

利用基因演算法實作一個 LEGO Mindstorms NXT 兩輪自體平衡機器人

林哲英, 黃志鵬*, 和 洪瑞鍾

資訊科學系, 臺北市立大學,

台北市中正區愛國西路 1 號, 台北市 100, 台灣 R.O.C.

ponytony@seed.net.tw

摘要

本研究主要的目的是利用基因演算法(Genetic Algorithm, GA) 對LEGO兩輪自體平衡機器人的非線性系統做性能優化的控制器設計。為了使狀態響應最佳化並達到平衡控制的目的, 使用GA尋找一組較佳的狀態回授增益。基於LEGO Mindstorms NXT所提供的動態模型, 系統狀態的性能可以被特徵化成: 上升時間(RS)、最大超越量(OS)及穩定時間(IAE)。因此, GA中的適合函數則是藉由這三項性能指標做定義以達到尋求較佳性能響應的係數。為了顯示方法的有效性, 本論文所提出的方法與現有的方法做比較。藉由模擬結果驗證了本研究所提出經GA調整的優化控制器具有較佳的效能。

關鍵詞: 基因演算法、適合函數、兩輪自體平衡機器人、動態模型

1 緒論

近年來, 兩輪自體平衡機器人(Two Wheeled Self-Balancing Robot, 2WSBR)的控制已被許多學者研究[1-6]。雖然有部分學者進行實作, 但大部分的學者則是提出了理論分析和模擬結果。2WSBR常見的實作有兩輪移動式機器人(Two wheeled mobile robot)被廣泛的應用在教育中, 而另一個常見的則是 LEGO Mindstorms NXT 模組, LEGO 模型除了在教育界中廣為使用, 在一般家庭中也隨處

可見, 其優點除了在於零件取得容易外, 對於模型的原型設計也容易許多。本篇論文利用 LEGO 所製作出來的 2WSBR 近似於倒單擺系統, 在系統不穩定的情況下, 設計控制系統使之平衡。在控制器的設計上, 主要的研究方向可以從系統模型上出發, 分別有李亞普諾夫理論(Lyapunov theory)及狀態回授控制(State feedback control)。而在狀態回授控制中, 二次線性調整器(Linear quadratic regulator, LQR)經常被應用於調整回授增益(Feedback gain)。然而, 其嘗試錯誤與經驗法則的調整方式會耗費太多的時間與精力來尋求一組較為穩定的參數。因此, 本論文利用基因演算法(GA)來做出最佳化調整。

基因演算法(GA), 是一個全域搜尋演算法, 參考生物演化的方式可分為選擇、交配及突變的階段找出最佳解[6]。GA 被廣泛的應用在許多領域內, 而本篇論文則將 GA 應用在設計狀態回授控制器的參數上。本論文以系統動態模型進行分析, 由文獻[7]中所推測出來的狀態方程加以實作, 並藉由 GA 調整出狀態較佳的回授增益。根據 LQR 所調整出的回授增益與本論文所提出的 GA 最佳化回授增益參數做比較, 其模擬結果顯示出系統的狀態響應(Step response)在上升時間(RS)、最大超越量(OS)及穩定時間(IAE)上有著較佳的效能。本篇論



圖一、兩輪自體平衡機器人結構

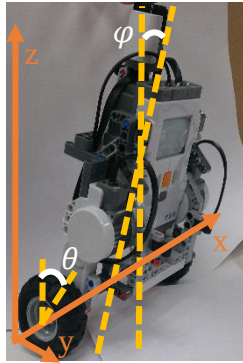
* 本篇論文是由國家科學委員會計畫支援, 計畫編號: NSC 102-2221-E-845 -001 -

文將以以下段落組成。段落 2 將描述系統模型，段落 3 提出控制理論的分析，段落 4 我們將所提出的方法效能，而結論(效能)則在段落 5 被提到。

2 描述系統模型

本篇論文中，圖一顯示出用 LEGO Mindstorms NXT 的套件所做出的 2WSBR 的模型結構。本 2WSBR 含有陀螺儀感測器、輪轉角度與輪轉速度感測器。為了系統狀態方程的推估與測量，我們將感測器所得到的數據轉化成身體傾倒角度、身體傾倒角速度、輪轉角度及輪轉角速度四個數值當作是系統的輸出。

