \begin{pmatrix} -(\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ -(\rho_2 + \eta_2) \cdot \text{sgn}(s_2) \\ -(\rho_3 + \eta_3) \cdot \text{sgn}(s_3) \end{pmatrix}$$

$$\delta_4^{re} = -\eta_4 \cdot \text{sgn}(s^T g_4)$$

(2) 主動式可靠度控制器

a. 針對 δ_1 故障之狀況設計

$$\begin{pmatrix} \delta_2^{eq} \\ \delta_3^{eq} \\ \delta_4^{eq} \end{pmatrix} = (g_2 \quad g_3 \quad g_4)^{-1} \begin{pmatrix} -f_1(x) + \dot{P}_{sc} - e_1 - g_{11}\delta_1^0 \\ -f_2(x) + \ddot{\alpha}_c - \dot{e}_2 - a_{21}e_2(t) - g_{21}\delta_1^0 \\ -f_3(x) + \ddot{\beta}_c - \dot{e}_3 - a_{31}e_3(t) - g_{31}\delta_1^0 \end{pmatrix}$$

$$\begin{pmatrix} \delta_2^{re} \\ \delta_3^{re} \\ \delta_4^{re} \end{pmatrix} = (g_2 \quad g_3 \quad g_4)^{-1} \begin{pmatrix} -(\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ -(\rho_2 + \eta_2) \cdot \text{sgn}(s_2) \\ -(\rho_3 + \eta_3) \cdot \text{sgn}(s_3) \end{pmatrix}$$

其中 δ_1^0 為 δ_1 卡死的角度。

b. 針對 δ_2 故障之狀況設計

$$\begin{pmatrix} \delta_1^{eq} \\ \delta_3^{eq} \\ \delta_4^{eq} \end{pmatrix} = (g_1 \quad g_3 \quad g_4)^{-1} \begin{pmatrix} -f_1(x) + \dot{P}_{sc} - e_1 - g_{12}\delta_2^0 \\ -f_2(x) + \ddot{\alpha}_c - \dot{e}_2 - a_{21}e_2(t) - g_{22}\delta_2^0 \\ -f_3(x) + \ddot{\beta}_c - \dot{e}_3 - a_{31}e_3(t) - g_{32}\delta_2^0 \end{pmatrix}$$

$$\begin{pmatrix} \delta_1^{re} \\ \delta_3^{re} \\ \delta_4^{re} \end{pmatrix} = (g_1 \quad g_3 \quad g_4)^{-1} \begin{pmatrix} -(\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ -(\rho_2 + \eta_2) \cdot \text{sgn}(s_2) \\ -(\rho_3 + \eta_3) \cdot \text{sgn}(s_3) \end{pmatrix}$$

其中 δ_2^0 為 δ_2 卡死的角度。

c. 針對 δ_3 故障之狀況設計

$$\begin{pmatrix} \delta_1^{re} \\ \delta_2^{re} \\ \delta_4^{re} \end{pmatrix} = (g_1 \quad g_2 \quad g_4)^{-1} \begin{pmatrix} -(\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ -(\rho_2 + \eta_2) \cdot \text{sgn}(s_2) \\ -(\rho_3 + \eta_3) \cdot \text{sgn}(s_3) \end{pmatrix}$$

$$\begin{pmatrix} \delta_1^{eq} \\ \delta_2^{eq} \\ \delta_4^{eq} \end{pmatrix} = (g_1 \quad g_2 \quad g_4)^{-1} \begin{pmatrix} -f_1(x) + \dot{P}_{sc} - e_1 - g_{13}\delta_3^0 \\ -f_2(x) + \ddot{\alpha}_c - \dot{e}_2 - a_{21}e_2(t) - g_{23}\delta_3^0 \\ -f_3(x) + \ddot{\beta}_c - \dot{e}_3 - a_{31}e_3(t) - g_{33}\delta_3^0 \end{pmatrix}$$

其中 δ_3^0 為 δ_3 卡死的角度。

d. 針對 δ_4 故障之情況設計

$$\begin{pmatrix} \delta_1^{eq} \\ \delta_2^{eq} \\ \delta_3^{eq} \end{pmatrix} = (g_1 \quad g_2 \quad g_3)^{-1} \begin{pmatrix} -f_1(x) + \dot{P}_{sc} - e_1 - g_{14}\delta_4^0 \\ -f_2(x) + \ddot{\alpha}_c - \dot{e}_2 - a_{21}e_2(t) - g_{24}\delta_4^0 \\ -f_3(x) + \ddot{\beta}_c - \dot{e}_3 - a_{31}e_3(t) - g_{34}\delta_4^0 \end{pmatrix}$$

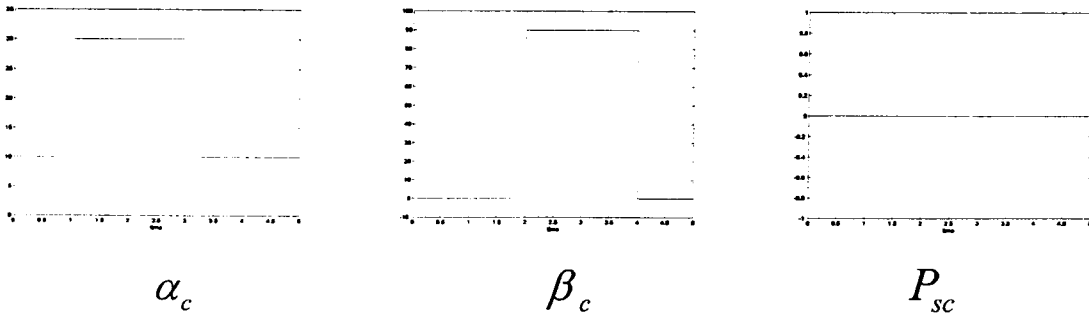
$$\begin{pmatrix} \delta_1^{re} \\ \delta_2^{re} \\ \delta_3^{re} \end{pmatrix} = (g_1 \quad g_2 \quad g_3)^{-1} \begin{pmatrix} -(\rho_1 + \eta_1) \cdot \text{sgn}(s_1) \\ -(\rho_2 + \eta_2) \cdot \text{sgn}(s_2) \\ -(\rho_3 + \eta_3) \cdot \text{sgn}(s_3) \end{pmatrix}$$

其中 δ_4^0 為 δ_4 卡死的角度。

5.4 控制器參數及初始值設定

在設計完可靠度控制器之後，以下將對控制器參數及系統狀態之初始值進行設定：

- (1) 選定順滑表面中的參數為 $a_{21} = 20$, $a_{31} = 8$
- (2) 為了避免利用 sign 函數所產生的切換現象(chattering) , 因此將 sign 函數轉成 saturation 函數的形式, 即 $\text{sgn}(s_i)$ 用 $\text{sat}(s_i / \varepsilon_i)$ 取代, 其中邊界層厚度選取分別為 $\varepsilon_1 = 1.0$, $\varepsilon_2 = 0.5$ 及 $\varepsilon_3 = 0.5$
- (3) 系統狀態初始值除飛彈速度 $V_m(0) = 2662 \text{ ft/sec}$ 和攻角為 $\alpha(0) = 10(\text{deg})$ 之外, 其餘皆設為零
- (4) 假設飛彈為定速飛行, 故飛彈速度保持在 $V_m = 2662 \text{ ft/sec}$ (2.75 馬赫)
- (5) 飛彈尾翼擺動角度之物理限制為 $-30^\circ \leq \delta_i \leq 30^\circ$ for $i=1, 2, 3, 4$
- (6) 設定飛彈輸出之命令軌跡為下所示:



$$\alpha_c = \begin{cases} 30(\text{deg}) & 1 \leq t \leq 3 \\ 10 & t < 1 \text{ or } t > 3 \end{cases}, P_{sc} = \begin{cases} 90(\text{deg/s}) & 2 \leq t \leq 4 \\ 0 & t < 2 \text{ or } t > 4 \end{cases}, \beta_c = 0$$

α_c 為攻角之命令軌跡

β_c 為側滑角之命令軌跡

P_{sc} 為滾動速率之命令軌跡

5.5 模擬結果與討論

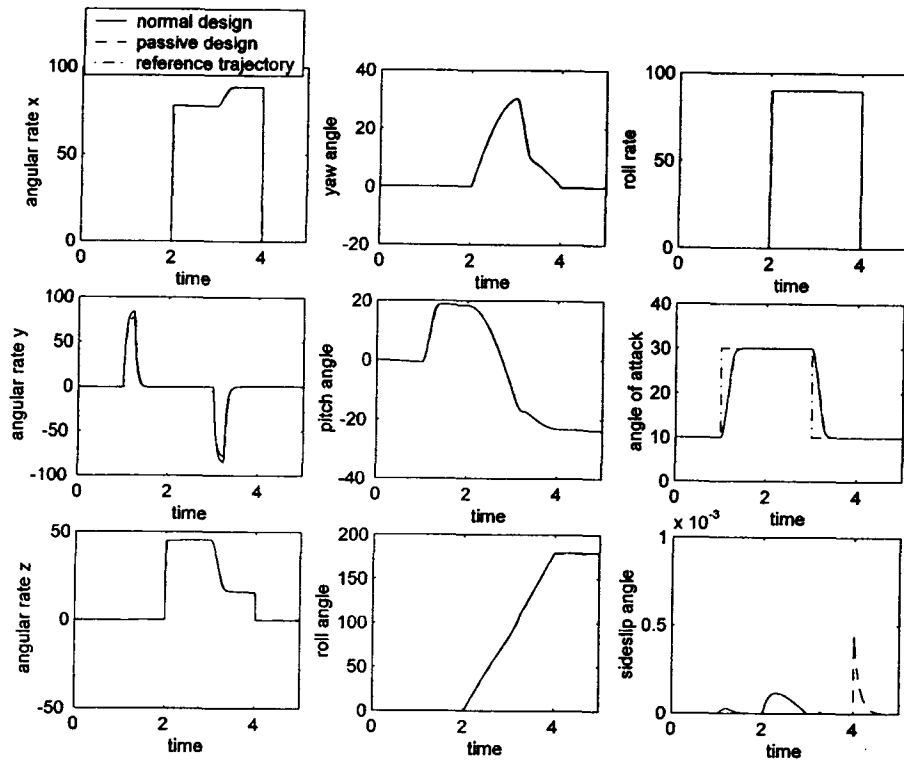
飛彈系統尾翼轉向控制模擬結果列於圖 5.1-5.20，模擬圖形主要包括正規設計的控制器模擬結果以及針對致動器完全正常或只有一致動器故障時，運用可靠度控制器和錯誤診斷機制所模擬的結果，由結果可以看出，無論利用被動式可靠度控制器或經由偵測器偵測錯誤訊號並即時切換至主動式控制器，系統都能快速追蹤到命令軌跡且達到穩定的結果。以下幾點是針對模擬圖所做的討論：

- a. 圖 5.1-5.4 分別為正規設計及針對 δ_1 、 δ_2 、 δ_3 和 δ_4 故障所設計之被動式控制器性能比較圖(模擬全部制動器皆正常運作的情況)，由模擬圖可以明顯地看出針對全部致動器皆正常模擬時，無論是利用正規設計或被動式設計，滾動角速率、攻角及側滑角皆能快速且在不耗費很大能量下追蹤到命令軌跡，然而，在暫態響應方面，正規設計又比被動式設計更快，且側滑角均能保持在 $\pm 2 \times 10^{-4}$ 度的範圍之內，唯針對 δ_2 及 δ_3 卡死設計之被動式控制器其側滑角擺動幅度約在 0.1-0.2 度左右，其餘皆在 $\pm 2 \times 10^{-3}$ 度範圍之內。因此，整體而言，

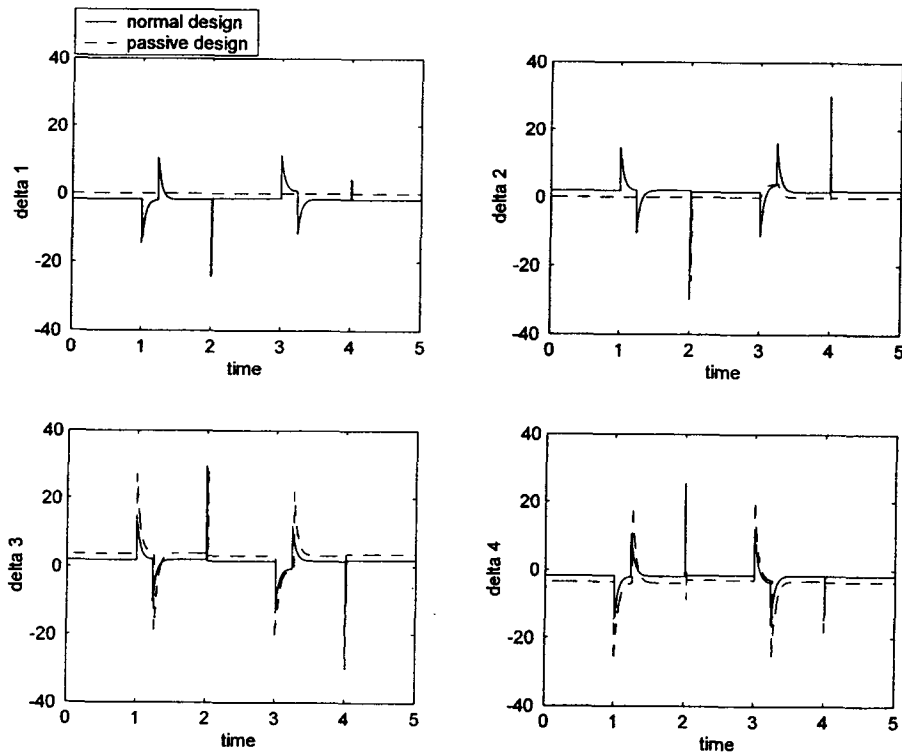
在圖 5.1-5.4 中，正規設計和被動式設計都展現了好的性能表現，但正規設計的性能較被動式設計好，這和理論分析結果相符。

- b. 在圖 5.5-5.8 針對正規控制器設計來模擬，致動器 $\delta_1 \sim \delta_4$ 在 $t=1.1$ 秒發生故障的現象。在沒有切換控制律的情況下，很明顯的可以觀察到系統所有輸出狀態均發散。而所設計的觀察器在 $t=1.1$ 秒時可成功的偵測到故障的發生及類型。
- c. 圖 5.9-5.12 分別為主動式控制針對致動器 $\delta_1 \sim \delta_4$ 在 $t=1.1$ 秒卡死時的模擬結果。由模擬圖中可明顯的觀察到觀察器能正確地偵測出錯誤訊號，並根據 residual signals 不同的產生組合，作出正確的控制律切換決策，使系統展現良好的性能表現。由圖中可看出無論在滾動角速率、攻角等方面皆能很快的追蹤到所設計的軌跡，而側滑角度也保持在相當小的範圍之內。
- d. 圖 5.13-5.16 為針對致動器 $\delta_1 \sim \delta_4$ 故障設計被動式控制器之模擬圖，當致動器 δ_4 在 $t=1.1$ 秒卡死時，被動式控制器仍能發揮作用，且讓系統達到良好的性能，可以明顯看出滾動角速率及攻角皆能很快地追蹤到所要的軌跡，且側滑角能保持在很小的範圍之內，這也證實了我們所設計的被動式控制器是確實可行的。
- e. 最後，在圖 5.17-5.20 中，我們比較被動式及主動式控制器之性能表現。在滾動角速率及攻角追蹤方面，主動式控制器與被動式控制

