

東海大學應用數學系
碩士論文

矩陣型 Lambert W Function 解
時滯微分系統
Solution and Stability Analysis of Delay
Systems via Matrical Lambert W Functions

研究生：楊秉諺
指導教授：黃皇男

中華民國 105 年 6 月 29 日

東海大學
應用數學系
碩士學位口試委員審定書

本系碩士班 楊秉諺 君

所提論文 SOLUTION AND STABILITY ANALYSIS OF DELAY
SYSTEMS VIA MATRICAL LAMBERT W FUNCTIONS
(矩陣型 LAMBERT W FUNCTION 解時滯微分系統)

合於碩士班資格水準，業經本委員會評審通過，特此證明。

口試委員：

吳旭堯

陳淑汀

陳碩聰

指導教授：

黃皇男

所長：

楊志煌

中華民國 一〇五 年 七 月 六 日

The image features a large, semi-transparent watermark of the Tsinghua University seal in the background. The seal is circular with a scalloped edge. It contains the university's name in Chinese characters '清華大學' at the top and 'TSINGHAI UNIVERSITY' in English around the bottom. The year '1911' is at the very bottom. In the center of the seal is a stylized emblem. Overlaid on the seal is the text '中文摘要' (Chinese Abstract) in a bold, black font.

中文摘要

考慮線性時滯微分系統，若由狀態轉移函數擬合時間響應，因是由一段段時間積分而成，因此時間響應將成變成分段可微的連續函數。但若能以 Lambert W function 擬合系統響應，則此響應乃是由可數無限多個作用在系統特徵值上的指數函數合成，因此必成為可微函數。本文針對多個時滯系統，從拉氏變換著手，得到轉移函數的拉氏變換，並運用 Laurent 級數，計算其由 Lambert W function 組合的轉移函數，以得到系統響應，這個響應為可微函數，將可供系統進一步進行穩定性分析與控制設計使用。

關鍵字：時滯系統、Lambert W function、狀態轉移函數、拉氏變換

The seal of Tonghai University is a circular emblem with a scalloped outer edge. It features a central cross-like symbol above a horizontal line, with the university's name in Chinese characters '通海大学' at the top and 'TONGHAI UNIVERSITY' in English around the bottom. The year '1955' is inscribed at the very bottom of the seal.

Abstract

For single delay systems, since the state transition function is consecutively obtained by integrating interval by interval and hence the response constructed by the state transition function is only piecewise differentiable. On the other hand, the response of single delay system will be differentiable one if it is constructed via the Lambert W function. In this work, we construct the state transition function by Lambert W function with the aid of Laplace transform together with Laurent series. After all the time response function for multiple delay systems is then constructed. This function can be further applied to the stability analysis and controller synthesis.

Keywords: time-delay system, Lambert W function, state transition function, Laplace transform

目錄

1	簡介	3
2	數學基礎知識	4
2.1	Lambert W Function	4
2.2	矩陣函數及其性質	4
2.3	拉氏變換	5
3	線性時滯微分方程求解與分析	8
3.1	單一時滯微分方程	8
3.1.1	狀態轉移函數	8
3.1.2	時域響應	9
3.1.3	以 Lambert W Function 求系統近似解	10
3.1.4	範例說明	12
3.2	兩項延遲時滯微分方程	12
3.2.1	狀態轉移函數	13
3.2.2	以 Lambert W Function 求系統近似解	14
3.2.3	範例說明	15
4	單一時滯微分系統	17
4.1	矩陣方程	17
4.2	狀態轉移矩陣	18
4.3	系統響應	19
4.4	範例說明	19
A	MATLAB 代碼	23
A.1	章節 3.1.4 代碼	23
A.2	章節 3.2.3 代碼	24
A.3	章節 4.4 代碼	25

圖目錄

2.1	Lambert W function 是 $f(x) = xe^x$ 的反函數，因在 $[-e^{-1}, 0)$ 的範圍內有兩個值，故定義值域大於 -1 的部分為 branch 0，其餘的部分為 branch -1	7
2.2	各個 branch 的 Lambert W function 的值域幾乎沒有交集，有交集的點如 $W_0(-e^{-1}) = W_{-1}(-e^{-1}) = -1$ 。	7
3.1	章節 3.1.4 結果	13
3.2	章節 3.2.3 結果，虛線部分為 Matlab 內建指令 <code>dde23</code> 得到的近似解	16
4.1	章節 4.4 響應，輸入為 0	21
4.2	章節 4.4 響應，上圖輸入為 $Bu(t) = (\cos t, 0)^T$ ，下圖輸入為 $Bu(t) = (\cos t, \sin t)^T$	22



第 1 章

簡介

時滯系統存在各大領域當中，如動態系統、大自然的生態、或物理、化學領域中皆存在需要以時滯方程解決的問題，考慮線性時滯系統，Malek-Zavarei 和 Jamshidi 以迭代的方式得到系統狀態轉移函數，再以狀態轉移函數得到系統解析解 [5]，Corless 以 Lambert W function 得到單一時滯方程齊次項的通解，因為由 Lambert W function 得到的特徵方程的解不重根，因此該解析解的形式為指數函數的線性組合 [2, 3]，Asl 和 Ulsoy 再將此法推廣至非齊次單一時滯方程，並以此法計算單一時滯系統的簡單例子，他們在附錄中提供系統常數項係數的計算方法 [1]，Yi 將此法完整系統化，可作用於任意單一時滯系統，在計算常數項係數方面，除了使用 Asl 和 Ulsoy 以矩陣的方式取得係數的近似值，也能使用拉氏變換得到常數項係數 [7]。

本文以拉氏變換得到多個時滯系統的狀態轉移函數，並以 Lambert W function 進行擬合表示，得到狀態轉移函數的解析形式，因此得對應系統的時間響應函數，其中對應起始條件的係數係利用拉氏變換與極限的觀念計算而得。

第二章簡單介紹閱讀本文所需要的基礎知識，包含 Lambert W function、矩陣函數與拉氏變換。第三章考慮與時滯方程相關的問題，過去 Lambert W function 只能擬合單一時滯系統的解析解，在方程方面，本文以與 Lambert W function 相關的迭代式以取得多項延遲方程的特徵方程的解，接下來便能使用與單一延遲方程相似的手法得到系統解析解。第四章考慮時滯系統，目前還無法解決多項延遲的問題，本文將著重討論矩陣型 Lambert W function 的運算方式，以及解決矩陣方程的方法。附錄則收錄本文使用於數值計算的 Matlab 程式碼。

第 2 章

數學基礎知識

為處理本文所討論的時滯系統，需要本章的前導知識。

2.1 Lambert W Function

Lambert W function 最早於 17 世紀被 Lambert 和 Euler 討論 [3]，其被定義為滿足

$$W(z)e^{W(z)} = z \quad (2.1)$$

的函數 W ，對某些 $z \in \mathbb{C}$ 。考慮定義在 \mathbb{R} 上的 W ，因其為 $f(t) = te^t$ 的反函數，因此在 $[-e^{-1}, 0)$ 上有兩個值，故定義值域大於 -1 的部分為 branch 0，其餘的部分為 branch -1 ，其中 $W_0(-e^{-1}) = W_{-1}(-e^{-1}) = -1$ [10]。考慮定義在 \mathbb{C} 上的 W ，因定義在 \mathbb{C} 上的對數函數可以有可數個值 [4]，因此 W 存在可數個 branch，記為 $W_k, \forall k \in \mathbb{Z}$ ，其中各項 branch 的值域除了邊界以外彼此互不相交。

Lambert W function 的微分

將 (2.1) 對 z 微分得

$$\dot{W}(z)e^{W(z)} + \dot{W}(z)W(z)e^{W(z)} = 1$$

因此有

$$\begin{aligned} \dot{W}(z) &= \frac{1}{1+W(z)} e^{-W(z)} \\ &= \frac{W(z)}{z(1+W(z))}, \quad \text{此時 } z \neq 0 \end{aligned}$$

此外，若反覆對 (2.1) 微分，會得到關係式

$$W^{(n)}(z) = \frac{p_n(w)}{(1+w)^{2n+1}} e^{-nw}$$

其中 $w = W(z)$ 且 p_n 為 w 的多項式，因此多項式 p_n 有下列關係式

$$p_{n+1}(w) = -(nw + 3n - 1)p_n(w) + (w + 1)p'_n(w)$$

對 $n \geq 1$ 且 $p_1(w) = 1$ 。Lambert W function 的其他性質在 [3] 中可以瞭解。

2.2 矩陣函數及其性質

顧名思義，矩陣函數是由矩陣對應至矩陣的函數。考慮指數函數，其以泰勒級數展開為

$$e^x = \sum_{k=0}^{\infty} \frac{1}{k!} x^k$$

故對任意方陣 $A \in M_n$ ，定義矩陣型指數

$$e^A = \sum_{k=0}^{\infty} \frac{1}{k!} A^k$$

顯然 $e^A \in M_n$ 且此時 A 為矩陣型指數的變數，矩陣型指數滿足下列性質 [10]：

性質 2.1. 令 $A, B \in M_n(\mathbb{C}), I, O \in M_n$ 分別為單位矩陣與零矩陣。

1. $e^O = I$
2. 對任意 A 與 e^A 具有乘法交換性，即 $Ae^A = e^A A$ 。
3. 若 A, B 具有乘法交換性，則 e^A, e^B 同樣也具有乘法交換性，且 $e^{A+B} = e^A e^B$ 。
4. 不論 A 是否存在可逆矩陣，對任意 e^A 皆可逆，且 $(e^A)^{-1} = e^{-A}$ 。

矩陣型 Lambert W Function

以矩陣型指數的觀念進一步定義矩陣型 Lambert W function，對某些 A 所構的 $n \times n$ 矩陣子集 \mathcal{A} ，定義 $W : \mathcal{A} \rightarrow M_n(\mathbb{C})$ 滿足

$$W(A)e^{W(A)} = A \quad (2.2)$$

則稱此 W 為矩陣型 Lambert W function。Yi 藉由 Pease 可以計算各項 branch 的矩陣型 Lambert W function [6, 9]，假設 $A \in M_n(\mathbb{C})$ 相似於 Jordan 矩陣

$$J = \text{diag}(J_1(\lambda_1), \dots, J_p(\lambda_p))$$

其中各項 $J_r(\lambda_r)$ 為特徵值 λ_r 的 n_r 階 Jordan block，其中 $r \in \{1, 2, \dots, p\}$ ， $1 \leq p \leq n$ 且 $\sum_{r=1}^p n_r = n$ 。因 $A = ZJZ^{-1}$ ，記 $W_{kr} = W_k(J_r(\lambda_r))$ ，因此

$$W_k(A) = Z \text{diag}(W_{k1}, \dots, W_{kp}) Z^{-1}$$

其中 W_k 作用於 Jordan block 的算法為

$$W_{kr} = \begin{pmatrix} W_k(\lambda_r) & \dot{W}_k(\lambda_r) & \cdots & \frac{1}{(n_r-1)!} W_k^{(n_r-1)}(\lambda_r) \\ & W_k(\lambda_r) & \ddots & \vdots \\ & & \ddots & \dot{W}_k(\lambda_r) \\ & & & W_k(\lambda_r) \end{pmatrix}$$