2WSBR 的座標系統及狀態方程相關參數在圖二及表一中呈現。其中，圖二的座標系統參數說明



圖二、兩輪自體平衡機器人座標系

如下： φ 是身體傾倒角度， θ 是輪轉角度。在表一中，呈現了系統模型相關係數及說明。

我們根據[7]所推估的方法來為本系統進行推導，並將左右輪設定為同步且輸入假設為 u ，因此本論文所提出的 2WSBR 的狀態方程式以下列表示：

$$\begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\varphi} \end{bmatrix} + \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} \theta \\ \varphi \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} u \quad (1)$$

$$\text{又 } E = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix}, F = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix},$$

$$G = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}, H = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}$$

其中， $e_{11} = (2m + M)R^2 + 2J_w + J_d$

$$e_{12} = e_{21} = MLR - 2n^2J_d$$

表一、系統相關參數

輪子重量	m	0.0168 (kg)
身體重量	M	0.553 (kg)
輪子半徑	R	0.056 (m)
身體深度	D	0.094 (m)
身體寬度	W	0.185 (m)
輪軸與身體質心的距離	L	0.082 (m)
輪子的轉動慣量	J_w	2.6×10^{-5} (kgm ²)
身體的轉動慣量	J_b	1.2×10^{-3} (kgm ²)
直流馬達的轉動慣量	J_d	1×10^{-5} (kgm ²)
直流馬達的電阻	R_d	6.69 (Ω)
直流馬達的 back EMF 常數	K_e	0.468 (V sec/rad)
直流馬達的扭力常數	K_t	0.317 (Nm/A)
地心引力	g	9.81 (m/sec ²)
齒輪比	n	1
身體與直流馬達的摩擦係數	f_d	0.0022
輪子與地面的摩擦係數	f_w	0

$$e_{22} = ML^2 + J_b + 2n^2J_d$$

$$f_{11} = 2(\beta + f_w)$$

$$f_{12} = f_{21} = -2\beta$$

$$f_{22} = 2\beta$$

$$g_{11} = g_{12} = g_{21} = 0$$

$$g_{22} = -MgL$$

$$h_1 = \alpha, h_2 = -\alpha$$

$$\alpha = \frac{nK_t}{R_d}, \beta = f_d + \frac{nK_eK_t}{R_d}$$

接著，讓狀態 $[\theta \ \varphi \ \dot{\theta} \ \dot{\varphi}]^T$ 當成 x 則可以得到方程式(2)：

$$\begin{aligned} \dot{x} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} x + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} u, \\ y &= \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix} x + \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} u \quad (2) \\ \text{又 } A &= \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}, B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}, \end{aligned}$$

$$C = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix}, D = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix}$$

其中, $\det(E) = e_{11}e_{22} - e_{12}e_{21}$,

$$a_{32} = \frac{g_{22}e_{12}}{\det(E)}, a_{42} = -\frac{g_{22}e_{11}}{\det(E)},$$

$$a_{33} = -\frac{f_{11}e_{22}}{\det(E)} + \frac{f_{12}e_{12}}{\det(E)},$$

$$a_{43} = \frac{f_{11}e_{12}}{\det(E)} + \frac{f_{22}e_{11}}{\det(E)},$$

$$a_{34} = \frac{f_{22}e_{22}}{\det(E)} + \frac{f_{22}e_{12}}{\det(E)},$$

$$a_{44} = \frac{f_{12}e_{11}}{\det(E)} + \frac{f_{12}e_{12}}{\det(E)},$$

$$a_{11} = a_{21} = a_{31} = a_{41} = 0,$$

$$a_{12} = a_{14} = a_{22} = a_{23} = 0$$

$$b_3 = \frac{h_1e_{22}}{\det(E)} + \frac{h_1e_{12}}{\det(E)}$$

$$b_4 = \frac{h_2e_{11}}{\det(E)} + \frac{h_2e_{12}}{\det(E)}$$

$$b_{11} = b_{12} = b_{21} = b_{22} = 0$$

$C = \text{Identity Matrix}, D = \text{Zero Matrix}$

因此, 系統狀態方程可以下列(3) 表示, 做為系統控制的受控體。

$$\dot{x} = Ax + Bu, \quad y = Cx + Du \quad (3)$$

3 GA 優化的控制器設計

在本段中, 我們將敘述本論文所提的控制方法, 用基因演算法調整狀態回授控制的回授增益。而適合函數將以上升時間(RS)、最大超越量(OS)及穩定時間(IAE)來對系統狀態進行效能調整。