器性能接近，而在側滑角方面，主動式控制器擺動角度則遠小於被動式控制器。因此，結合上述結果可知主動式控制器的整體性能比被動式控制器略佳，這也和我們所提出的理論相符。

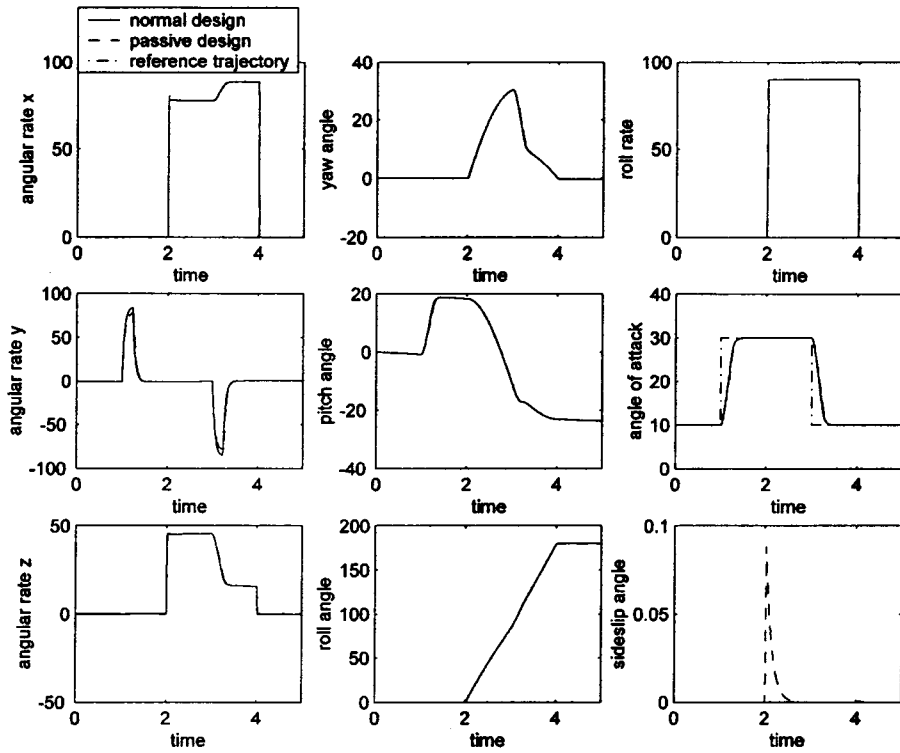


(a)

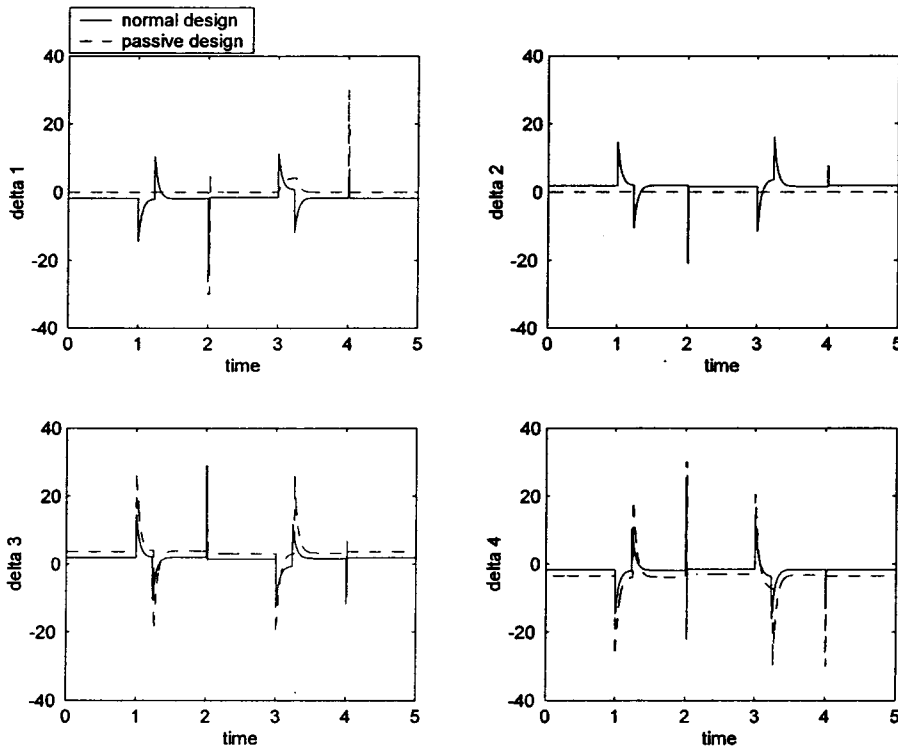


(b)

圖 5.1 正規設計與針對致動器 1 卡死所設計之被動式控制器(模擬全部致動器正常)之(a)狀態比較圖 (b)輸出角度比較圖

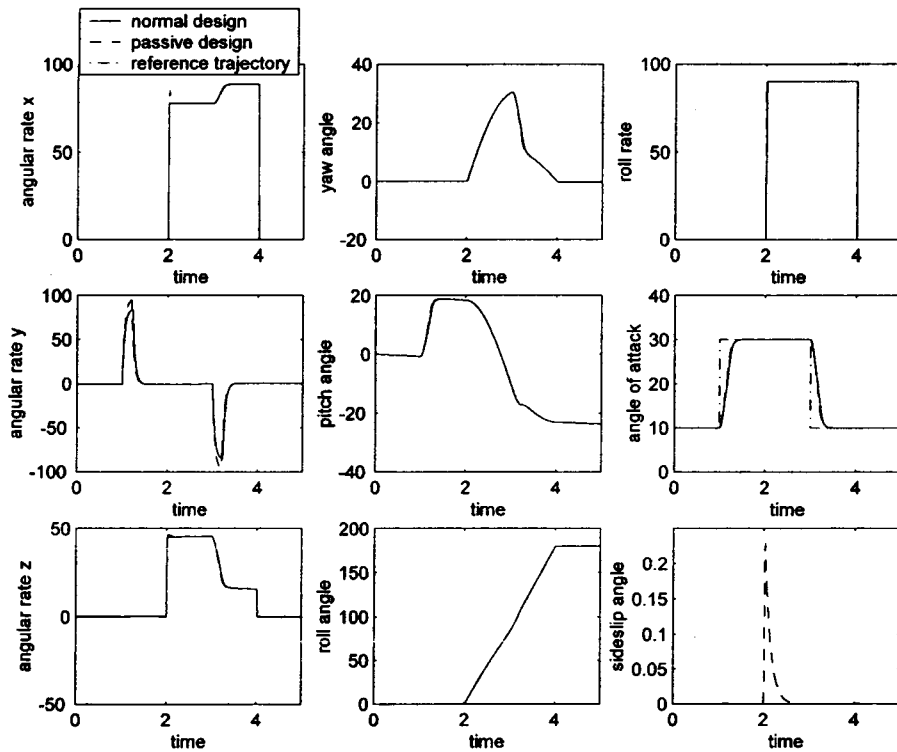


(a)

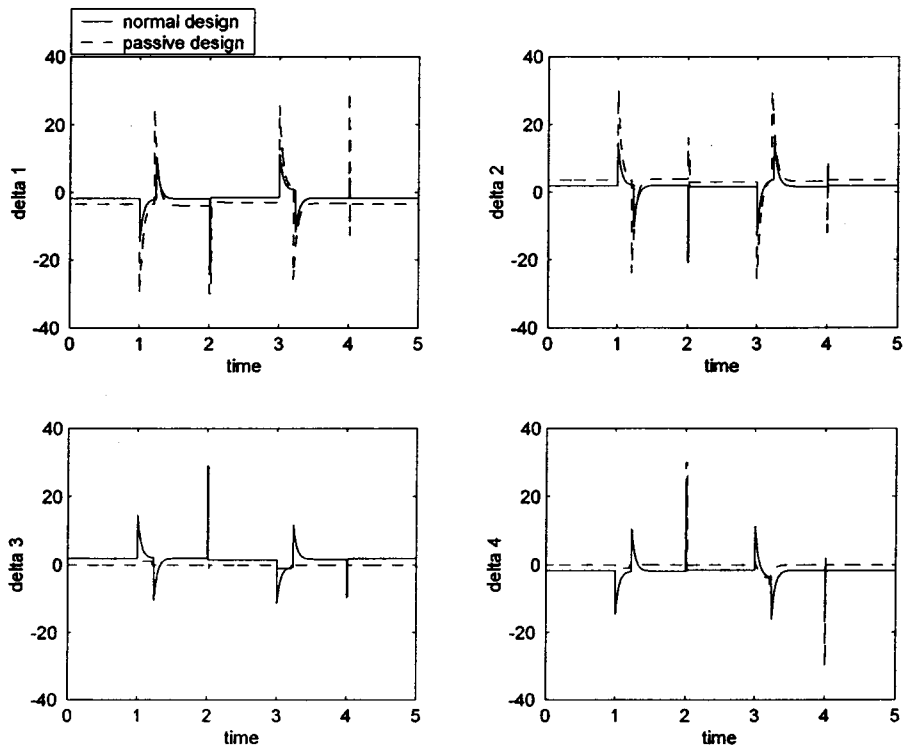


(b)

圖 5.2 正規設計與針對致動器 2 卡死所設計之被動式控制器(模擬全部致動器正常)之(a)狀態比較圖 (b)輸出角度比較圖

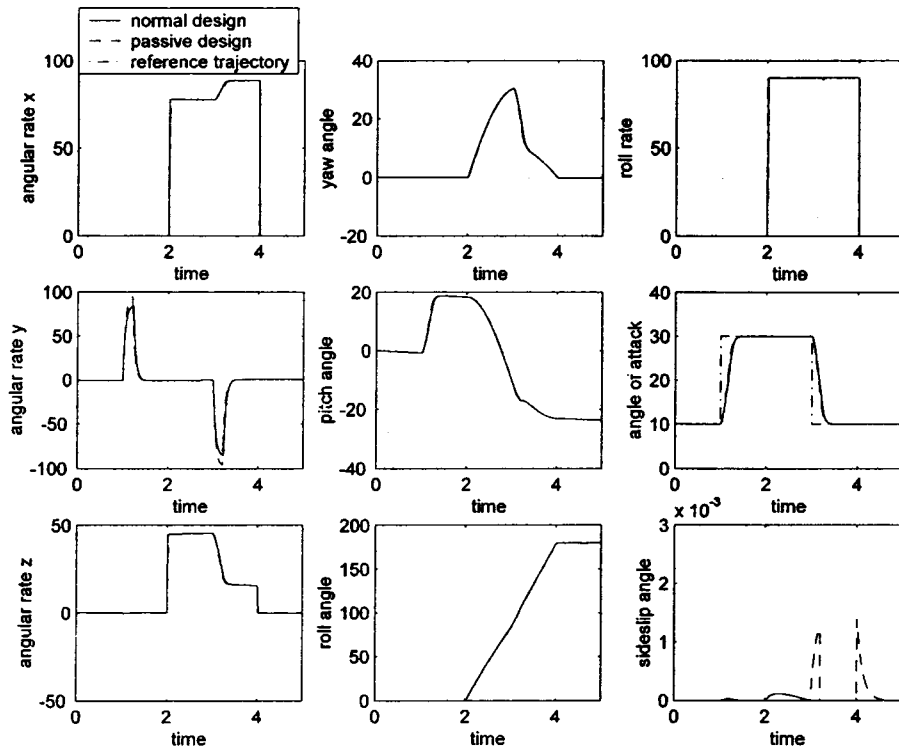


(a)

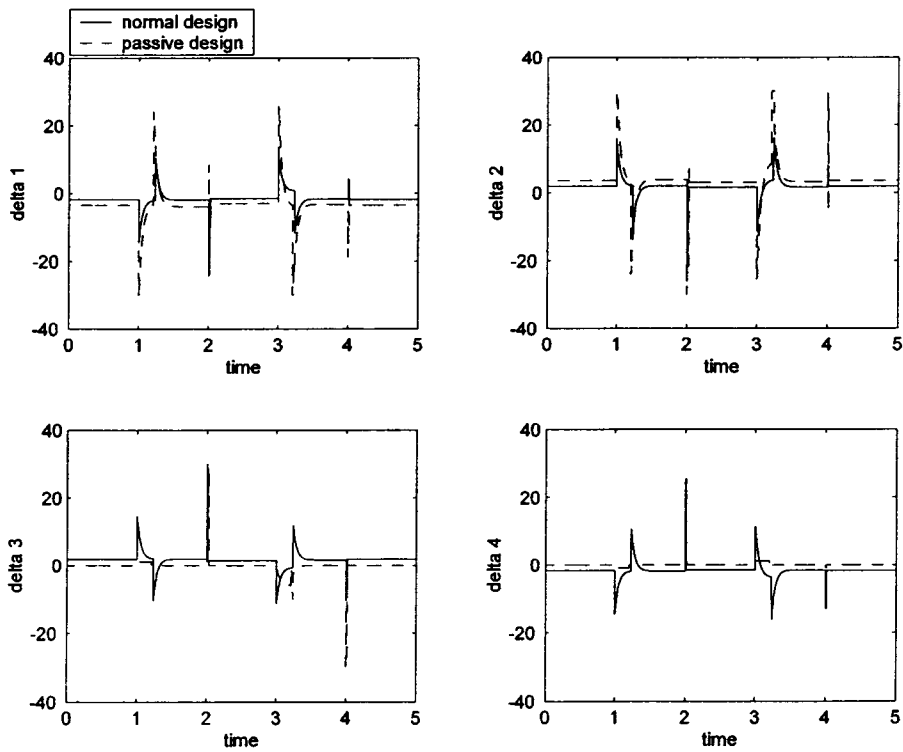


(b)

圖 5.3 正規設計與針對致動器 3 卡死所設計之被動式控制器(模擬全部致動器正常)之(a)狀態比較圖 (b)輸出角度比較圖



(a)



(b)

圖 5.4 正規設計與針對致動器 4 卡死所設計之被動式控制器(模擬全部致動器正常)之(a)狀態比較圖 (b)輸出角度比較圖

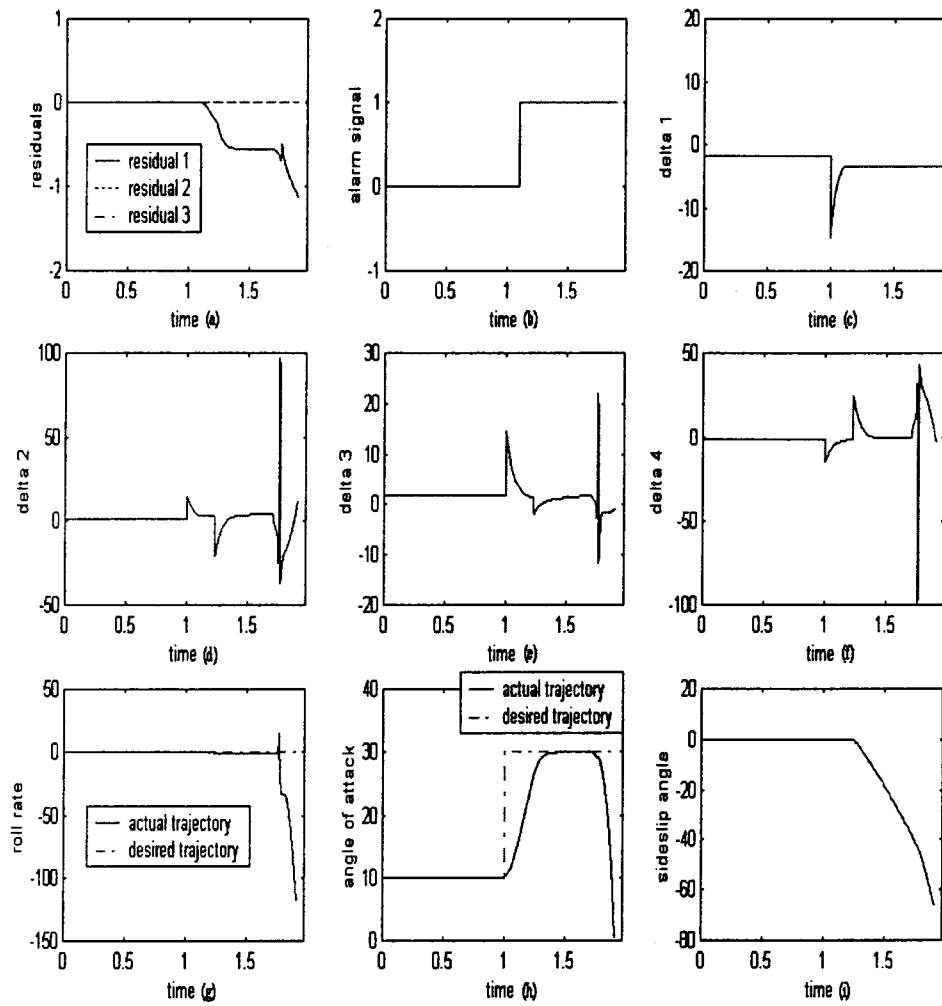


圖 5.5 針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第一個致動器 δ_1 發生故障：(a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

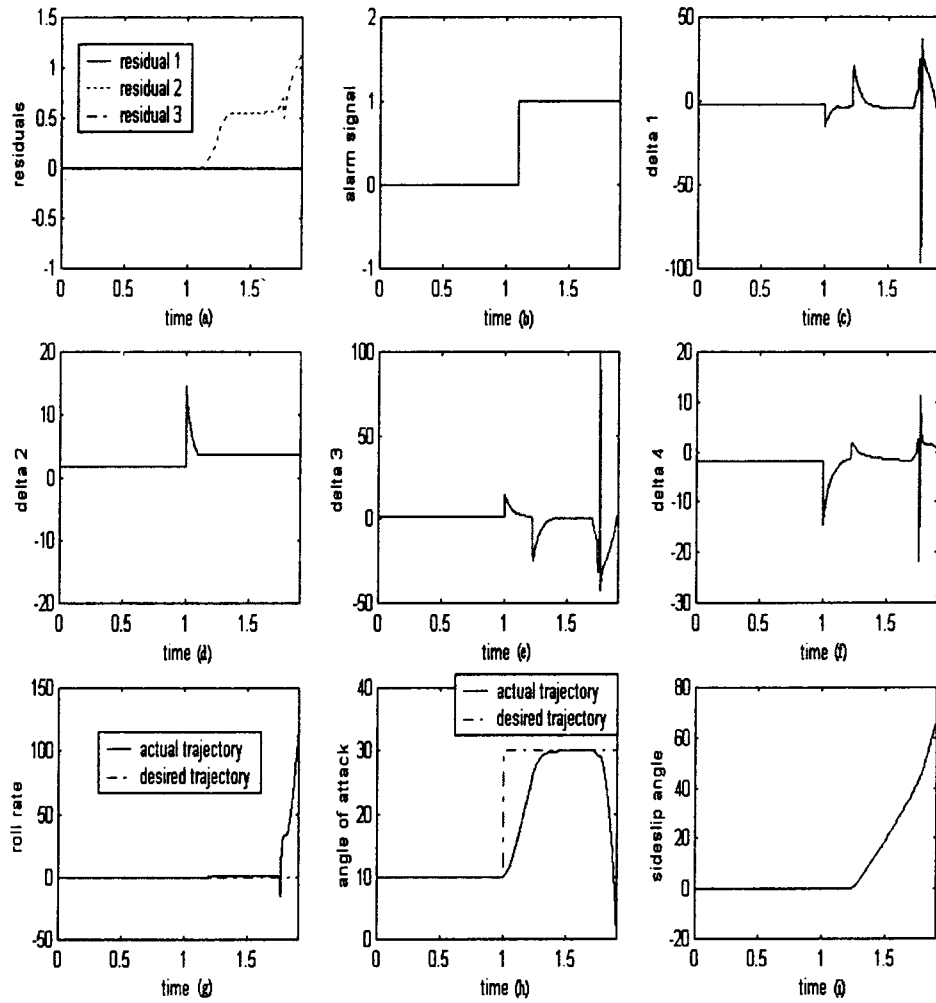


圖 5.6 針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第二個致動器 δ_2 發生故障: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

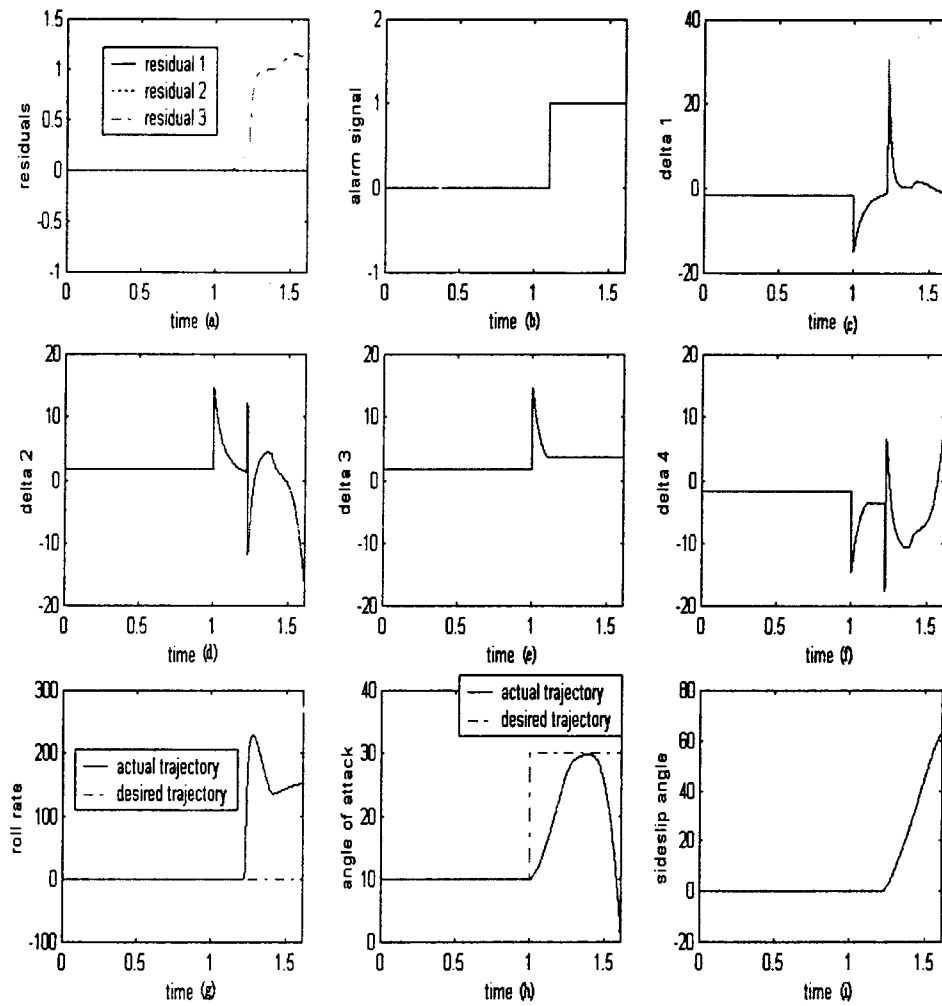


圖 5.7 針對正規控制器設計，模擬 $t=1.1\text{sec}$ 第三個致動器 δ_3 發生故障：(a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

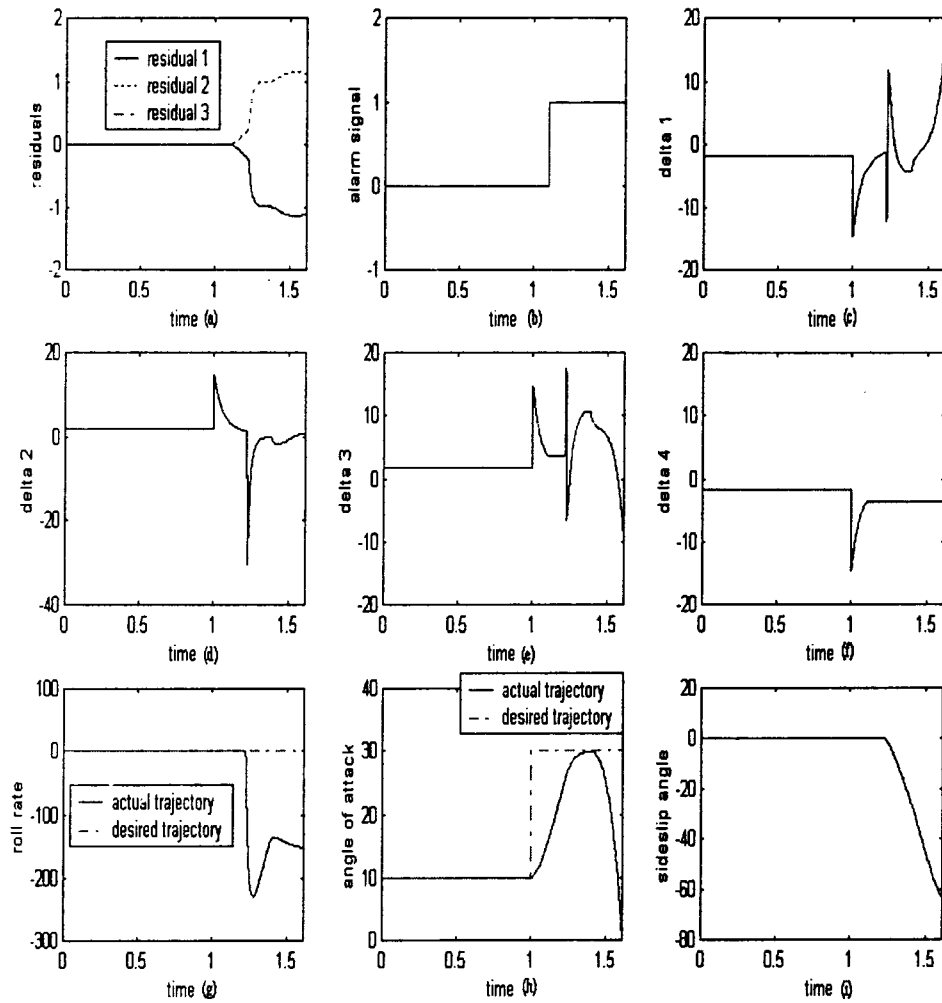


圖 5.8 針對正規控制器設計, 模擬 $t=1.1\text{sec}$ 第四個致動器 δ_4 發生故障: (a) residual signals (b) alarm signal (c) 第一個致動器 δ_1 (d) 第二個致動器 δ_2 (e) 第三個致動器 δ_3 (f) 第四個致動器 δ_4 (g) roll rate (f) angle of attack (g) sideslip angle

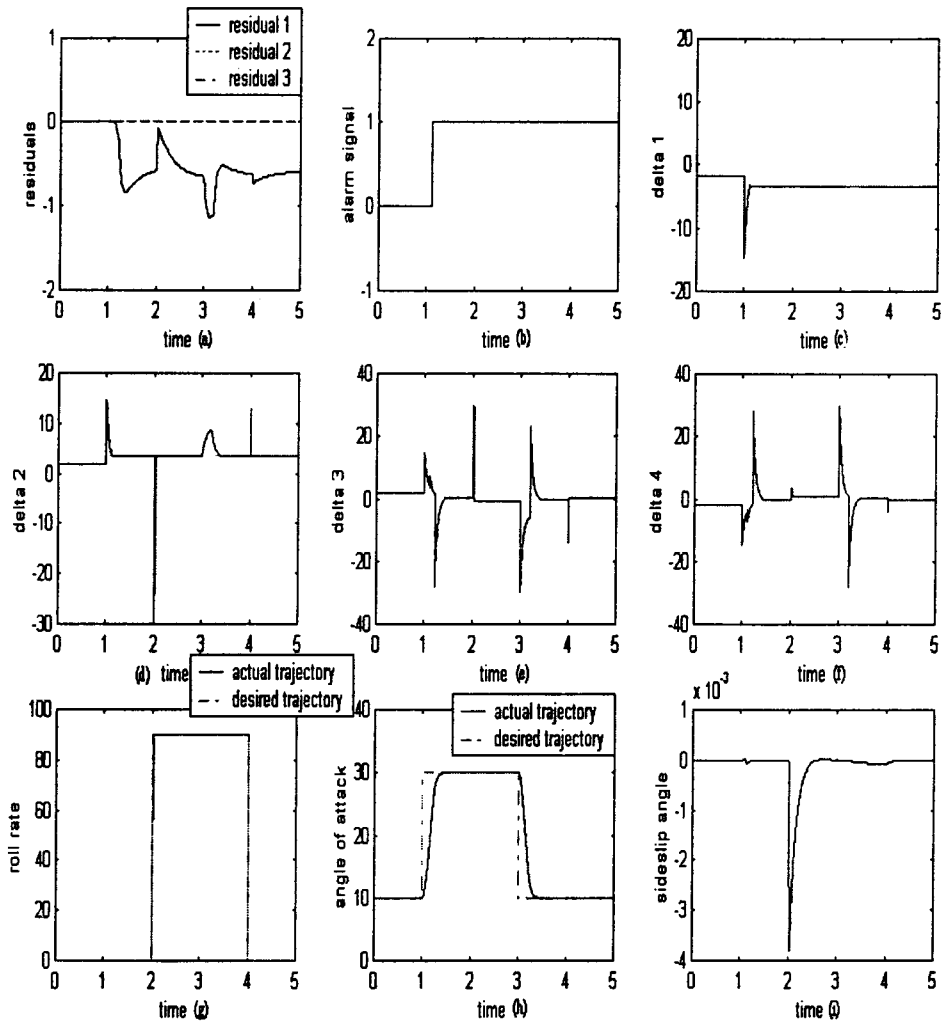


圖 5.9 主動式控制針對第一個致動器 δ_1 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

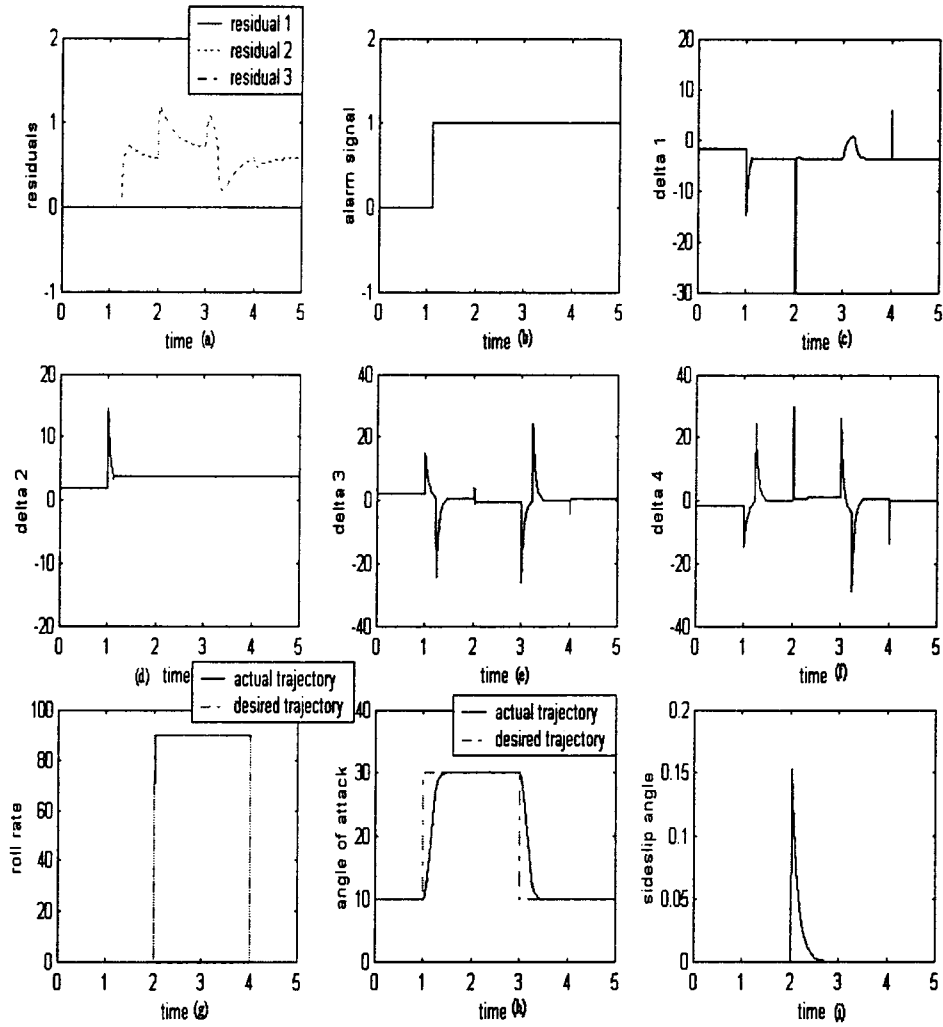


圖 5.10 主動式控制針對第二個致動器 δ_2 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (h)angle of attack (i) sideslip angle

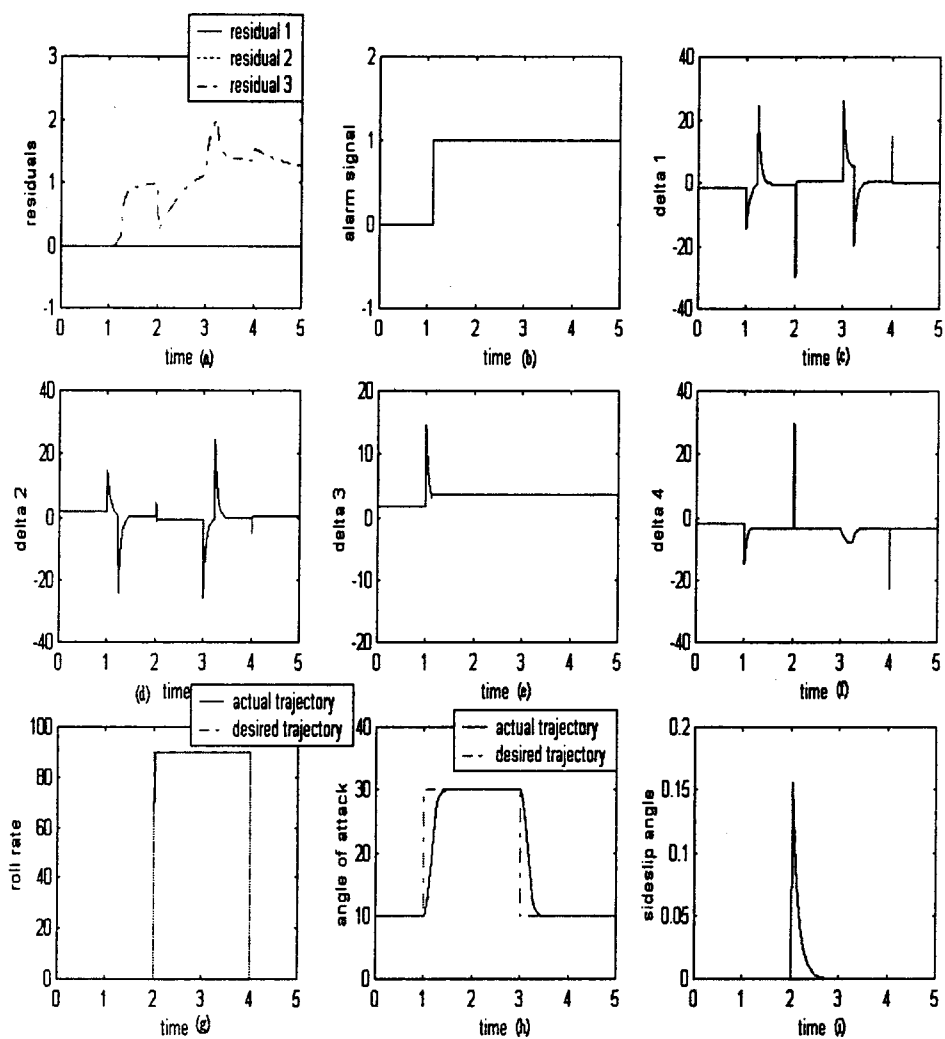


圖 5.11 主動式控制針對第三個致動器 δ_3 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (h)angle of attack (i) sideslip angle

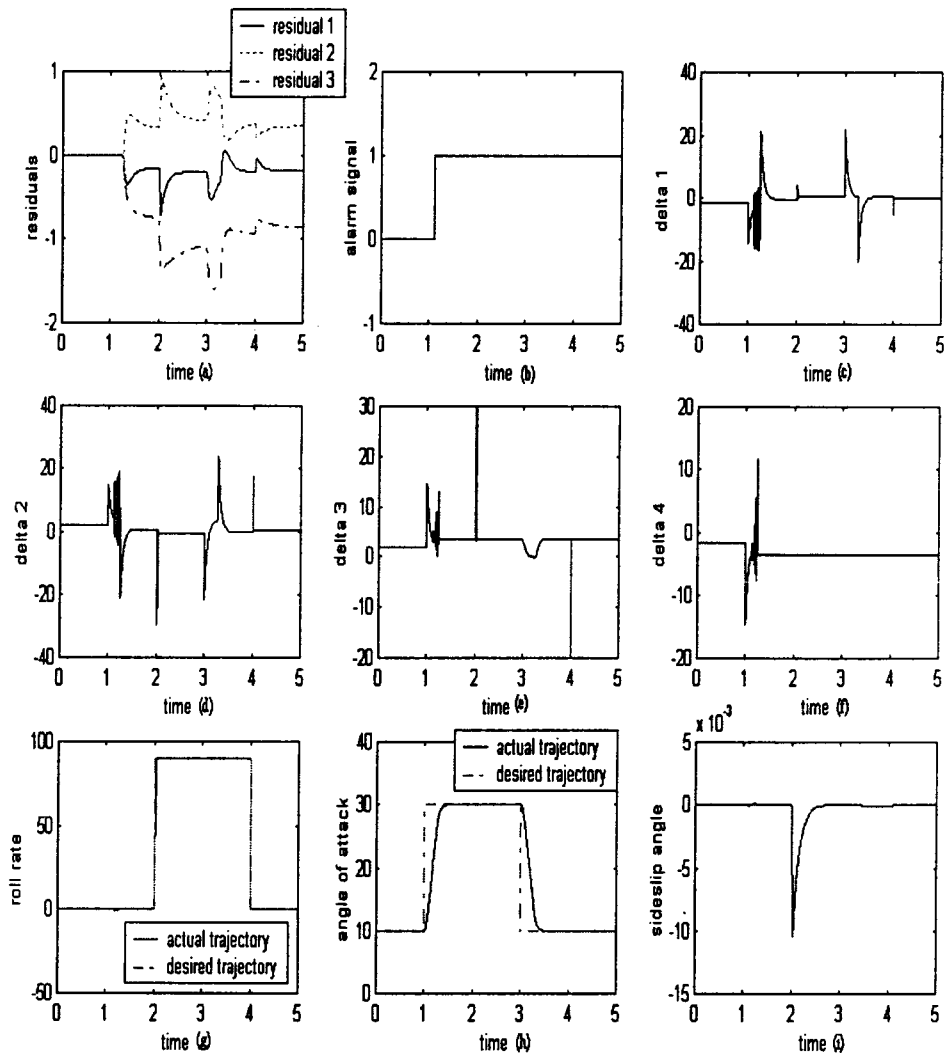


圖 5.12 主動式控制針對第四個致動器 δ_4 發生故障之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (h)angle of attack (i) sideslip angle

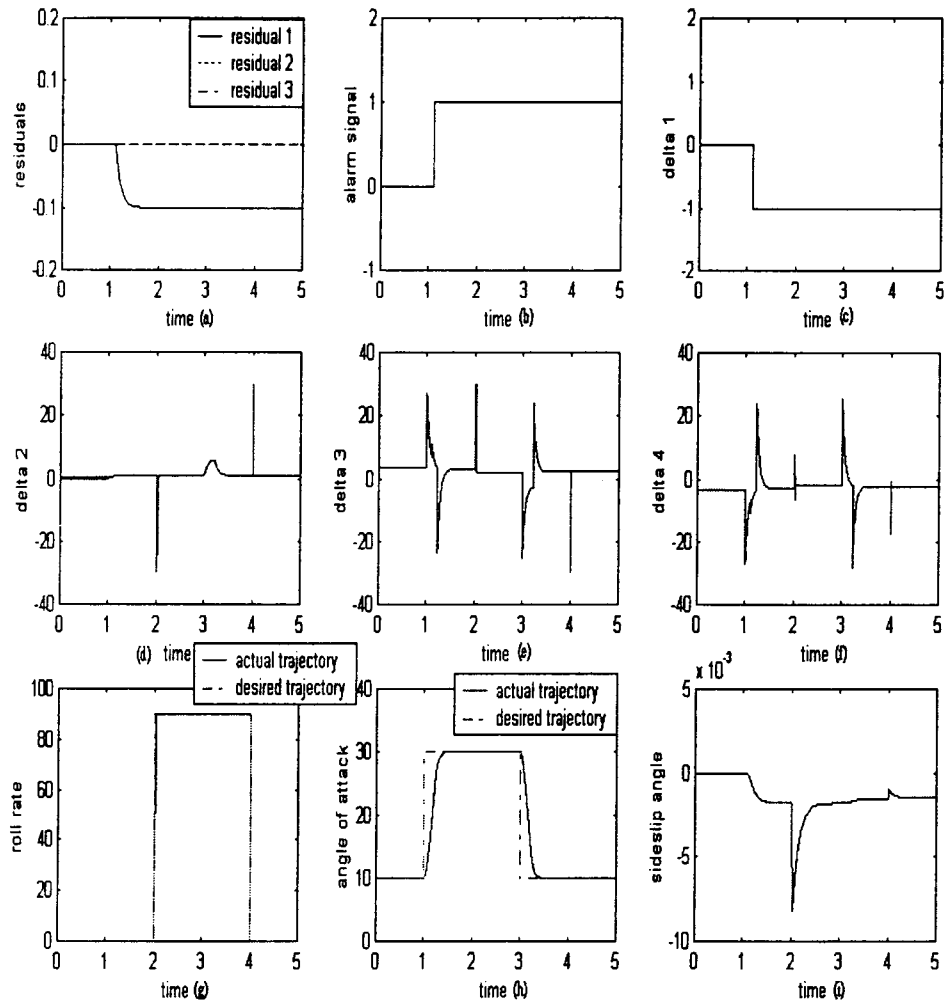


圖 5.13 針對第一個致動器 δ_1 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

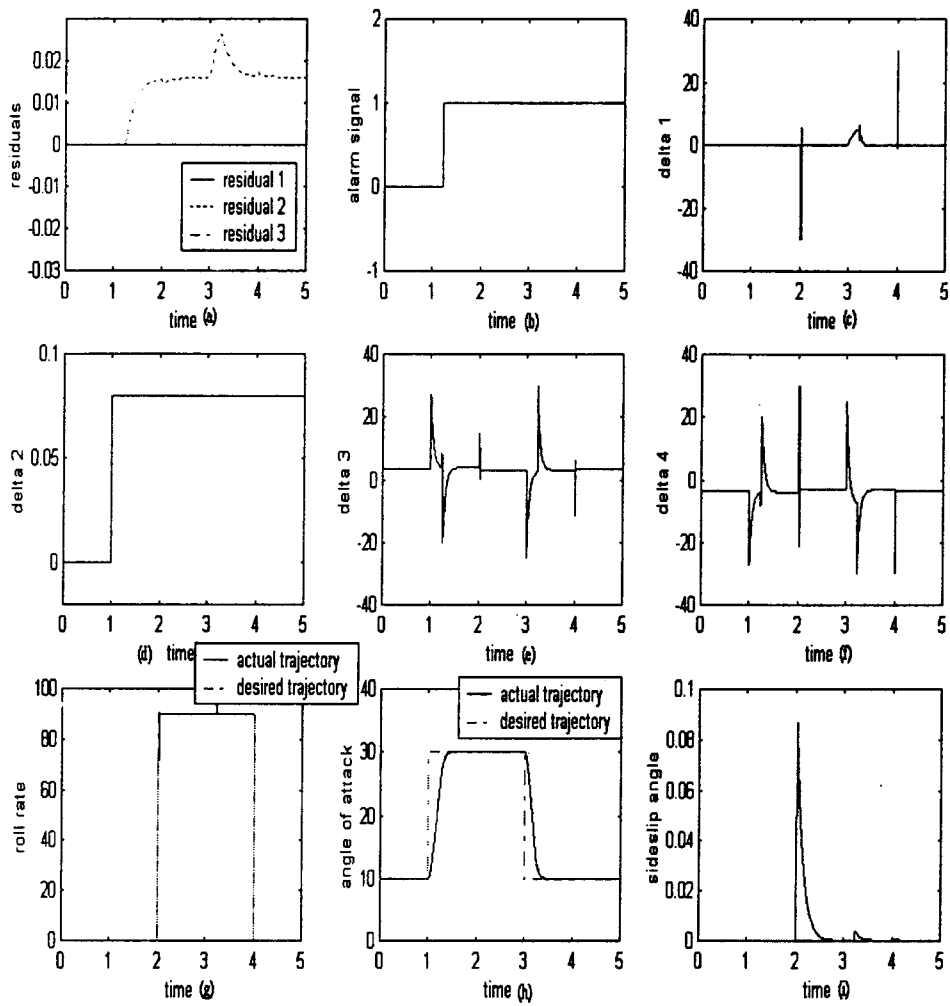


圖 5.14 針對第二個致動器 δ_2 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

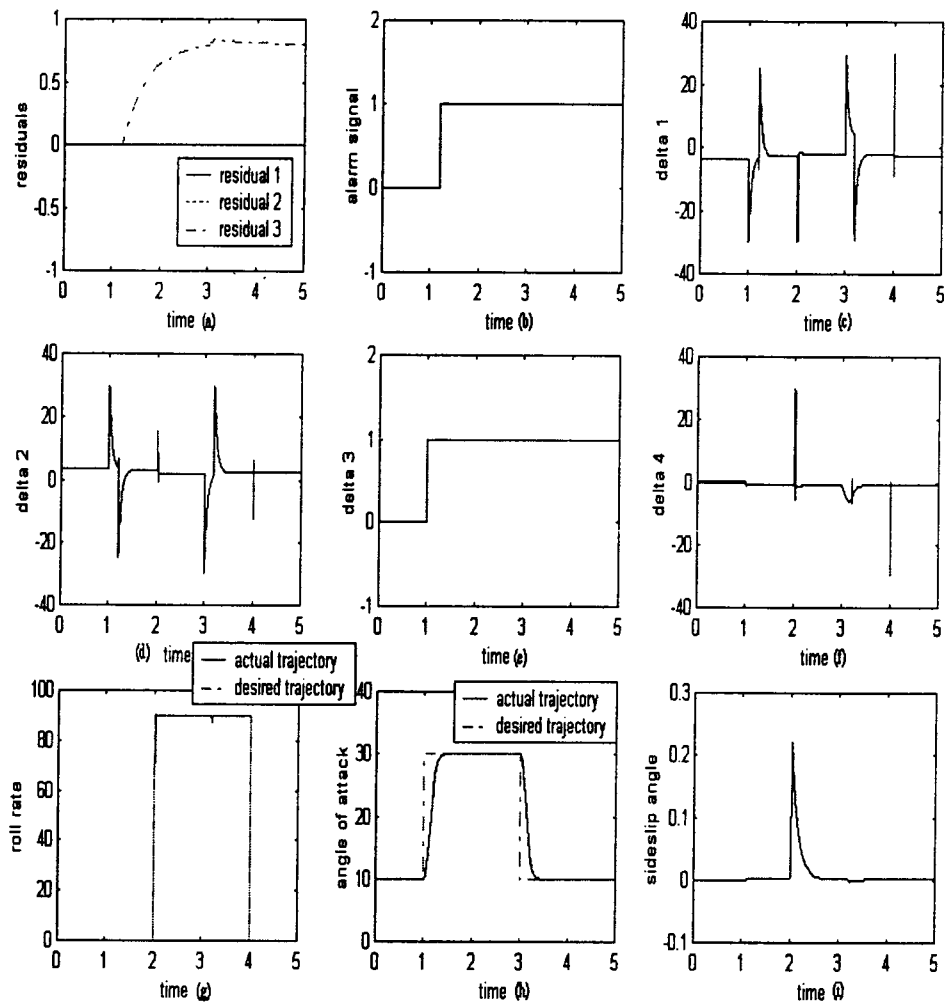


圖 5.15 針對第三個致動器 δ_3 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

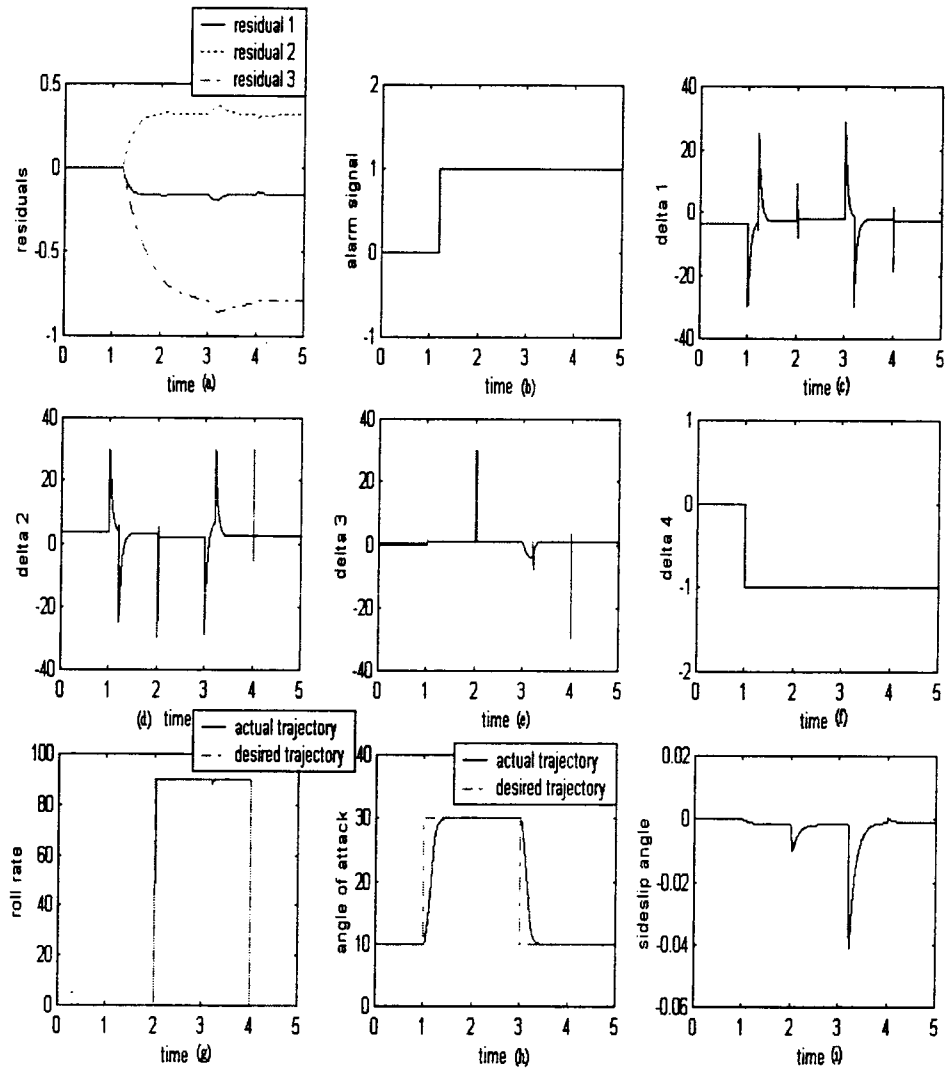


圖 5.16 針對第四個致動器 δ_4 發生故障的被動式可靠度控制之模擬結果: (a)residual signals (b)alarm signal (c)第一個致動器 δ_1 (d)第二個致動器 δ_2 (e)第三個致動器 δ_3 (f)第四個致動器 δ_4 (g)roll rate (f)angle of attack (g) sideslip angle

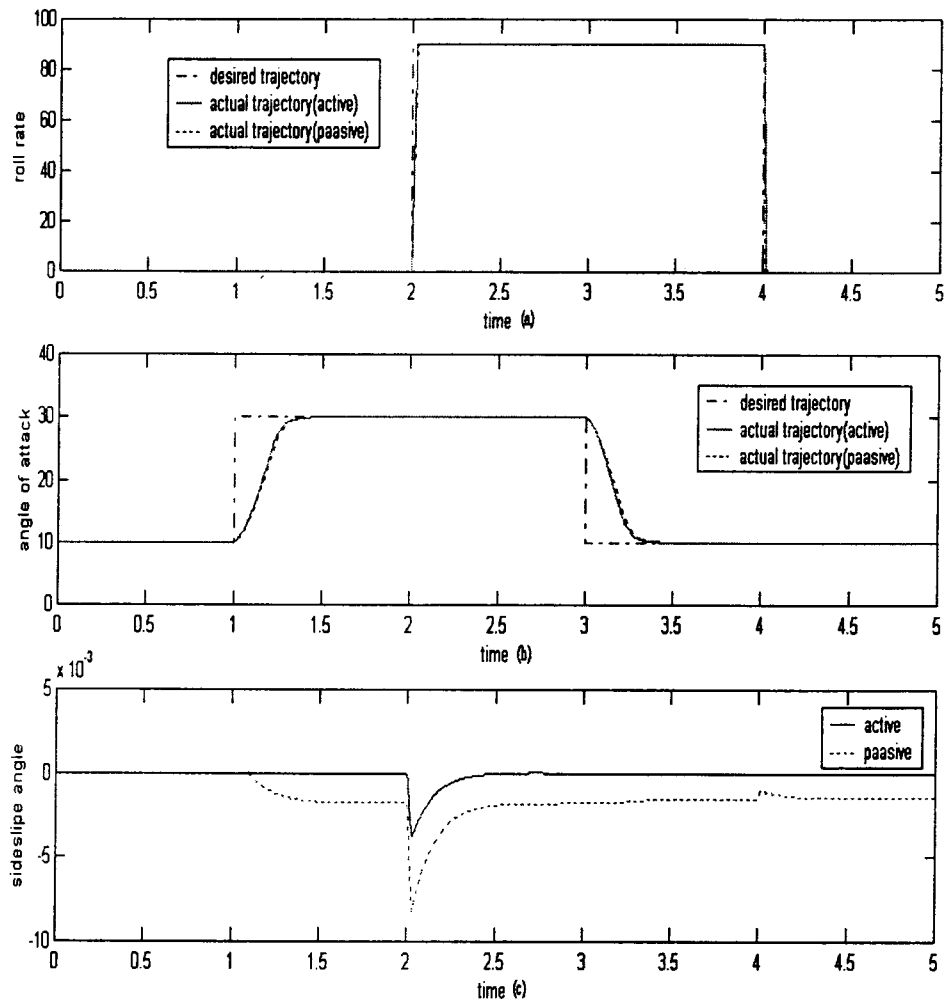


圖 5.17 主動式及被動式可靠度控制器在第一個致動器 δ_1 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle

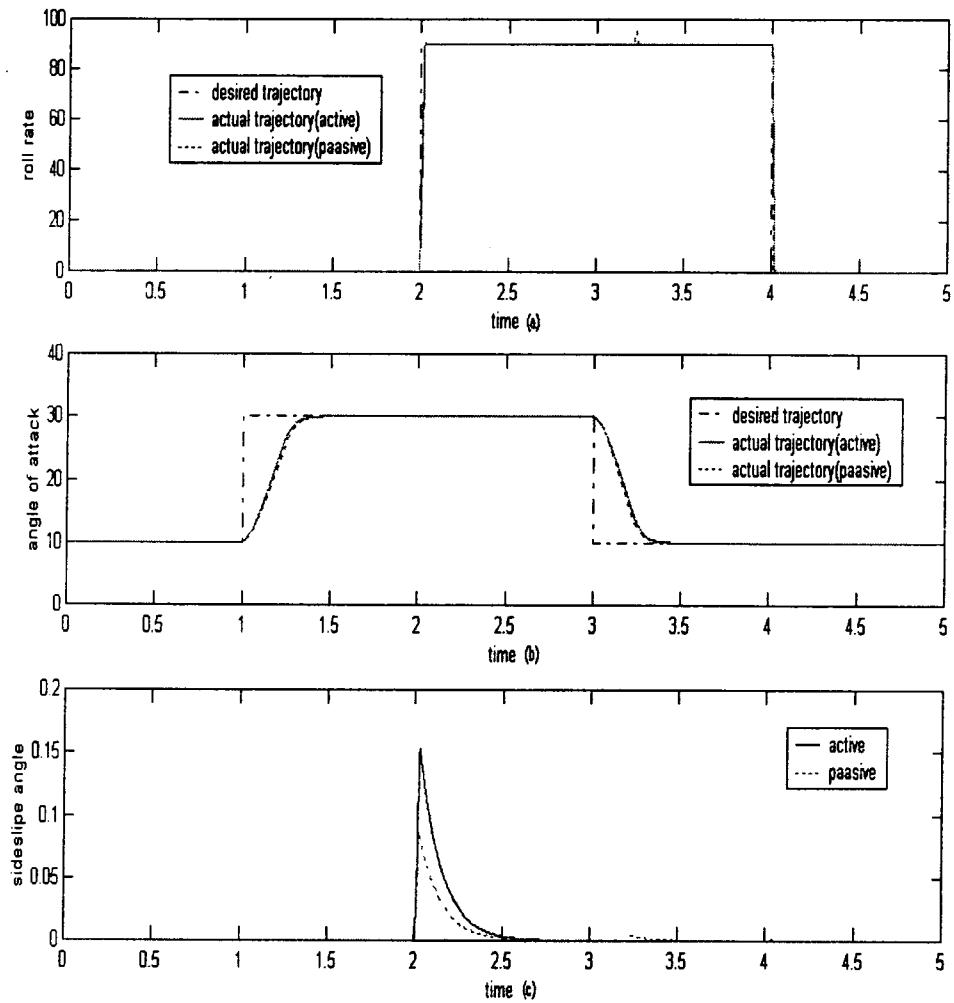


圖 5.18 主動式及被動式可靠度控制器在第二個致動器 δ_2 發生故障時的比較：(a)roll rate (b)angle of attack (c)sideslip angle

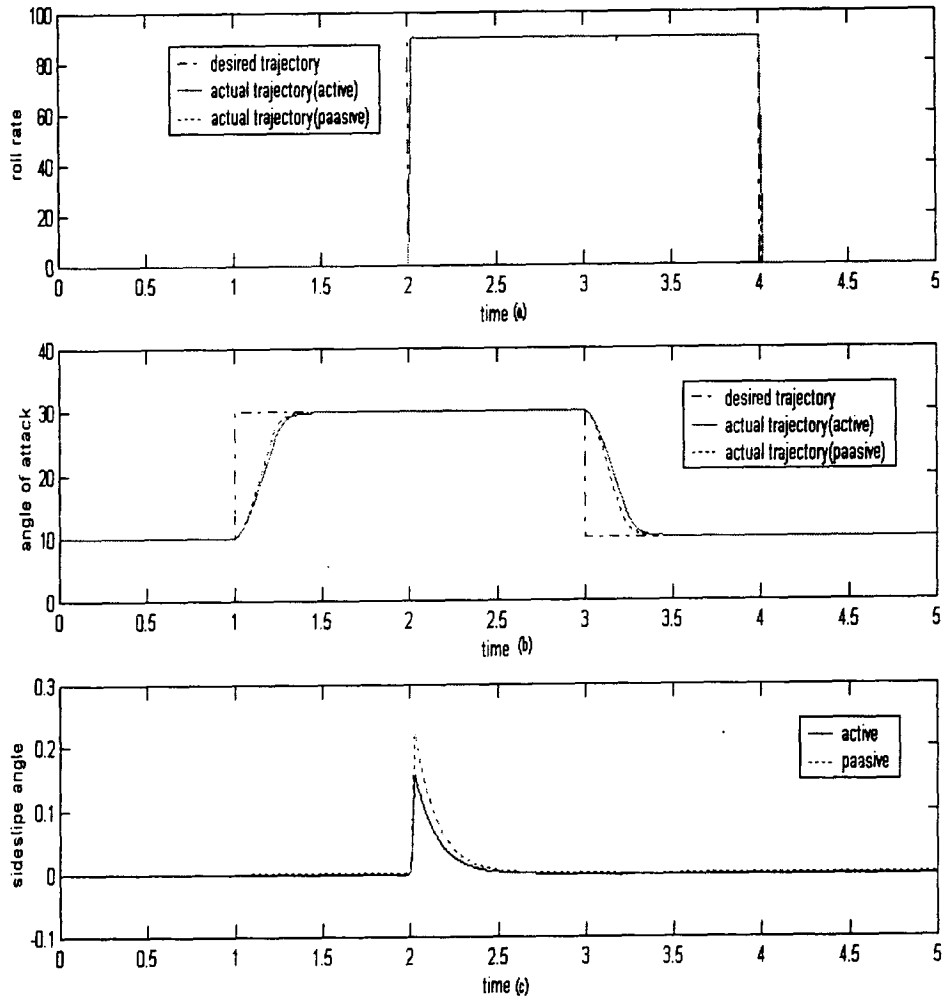


圖 5.19 主動式及被動式可靠度控制器在第三個致動器 δ_3 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle

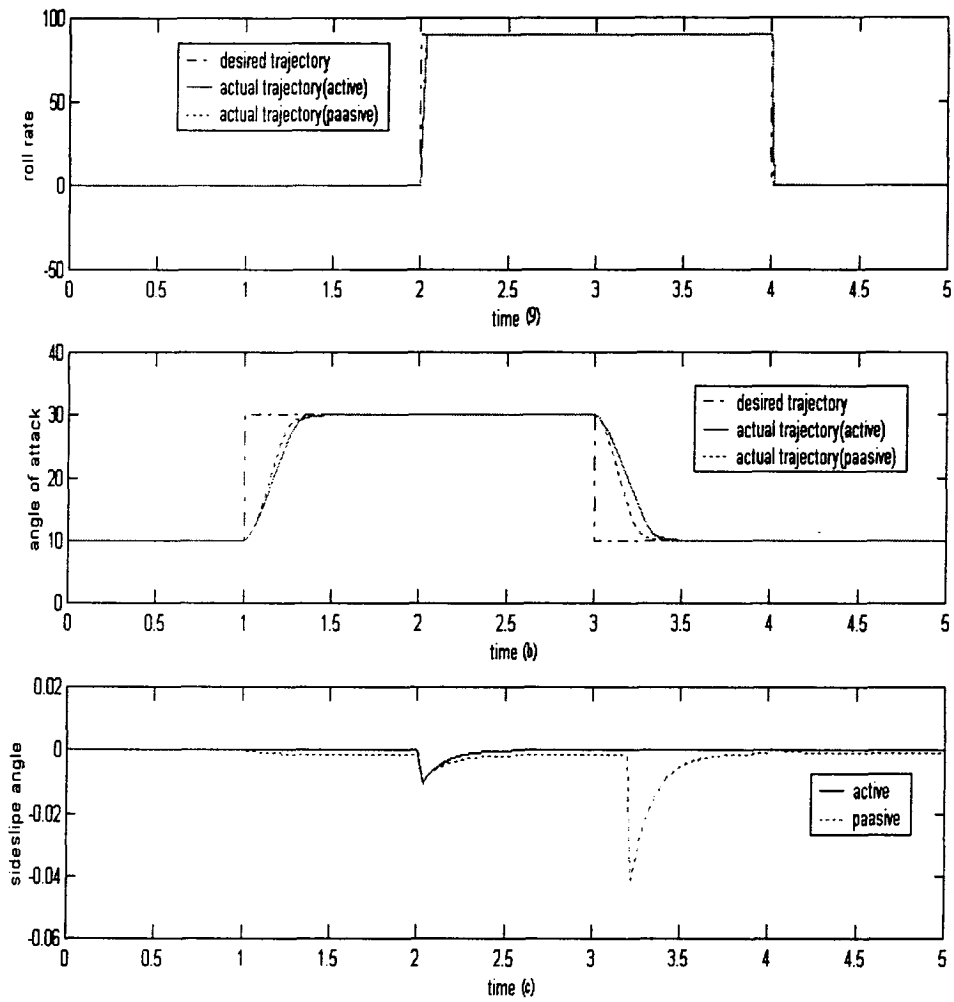


圖 5.20 主動式及被動式可靠度控制器在第四個致動器 δ_4 發生故障時的比較: (a)roll rate (b)angle of attack (c)sideslip angle

6 結論

本計畫的主要目的是在探討如何設計錯誤偵測與診斷的機制及可靠度控制器，並將理論結果應用於飛彈系統轉向控制上，最後分析其性能表現。所謂可靠度控制系統是指經由適當設計控制器，使系統能忍受特定致動器發生故障或異常的情況下，仍能保持穩定運作的系統，而錯誤診斷與偵測機制目的在於能偵測出系統當時的狀況並即時切換至可靠度控制器，使系統能有良好的性能表現。

在本計畫中我們已完成了下列事項：

- 提出了彈翼之錯誤偵測與診斷觀察器的設計方式
- 提出了被動式及主動式可靠度控制器的設計法則
- 將所提出的方法應用於 HAVE DASH II 飛彈系統上並進行數值分析與模擬

在本次計劃中的錯誤偵測與診斷部分，我們針對飛彈系統設計一觀察器，目的是希望一旦致動器發生故障時能即時偵測出來，並且作控制律的切換，使系統能夠穩定並達到預期的性能。由理論分析及數值模擬驗證了觀察器所產生的 residual signals 能夠成功且及時的

偵測及診斷出發生故障的致動器，提供系統及時的警告資訊。而在可靠度控制方面，我們利用可變結構控制(VSC)的概念來設計可靠度控制器，利用 VSC 來設計可靠度控制器的好處為反應速度快、對系統數學模式不確定性及外界干擾的抑制性能佳 (Robustness)，且可大幅減輕傳統上必須求解 Hamilton-Jacobi 方程式之計算負擔。由模擬結果可看出無論是主動式或被動式可靠度控制器都能有效讓系統達成良好的性能表現，然而，主動式可靠度控制在系統發生錯誤前，可充分應用所有控制器，當錯誤發生後又包含有錯誤偵測與診斷之結果，因此，其所需耗費的控制能量及性能表現在系統正常及異常時通常都較被動式可靠度控制佳。

參考文獻

- [1] Y.-W. Liang, D.-C. Liaw and T.-C Lee, "Reliable control of nonlinear systems," *IEEE Transactions on automatic control*, vol. 45, No. 4, pp.706-710, 2000.
- [2] D. -C. Liaw, Y. -W. Liang, and T. -C. Lee, "Minimum energy design of nonlinear systems," Proc. 1997 International Symposium on Nonlinear Theory and its Applications, Hawaii, pp. 61-64, 1997.
- [3] Vidyasagar M. and Viswanadham N., "Algebraic design techniques for reliable stabilization," *IEEE Transaction Automatic Control*, pp.1085-1095, 1982.
- [4] R. J. Veillette, "Reliable linear-quadratic state-feedback control," *Automatica*, vol. 31, pp.137-143, 1995.
- [5] G. -H. Yang, J. Lam, and J. Wang, "Reliable controller design for nonlinear system," *Proc. 35th IEEE Conf. on Decision and Control*, pp. 112-117, 1996.
- [6] G. -H. Yang, Y. Liu, and J. Wang, and C. B. Soh, "Reliable nonlinear controller system design using redundant control element," *Proc. 35th IEEE Conf. on Decision and Control*, pp. 118-123, 1996.
- [7] J. Huang and C. -F. Lin, "Application of Sliding Mode Control to Bank-To-Turn Missile Systems," *Proceedings of the First IEEE Regional Conference on Aerospace Control Systems*, pp. 569-573, 1993.
- [8], J. -C. Juang, C. -F. Lin, J. R. Cloutier and J. H. Evers,

- “Robust full-envelope missile autopilot design,” *Proceedings of the First IEEE Conference on Control Applications*, vol 1, pp. 231-236, 1992.
- [9] F. Liao, J. L. Wang and G. -H. Yang, “Reliable robust flight tracking control: an LMI approach,” *IEEE Transactions on Control Systems Technology*, Vol 10, Issue 1, pp. 76-89, 2002.
- [10] D. K. Chwa and J. Y. Choi, “New parametric affine modeling and control for skid-to-turn missiles,” *IEEE Transactions on Control Systems Technology*, Vol 9, Issue 2, pp. 335-347, 2001.
- [11] J. Huang, C. -F. Lin, J. R. Cloutier, J. H. Evers, C. D’ Souza, “Robust feedback linearization approach to autopilot design,” *Proceedings of the First IEEE Regional Conference on Aerospace Control Systems*, vol 1, pp. 220-225, 1992.
- [12] G. -H. Yang, S. -Y. Zhang, J. Lam and J. L. Wang, “Reliable control using redundant controllers,” *IEEE Transactions on Automatic Control*, Vol 43, Issue 11, pp. 1588-1593, 1998.

附錄 A HAVE DASH II 飛彈系統模 擬之 SIMNON 和 MATLAB 程式

Passive design

```

CONTINUOUS SYSTEM PMMAC11
" passive control
"=====symbol define=====
"x1:P x2:Q x3:R x4:fi x5:thita x6:fiine x7:Vm x8:afa x9:beta
"=====
STATE x1 x2 x3 x4 x5 x6 x8 x9 Vm
DER dx1 dx2 dx3 dx4 dx5 dx6 dx8 dx9 dVm
TIME t
"=====
" parameter set
pi:3.1415926
afacdot:0
afacddot:0
Pscdot:0
t11=30*pi/180
t111=10*pi/180
afac=IF t<1 THEN t11 ELSE IF t>3 THEN t111 ELSE t11
t33=(90)*pi/180
Psc=IF t<2 THEN 0 ELSE IF t>4 THEN 0 ELSE t33
"=====
gx=-ma*g*sin(x5)
gy=ma*g*cos(x5)*sin(x6)
gz=ma*g*cos(x5)*cos(x6)
ax=km*lo*Vm*Vm
af=kf*lo*Vm*Vm
"=====
Cx0=-0.57+0.0083*x8
Cy0=-0.21*x9
Cz1=-0.0015*(x8*x8*x8)+0.0125*(x8*x8)-0.5052*x8+0.0429
Cz2=0.0006*(x8*x8*x8)-0.0138*(x8*x8)+0.1230*x8-0.0191
Cz0=Cz1+Cz2*(Vm/c)
Fx=af*Cx0
Fy=af*Cy0
Fz=af*Cz0
"=====
C10=0.116*x9
Cm1=-0.0055*((x8)^3)+0.2131*((x8)^2)-2.7419*x8-0.0381
Cm2=0.0014*((x8)^3)-0.0623*((x8)^2)+0.8715*x8-0.4041
Cm0=Cm1+Cm2*(Vm/c)
Cn0=0.08*x9
"=====
d1=-(lzz-lyy)/lxx*x2*x3
d2=-(lxx-lzz)/lyy*x1*x3
d3=-(lyy-lxx)/lzz*x1*x2
d5=cos(x6)*x2-sin(x6)*x3
d6=x1+sin(x6)*tan(x5)*x2+cos(x6)*tan(x5)*x3
d81=x2-(x1*cos(x8)+x3*sin(x8))*tan(x9)
d82=(1/(ma*Vm*cos(x9)))*(-sin(x8)*(Fx+gx)+cos(x8)*(Fz+gz))

```

```

d8=d81+d82
d91=x1*sin(x8)-x3*cos(x8)+1/(ma*Vm)*(-cos(x8)*sin(x9)*(Fx+gx))
d92=(1/(ma*Vm))*(cos(x9)*(Fy+gy)-sin(x8)*sin(x9)*(Fz+gz))
d9=d91+d92
" =====
Cx0dot=0.0083*d8
Cy0dot=-0.21*d9
Cz1dot=(-0.0045*x8*x8+0.025*x8-0.5052)*d8
Cz2dot=(0.0018*x8*x8-0.0276*x8+0.123)*d8*(Vm/c)
Cz0dot=Cz1dot+Cz2dot
" =====
gxdot=-ma*g*cos(x5)*d5
gydot=ma*g*(-sin(x5)*sin(x6)*d5+cos(x5)*cos(x6)*d6)
gzdot=-ma*g*(sin(x5)*cos(x6)*d5+cos(x5)*sin(x6)*d6)
" =====
B11=d1*cos(x8)+d3*sin(x8)
B12=(-x1*sin(x8)+x3*cos(x8))*d8
B13=ax*(cos(x8)/Ixx*C10+sin(x8)/Izz*Cn0)
B1=B11+B12+B13
B21=-(x1*cos(x8)+x3*sin(x8))/((cos(x9))^2)*d9
B22=-(x1*sin(x8)+x3*cos(x8))*tan(x9)*d8
B231=-(1/(ma*Vm*cos(x9)*cos(x9)))*(-sin(x8)*(Fx+gx))*d9
B232=-(1/(ma*Vm*cos(x9)*cos(x9)))*(cos(x8)*(Fz+gz))*d9
B23=B231+B232
B241=1/(ma*Vm*cos(x9))*(-cos(x8)*(Fx+gx))*d8
B242=1/(ma*Vm*cos(x9))*(sin(x8)*(Fz+gz))*d8
B24=B241+B242
B251=1/(ma*Vm*cos(x9))*(-sin(x8)*(af*Cx0dot+gxdot))
B252=1/(ma*Vm*cos(x9))*(cos(x8)*(af*Cz0dot+gzdot))
B25=B251+B252
B26=d2-(d1*cos(x8)*tan(x9)+d3*sin(x8)*tan(x9))
B27=ax*(-cos(x8)*tan(x9)/Ixx*C10+Cm0/Iyy-sin(x8)*tan(x9)/Izz*Cn0)
B2=B21+B22+B23+B24+B25+B26+B27
B31=-(-sin(x8)*cos(x9)*d8+cos(x8)*sin(x9)*d9)*(Fx+gx)
B32=-sin(x9)*d9*(Fy+gy)
B33=-(cos(x8)*sin(x9)*d8+sin(x8)*cos(x9)*d9)*(Fz+gz)
B34=-cos(x8)*cos(x9)*(af*Cx0dot+gxdot)
B35=cos(x9)*(af*Cy0dot+gydot)
B36=-sin(x8)*sin(x9)*(af*Cz0dot+gzdot)
B37=d1*sin(x8)-d3*cos(x8)
B38=ax*(sin(x8)/Ixx*C10-cos(x8)/Izz*Cn0)
B39=(x1*cos(x8)+x3*sin(x8))*d8
B3=-1/(ma*Vm)*(B31+B32+B33+B34+B35+B36)+B37+B38+B39
" =====
" find A(x) and A(x) inverse
"(g1 g2 g3)^-1(x)=[g1 g2 g3;g4 g5 g6;g7 g8 g9];
" =====
a11=0*ax
a12=cos(x8)/Ixx*(-0.127)*ax
a13=sin(x8)/Iyy*(-0.584)*ax
a21=(1/Iyy)*(-0.675)*ax
a22=-cos(x8)*tan(x9)/Ixx*(-0.127)*ax
a23=-sin(x8)*tan(x9)/Izz*(-0.584)*ax
a31=0*ax
a32=sin(x8)/Ixx*(-0.127)*ax
a33=-cos(x8)/Izz*(-0.584)*ax
g11=a11*0.25-a12*0.25+a13*0.25
g12=-a11*0.25-a12*0.25+a13*0.25

```

```

g13=-a11*0.25-a12*0.25-a13*0.25
g14=a11*0.25-a12*0.25-a13*0.25
g21=a21*0.25-a22*0.25+a23*0.25
g22=-a21*0.25-a22*0.25+a23*0.25
g23=-a21*0.25-a22*0.25-a23*0.25
g24=a21*0.25-a22*0.25-a23*0.25

g31=a31*0.25-a32*0.25+a33*0.25
g32=-a31*0.25-a32*0.25+a33*0.25
g33=-a31*0.25-a32*0.25-a33*0.25
g34=a31*0.25-a32*0.25-a33*0.25
"=====find the upper bound of g4*u4
"kk1=(abs(a11)+abs(a12)+abs(a13))*0.25
"kk2=(abs(a21)+abs(a22)+abs(a23))*0.25
"kk3=(abs(a31)+abs(a32)+abs(a33))*0.25
"kc1=kk1*abs(u40)
"kc2=kk2*abs(u40)
"kc3=kk3*abs(u40)
"=====without upper bound
con:30
kc1=abs(g11*con)
kc2=abs(g21*con)
kc3=abs(g31*con)
delta=(g12*g23*g34+g13*g24*g32+g14*g22*g33-g32*g23*g14-g12*g33*g24-g22*g13*g34)
g1=(g23*g34-g33*g24)/delta
g2=-((g13*g34-g33*g14)/delta
g3=(g13*g24-g23*g14)/delta
g4=-((g22*g34-g32*g24)/delta
g5=-((g12*g34-g32*g14)/delta
g6=-((g12*g24-g22*g14)/delta
g7=-(g22*g33-g32*g23)/delta
g8=-((g12*g33-g32*g13)/delta
g9=(g12*g23-g22*g13)/delta
Ps=x1*cos(x8)+x3*sin(x8)
e1=Ps-Psc
e2=x8-afac
e3=x9
e2dot=d8-afacdot
e3dot=d9
s1=e1
s2=e2dot+k21*e2
s3=e3dot+k31*e3
s11=-Pscdot
s22=k21*e2dot-afacddot
s33=k31*d9
"=====
"calculate control law
"=====
t1=-s11-B1
t2=-s22-B2
t3=-s33-B3
"=====calculate u1eq u2eq u3eq
u2eq=g1*t1+g2*t2+g3*t3
u3eq=g4*t1+g5*t2+g6*t3
u4eq=g7*t1+g8*t2+g9*t3
"=====setting which actuator fail and gain=====
p1=1
p2=1

```

```

p3=1
p4=1
" =====
" delta_re=(gx)^-1(-eta*sgn(s)-g4*u40)
" u40 imply the actuator u4 fixed at fixed degree
" g4=[g14 g24 g34]^T

sat1=IF abs(s1/eps1)>1 THEN sign(s1/eps1) ELSE s1/eps1
sat2=IF abs(s2/eps2)>1 THEN sign(s2/eps2) ELSE s2/eps2
sat3=IF abs(s3/eps3)>1 THEN sign(s3/eps3) ELSE s3/eps3
"ul fail
ss4=s1*g11+s2*g21+s3*g31
sat4=IF abs(ss4/eps3)>1 THEN sign(ss4/eps3) ELSE ss4/eps3
ttl=-(kcl+etal)*sat1
tt2=-(kc2+eta2)*sat2
tt3=-(kc3+eta3)*sat3
u2re=g1*ttl+g2*tt2+g3*tt3
u3re=g4*ttl+g5*tt2+g6*tt3
u4re=g7*ttl+g8*tt2+g9*tt3
" =====
ulre=-eta4*sat4
"ulre=1
" =====
u11=ulre
u333=u3eq+u3re
u444=u4eq+u4re
u222=u2eq+u2re
" =====
limit=30
u11=IF abs(u11)>limit THEN sign(u11)*limit ELSE u11
u22=IF abs(u22)>limit THEN sign(u22)*limit ELSE u22
u44=IF abs(u44)>limit THEN sign(u44)*limit ELSE u44
u33=IF abs(u333)>limit THEN sign(u333)*limit ELSE u333
" =====
C1=C10-0.127*(0.25*(-p1*u11-p2*u22-p3*u33-p4*u44))
Cm=Cm0-0.675*(0.25*(p1*u11-p2*u22-p3*u33+p4*u44))
Cn=Cn0-0.584*(0.25*(p1*u11+p2*u22-p3*u33-p4*u44))
L=ax*C1
M=ax*Cm
N=ax*Cn
" =====
Vm1=cos(x8)*cos(x9)*(Fx+gx)
Vm2=sin(x9)*(Fy+gy)+sin(x8)*cos(x9)*(Fz+gz)
dx1=-(-Izz-Iyy)/Ixx*x2*x3+L/Ixx
dx2=-(-Ixx-Izz)/Iyy*x1*x3+M/Iyy
dx3=-(-Iyy-Ixx)/Izz*x1*x2+N/Izz
dx4=sin(x6)/cos(x5)*x2+cos(x6)/cos(x5)*x3
dx5=cos(x6)*x2-sin(x6)*x3
dx6=x1+sin(x6)*tan(x5)*x2+cos(x6)*tan(x5)*x3
dVm=1/m*(Vm1+Vm2)*0
dx8=d8
dx9=d9
" =====
" true value (degree)
tx1=x1*180/pi
tx2=x2*180/pi
tx3=x3*180/pi
tx4=x4*180/pi

```

```
tx5=x5*180/pi
tx6=x6*180/pi
tx8=x8*180/pi
tx9=x9*180/pi
tPs=Ps*180/pi
tafac=afac*180/pi
tPsc=Psc*180/pi
"=====
g:32.174
ma:9.89
c:968
kf:0.1534
km:0.0959
lo:0.0005124
lxx:1.1913
lyy:100.51
lzz:100.57
k21:20
k31:8
eta1:15
eta2:10
eta3:28
eta4:0.01
eps1=1*pi/180
eps2=0.5*pi/180
eps3=0.5*pi/180
u10:10
"=====
END
```

Active design

```

CONTINUOUS SYSTEM AACMISL
"=====
"-The forth actuator is fault at t=1.1sec and the control law will =
"=be changed to active form from normal form. =
"=====
"=====symbol define=====
"x1:P x2:Q x3:R x4:fi x5:thita x6:fiine x7:Vm x8:afa x9:beta
"=====
STATE x1 x2 x3 x4 x5 x6 x8 x9 Vm z1 z2 z3 o1 o2 o3
DER dx1 dx2 dx3 dx4 dx5 dx6 dx8 dx9 dVm dz1 dz2 dz3 dol do2 do3
TIME t
"=====
"parameter set
pi:3.1415926
afacdot:0
afacddot:0
Pscdot:0
t11=30*pi/180
t111=10*pi/180
afac=IF t<1 THEN t111 ELSE IF t>3 THEN t111 ELSE t11
t33=(90)*pi/180
Psc=IF t<2 THEN 0 ELSE IF t>4 THEN 0 ELSE t33
"=====
gx=-ma*g*sin(x5)
gy=ma*g*cos(x5)*sin(x6)
gz=ma*g*cos(x5)*cos(x6)
ax=-km*1o*Vm*Vm
af=kf*1o*Vm*Vm
"=====
Cx0=-0.57+0.0083*x8
Cy0=-0.21*x9
Cz1=-0.0015*(x8*x8*x8)+0.0125*(x8*x8)-0.5052*x8+0.0429
Cz2=0.0006*(x8*x8*x8)-0.0138*(x8*x8)+0.1230*x8-0.0191
Cz0=Cz1+Cz2*(Vm/c)
Fx=af*Cx0
Fy=af*Cy0
Fz=af*Cz0
"=====
C10=0.116*x9
Cm1=-0.0055*((x8)^3)+0.2131*((x8)^2)-2.7419*x8-0.0381
Cm2=0.0014*((x8)^3)-0.0623*((x8)^2)+0.8715*x8-0.4041
Cm0=Cm1+Cm2*(Vm/c)
Cn0=0.08*x9
"=====
d1=-((Izz-Iyy)/Ixx*x2*x3)
d2=-((Ixx-Izz)/Iyy*x1*x3)
d3=-((Iyy-Ixx)/Izz*x1*x2)
d5=cos(x6)*x2-sin(x6)*x3
d6=x1+sin(x6)*tan(x5)*x2+cos(x6)*tan(x5)*x3
d81=x2-(x1*cos(x8)+x3*sin(x8))*tan(x9)
d82=(1/(ma*Vm*cos(x9)))*(-sin(x8)*(Fx+gx)+cos(x8)*(Fz+gz))
d8=d81+d82
d91=x1*sin(x8)-x3*cos(x8)+1/(ma*Vm)*(-cos(x8)*sin(x9)*(Fx+gx))
d92=(1/(ma*Vm))*(cos(x9)*(Fy+gy)-sin(x8)*sin(x9)*(Fz+gz))

```

```

d9=d91+d92
"=====
Cx0dot=0.0083*d8
Cy0dot=-0.21*d9
Cz1dot=(-0.0045*x8*x8+0.025*x8-0.5052)*d8
Cz2dot=(0.0018*x8*x8-0.0276*x8+0.123)*d8*(Vm/c)
Cz0dot=Cz1dot+Cz2dot
"=====
gxdot=-ma*g*cos(x5)*d5
gydot=ma*g*(-sin(x5)*sin(x6)*d5+cos(x5)*cos(x6)*d6)
gzdot=-ma*g*(sin(x5)*cos(x6)*d5+cos(x5)*sin(x6)*d6)
"=====
B11=d1*cos(x8)+d3*sin(x8)
B12=(-x1*sin(x8)+x3*cos(x8))*d8
B13=ax*(cos(x8)/Ixx*C10+sin(x8)/Izz*Cn0)
B1=B11+B12+B13
B21=-(x1*cos(x8)+x3*sin(x8))/((cos(x9))^2)*d9
B22=-(-x1*sin(x8)+x3*cos(x8))*tan(x9)*d8
B231=-1/(ma*Vm*cos(x9)*cos(x9))*(-sin(x8)*(Fx+gx))*d9
B232=-1/(ma*Vm*cos(x9)*cos(x9))*(cos(x8)*(Fz+gz))*d9
B23=B231+B232
B241=1/(ma*Vm*cos(x9))*(-cos(x8)*(Fx+gx))*d8
B242=1/(ma*Vm*cos(x9))*(sin(x8)*(Fz+gz))*d8
B24=B241+B242
B251=1/(ma*Vm*cos(x9))*(-sin(x8)*(af*Cx0dot+gxdot))
B252=1/(ma*Vm*cos(x9))*(cos(x8)*(af*Cz0dot+gzdot))
B25=B251+B252
B26=d2-(d1*cos(x8)*tan(x9)+d3*sin(x8)*tan(x9))
B27=ax*(-cos(x8)*tan(x9)/Ixx*C10+Cm0/Iyy-sin(x8)*tan(x9)/Izz*Cn0)
B2=B21+B22+B23+B24+B25+B26+B27
B31=-(-sin(x8)*cos(x9)*d8+cos(x8)*sin(x9)*d9)*(Fx+gx)
B32=-sin(x9)*d9*(Fy+gy)
B33=-cos(x8)*sin(x9)*d8+sin(x8)*cos(x9)*d9*(Fz+gz)
B34=-cos(x8)*cos(x9)*(af*Cx0dot+gxdot)
B35=-cos(x9)*(af*Cy0dot+gydot)
B36=-sin(x8)*sin(x9)*(af*Cz0dot+gzdot)
B37=d1*sin(x8)-d3*cos(x8)
B38=ax*(sin(x8)/Ixx*C10-cos(x8)/Izz*Cn0)
B39=(x1*cos(x8)+x3*sin(x8))*d8
B3=-1/(ma*Vm)*(B31+B32+B33+B34+B35+B36)+B37+B38+B39
"=====
" find A(x) and A(x) inverse
"A^-1(x)=[a1 a2 a3;a4 a5 a6;a7 a8 a9];
"=====
a11=0*ax
a12=cos(x8)/Ixx*(-0.127)*ax
a13=sin(x8)/Iyy*(-0.584)*ax
a21=(1/Iyy)*(-0.675)*ax
a22=-cos(x8)*tan(x9)/Ixx*(-0.127)*ax
a23=-sin(x8)*tan(x9)/Izz*(-0.584)*ax
a31=0*ax
a32=sin(x8)/Ixx*(-0.127)*ax
a33=-cos(x8)/Izz*(-0.584)*ax
delta=(a11*a22*a33+a12*a23*a31+a13*a21*a32-a31*a22*a13-a11*a32*a23-a21*a12*a33)
a1=(a22*a33-a23*a32)/delta
a2=-a12*a33-a13*a32/delta
a3=(a12*a23-a13*a22)/delta
a4=-a21*a33-a31*a23/delta

```

$a5 = (a11*a33 - a13*a31) / \delta$
 $a6 = -(a11*a23 - a21*a13) / \delta$
 $a7 = (a21*a32 - a22*a31) / \delta$
 $a8 = -(a11*a32 - a31*a12) / \delta$
 $a9 = (a11*a22 - a21*a12) / \delta$
 $g11 = a11*0.25 - a12*0.25 + a13*0.25$
 $g12 = -a11*0.25 - a12*0.25 + a13*0.25$
 $g13 = -a11*0.25 - a12*0.25 - a13*0.25$
 $g14 = a11*0.25 - a12*0.25 - a13*0.25$
 $g21 = a21*0.25 - a22*0.25 + a23*0.25$
 $g22 = -a21*0.25 - a22*0.25 + a23*0.25$
 $g23 = -a21*0.25 - a22*0.25 - a23*0.25$
 $g24 = a21*0.25 - a22*0.25 - a23*0.25$
 $g31 = a31*0.25 - a32*0.25 + a33*0.25$
 $g32 = -a31*0.25 - a32*0.25 + a33*0.25$
 $g33 = -a31*0.25 - a32*0.25 - a33*0.25$
 $g34 = a31*0.25 - a32*0.25 - a33*0.25$
 $\delta1\text{taa} = (g12*g23*g34 + g13*g24*g32 + g14*g22*g33 - g32*g23*g14 - g12*g33*g24 - g22*g13*g34)$
 $g1a = (g23*g34 - g24*g33) / \delta1\text{taa}$
 $g2a = -(g13*g34 - g14*g33) / \delta1\text{taa}$
 $g3a = (g13*g24 - g14*g23) / \delta1\text{taa}$
 $g4a = -(g22*g34 - g32*g24) / \delta1\text{taa}$
 $g5a = (g12*g34 - g14*g32) / \delta1\text{taa}$
 $g6a = -(g12*g24 - g22*g14) / \delta1\text{taa}$
 $g7a = (g22*g33 - g23*g32) / \delta1\text{taa}$
 $g8a = -(g12*g33 - g32*g13) / \delta1\text{taa}$
 $g9a = (g12*g23 - g22*g13) / \delta1\text{taa}$
 $\delta1\text{tab} = (g11*g23*g34 + g13*g24*g31 + g14*g21*g33 - g31*g23*g14 - g11*g33*g24 - g21*g13*g34)$
 $g1b = (g23*g34 - g24*g33) / \delta1\text{tab}$
 $g2b = -(g13*g34 - g14*g33) / \delta1\text{tab}$
 $g3b = (g13*g24 - g14*g23) / \delta1\text{tab}$
 $g4b = -(g21*g34 - g31*g24) / \delta1\text{tab}$
 $g5b = (g11*g34 - g14*g31) / \delta1\text{tab}$
 $g6b = -(g11*g24 - g21*g14) / \delta1\text{tab}$
 $g7b = (g21*g33 - g23*g31) / \delta1\text{tab}$
 $g8b = -(g11*g33 - g31*g13) / \delta1\text{tab}$
 $g9b = (g11*g23 - g21*g13) / \delta1\text{tab}$
 $\delta1\text{tac} = (g11*g22*g34 + g12*g24*g31 + g14*g21*g32 - g31*g22*g14 - g11*g32*g24 - g21*g12*g34)$
 $g1c = (g22*g34 - g24*g32) / \delta1\text{tac}$
 $g2c = -(g12*g34 - g14*g32) / \delta1\text{tac}$
 $g3c = (g12*g24 - g14*g22) / \delta1\text{tac}$
 $g4c = -(g21*g34 - g31*g24) / \delta1\text{tac}$
 $g5c = (g11*g34 - g14*g31) / \delta1\text{tac}$
 $g6c = -(g11*g24 - g21*g14) / \delta1\text{tac}$
 $g7c = (g21*g32 - g22*g31) / \delta1\text{tac}$
 $g8c = -(g11*g32 - g31*g12) / \delta1\text{tac}$
 $g9c = (g11*g22 - g21*g12) / \delta1\text{tac}$
 $\delta1\text{tad} = (g11*g22*g33 + g12*g23*g31 + g13*g21*g32 - g31*g22*g13 - g11*g32*g23 - g21*g12*g33)$
 $g1d = (g22*g33 - g23*g32) / \delta1\text{tad}$
 $g2d = -(g12*g33 - g13*g32) / \delta1\text{tad}$
 $g3d = (g12*g23 - g13*g22) / \delta1\text{tad}$
 $g4d = -(g21*g33 - g31*g23) / \delta1\text{tad}$
 $g5d = (g11*g33 - g13*g31) / \delta1\text{tad}$
 $g6d = -(g11*g23 - g21*g13) / \delta1\text{tad}$
 $g7d = (g21*g32 - g22*g31) / \delta1\text{tad}$
 $g8d = -(g11*g32 - g31*g12) / \delta1\text{tad}$
 $g9d = (g11*g22 - g21*g12) / \delta1\text{tad}$

" =====


```

Ps=x1*cos(x8)+x3*sin(x8)
e1=Ps-Psc

e2=x8-afac
e3=x9
e2dot=d8-afacdot
e3dot=d9
s1=e1
s2=e2dot+k21*e2
s3=e3dot+k31*e3
s11=-Pscdot
s22=k21*e2dot-afacddot
s33=k31*d9
"=====
"calculate control law
"=====
sat1=IF abs(s1/eps1)>1 THEN sign(s1/eps1) ELSE s1/eps1
sat2=IF abs(s2/eps2)>1 THEN sign(s2/eps2) ELSE s2/eps2
sat3=IF abs(s3/eps3)>1 THEN sign(s3/eps3) ELSE s3/eps3
t1=-s11-B1-eta1*sat1
t2=-s22-B2-eta2*sat2
t3=-s33-B3-eta3*sat3
"=====actual control law delta_e delta_a delta_r
u1=a1*t1+a2*t2+a3*t3
u2=a4*t1+a5*t2+a6*t3
u3=a7*t1+a8*t2+a9*t3
u11=1*u1-1*u2+1*u3
u22=-1*u1-1*u2+1*u3
u33=-1*u1-1*u2-1*u3
u44=1*u1-1*u2-1*u3
ud1=-3.5619623
u1ac=IF t<ts THEN u11 ELSE u11
ud2=3.5619131
u2ac=IF t<ts THEN u22 ELSE u22
ud3=3.5619692
u3ac=IF t<ts THEN u33 ELSE u33
ud4=-3.5619218
u4ac=IF t<ts THEN u44 ELSE ud4
t1a=-s11-B1-g11*u1ac
t2a=-s22-B2-g21*u1ac
t3a=-s33-B3-g31*u1ac
t1b=-s11-B1-g12*u2ac
t2b=-s22-B2-g22*u2ac
t3b=-s33-B3-g32*u2ac
t1c=-s11-B1-g13*u3ac
t2c=-s22-B2-g23*u3ac
t3c=-s33-B3-g33*u3ac
t1d=-s11-B1-g14*u4ac
t2d=-s22-B2-g24*u4ac
t3d=-s33-B3-g34*u4ac
"=====
"calculate ueq & ure
"=====
u2eqa=g1a*t1a+g2a*t2a+g3a*t3a
u3eqa=g4a*t1a+g5a*t2a+g6a*t3a
u4eqa=g7a*t1a+g8a*t2a+g9a*t3a
u1eqb=g1b*t1b+g2b*t2b+g3b*t3b
u3eqb=g4b*t1b+g5b*t2b+g6b*t3b

```

```

u4eqb=g7b*t1b+g8b*t2b+g9b*t3b
uleqc=g1c*t1c+g2c*t2c+g3c*t3c
u2eqc=g4c*t1c+g5c*t2c+g6c*t3c
u4eqc=g7c*t1c+g8c*t2c+g9c*t3c
uleqd=g1d*t1d+g2d*t2d+g3d*t3d
u2eqd=g4d*t1d+g5d*t2d+g6d*t3d
u3eqd=g7d*t1d+g8d*t2d+g9d*t3d
tt1a=-(0+eta1a)*sat1
tt2a=-(0+eta2a)*sat2
tt3a=-(0+eta3a)*sat3
tt1b=-(0+eta1b)*sat1
tt2b=-(0+eta2b)*sat2
tt3b=-(0+eta3b)*sat3
tt1c=-(0+eta1c)*sat1
tt2c=-(0+eta2c)*sat2
tt3c=-(0+eta3c)*sat3
tt1d=-(0+eta1d)*sat1
tt2d=-(0+eta2d)*sat2
tt3d=-(0+eta3d)*sat3
u2rea=g1a*tt1a+g2a*tt2a+g3a*tt3a
u3rea=g4a*tt1a+g5a*tt2a+g6a*tt3a
u4rea=g7a*tt1a+g8a*tt2a+g9a*tt3a
ulreb=g1b*tt1b+g2b*tt2b+g3b*tt3b
u3reb=g4b*tt1b+g5b*tt2b+g6b*tt3b
u4reb=g7b*tt1b+g8b*tt2b+g9b*tt3b
ulrec=g1c*tt1c+g2c*tt2c+g3c*tt3c
u2rec=g4c*tt1c+g5c*tt2c+g6c*tt3c
u4rec=g7c*tt1c+g8c*tt2c+g9c*tt3c
ulred=g1d*tt1d+g2d*tt2d+g3d*tt3d
u2red=g4d*tt1d+g5d*tt2d+g6d*tt3d
u3red=g7d*tt1d+g8d*tt2d+g9d*tt3d
u111a=u1ac
u222a=u2eqa+u2rea
u333a=u3eqa+u3rea
u444a=u4eqa+u4rea
u111b=u1eqb+ulreb
u222b=u2ac
u333b=u3eqb+u3reb
u444b=u4eqb+u4reb
u111c=u1eqc+ulrec
u222c=u2eqc+u2rec
u333c=u3ac
u444c=u4eqc+u4rec
u111d=u1eqd+ulred
u222d=u2eqd+u2red
u333d=u3eqd+u3red
u444d=u4ac
limit=30
u222a=IF abs(u222a)>limit THEN sign(u222a)*limit ELSE u222a
u333a=IF abs(u333a)>limit THEN sign(u333a)*limit ELSE u333a
u444a=IF abs(u444a)>limit THEN sign(u444a)*limit ELSE u444a
u111b=IF abs(u111b)>limit THEN sign(u111b)*limit ELSE u111b
u333b=IF abs(u333b)>limit THEN sign(u333b)*limit ELSE u333b
u444b=IF abs(u444b)>limit THEN sign(u444b)*limit ELSE u444b
u111c=IF abs(u111c)>limit THEN sign(u111c)*limit ELSE u111c
u222c=IF abs(u222c)>limit THEN sign(u222c)*limit ELSE u222c
u444c=IF abs(u444c)>limit THEN sign(u444c)*limit ELSE u444c
u111d=IF abs(u111d)>limit THEN sign(u111d)*limit ELSE u111d

```

```

u222d=IF abs(u222d)>limit THEN sign(u222d)*limit ELSE u222d
u333d=IF abs(u333d)>limit THEN sign(u333d)*limit ELSE u333d
" =====
"switch the control law
" =====
thre=0.001
up1=IF abs(r2)>thre THEN u111d ELSE ulac
up11=IF abs(r2)>thre THEN u111b ELSE IF abs(r3)>thre THEN u111c ELSE u11
uu1=IF abs(r1)>thre THEN up1 ELSE up11
up2=IF abs(r2)>thre THEN u222d ELSE u222a
up22=IF abs(r2)>thre THEN u2ac ELSE IF abs(r3)>thre THEN u222c ELSE u22
uu2=IF abs(r1)>thre THEN up2 ELSE up22
up3=IF abs(r2)>thre THEN u333d ELSE u333a
up33=IF abs(r2)>thre THEN u333b ELSE IF abs(r3)>thre THEN u3ac ELSE u33
uu3=IF abs(r1)>thre THEN up3 ELSE up33
up4=IF abs(r2)>thre THEN u4ac ELSE u444a
up44=IF abs(r2)>thre THEN u444b ELSE IF abs(r3)>thre THEN u444c ELSE u44
uu4=IF abs(r1)>thre THEN up4 ELSE up44
" =====
"change the model to f+g*u
" =====
Vm1=cos(x8)*cos(x9)*(Fx+gx)
Vm2=sin(x9)*(Fy+gy)+sin(x8)*cos(x9)*(Fz+gz)
ti1=0.25*0.127*ax*(uu1+uu2+uu3+u4ac)/Ixx
ti2=0.25*(-0.675)*ax*(uu1-uu2-uu3+u4ac)/Iyy
ti3=0.25*(-0.584)*ax*(uu1+uu2-uu3-u4ac)/Izz
dx1=-(Izz-Iyy)/Ixx*x2*x3+ax*C10/Ixx+ti1
dx2=-(Ixx-Izz)/Iyy*x1*x3+ax*Cm0/Iyy+ti2
dx3=-(Iyy-Ixx)/Izz*x1*x2+ax*Cn0/Izz+ti3
dx4=sin(x6)/cos(x5)*x2+cos(x6)/cos(x5)*x3
dx5=cos(x6)*x2-sin(x6)*x3
dx6=x1+sin(x6)*tan(x5)*x2+cos(x6)*tan(x5)*x3
dVm=1/m*(Vm1+Vm2)*0
dx8=d8
dx9=d9
" =====
"Design the observer
" =====
f1=-(Izz-Iyy)/Ixx*x2*x3+ax*C10/Ixx
f2=-(Ixx-Izz)/Iyy*x1*x3+ax*Cm0/Iyy
f3=-(Iyy-Ixx)/Izz*x1*x2+ax*Cn0/Izz
dz1=0.0539*f1-0.8552*f2+uu1+u4ac
dz2=0.8552*f2-0.9891*f3+uu2-u4ac
dz3=0.0539*f1+0.9891*f3+uu3+u4ac
do1=0.0539*f1-0.8552*f2+uu1+u44+K1*(z1-o1)
do2=0.8552*f2-0.9891*f3+uu2-u44+K2*(z2-o2)
do3=0.0539*f1+0.9891*f3+uu3+u44+K3*(z3-o3)
r1=z1-o1
r2=z2-o2
r3=z3-o3
" =====
"alarm signal
" =====
ap=IF abs(r1)>0 THEN 1 ELSE IF abs(r2)>0 THEN 1 ELSE 0
alarm=IF abs(r3)>0 THEN 1 ELSE ap
" =====
" true value (degree)
tx1=x1*180/pi

```

```
tx2=x2*180/pi
tx3=x3*180/pi
tx4=x4*180/pi
tx5=x5*180/pi
tx6=x6*180/pi
tx8=x8*180/pi
tx9=x9*180/pi
tPs=Ps*180/pi
tafac=afac*180/pi
tPsc=Psc*180/pi
```

```
"=====
" parameters setting
"=====
```

```
g:32.174
ma:9.89
c:968
kf:0.1534
km:0.0959
lo:0.0005124
lxx:1.1913
lyy:100.51
lzz:100.57
k21:20
k31:8
etal:100
eta2:30
eta3:16
etald:100
eta2d:25
eta3d:35
etala:80
eta2a:35
eta3a:30
etalb:80
eta2b:30
eta3b:20
etalc:85
eta2c:30
eta3c:25
eps1=1*pi/180
eps2=0.5*pi/180
eps3=0.5*pi/180
m=9.89
K1:3
K2:3
K3:1.5
ts:1.1
```

```
"=====
END
```

Macro file

```
MACRO MAACMISL
SYST AACMISL
INIT x1:0
INIT x2:0
INIT x3:0
INIT x4:0
init x5:0
init x6:0
init x8:0.1745
init x9:0
init Vm:2662
STORE tx1 tx2 tx3 tx4 tx5 tx6 tPs tPsc tx8 tx9 tafac uul uu2 uu3 uu4 r1 r2 r3 alarm
error 1e-10
simu 0 5 /dl
"export f41<dl
newplot
split 3 2
  ashow tx1 /dl
  text 'angular rate x'
  ashow tx2 /dl
  show tx3 /dl
  text 'angular rate z'
  ashow tx4
  text 'yaw angle'
  ashow tx5
  text 'pitch angle'
  ashow tx6
  text 'roll angle'
  ashow tPs tPsc /dl
  text 'roll rate'
  ashow tx8 tafac /dl
  text 'angle of attack'
  ashow tx9 /dl
newplot
split 2 2
ashow uul /dl
text 'delta 1'
ashow uu2 /dl
text 'delta 2'
ashow uu3 /dl
text 'delta 3'
ashow uu4 /dl
text 'delta 4'
newplot
split 2 2
ashow r1 /dl
text 'residual 1'
ashow r2 /dl
text 'residual 2'
ashow r3 /dl
text 'residual 3'
ashow alarm /dl
text 'alarm signal'
switch color reset
END
```