第四章的矩陣型 Lambert W function 將以此法做計算， W_k 的導函數的計算方式已於上一小節提及。

2.3 拉氏變換

給定函數 $f : \mathbb{R}^+ \rightarrow \mathbb{R}$ ，定義 f 的拉氏變換為

$$\mathcal{L}[f(t); s] = \int_0^{\infty} f(t)e^{-st} dt$$

是 \mathcal{L} 對算子 f 的函數。考慮 s 為複數，若 $|\mathcal{L}[f(t); s]| < +\infty$ ，則稱 f 為可拉氏變換函數。

拉氏變換有幾項簡單的性質：

1. 對任意常數 α 、可拉氏變換函數 f, g

$$\mathcal{L}[\alpha f(t) + g(t); s] = \alpha \mathcal{L}[f(t); s] + \mathcal{L}[g(t); s]$$

2. 假設函數 f 可進行拉氏變換且可微分

$$\mathcal{L}[\dot{f}(t); s] = s\mathcal{L}[f(t); s] - \lim_{t \rightarrow 0^+} f(t)$$

3. 令 $h > 0$, f 可進行拉氏變換

$$\mathcal{L}[f(t-h); s] = e^{-sh}\mathcal{L}[f(t); s] + e^{-sh} \int_{-h}^0 f(t)e^{-st} dt$$

4. 令 f, g 為可拉氏變換函數且定義

$$(f * g)(t) = \int_0^t f(t-\tau)g(\tau) d\tau$$

因此

$$\mathcal{L}[(f * g)(t); s] = \mathcal{L}[f(t); s]\mathcal{L}[g(t); s]$$

下面定理考慮拉氏變換的存在性問題。

定理 2.2. 令 $f: \mathbb{R}^+ \rightarrow \mathbb{R}$ 為分段連續函數且存在 $M > 0, s_0 \in \mathbb{R}$ 使得

$$|f(t)| \leq Me^{s_0 t}, \quad \forall t \geq 0$$

則 f 的拉氏變換存在條件為 $\operatorname{Re}(s) > s_0$ 。

拉氏變換定義在向量值、矩陣值函數

考慮向量值函數 (vector-valued function)

$$f(t) = (f_1(t), f_2(t), \dots, f_n(t))^T$$

其中各項 $f_i: \mathbb{R}^+ \rightarrow \mathbb{R}$ 皆可進行拉氏變換。定義其拉氏變換

$$\mathcal{L}[f(t); s] = (\mathcal{L}[f_1(t); s], \dots, \mathcal{L}[f_n(t); s])^T$$

此拉氏變換也滿足上述常數型拉氏變換的各項性質。考慮矩陣值函數 (matrix-valued function)

$$f(t) = (f_{ij}(t))_{n \times n}$$

其中各項 $f_{ij}: \mathbb{R}^+ \rightarrow \mathbb{R}$ 皆可進行拉氏變換，因此可定義拉氏變換為

$$\mathcal{L}[f(t); s] = (\mathcal{L}[f_{ij}(t); s])_{n \times n}$$

性質 2.3. 給定 $A \in M_n(\mathbb{R})$, 已知矩陣值函數 e^{At} 可表示為

$$I + At + \frac{1}{2!}A^2t^2 + \frac{1}{3!}A^3t^3 + \dots$$

其中 I 為 $n \times n$ 單位矩陣, 因此 e^{At} 的拉氏變換為

$$\mathcal{L}[e^{At}; s] = (sI - A)^{-1}$$

若 s 為矩陣 A 的特徵值, 則 e^{At} 的拉氏變換不可解析。

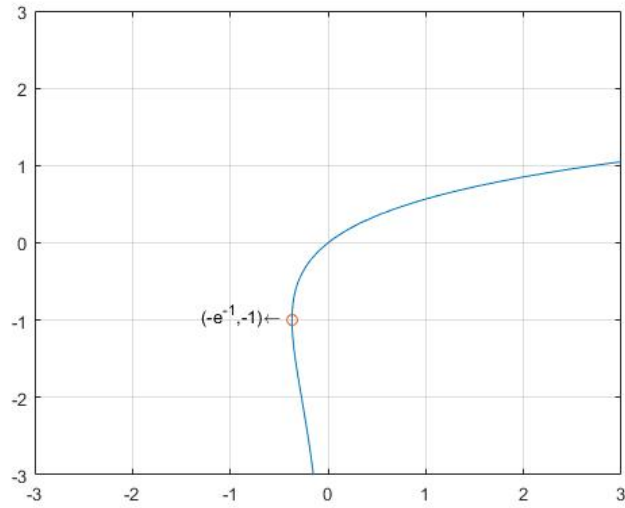


圖 2.1: Lambert W function 是 $f(x) = xe^x$ 的反函數，因在 $[-e^{-1}, 0)$ 的範圍內有兩個值，故定義值域大於 -1 的部分為 branch 0，其餘的部分為 branch -1

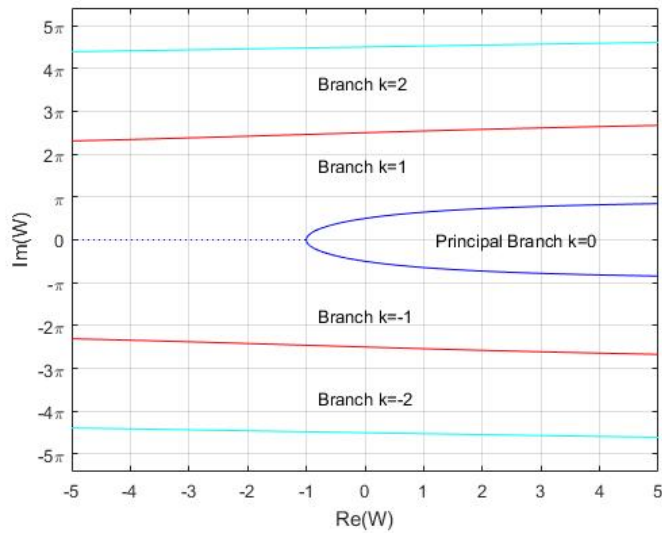


圖 2.2: 各個 branch 的 Lambert W function 的值域幾乎沒有交集，有交集的點如 $W_0(-e^{-1}) = W_{-1}(-e^{-1}) = -1$ 。

第 3 章

線性時滯微分方程求解與分析

令 $h > 0, a, a_1, \dots, a_M, x_0 \in \mathbb{R}$ ，考慮線性時滯微分方程

$$\begin{aligned} \dot{x}(t) &= ax(t) + \sum_{r=1}^M a_r x(t-rh) + u(t), & t > 0 \\ x(0) &= x_0 \\ x(t) &= \phi(t), & t \in [-Mh, 0) \end{aligned} \quad (3.1)$$

其中 u 為系統輸入， ϕ 為前導函數，在 $[-Mh, 0]$ 連續，該系統在 $t = 0$ 時不必連續，即 $\phi(0)$ 不必等於 x_0 。系統穩定的條件為系統特徵方程的根實部皆小於 0，也就是說，特徵方程的根皆在複數平面的左半平面。

3.1 單一時滯微分方程

令 $a_2 = a_3 = \dots = a_M = 0$ ，則 (3.1) 為單一時滯微分方程，因前導條件不變，只需考慮方程

$$\dot{x}(t) = ax(t) + a_1 x(t-h) + u(t) \quad (3.2)$$

假設輸入為 0，因此有特徵方程為

$$s - a - a_1 e^{-sh} = 0 \quad (3.3)$$

經整理後得到

$$(s-a)he^{(s-a)h} = a_1 h e^{-ah} \quad (3.4)$$

因上式剛好滿足 Lambert W function 的關係式，也就是 $(s-a)h = W(a_1 h e^{-ah})$ ，且因為 Lambert W function 有可數個 branch，因此上式有解

$$S_k = \frac{1}{h} W_k(a_1 h e^{-ah}) + a, \quad \forall k \in \mathbb{Z}$$

已知 Lambert W function 的值域除各 branch 邊界外皆互斥，因此若 $W_k(a_1 h e^{-ah})$ 的值不位在該 branch 的邊界，則各項 S_k 皆相異，即不存在重根問題，因此該齊次方程的解析解可表示成可數個指數函數的線性組合

$$x(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k$$

其中各項常數 C_k 由前導條件 $x(0) = x_0, x(t) = \phi(t)$ 對 $t \in [-h, 0)$ 決定 [7, 8]。

3.1.1 狀態轉移函數

定義當 $t = 0$ 時， $\delta(t) = 1$ ，反之， $\delta(t) = 0$ ，因此給定下面定義。

定義 3.1. 當函數 Ψ 滿足

$$\begin{aligned} \frac{d}{dt}\Psi(t) &= a\Psi(t) + a_1\Psi(t-h), \quad t > 0 \\ \Psi(t) &= \delta(t), \quad t \leq 0 \end{aligned}$$

則稱 Ψ 為系統 (3.2) 的狀態轉移函數。

令 $X(s) = \mathcal{L}[x(t); s], U(s) = \mathcal{L}[u(t); s]$, 對 (3.2) 進行拉式變換

$$sX(s) - x_0 = aX(s) + a_1e^{-sh}X(s) + a_1e^{-sh}\Phi(s) + U(s) \quad (3.5)$$

其中 Φ 被定義為 $\Phi(s) = \int_{-h}^0 \phi(t)e^{-st}dt$ 。將上式整理後得到

$$(s - a - a_1e^{-sh})X(s) = x_0 + a_1e^{-sh}\Phi(s) + U(s) \quad (3.6)$$

假設存在 Ψ 滿足

$$\mathcal{L}[\Psi(t); s] = \frac{1}{s - a - a_1e^{-sh}} \quad (3.7)$$

並限制條件 $\Psi(t) = \delta(t)$ 對所有 $t \leq 0$, 則

$$\begin{aligned} (s\mathcal{L}[\Psi(t); s] - \Psi(0)) - a\mathcal{L}[\Psi(t); s] - a_1e^{-sh}\mathcal{L}[\Psi(t); s] \\ = (s - a - a_1e^{-sh})\mathcal{L}[\Psi(t); s] - 1 \\ = 0 \end{aligned}$$

因此

$$s\mathcal{L}[\Psi(t); s] - \Psi(0) = a\mathcal{L}[\Psi(t); s] + a_1e^{-sh}\mathcal{L}[\Psi(t); s] \quad (3.8)$$

若將上式拉回時域上, 當 $t > 0$

$$\frac{d}{dt}\Psi(t) = a\Psi(t) + a_1\Psi(t-h) \quad (3.9)$$

且在 $t \leq 0$ 時滿足 $\Psi(t) = \delta(t)$, 因此 Ψ 為 (3.2) 的狀態轉移函數。

3.1.2 時域響應

假設存在 Ψ 滿足 (3.7) 且令 $\bar{\Psi}(s)$ 為 $\Psi(t)$ 的拉式變換, 因此從 (3.6) 得到

$$X(s) = \bar{\Psi}(s)x_0 + \bar{\Psi}(s)a_1e^{-sh}\Phi(s) + \bar{\Psi}(s)U(s) \quad (3.10)$$

觀察 (3.10) 的第二項, 擴增前導函數 ϕ 的定義域並定義

$$\phi_e(t) = \begin{cases} \phi(t), & t \in [-h, 0] \\ 0, & \text{otherwise} \end{cases}$$

考慮函數 $\int_0^t \Psi(t-\tau)a_1\phi_e(\tau-h)d\tau$ 的拉式變換

$$\mathcal{L}\left[\int_0^t \Psi(t-\tau)a_1\phi_e(\tau-h)d\tau; s\right] = \int_0^\infty \int_0^t \Psi(t-\tau)a_1\phi_e(\tau-h)e^{-st}d\tau dt$$

若對調其積分的順序, 得到

$$\begin{aligned} \int_{\tau=0}^\infty \int_{t=\tau}^\infty \Psi(t-\tau)a_1\phi_e(\tau-h)e^{-st}dtd\tau &= \int_{\tau=0}^\infty \left[\int_{t=\tau}^\infty \Psi(t-\tau)e^{-st}dt \right] a_1\phi_e(\tau-h)d\tau \\ &= \bar{\Psi}(s) \int_{\tau=0}^\infty e^{-s\tau}a_1\phi_e(\tau-h)d\tau \\ &= \bar{\Psi}(s)a_1e^{-sh} \int_{-h}^\infty e^{-s\tau}\phi_e(\tau)d\tau \end{aligned}$$

因 ϕ_e 在 $[-h, 0]$ 以外的地方為 0，因此 $\Phi(s) = \int_{-h}^0 \phi(\tau)e^{-s\tau}d\tau = \int_{-h}^{\infty} \phi_e(\tau)e^{-s\tau}d\tau$ ，最後得到

$$\mathcal{L}\left[\int_0^t \Psi(t-\tau)a_1\phi_e(\tau-h)d\tau; s\right] = \bar{\Psi}(s)a_1e^{-sh}\Phi(s)$$

又因 $\int_0^t \Psi(t-\tau)a_1\phi_e(\tau-h)d\tau = \int_0^{\min\{0, t-h\}} \Psi(t-\tau-h)a_1\phi(\tau)d\tau$ ，因此知道

$$\mathcal{L}\left[\int_0^{\min\{0, t-h\}} \Psi(t-\tau-h)a_1\phi(\tau)d\tau; s\right] = \bar{\Psi}(s)a_1e^{-sh}\Phi(s)$$

綜上所述，系統 (3.2) 在時域上的響應為

$$x(t) = \Psi(t)x_0 + \int_0^{\min\{0, t-h\}} \Psi(t-\tau-h)a_1\phi(\tau)d\tau + \int_0^t \Psi(t-\tau)u(\tau)d\tau$$

其中 Ψ 為 (3.2) 的狀態轉移函數。

定理 3.1. 令 Ψ 為 (3.2) 的狀態轉移函數，則 (3.2) 的時域響應為

$$x(t) = \Psi(t)x_0 + \int_0^{\min\{0, t-h\}} \Psi(t-\tau-h)a_1\phi(\tau)d\tau + \int_0^t \Psi(t-\tau)u(\tau)d\tau$$

3.1.3 以 Lambert W Function 求系統近似解

回到 (3.10)，(3.2) 在頻域上的響應為

$$X(s) = \bar{\Psi}(s)x_0 + \bar{\Psi}(s)a_1e^{-sh}\Phi(s) + \bar{\Psi}(s)U(s)$$

其中

$$\bar{\Psi}(s) = \frac{1}{s - a - a_1e^{-sh}}$$

為 (3.2) 狀態轉移函數的拉式變換。

引理 3.2. 令 $h > 0, a, a_1 \in \mathbb{R}$ 對任意 $s \in \mathbb{C}$

$$\frac{1}{s - a - a_1e^{-sh}} = \sum_{k=-\infty}^{\infty} \frac{C_k}{s - S_k}, \quad C_k = \frac{1}{1 + a_1he^{-S_k h}}$$

其中

$$S_k = \frac{1}{h}W_k(a_1he^{-ah}) + a$$

$\forall k \in \mathbb{Z}$ 。

Proof. 已知對所有 $k \in \mathbb{Z}$ ， S_k 滿足

$$S_k - a - a_1e^{-S_k h} = 0$$

且當 $k \neq \ell$ 時

$$S_k \neq S_\ell$$

因此對任意 $\ell \in \mathbb{Z}$ ，我們有

$$\lim_{s \rightarrow S_\ell} \sum_{k=-\infty}^{\infty} \frac{s - a - a_1e^{-sh}}{s - S_k} = \lim_{s \rightarrow S_\ell} \frac{s - a - a_1e^{-sh}}{s - S_k} = 1 + a_1he^{-S_\ell h}$$

取任意 $\ell \in \mathbb{Z}$ ，令 $s \rightarrow S_\ell$

$$\begin{aligned} \lim_{s \rightarrow S_\ell} (s - a - a_1e^{-sh}) \sum_{k=-\infty}^{\infty} \frac{C_k}{s - S_k} &= \left(\lim_{s \rightarrow S_\ell} \sum_{k=-\infty}^{\infty} \frac{s - a - a_1e^{-sh}}{s - S_k} \right) \frac{1}{1 + a_1he^{-S_\ell h}} \\ &= \left(\lim_{s \rightarrow S_\ell} \frac{s - a - a_1e^{-sh}}{s - S_k} \right) \frac{1}{1 + a_1he^{-S_\ell h}} \\ &= 1 \end{aligned}$$

令 $f(s) = s - a - a_1 e^{-sh}$ ，因對任意 $S_\ell, \ell \in \mathbb{Z}$ 皆為 f 的單根，因此 $[f(s)]^{-1}$ 在 $s = S_\ell$ 的 Laurent 展開為

$$\frac{1}{s - a - a_1 e^{-sh}} = \frac{a_{-1}}{s - S_\ell} + a_0 + a_1(s - S_\ell) + a_2(s - S_\ell)^2 + \cdots$$

因此

$$a_{-1} = \lim_{s \rightarrow S_\ell} \frac{s - S_\ell}{s - a - a_1 e^{-sh}} = C_\ell$$

對任意 $k \neq \ell$

$$\begin{aligned} \frac{1}{S_k - s} &= \frac{1}{S_k - S_\ell - (s - S_\ell)} = \frac{1}{S_k - S_\ell} \frac{1}{1 - \frac{s - S_\ell}{S_k - S_\ell}} \\ &= \frac{1}{S_k - S_\ell} \left[1 + \frac{s - S_\ell}{S_k - S_\ell} + \frac{(s - S_\ell)^2}{(S_k - S_\ell)^2} + \cdots \right] \\ &= \sum_{j=0}^{\infty} \frac{1}{(S_k - S_\ell)^{j+1}} (s - S_\ell)^j \end{aligned}$$

我們有

$$\begin{aligned} \sum_{k=-\infty}^{\infty} \frac{C_k}{s - S_k} &= \frac{C_\ell}{s - S_\ell} + \sum_{\substack{k=-\infty \\ k \neq \ell}}^{\infty} C_k \frac{1}{s - S_k} \\ &= \frac{C_\ell}{s - S_\ell} + \sum_{\substack{k=-\infty \\ k \neq \ell}}^{\infty} (-C_k) \frac{1}{S_k - s} \\ &= \frac{C_\ell}{s - S_\ell} + \sum_{\substack{k=-\infty \\ k \neq \ell}}^{\infty} (-C_k) \sum_{j=0}^{\infty} \frac{1}{(S_k - S_\ell)^{j+1}} (s - S_\ell)^j \\ &= \frac{C_\ell}{s - S_\ell} + \sum_{j=0}^{\infty} \left[\sum_{\substack{k=-\infty \\ k \neq \ell}}^{\infty} \frac{-C_k}{(S_k - S_\ell)^{j+1}} \right] (s - S_\ell)^j \end{aligned}$$

這是 $\sum_{k=-\infty}^{\infty} \frac{C_k}{s - S_k}$ 在 $s = S_\ell$ 的 Laurent 級數，因 Laurent 級數在同一點的展開是惟一的，因此

$$\frac{1}{s - a - a_1 e^{-sh}} = \sum_{k=-\infty}^{\infty} \frac{C_k}{s - S_k}, \quad C_k = \frac{1}{1 + a_1 h e^{-S_k h}}$$

對任意 $s \neq S_k, \forall k \in \mathbb{Z}$ 皆成立。

□

由引理 3.2 得 (3.2) 的狀態轉移函數為

$$\Psi(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k \quad (3.11)$$

其中 S_k 與 C_k 如引理 3.2 所述。考慮 (3.10) 的第二項，假設存在 $\{C_k^I \in \mathbb{C} \mid \forall k \in \mathbb{Z}\}$ 滿足

$$\bar{\Psi}(s) a_1 e^{-sh} \Phi(s) = \sum_{k=-\infty}^{\infty} \frac{C_k^I}{s - S_k} \quad (3.12)$$

因為

$$C_k^I = \lim_{s \rightarrow S_k} \frac{(s - S_k) a_1 e^{-sh} \Phi(s)}{s - a - a_1 e^{-sh}} = \frac{a_1 e^{-S_k h} \Phi(S_k)}{1 + a_1 h e^{-S_k h}}$$

為常數，因此若將 (3.12) 轉移至時域上，得到

$$\sum_{k=-\infty}^{\infty} e^{S_k t} C_k^I \quad (3.13)$$

若用 Lambert W function 擬合 (3.2) 的時域響應會得到

$$x(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-\infty}^{\infty} e^{S_k(t-\tau)} C_k b u(\tau) d\tau \quad (3.14)$$

其中係數 C_k, C_k^I 如上所述。

在實際運算上，取足夠大的 N 使得

$$x(t) \approx \sum_{k=-N}^N e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-N}^N e^{S_k(t-\tau)} C_k b u(\tau) d\tau$$

因 Lambert W function 在特定條件下具有共軛性，此共軛性質連帶影響到係數 C_k 與 C_k^I ，因此只須計算 $N/2$ 項或 $(N-1)/2$ 項就能推測全部 N 項所造成的結果。

3.1.4 範例說明

考慮系統

$$\begin{aligned} \dot{x}(t) &= -x(t) - x(t-1) + \cos t, & t > 0 \\ x(0) &= 1 & t \in [-1, 0] \end{aligned}$$

觀察 S_0

$$S_0 = W_0(-e) - 1 \approx -0.6050 + 1.7882i$$

因其他 branch 的 S_k 實部都比 S_0 小，該系統穩定，取 branch $k = -3, -2, -1, 0, 1, 2$

$$\begin{aligned} S_{0,-1} &= -0.6050 \pm 1.7882i & C_{0,-1} &= 0.2712 \mp 0.3477i \\ S_{1,-2} &= -2.0528 \pm 7.7184i & C_{1,-2} &= -0.0009 \mp 0.1296i \\ S_{2,-3} &= -2.6474 \pm 14.0202i & C_{2,-3} &= -0.0033 \mp 0.0712i \end{aligned}$$

因

$$C_k^I = \frac{a_1 e^{-S_k h} \Phi(S_k)}{1 + a_1 h e^{-S_k h}} = (1 - C_k) \Phi(S_k)$$

系統響應的近似值為

$$x(t) \approx \sum_{k=-3}^2 e^{S_k t} (C_k^N x_0 + C_k^{IN}) + \int_0^t \sum_{k=-3}^2 e^{S_k(t-\tau)} C_k^N b u(\tau) d\tau$$

其中 $\Phi(s) = \int_{-1}^0 e^{-st} dt$ ，實際計算的結果在圖 3.1。

3.2 兩項延遲時滯微分方程

兩項延遲時滯方程的特徵方程為

$$s - a - a_1 e^{-sh} - a_2 e^{-2sh} = 0 \quad (3.15)$$

該方程無法以 Lambert W function 直接求解，故將其改寫為

$$(s - a) h e^{(s-a)h} = a_1 h e^{-ah} + a_2 h e^{-ah} e^{-sh} \quad (3.16)$$

若要求上式的根，以迭代式

$$\begin{aligned} S_k^{(n)} &= \frac{1}{h} W_k(a_1 h e^{-ah} + a_2 h e^{-ah} e^{-S_k^{(n-1)} h}) + a \\ S_k^{(0)} &= \frac{1}{h} W_k(a_1 h e^{-ah}) + a \end{aligned}$$

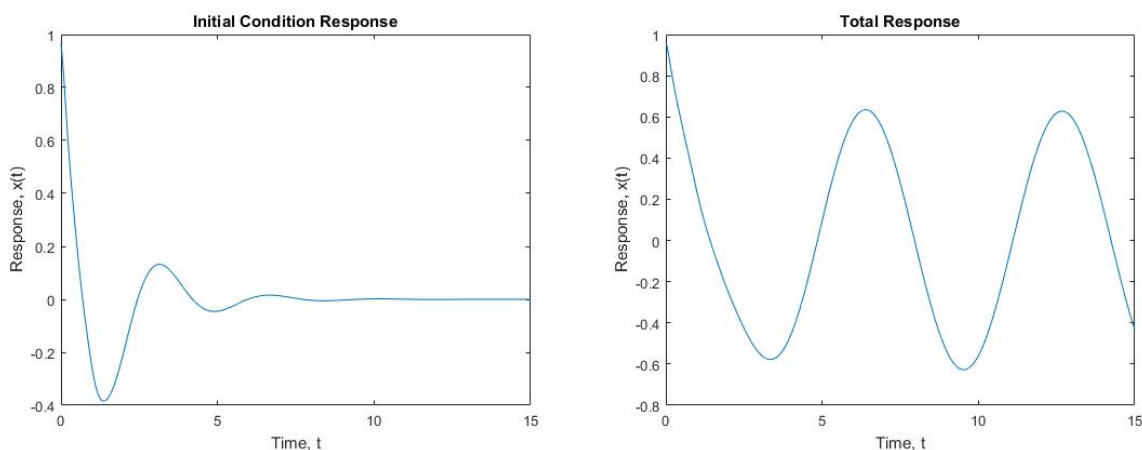


圖 3.1: 章節 3.1.4 結果

對所有 $n \in \mathbb{N}$ 。若這可數個根彼此相異 (或彼此在複數平面上的距離「足夠」遠)，則該系統的齊次方程有解

$$x(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k$$

其中 C_k 為常數，被前導條件 $x(0) = x_0$ ， $x(t) = \phi(t)$ 對 $t \in [-2h, 0)$ 決定。上述迭代法的收斂條件為迭代過程中的各項 s 均滿足 $|F'(s)| < 1$ ，其中

$$F(s) = \frac{1}{h} W(a_1 h e^{-ah} + a_2 h e^{-ah} e^{-sh}) + a$$

且 W 為任意 branch 的 Lambert W function，因 W 的微分的特殊性質，因此 $|F'(s)| < 1$ 可改寫為

$$\left| \frac{a_2 h e^{-ah} e^{-sh}}{a_1 h e^{-ah} + a_2 h e^{-ah} e^{-sh}} \frac{W(a_1 h e^{-ah} + a_2 h e^{-ah} e^{-sh})}{1 + W(a_1 h e^{-ah} + a_2 h e^{-ah} e^{-sh})} \right| < 1$$

3.2.1 狀態轉移函數

考慮兩項延遲的線性時滯方程

$$\dot{x}(t) = ax(t) + a_1 x(t-h) + a_2 x(t-2h) + u(t) \quad (3.17)$$

且系統的前導條件不變。

定義 3.2. 當函數 Ψ 滿足

$$\begin{aligned} \frac{d}{dt} \Psi(t) &= a\Psi(t) + a_1 \Psi(t-h) + a_2 \Psi(t-2h), & t > 0 \\ \Psi(t) &= \delta(t), & t \leq 0 \end{aligned}$$

則稱 Ψ 為系統 (3.17) 的狀態轉移函數。

對 (3.17) 進行拉式變換

$$sX(s) - x_0 = aX(s) + a_1 e^{-sh} X(s) + a_2 e^{-2sh} X(s) + a_1 e^{-sh} \Phi_1(s) + a_2 e^{-2sh} \Phi_2(s) + U(s) \quad (3.18)$$

其中定義為 $\Phi_1(s) = \int_{-h}^0 \phi(t) e^{-st} dt$ 且 $\Phi_2(s) = \int_{-2h}^0 \phi(t) e^{-st} dt$ 。將上式整理後得到

$$(s - a - a_1 e^{-sh} - a_2 e^{-2sh}) X(s) = x_0 + a_1 e^{-sh} \Phi_1(s) + a_2 e^{-2sh} \Phi_2(s) + U(s) \quad (3.19)$$

假設存在函數 Ψ 滿足

$$\mathcal{L}[\Psi(t); s] = \frac{1}{s - a - a_1 e^{-sh} - a_2 e^{-2sh}} \quad (3.20)$$

並設置前導條件 $\Psi(t) = \delta(t)$ ，對所有 $t \leq 0$ ，則

$$\begin{aligned} & (s\mathcal{L}[\Psi(t); s] - \Psi(0)) - a\mathcal{L}[\Psi(t); s] - a_1 e^{-sh}\mathcal{L}[\Psi(t); s] - a_2 e^{-2sh}\mathcal{L}[\Psi(t); s] \\ &= (s - a - a_1 e^{-sh} - a_2 e^{-2sh})\mathcal{L}[\Psi(t); s] - 1 \\ &= 0 \end{aligned}$$

換句話說

$$s\mathcal{L}[\Psi(t); s] - \Psi(0) = a\mathcal{L}[\Psi(t); s] + a_1 e^{-sh}\mathcal{L}[\Psi(t); s] + a_2 e^{-2sh}\mathcal{L}[\Psi(t); s] \quad (3.21)$$

若將上式拉回時域上，當 $t > 0$

$$\frac{d}{dt}\Psi(t) = a\Psi(t) + \sum_{r=1}^M a_r \Psi(t - rh) \quad (3.22)$$

且在 $t \leq 0$ 時滿足 $\Psi(t) = \delta(t)$ ，因此 Ψ 為 (3.17) 的狀態轉移函數。

3.2.2 以 Lambert W Function 求系統近似解

令 $\bar{\Psi}(s)$ 為 (3.17) 狀態轉移函數 $\Psi(t)$ 的拉式變換，因此 (3.19) 可改寫為

$$X(s) = \bar{\Psi}(s)x_0 + \bar{\Psi}(s)(a_1 e^{-sh}\Phi_1(s) + a_2 e^{-2sh}\Phi_2(s)) + \bar{\Psi}(s)U(s) \quad (3.23)$$

其中 $\Phi_1(s) = \int_{-h}^0 \phi(t)e^{-st}dt$ 且 $\Phi_2(s) = \int_{-2h}^0 \phi(t)e^{-st}dt$ 。已知該系統的特徵方程有可數個根，即 $\{S_k \in \mathbb{C} \mid \forall k \in \mathbb{Z}\}$ ，因此

$$\bar{\Psi}(s) = \prod_{k=-\infty}^{\infty} \frac{C_k}{s - S_k} \quad (3.24)$$

因此有

$$\Psi(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k \quad (3.25)$$

其中

$$C_k = \lim_{s \rightarrow S_k} \frac{s - S_k}{s - a - a_1 e^{-sh} - a_2 e^{-2sh}} = \frac{1}{1 + a_1 h e^{-S_k h} + 2a_2 h e^{-2S_k h}}$$

考慮 (3.23) 的第二項，假設存在 $C_k^I \in \mathbb{C} \mid \forall k \in \mathbb{Z}$ 滿足

$$\bar{\Psi}(s)(a_1 e^{-sh}\Phi_1(s) + a_2 e^{-2sh}\Phi_2(s)) = \sum_{k=-\infty}^{\infty} \frac{C_k^I}{s - S_k}$$

其中

$$C_k^I = \frac{a_1 e^{-S_k h}\Phi_1(S_k) + a_2 e^{-2S_k h}\Phi_2(S_k)}{1 + a_1 h e^{-S_k h} + 2a_2 h e^{-2S_k h}}$$

最後得到 (3.23) 的時域響應為

$$x(t) = \sum_{k=-\infty}^{\infty} (C_k x_0 + C_k^I) e^{S_k t} + \int_0^t \sum_{k=-\infty}^{\infty} e^{S_k(t-\tau)} C_k u(\tau) d\tau \quad (3.26)$$

其中係數 C_k 與 C_k^I 如上所述。在實際運算上，取足夠大的 N 使得

$$x(t) \approx \sum_{k=-N}^N e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-N}^N e^{S_k(t-\tau)} C_k u(\tau) d\tau$$

因 Lambert W function 在特定條件下具有共軛性，此共軛性質回連帶影響到係數 C_k 與 C_k^I ，因此只須計算 $N/2$ 項或 $(N-1)/2$ 項就能推測全部 N 項所造成的結果。

3.2.3 範例說明

考慮兩項延遲系統

$$\begin{aligned} \dot{x}(t) &= -x(t) - x(t-1) - 0.5x(t-2) + \cos t, & t > 0 \\ x(t) &= 1, & t \in [-2, 0] \end{aligned}$$

$\forall n \in \mathbb{N}$, Lambert W function 的任意 branch k , S_k 的迭代式如下

$$\begin{aligned} S_k^{(n)} &= W_k([-1 - 0.5e^{-S_k^{(n-1)}}]e) - 1 \\ S_k^{(0)} &= W_k(-e) - 1 \end{aligned}$$

已知 S_k , 因此可求 C_k 與 C_k^I

$$\begin{aligned} C_k &= \frac{1}{1 - e^{-S_k} - e^{-2S_k}} \\ C_k^I &= \frac{-e^{-S_k}\Phi_1(S_k) - 0.5e^{-2S_k}\Phi_2(S_k)}{1 - e^{-S_k} - e^{-2S_k}} \end{aligned}$$

其中 $\Phi_1(s) = \int_{-1}^0 e^{-st} dt$, $\Phi_2(s) = \int_{-2}^0 e^{-st} dt$. 取 $N = 5$, 我們有

$$\begin{aligned} S_0 &= -0.2750 + 1.4752i & C_0 &= 0.2763 - 0.1759i & C_0^I &= 0.0961 + 0.5746i \\ S_{\pm 1} &= -1.1468 \pm 7.2401i & C_{\pm 1} &= 0.0170 \mp 0.0804i & C_{\pm 1}^I &= 0.0110 \pm 0.0818i \\ S_{\pm 2} &= -1.5075 \pm 13.4657i & C_{\pm 2} &= 0.0050 \mp 0.0421i & C_{\pm 2}^I &= 0.0028 \pm 0.0422i \\ S_{\pm 3} &= -1.7229 \pm 19.7226i & C_{\pm 3} &= 0.0025 \mp 0.0282i & C_{\pm 3}^I &= 0.0011 \pm 0.0282i \\ S_{\pm 4} &= -1.8759 \pm 25.9973i & C_{\pm 4} &= 0.0016 \mp 0.0211i & C_{\pm 4}^I &= 0.0005 \pm 0.0212i \\ S_{\pm 5} &= -1.9944 \pm 32.2723i & C_{\pm 5} &= 0.0011 \mp 0.0169i & C_{\pm 5}^I &= 0.0002 \pm 0.0169i \end{aligned}$$

系統響應的近似值為

$$x(t) \approx \sum_{k=-5}^5 e^{S_k t} (C_k^N x_0 + C_k^{IN}) + \int_0^t \sum_{k=-5}^5 e^{S_k(t-\tau)} C_k^N b u(\tau) d\tau$$

其中 $\Phi_1(s) = \int_{-1}^0 e^{-st} dt$ 且 $\Phi_2(s) = \int_{-2}^0 e^{-st} dt$, 實際計算的結果在圖 3.2。

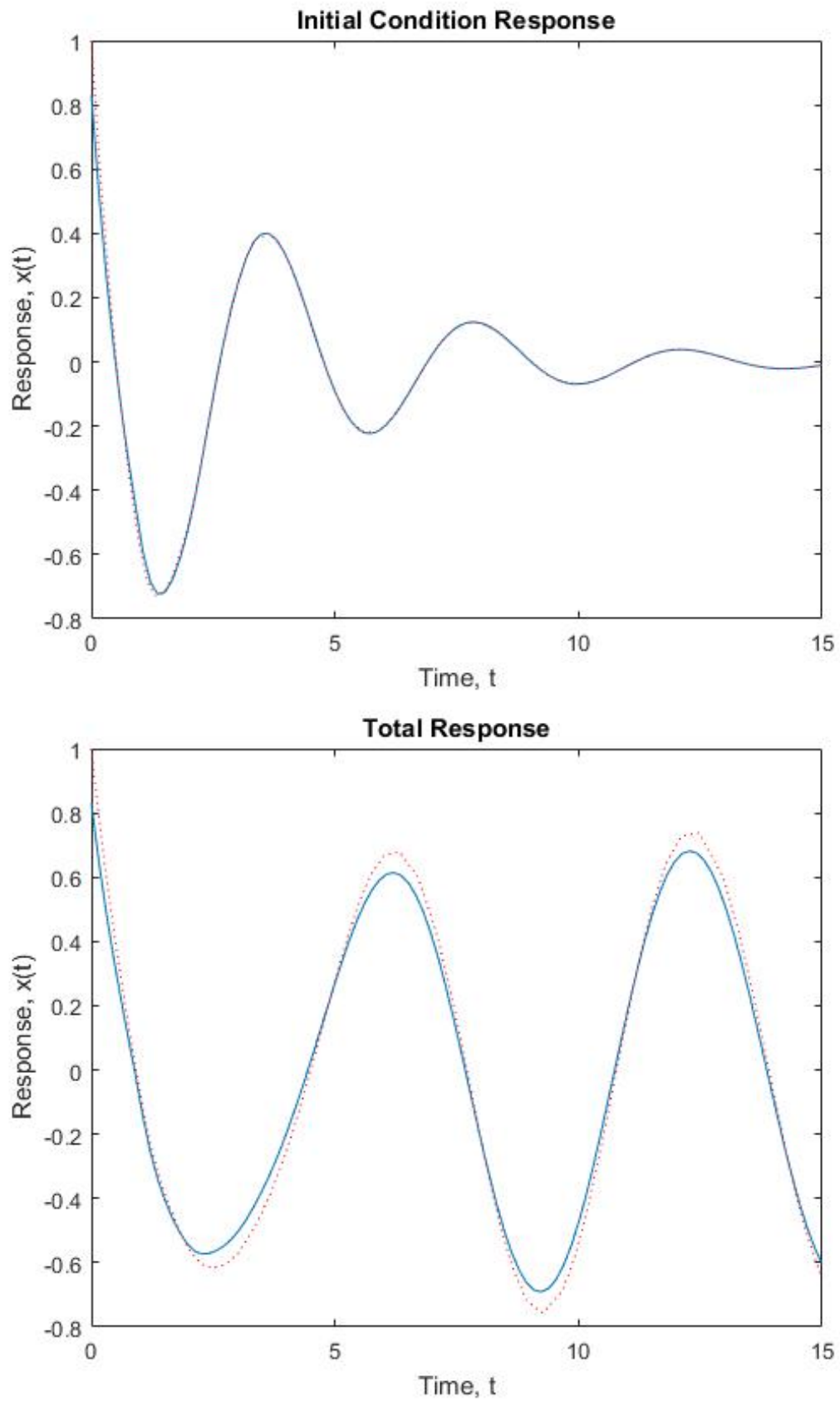


圖 3.2: 章節 3.2.3 結果，虛線部分為 Matlab 內建指令 `dde23` 得到的近似解

第 4 章

單一時滯微分系統

令 $h > 0, A, A_d \in M_n(\mathbb{R}), B \in M_{n \times p}(\mathbb{R}), x_0 \in \mathbb{R}^n$ ，給定單一時滯微分系統

$$\begin{aligned} \dot{x}(t) &= Ax(t) + A_dx(t-h) + Bu(t), & t > 0 \\ x(0) &= x_0 \\ x(t) &= \phi(t), & t \in [-h, 0) \end{aligned} \quad (4.1)$$

其中 $\phi \in C([-h, 0], \mathbb{R}^n)$ 為前導函數, $u \in C((0, \infty), \mathbb{R}^p)$ 為系統輸入, 此系統在 $t = 0$ 時不必連續, 換句話說, $\phi(0)$ 不必等於 x_0 。系統穩定的條件為系統特徵值實部均小於 0, 即系統特徵值位於複數平面上的左半平面。

4.1 矩陣方程

令 (4.1) 的輸入為零, 則其特徵方程為

$$S - A - A_de^{-Sh} = O \quad (4.2)$$

其中 O 為 $n \times n$ 零矩陣。上式等價於

$$(S - A)he^{Sh} = A_dh \quad (4.3)$$

Yi 在其附錄中提到 [9], 若 A 與 A_d 可交換, 則

$$(S - A)he^{(S-A)h} = (S - A)he^{Sh}e^{-Ah} = A_dhe^{-Ah}$$

因此得到

$$S = \frac{1}{h}W(A_dhe^{-Ah}) + A$$

但一般情況下 A 與 A_d 不可交換, 因此假設存在 $Q \in M_n(\mathbb{C})$ 使得

$$(S - A)he^{(S-A)h} = A_dhQ$$

因矩陣型 Lambert W function 也有可數個 branch, 因此有可數個 S_k 滿足上式並有下列關係

$$S_k = \frac{1}{h}W_k(A_dhQ_k) + A \quad (4.4)$$

其中由 Q_k 決定 S_k , 對所有 $k \in \mathbb{Z}$ 。若將 (4.4) 帶入 (4.3), 得到

$$W_k(A_dhQ_k)e^{A_dhQ_k + Ah} = A_dh \quad (4.5)$$

顯然當 A, A_d 可交換時, 每項 Q_k 皆可以是 e^{-Ah} 。若 A, A_d 不可交換, 令 $H_k = A_dhQ_k$, 解矩陣方程 4.5, 以 Matlab 指令 `fsolve`

```
% Initial Condition: A, Ad, Xini, k, h
F=@(k,H) lambertd(k,H)*expm(lambertd(k,H)+A*h) - Ad*h;
H_k=fsolve(@(H) F(k,H), Xini);
S_k=A+lambertd(k,H_k)/h;
```

其中

```
function y=lambertd(k,X)
% 只能用在可對角化矩陣
[V,D]=eig(X);
d=diag(D);
y=V*diag(lambertw(k,d))/V;
```

4.2 狀態轉移矩陣

已經知道各項 S_k ，透過矩陣型 Lambert W function 擬合系統的狀態轉移矩陣。

定義 4.1. 若函數 $\Psi: \mathbb{R} \rightarrow M_n(\mathbb{R})$ 滿足

$$\begin{aligned} \frac{d}{dt}\Psi(t) &= A\Psi(t) + A_d\Psi(t-h), & t > 0 \\ \Psi(t) &= I\delta(t), & t \leq 0 \end{aligned}$$

則稱 Ψ 為 (4.1) 的狀態轉移矩陣。

假設存在 Ψ 滿足

$$\mathcal{L}[\Psi(t); s] = (sI - A - A_d e^{-sh})^{-1} \quad (4.6)$$

並設置前導條件為 $\Psi(0) = I$ 且 $\Psi(t) = O$ ，對所有 $t < 0$ 。對 4.1 進行拉氏變換

$$\begin{aligned} (s\mathcal{L}[\Psi(t); s] - \Psi(0)) &= (A\mathcal{L}[\Psi(t); s] + A_d e^{-sh}\mathcal{L}[\Psi(t); s]) \\ &= (sI - A - A_d e^{-sh})\mathcal{L}[\Psi(t); s] - I = O \end{aligned}$$

因此得知 Ψ 是 (4.1) 的狀態轉移矩陣，同第三章的推論，系統的時域響應為

$$x(t) = \Psi(t)x_0 + \int_0^{\min\{0, t-h\}} \Psi(t-\tau-h)A_d\phi(\tau)d\tau + \int_0^t \Psi(t-\tau)Bu(\tau)d\tau$$

已知各項 S_k ， Y_i 透過拉氏變換得到 [9]

$$(sI - A - A_d e^{-sh})^{-1} = \sum_{k=-\infty}^{\infty} (sI - S_k)^{-1} C_k \quad (4.7)$$

其中 $C_k \in M_n(\mathbb{C})$ 為常數矩陣， $\forall k \in \mathbb{Z}$ 。已經知道 $(sI - A - A_d e^{-sh})^{-1}$ 是 (4.1) 狀態轉移矩陣的拉氏變換，因 (4.1) 的狀態轉移矩陣 Ψ 可以用下列的方式擬合

$$\Psi(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k \quad (4.8)$$

考慮 $n \geq 2$ ，計算 C_k 。假設所有的 S_k 皆可對角化，因此

$$(sI - A - A_d e^{-sh})^{-1} = \frac{1}{\Upsilon(s)} \text{Adj}(sI - A - A_d e^{-sh})$$

其中 $\Upsilon(s) = \det(sI - A - A_d e^{-sh})$ ， $\text{Adj}(sI - A - A_d e^{-sh})$ 為 $sI - A - A_d e^{-sh}$ 的伴隨矩陣。令 S_k 的特徵值為 $\lambda_{k1}, \dots, \lambda_{kp}$ ，其中 $p \leq n$ 。假設單位矩陣 I_r 的階數為特徵值 λ_{kr} 的重根數，因 S_k 可對角化，所以存在 $V_k \in M_n(\mathbb{C})$ 滿足

$$(sI - S_k)^{-1} = V_k \text{diag}\left(\frac{1}{s - \lambda_{k1}} I_1, \dots, \frac{1}{s - \lambda_{kp}} I_p\right) V_k^{-1}$$

已知

$$(sI - A - A_d e^{-sh})^{-1} = \sum_{k=-\infty}^{\infty} (sI - S_k)^{-1} C_k$$

恆成立，若將兩邊同乘 $s - \lambda_{kr}$ 並取極限，得到

$$\begin{aligned} V_k \text{diag}(\cdots, I_r, \cdots) V_k^{-1} C_k &= \lim_{s \rightarrow \lambda_{kr}} \frac{s - \lambda_{kr}}{\Upsilon(s)} \text{Adj}(sI - A - A_d e^{-sh}) \\ &= \frac{1}{\tilde{\Upsilon}(\lambda_{kr})} \text{Adj}(\lambda_{kr} I - A - A_d e^{-\lambda_{kr} h}) \end{aligned}$$

對所有 $r \in \{1, 2, \dots, p\}$ 將上式分別帶入 $r = 1, 2, \dots, p$ 並加總，得到

$$C_k = \sum_{r=1}^p \frac{1}{\tilde{\Upsilon}(\lambda_{kr})} \text{Adj}(\lambda_{kr} I - A - A_d e^{-\lambda_{kr} h})$$

4.3 系統響應

令 $\bar{\Psi}(s) = (sI - A - A_d e^{-sh})^{-1}$ ，同第三章的手法，系統的頻域響應為

$$X(s) = \bar{\Psi}(s)x_0 + \bar{\Psi}(s)A_d e^{-sh}\Phi(s) + \bar{\Psi}(s)BU(s) \quad (4.9)$$

其中 $\Phi(s) = \int_{-h}^0 \phi(t)e^{-st}dt$ 。考慮 (4.9) 的第二項，假設存在向量常數 $\{C_k^I \in \mathbb{C}^n \mid \forall k \in \mathbb{Z}\}$ 滿足

$$\bar{\Psi}(s)A_d e^{-sh}\Phi(s) = \sum_{k=-\infty}^{\infty} e^{S_k t} C_k^I \quad (4.10)$$

因此得到

$$C_k^I = \sum_{r=1}^p \frac{1}{\tilde{\Upsilon}(\lambda_{kr})} \text{Adj}(\lambda_{kr} I - A - A_d e^{-\lambda_{kr} h}) A_d e^{-\lambda_{kr} h} \Phi(\lambda_{kr})$$

與第三章的推論相同，最後得到的系統響應為

$$x(t) = \sum_{k=-\infty}^{\infty} e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-\infty}^{\infty} e^{S_k(t-\tau)} C_k B u(\tau) d\tau$$

在實際運算上，取足夠大的 N 使得

$$x(t) \approx \sum_{k=-N}^N e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-N}^N e^{S_k(t-\tau)} C_k B u(\tau) d\tau$$

因 Lambert W function 在特定條件下具有共軛性，此共軛性質回連帶影響到係數 C_k 與 C_k^I ，因此只須計算 $N/2$ 項或 $(N-1)/2$ 項就能推測全部 N 項所造成的結果。

4.4 範例說明

考慮系統 (4.1)，令 $h = 1$ 且

$$A = \begin{pmatrix} -1 & -3 \\ 2 & -5 \end{pmatrix}, \quad A_d = \begin{pmatrix} 1.66 & -0.697 \\ 0.93 & -0.330 \end{pmatrix},$$

其中前導條件為 $x_0 = (1, 0)^T$ ， $\phi(t) = (1, 0)^T$ 。取 $N = 3$ ，由 Matlab 指令 `fsolve` 解矩陣函數

$$F_k(X) = W_k(X) e^{W_k(X) + A_h} A_d h$$

得到

$$H_0 = \begin{pmatrix} 6.3796 & 19.1537 \\ 1.5916 & 11.1283 \end{pmatrix}$$

$$H_{\pm 1} = \begin{pmatrix} 11.3986 \mp 0.4665i & 9.3815 \mp 3.5768i \\ 2.6881 \mp 1.6938i & 5.3173 \mp 2.0199i \end{pmatrix}$$

$$H_{\pm 2} = \begin{pmatrix} 10.1386 \mp 0.5348i & 9.1198 \mp 1.5138i \\ 1.4792 \mp 1.0008i & 5.2210 \mp 0.8456i \end{pmatrix}$$

$$H_{\pm 3} = \begin{pmatrix} 9.9367 \mp 0.3573i & 9.1513 \mp 0.9567i \\ 1.3161 \mp 0.6475i & 5.2456 \mp 0.5344i \end{pmatrix}$$

其中 W_k 為 branch k 的矩陣型 Lambert W function。再將上述 H_k 代入矩陣型 Lambert W function 得到

$$S_0 = \begin{pmatrix} 0.3055 & -1.4150 \\ 2.1317 & -3.3015 \end{pmatrix}$$

$$S_{\pm 1} = \begin{pmatrix} -0.3499 \pm 4.9081i & -1.6253 \pm 0.1459i \\ 2.4175 \mp 0.1308i & -5.1048 \pm 4.5592i \end{pmatrix}$$

$$S_{\pm 2} = \begin{pmatrix} -1.1893 \pm 11.0140i & -1.6416 \mp 0.0549i \\ 2.2340 \mp 0.1191i & -5.9012 \pm 10.8791i \end{pmatrix}$$

$$S_{\pm 3} = \begin{pmatrix} -1.6528 \pm 17.2617i & -1.6392 \mp 0.0337i \\ 2.2017 \mp 0.0800i & -6.3426 \pm 17.1805i \end{pmatrix}$$

且已知常數矩陣 C_k, C_k^I 的算法

$$C_0 = \begin{pmatrix} 0.3424 & -0.0789 \\ -0.0563 & -0.4855 \end{pmatrix}$$

$$C_0^I = \begin{pmatrix} 0.6422 \\ 0.2725 \end{pmatrix}$$

$$C_{\pm 1} = \begin{pmatrix} 0.0767 \mp 0.1876i & 0.0142 \pm 0.0419i \\ 0.0239 \mp 0.0013i & 0.0196 \mp 0.1925i \end{pmatrix}$$

$$C_{\pm 1}^I = \begin{pmatrix} -0.0595 \pm 0.1886i \\ -0.0823 \pm 0.0598i \end{pmatrix}$$

$$C_{\pm 2} = \begin{pmatrix} 0.0073 \mp 0.0922i & 0.0102 \pm 0.0045i \\ 0.0026 \mp 0.0016i & 0.0005 \mp 0.0865i \end{pmatrix}$$

$$C_{\pm 2}^I = \begin{pmatrix} -0.0103 \pm 0.0922i \\ -0.0213 \pm 0.0110i \end{pmatrix}$$

$$C_{\pm 3} = \begin{pmatrix} 0.0012 \mp 0.0584i & 0.0045 \pm 0.0010i \\ 0.0007 \mp 0.0005i & -0.0012 \mp 0.0576i \end{pmatrix}$$

$$C_{\pm 3}^I = \begin{pmatrix} -0.0030 \pm 0.0526i \\ -0.0094 \pm 0.0036i \end{pmatrix}$$

此時系統零輸入響應為

$$x(t) \approx \sum_{k=-3}^3 e^{S_k t} (C_k x_0 + C_k^I)$$

若加入輸入 $Bu(t)$ ，則

$$x(t) \approx \sum_{k=-3}^3 e^{S_k t} (C_k x_0 + C_k^I) + \int_0^t \sum_{k=-3}^3 e^{S_k(t-\tau)} C_k B u(\tau) d\tau$$

其各項響應分別如圖 4.1、4.2 所示。

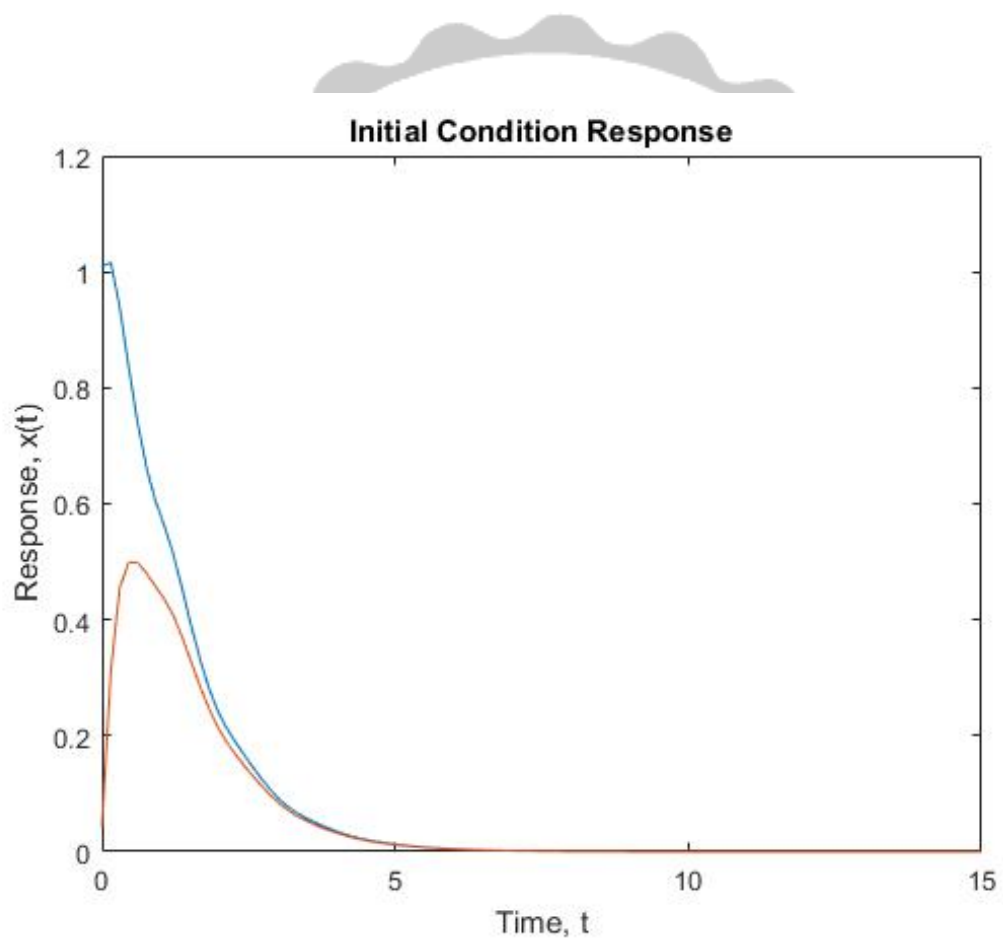


圖 4.1: 章節 4.4 響應，輸入為 0

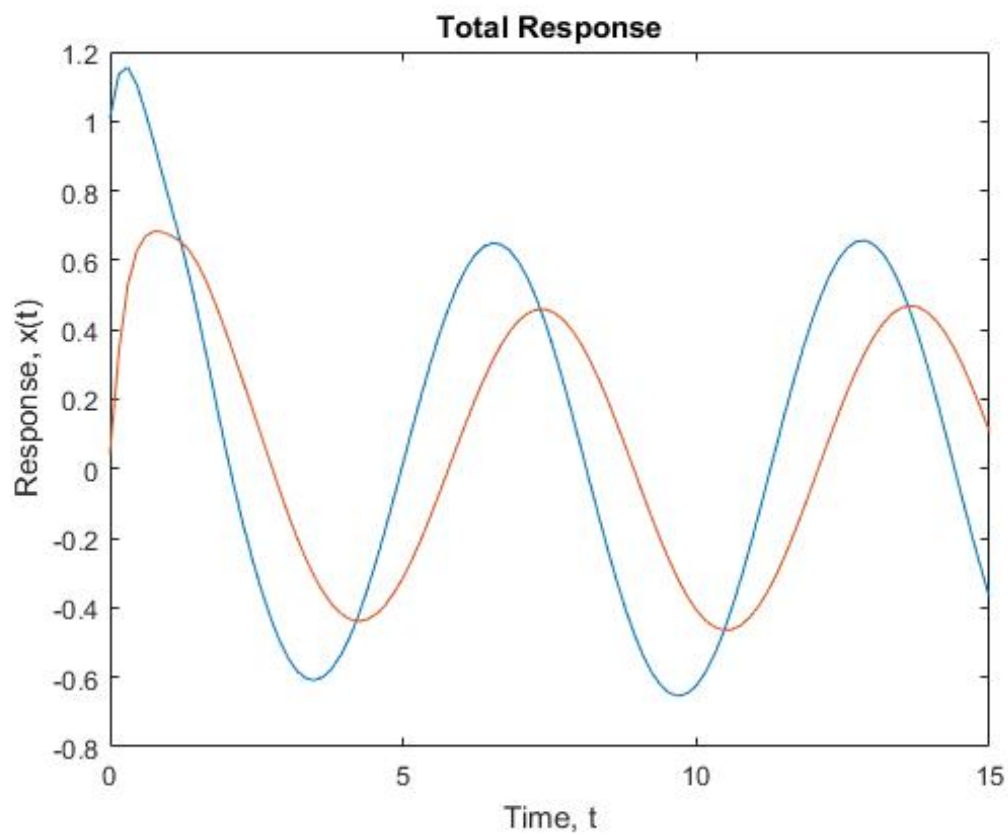
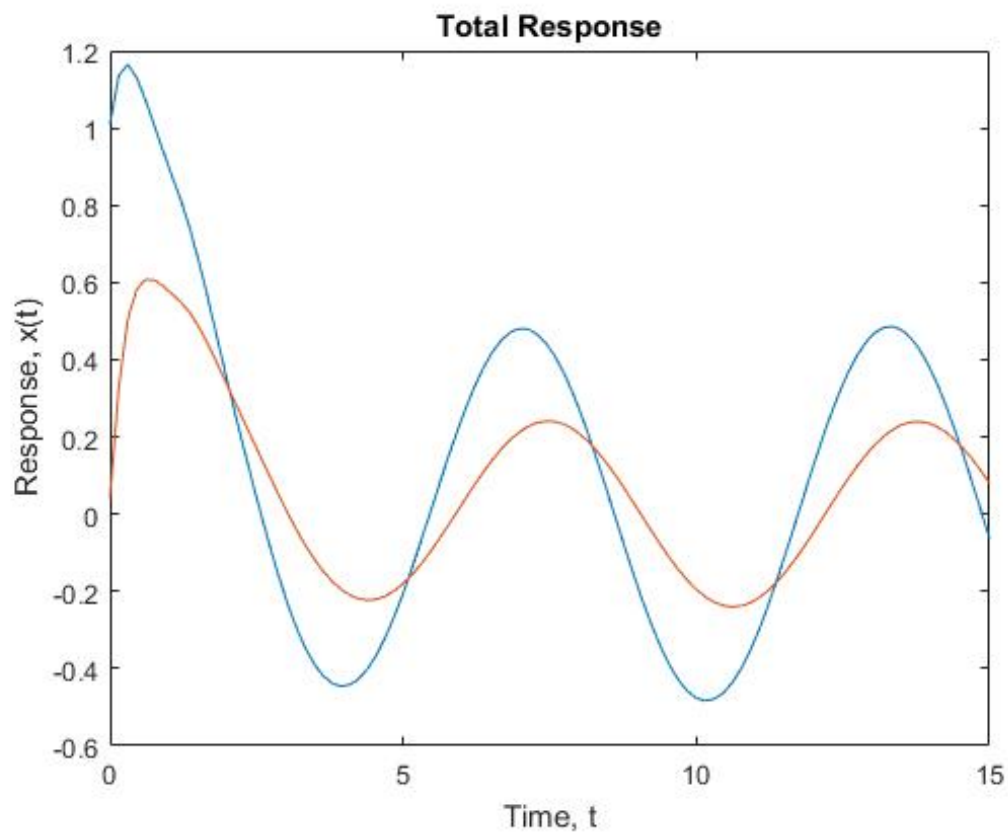


圖 4.2: 章節 4.4 響應，上圖輸入為 $Bu(t) = (\cos t, 0)^T$ ，下圖輸入為 $Bu(t) = (\cos t, \sin t)^T$

附錄 A

MATLAB 代碼

A.1 章節 3.1.4 代碼

```
clear; clc;
%-----%
h=1; a=-1; ad=-1; x0=1; tf=15;
u=@(t) cos(t);
%-----%
xfree=@(t,x,Z) a*x(1)+ad*Z(1,1);
xtol=@(t,x,Z) xfree(t,x,Z)+u(t);
hist=@(t) 1;
sol_1=dde23(xfree,1,hist,[0 tf]);
sol_2=dde23(xtol,1,hist,[0 tf]);

H=ad*h/exp(a*h);

s0=a+lambertw(H)/h;
s1=a+lambertw(1,H)/h;
s2=a+lambertw(2,H)/h;

c0=1/(1+ad*h/exp(s0*h));
c1=1/(1+ad*h/exp(s1*h));
c2=1/(1+ad*h/exp(s2*h));

t=linspace(0,tf,720);
p1= exp(s0*t)*c0 + exp(s1*t)*c1 + exp(s2*t)*c2;
p2= exp(s0*t)*(1-c0)*integral(@(t) exp(-s0*t),-h,0)...
    + exp(s1*t)*(1-c1)*integral(@(t) exp(-s1*t),-h,0)...
    + exp(s2*t)*(1-c2)*integral(@(t) exp(-s2*t),-h,0);
x= 2*real(p1)*x0 + 2*real(p2)/h;

plot(t,x), hold on;
% plot(sol_1.x,sol_1.y,'r:')
title('Initial Condition Response')
xlabel('Time, t')
ylabel('Response, x(t)')
```

```

z=[];
for i=1:length(t)
    temp=integral(@(xi) exp(s0*(t(i)-xi))*c0.*cos(xi),0,t(i))...
        +integral(@(xi) exp(s1*(t(i)-xi))*c1.*cos(xi),0,t(i))...
        +integral(@(xi) exp(s2*(t(i)-xi))*c2.*cos(xi),0,t(i));
    z=[z temp];
end
xx=x+2*real(z);

figure plot(t,xx), hold on
% plot(sol_2.x,sol_2.y,'r:')
title('Total Response')
xlabel('Time, t')
ylabel('Response, x(t)')

```

A.2 章節 3.2.3 代碼

```

clear; clc;
%-----%
h=1; a=-1; a1=-1; a2=-1/2; x0=1;
Phi_1=@(s) (exp(s)-1)/s;
Phi_2=@(s) (exp(2*s)-1)/s;
u=@(t) cos(t); phi=@(t) 1;
n=5; tf=15; err=1e-10;
%-----%
H=a1*h/exp(a*h);
t=linspace(0,tf);
xh=zeros(size(t));
xn=zeros(size(t));
% S=[]; C=[]; CI=[];
for k=0:n
    Sk_old=a+lambertw(k,H)/h;
    Serr=1;
    while Serr>err
        H1=H+a2*h/exp(a*h)/exp(Sk_old*h);
        Sk=a+lambertw(k,H1)/h;
        Serr=abs(Sk-Sk_old);
        Sk_old=Sk;
    end
    Sk=Sk_old;
    Ck=1/( 1 + a1*h/exp(Sk*h) + 2*a2*h/exp(2*Sk*h) );
    CIk=( a1/exp(Sk*h)*Phi_1(Sk) + a2/exp(2*Sk*h)*Phi_2(Sk) )*Ck;
% S=cat(1,S,Sk);
% C=cat(1,C,Ck);
% CI=cat(1,CI,CIk);
    xh=xh+ ( Ck*x0 + CIk )*exp(Sk*t);
    f=@(t,xi) exp(Sk*(t-xi))*Ck.*u(xi);
    temp=zeros(size(t));
    for j=1:length(t)

```

```

temp(j)=integral(@(xi) f(t(j),xi), 0,t(j));
end
xn=xn+temp;
end

```

```

fun_1=@(t,x,Z) a*x(1)+a1*Z(1,1)+a2*Z(1,2);
fun_2=@(t,x,Z) fun_1(t,x,Z)+u(t);
sol_1=dde23(fun_1,[1 2],phi,[0 tf]);
sol_2=dde23(fun_2,[1 2],phi,[0 tf]);

```

```

xh=2*real(xh);
plot(t,xh), hold on
plot(sol_1.x,sol_1.y,'r:')
title('Initial Condition Response')
xlabel('Time, t')
ylabel('Response, x(t)')

```

```

figure
xn=2*real(xn);
xx=xh+xn;
plot(t,xx), hold on
plot(sol_2.x,sol_2.y,'r:')
title('Total Response')
xlabel('Time, t')
ylabel('Response, x(t)')

```

A.3 章節 4.4 代碼

```

clear; clc;
%-----%
% h=1;
A=[-1 -3; 2 -5];
Ad=[1.66 -0.697; 0.93 -0.330];
x0=[1 0]';
%-----%

```

```

F=@(k,X) lambertd(k,X)*expm(lambertd(k,X)+A) - Ad;

```

```

Xini=-A/expm(A);
H0=fsolve(@(X) F(0,X),Xini);
H1=fsolve(@(X) F(1,X),Xini);
H2=fsolve(@(X) F(2,X),Xini);
H3=fsolve(@(X) F(3,X),Xini);

```

```

S0=A+lambertd(0,H0);
S1=A+lambertd(1,H1);
S2=A+lambertd(2,H2);
S3=A+lambertd(3,H3);

```

```

%-----%
Adj=@(s) [s-A(2,2)-Ad(2,2)/exp(s) A(1,2)+Ad(1,2)/exp(s);
A(2,1)+Ad(2,1)/exp(s) s-A(1,1)-Ad(1,1)/exp(s) ];
Us=@(s) det(s*eye(2)-A-Ad/exp(s));
Phi=@(s) [ (exp(s)-1)/s; 0 ];
tf=15; ep=1e-4; phi=@(t) [1 0]';
%-----%

s0=eig(S0); s01=s0(1); s02=s0(2);
s1=eig(S1); s11=s1(1); s12=s1(2);
s2=eig(S2); s21=s2(1); s22=s2(2);
s3=eig(S3); s31=s3(1); s32=s3(2);

c01=(Us(s01+ep)-Us(s01))/ep;    c02=(Us(s02+ep)-Us(s02))/ep;
c11=(Us(s11+ep)-Us(s11))/ep;    c12=(Us(s12+ep)-Us(s12))/ep;
c21=(Us(s21+ep)-Us(s21))/ep;    c22=(Us(s22+ep)-Us(s22))/ep;
c31=(Us(s31+ep)-Us(s31))/ep;    c32=(Us(s32+ep)-Us(s32))/ep;

C0=Adj(s01)/c01 + Adj(s02)/c02;
C1=Adj(s11)/c11 + Adj(s12)/c12;
C2=Adj(s21)/c21 + Adj(s22)/c22;
C3=Adj(s31)/c31 + Adj(s32)/c32;

CI0=Adj(s01)/c01*Ad/exp(s01)*Phi(s01) + Adj(s02)/c02*Ad/exp(s02)*Phi(s02);
CI1=Adj(s11)/c11*Ad/exp(s11)*Phi(s11) + Adj(s12)/c12*Ad/exp(s12)*Phi(s12);
CI2=Adj(s21)/c21*Ad/exp(s21)*Phi(s21) + Adj(s22)/c22*Ad/exp(s22)*Phi(s22);
CI3=Adj(s31)/c31*Ad/exp(s31)*Phi(s31) + Adj(s32)/c32*Ad/exp(s32)*Phi(s32);

% fun=@(t,x,Z) ...
% [ A(1,1)*x(1) + A(1,2)*x(2) + Ad(1,1)*Z(1,1) + Ad(1,2)*Z(2,1);
%   A(2,1)*x(1) + A(2,2)*x(2) + Ad(2,1)*Z(1,1) + Ad(2,2)*Z(2,1) ];
% sol=dde23(fun,1,hist,[0 tf]);

t=linspace(0,tf);
xh1=zeros(size(t));
xh2=zeros(size(t));
for i=1:length(t)
    term=expm(S0*t(i))*(C0*x0 + CI0) + 2*real(expm(S1*t(i))*(C1*x0 + CI1)) ...
        + 2*real(expm(S2*t(i))*(C2*x0 + CI2)) + 2*real(expm(S3*t(i))*(C3*x0 + CI3));
    xh1(i)=real(term(1));
    xh2(i)=real(term(2));
end
plot(t,xh1,t,xh2), hold on
% plot(sol.x,sol.y,':')
title('Initial Condition Response')
xlabel('Time, t')
ylabel('Response, x(t)')

for n=1:2

```

```

%-----%
switch n
    case 1
        u=@(t) [cos(t); 0];
    case 2
        u=@(t) [cos(t); sin(t)];
end
%-----%

xf1=zeros(size(t));
xf2=zeros(size(t));
for i=1:length(t)
    xi=linspace(0,t(i));
    xh=t(i)/length(xi);
    temp0=zeros(2,1); temp1=zeros(2,1);
    temp2=zeros(2,1); temp3=zeros(2,1);
    for j=1:length(xi)
        temp0=temp0+expm(S0*(t(i)-xi(j)))*C0*u(xi(j));
        temp1=temp1+expm(S1*(t(i)-xi(j)))*C1*u(xi(j));
        temp2=temp2+expm(S2*(t(i)-xi(j)))*C2*u(xi(j));
        temp3=temp3+expm(S3*(t(i)-xi(j)))*C3*u(xi(j));
    end
    temp=real(temp0)+2*(real(temp1)+real(temp2)+real(temp3));
    xf1(i)=xh*temp(1);
    xf2(i)=xh*temp(2);
end
x1=xh1+xf1;
x2=xh2+xf2;

fun1=@ fun(t,x,Z)+u(t);
sol1=dde23(fun1,1,hist,[0 tf]);

figure
plot(t,x1,t,x2), hold on
% plot(sol1.x,sol1.y,':')
title('Total Response')
xlabel('Time, t')
ylabel('Response, x(t)')
end

```

參考文獻

- [1] Asl, F. M. and Ulsoy, A. G. (2003). Analysis of a system of linear delay differential equations, *Journal of Dynamic Systems, Measurement and Control* **125**, 2, pp. 215-223.
- [2] Bellman, R. E. and Cooke, K. L. (1963). *Differential-Difference Equations* (Academic Press, New York).
- [3] Corless, R. M., Gonnet, G. H., Hare, D. E. G., Jeffrey, D. J. and Knuth, D. E. (1996). On the Lambert W function, *Advances in Computational Mathematics* **5**, 4, pp. 329-359.
- [4] Jeffrey, D. J., Hare, D. E. G. and Corless, R. M. (1996). Unwinding the branches of Lambert W function, *Mathematical Scientist*, **21**, pp. 1-7.
- [5] Malek-Zavarei, M. and Jamshidi, M. (1987). *Time-Delay Systems: Analysis, Optimization and Applications* (Elsevier Science Pub., New York, U.S.A.).
- [6] Pease, M. C. (1965). *Methods of Matrix Algebra* (Academic Press, New York).
- [7] Yi, S. and Ulsoy, A. G. (2006). Solution of a system of linear delay differential equations using the matrix Lambert function, in *Proceedings of 2006 American Control Conference, Minneapolis, MN, 2006 June*, pp. 2433-2438.
- [8] Yi, S., Ulsoy, A. G. and Nelson, P. W. (2006). Solution of systems of linear delay differential equations via Laplace transformation, in *Proceedings of 45th IEEE Conference on Decision and Control, San Diego, CA, DEC. 2006*, pp. 2535-2540.
- [9] Yi, S., Nelson, P. W. and Ulsoy, A. G. (2007). *Time-Delay Systems: Analysis and Control using the Lambert W Function* (World Scientific).
- [10] 郭威志 (2004) : 矩陣型的 Lambert W-函數及其應用於時滯系統穩定性分析 (東海大學碩士論文)