● 回授增益

如同第(3)式所提到的狀態方程式, 我們可以藉由狀態回授控制的方程式(4)來進行回授增益 k 的調整。

$$u = -k^T x \quad (4)$$

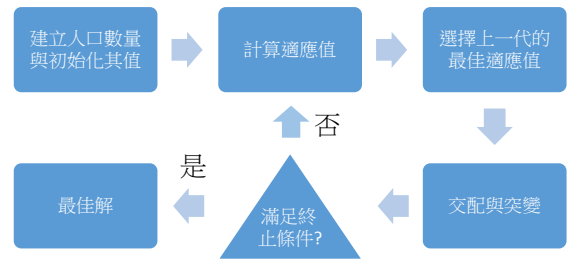
因此, 我們可以將(3)式替代成(5)：

$$\begin{aligned} \dot{x} &= (A - Bk^T)x \\ y &= (C - Dk^T)x \end{aligned} \quad (5)$$

藉由調整回授增益 k 即可改變 $A - Bk$ 的特徵值使系統穩定[7]。而在大部分的研究中, 回授增益的調整是藉由 LQR 的成本函數 (Cost function) 進行調整, 其 Q 跟 R 矩陣的設定是由經驗上的調整進行, 耗費人力與耗時的問題, 我們將用基因演算法的方式將回授增益 k 進行調整。

● 基因演算法調整

基因演算法是由生物演化的概念所提出的搜尋最佳解的演算法, 其演算法流程在圖三中

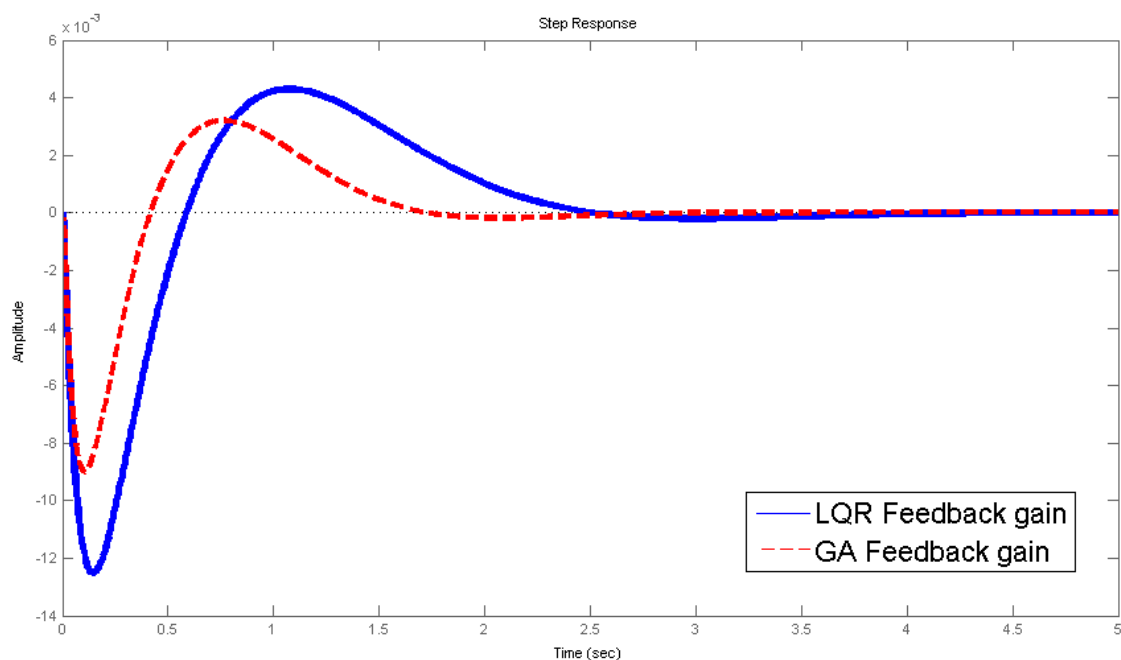


圖三、基因演算法流程圖

表示。本篇論文則是會利用 GA 來搜尋最佳的回授增益, 藉由系統的步階響應可得知系統的上升時間(RS)、最大超越量(OS)及穩定時間(IAE), 由這些參數可以得知系統效能。其中, 上升時間是指訊號從最大訊號振幅的 10% 轉換到 90% 的時間, 最大超越量是指信號超過目標的最大值, 穩定時間則是指訊號達到目標值 2% 範圍內所需的時間。我們將這三個參數放進適合函數中, 並得到下列適合函數的公式：

$$f_{sys} = fit(sys) = e^{-w_1 \cdot sys(RS)} \times e^{-w_2 \cdot sys(OS)} \times e^{-w_3 \cdot sys(IAE)} \quad (6)$$

其中 w_1 、 w_2 、 w_3 分別是上升時間、最大超越量及穩定時間的權重。而回授增益的最大值與最小值則如表二中所示, 藉由 $A - Bk$ 矩陣的特徵值可界定出穩定範圍。其中回授增益 $k = [k_1 \ k_2 \ k_3 \ k_4]^T$



圖四、兩輪自體平衡機器人結構

表二、回授增益最大與最小值

	最小值	最大值
k_1	-3	-1
k_2	-50	-40
k_3	-2	-1
k_4	-4	-3

4 模擬與實驗結果

在本段中，我們將展現本論文所提出的 GA 最佳化回授增益在系統模型上的效能。而根據 LQR 所調整出來的回授增益[7]將與本論文的結果進行比較，表三將回授增益值表示如下：

表三、回授增益

	LQR	GA
k_1	-0.8351	-2.33
k_2	-34.1896	-42.0464
k_3	-1.0995	-1.7888
k_4	-2.8141	-3.9555

而模擬結果在圖四中顯示，可以看出其上升時間、最大超越量及穩定時間上皆有較佳的結果，其中穩定時間更是減少了 59%。 w_1 、 w_2 及 w_3 的選擇上則是以原 LQR 求出回授增益的系統作為其參數。

5 結論

本論文提出了一個在 LEGO Mindstorms NXT 的 2WSBR 人上利用 GA 來調整狀態回授控制的回授增益。2WSBR 的廣泛應用讓許多種研究方法不斷地被提出，其中，我們利用易於廣泛被使用的 LEGO Mindstorms NXT 的套件做出 2WSBR 的模型並將其應用。而在控制器的範疇上，本篇論文以狀態回授控制進行控制使得系統穩定，而調整回授增益的方法則是使用基因演算法，有別於錯誤嘗試與經驗法則的 LQR，使得效能調整自動化與有較佳的結果。最後，根據模擬實驗結果顯示，使用本論文所提的方在系統響應的法上升時間、最大超越量及穩定時間有著較佳的結果。

References

- [1] C.-H. Huang, W.-J. Wang, and C.-H. Chiu. Design and implementation of fuzzy control on a two-wheel inverted pendulum. *Journal of IEEE Trans. Ind. Electron*, vol. 58, no.7, pp. 2988–3001, Jul. 2011.

- [2] F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer. JOE: A mobile, inverted pendulum. *Journal of IEEE Trans. Ind. Electron.*, vol. 49, no. 1, pp. 107–114, Feb. 2002.
- [3] K. Pathak, J. Franch, and S. K. Agrawal. Velocity and position control of a wheel inverted pendulum by partial feedback linearization. *Journal of IEEE Trans. Robot.*, vol. 21, no. 3, pp. 505–513, Jun. 2005.
- [4] P. Oryschuk, A. Salerno, A. M. Al-Husseini, and J. Angeles. Experimental validation of an underactuated two-wheeled mobile robot. *Journal of IEEE/ASME Trans. Mechatronics*, vol. 14, no. 2, pp. 252–257, Apr. 2009.
- [5] M. T. Ravichandran and A. D. Mahindrakar. Robust stabilization of a class of underactuated mechanical systems using time scaling and Lyapunov redesign. *Journal of IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4299–4313, Sep. 2011.
- [6] Y. T. Juang, Y. T. Chang, and C. P. Huang. Design of fuzzy PID controllers using modified triangular membership functions. *Journal of ELSEVIER Information Science*, 178, pp. 1325 – 1333, 2008.
- [7] The Math Works, NXTway-GS Model-Based Design—Control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT