

# 應用田口微分進化演算法於模式簡化

## Model Reduction Using Taguchi-Differential Evolution Algorithm

何文獻  
高雄醫學大學  
醫療資訊管理學系  
whho@kmu.edu.tw

蔡進聰  
屏東教育大學  
資訊科學系  
jttsai@mail.npue.edu.tw

周至宏  
國立高雄第一科技大學  
資訊系統與控制研究所  
choujh@ccms.nkfust.edu.tw

### 摘要

在本文中，提出了兩種改良式的微分進化演算法(differential evolution algorithm, DEA)，稱為田口微分進化演算法 (Taguchi-differential evolution algorithm, TDEA) 及可調整式田口微分進化演算法 (adjustable Taguchi-differential evolution algorithm, ATDEA)。其演算法應用於模式簡化 (model reduction)，降階模型的動作使得模擬、分析或是控制設計能夠更容易。TDEA 修改了傳統 DEA 突變之法則。利用田口實驗法具有的系統規劃參數設計理念，應用於突變機制中，取代了傳統 DEA 在突變機制中隨機產生擾動向量的動作，提升了避免搜索過程中陷入局部解的機率。然而在搜尋最佳解時，最大的問題在於無法得知其範圍是多大，故 ATDEA 引入了搜尋空間擴大機制 (search space expansion scheme)，讓搜尋範圍空間成為動態，使得搜索效率能提升。

最後利用 TDEA 及 ATDEA 去做模式簡化的應用，並與文獻中提出的 DEA 來做分析比較。

關鍵字：微分進化演算法，田口實驗法，搜尋空間擴大機制，模式簡化。

### Abstract

In this paper, the problems of model reduction are solved by using two improved differential evolution algorithms (DEA), which are called Taguchi-differential evolution algorithm (TDEA) and adjustable Taguchi-differential evolution algorithm (ATDEA). The use of a reduced-order model makes it easier to implement analyses, simulations and control designs. The proposed TDEA is modified from the traditional DEA with mutation operation. The systematic reasoning ability of the Taguchi method can promote the mutation efficiency. The Taguchi method applied into mutation operation and replaced the action of perturbed vectors were chosen randomly in the tradition DEA. TDEA can avoid premature convergence with controllable deteriorating probability.

However, the biggest problem in optimal solution is that we can't know the range of searching. Here, we incorporate a search space expansion scheme in the ATDEA approach and let search space dynamical. The efficiency will be promoted.

Finally, we use TDEA and ATDEA to solve the example of model reduction. Simulation results show that the proposed TDEA and ATDEA approaches can obtain better performances than the existing DEA reported recently in the literature.

Keywords: differential evolution algorithm, Taguchi method, search space expansion scheme, model reduction.

### 一、緒論

隨著控制技術與控制要求的提高，系統的階次也越來越高。高階次的模型會帶來繁重的計算負擔，同時也增加系統設計的複雜程度，例如一個受控體 (plant) 的轉移函數通常包含著時延 (time delay) 項次[1]，這會使得在做時域分析或是控制器的設計時，將會更加的困難。故藉由著高階系統轉成低階系統，這些問題將會相當的容易。我們將高階系統轉成低階系統的動作，稱為模式簡化 (model reduction)。

模式簡化的計算方法傳統上有 Pade approximation [2]、Routh approximation [3]、平衡截斷法 (balanced truncation) [4]等，這一類的方法屬於非性能指標導向的範圍。相反的，在以性能指標導向的方法裡，它藉由近似誤差準則的最小化來得到最佳降階模型，其方法像是有  $H^2$ -norm [5-6]、 $L^2$ -norm [7-8]、 $H^\infty$ -norm [9-10]及  $L^\infty$ -norm [11-12]。這一類的方法，它們擁有一套有效率的最佳化演算法去計算性能指標。以  $H^2$ -norm、 $L^2$ -norm 來講，常用的方法為梯度法為基礎 (gradient-based) 的搜尋法則，但不幸地，梯度法通常都會陷入局部的最佳解。同樣的， $H^\infty$ -norm 及  $L^\infty$ -norm 的困難點在於它們被選擇為性能指標時，就必須求其導數，但這兩者沒有任何一點可微分的性質，所以，在最小化  $H^\infty$ -norm 或是  $L^\infty$ -norm

的性能指標裡，仍然缺少有效的計算方法。

微分進化演算法 (differential evolution algorithm, DEA) 近年在非線性、非可微分以及非凸集合 (non-convex) 函數的最佳化上皆有成功的案例[13]。其特質大致與基因演算法類似，是個以族群 (population) 為基礎，同時也是一個平行搜尋的演算法，其演算法過程簡單且同時需要較少的參數。DEA的擾動向量 (perturb vector) 在突變過程中是以隨機產生，進而得到差異向量 (differential vector)，並在加上量化因子 (scaling factor)，成為新的染色體，對於跳脫局部最佳解有較大的可能性。

在本文中，我們在傳統DEA中引入了田口實驗法 (Taguchi method)，稱為田口微分進化演算法 (Taguchi-differential evolution algorithm, TDEA)。田口實驗法是一種能夠有效節省實驗成本的最佳化機制，在傳統DEA中，突變向量是以隨機方式產生，這將會使得產生出來的值較不穩定，同時也為了增加跳脫局部最佳解之機率，故本文藉由田口實驗法的品質最佳化機制，來提升DEA的突變效能，並期望能藉此TDEA的方法能夠應用在模式簡化中。但在尋找最佳參數解時，搜尋範圍大都由使用者決定，但最佳解是否在此範圍內，我們無法得知。故這裡再引入搜尋範圍擴大機制，稱為可調整式田口微分進化演算法 (adjustable Taguchi-differential evolution algorithm, ATDEA)。搜尋空間擴大機制在於經過某週期的世代下，一一檢查其當代最佳參數解，若符合條件時，則啟動此機制，故可預期的是找出來的最佳解不會落至局部最佳解。

本文分為五個部份，第一部份為緒論，說明本文的研究緣起及目的，並略述本文的研究方法及文章架構。第二部份為欲解決之問題描述。第三部份為田口微分進化演算法 (TDEA) 的介紹，說明如何整合田口實驗法以及DEA，來改善傳統DEA在突變過程中以隨機選取的方式，並提升搜尋效能並且增加跳脫局部最佳解的機會，並且也提出了搜尋空間擴大機制，稱為可調整式田口微分進化演算法 (ATDEA)，用來搜尋參數的最佳解。第四部份為模式簡化應用，在這裡將會描述模式簡化的數學問題，並引用第三部份的TDEA及ATDEA來做最佳化設計，並與其他文獻中提出DEA方法做比較，證實TDEA及ATDEA的方法較為優秀。第五部份為結論。

## 二、問題描述

我們考慮一個單輸入單輸出時延 (time delay) 的高階系統  $G(s)$ ，希望能轉換成近似模型  $H_m(s)$  如下[9]：

$$H_m(s) = \frac{b_0 + b_1s + \dots + b_{m-1}s^{m-1}}{a_0 + a_1s + \dots + a_{m-1}s^{m-1} + s^m} \cdot e^{-\tau_d s} \quad (2-1)$$

其中  $a_i, b_i, \tau_d$  為降階模型的係數， $i = 0, 1, \dots, m-1$ ，

$H_m(s)$  包含了原系統  $G(s)$  的特徵。在本研究中，我們的目標在於尋找最佳降階模型  $H_m(s)$ ，其依據頻率  $L^2$  誤差性能指標 ( $L^2$ -error performance index) 如下：

$$J = \sum_{i=0}^N |G(jw_i) - H_m(jw_i)|^2 \quad (2-2)$$

其中  $w_i$  為頻率點， $i = 1, 2, \dots, N$ 。因此，本文的目標希望在  $w_i$  點時，原系統  $G(s)$  與降階後模型  $H_m(s)$  的性能指標能越小越好。倘若  $G(s)$  為漸近穩定 (asymptotically stable)，則我們設定限制條件如下：

$$H_m(0) = G(0) \quad (2-3)$$

由此限制條件，我們可確保輸入訊號為單位步階 (unit-step) 響應時，原系統  $G(s)$  與降階模型  $H_m(s)$  的穩態誤差能夠相同。

參數的最佳選擇是依據 (2-2) 所得到的性能指標  $J$  之最小化為準，藉由第三章我們提出的 TDEA 及 ATDEA 去尋找最佳參數  $a_i, b_i, \tau_d$ 。在第四部份中，本文將會舉例去驗證所提出的 TDEA 及 ATDEA 將會比傳統的 DEA 的結果還要好。

## 三、田口微分進化演算法

微分進化演算法 (differential evolution algorithm, DEA) 最初是由 Storn 和 Price 所提出 [14]，而本文中將介紹的兩種改良式 DEA，分別為田口微分進化演算法 (Taguchi-differential evolution algorithm, TDEA) 及可調整式田口微分進化演算法 (adjustable Taguchi-differential evolution algorithm, ATDEA)。其中在 TDEA 中，我們引入了輪盤式選擇 [15] (roulette wheel selection) 及田口實驗法。而在 ATDEA 中，除了輪盤式選擇及田口實驗法外，也整合了搜尋空間擴大機制[9]。

輪盤式選擇的目的在於讓適應值較佳的染色體有較高的機會能夠被挑選出來，並配置於直交表中。

由於傳統 DEA 突變方式是以隨機選取並產生擾動向量，對於間接得到的子代染色體品質不容易提升。而田口實驗法是一種能夠有效地節省實驗成本的最佳化機制，故本文引入田口實驗法並將之融入突變機制中，將挑選出來的染色體配置於直交表，產生出新的染色體，於是產生優良品質的染色體的機率便可以大幅提升。

搜尋空間擴大機制的的作用在於避免收斂值跑到局部最佳解所應映的策略，在一般的演算法中，都會自訂一固定範圍，但這將會存在一個問題就是，倘若範圍設定的較大，會使得適應值的收斂速度會比較慢且可能會陷入局部最佳解；相對地，範圍設定的較小，很有可能最後找到的最佳解不是全域最佳解，而只是局部最佳解，故針對這問題，此

範圍的擴大必須為動態的。其觀念為先設定非常小的範圍，接著在每一個週期世代交替過程中，一一檢查其當代最佳參數解來決定範圍是否要擴大，如此經過幾次反覆運算時，不僅收斂速度較快，且得到全域最佳解的機會也較大。

### 3.1 TDEA 流程

TDEA 成為一種加強式的改良型微分進化演算法，其操作方式與傳統 DEA 大致上雷同，只在於在突變機制引入田口法的運用，如圖 3-1 為 TDEA 之流程圖。以下將列出 TDEA 的應用步驟：

- 步驟一：參數設定，分別為染色體數目，交配率、量化因子及世代代數。
- 步驟二：設定初始族群，也就是母代。
- 步驟三：採取輪盤法挑選出染色體。
- 步驟四：進入突變機制。挑選出的染色體進行田口實驗法。
- 步驟五：選定合適的直交表。
- 步驟六：計算直交表中所有組合的適應值。
- 步驟七：計算 S/N 比。
- 步驟八：製作回應表。
- 步驟九：推估最佳組合之染色體，成為新的擾動向量  $V_{ji}^{G+1}$ 。
- 步驟十：進入交配機制，決定是否交配？若否，跳至步驟十二。
- 步驟十一：新的擾動向量  $V_{ji}^{G+1}$  與原始族群  $X_i^G$  交配，產生出新的嘗試向量  $U_{ji}^{G+1}$ ，也就是子代。
- 步驟十二：將母代與子代混合，重新依照適應值優劣排列。
- 步驟十三：選出優良的族群作為下一代的母代。
- 步驟十四：是否達到中止條件或是世代執行結束？若有，則跳至步驟十五；若否，則回到步驟三至步驟十四。
- 步驟十五：顯示最佳染色體以及最佳適應值。

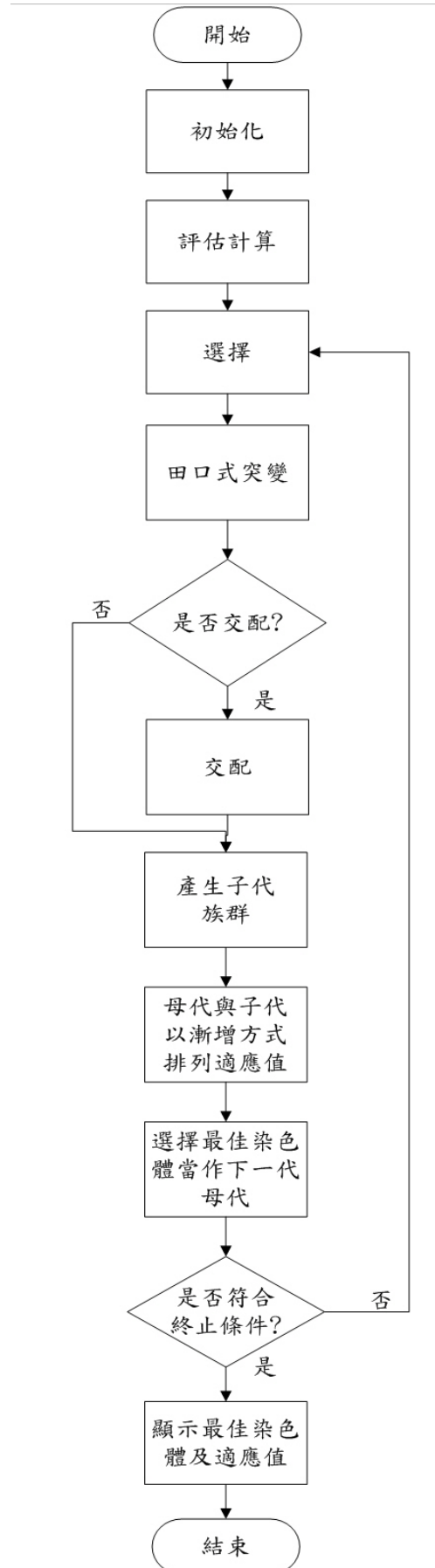


圖 3-1 TDEA 之流程圖

### 3.2 ATDEA 流程

ATDEA 與 TDEA 不同的地方,在於經過  $N_c$  個世代後,加入了搜尋空間擴大機制的判斷,圖 3-2 為 ATDEA 之流程圖。以下將列出 ATDEA 的應用步驟:

- 步驟一: 參數設定,分別為染色體數目,交配率、量化因子及世代代數。
- 步驟二: 設定初始族群,也就是母代。
- 步驟三: 採取輪盤法挑選出染色體。
- 步驟四: 進入突變機制。挑選出的染色體進行田口實驗法。
- 步驟五: 選定合適的直交表。
- 步驟六: 計算直交表中所有組合的適應值。
- 步驟七: 計算 S/N 比。
- 步驟八: 製作回應表。
- 步驟九: 推估最佳組合之染色體,成為新的擾動向量  $V_{ji}^{G+1}$ 。
- 步驟十: 進入交配機制,決定是否交配?若否,跳至步驟十二。
- 步驟十一: 新的擾動向量  $V_{ji}^{G+1}$  與原始族群  $X_i^G$  交配,產生出新的嘗試向量  $U_{ji}^{G+1}$ ,也就是子代。
- 步驟十二: 將母代與子代混合,重新依照適應值優劣排列。
- 步驟十三: 選出優良的族群作為下一代的母代。
- 步驟十四: 倘若達到  $N_c$  個世代過後,執行搜尋範圍擴大機制。
- 步驟十五: 是否達到中止條件或是世代執行結束?若有,則跳至步驟十六;若否,則回到步驟三至步驟十五。
- 步驟十六: 顯示最佳染色體以及最佳適應值。

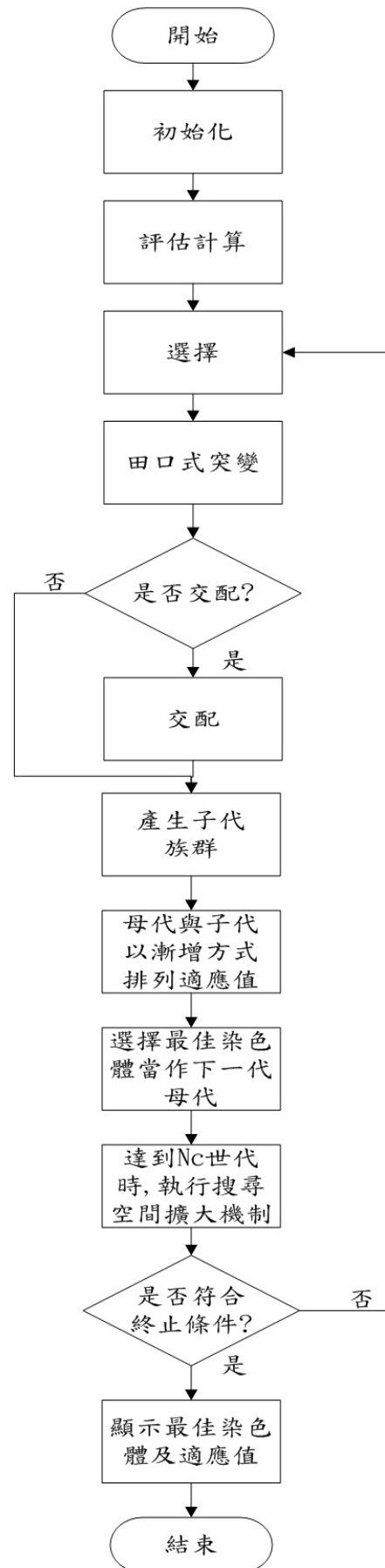


圖 3-2 ATDEA 之流程圖

#### 四、模式簡化應用

此系統取自於[9]，為了能與[9]公平比較，在此設定的條件式皆與[9]相同。此系統  $G(s)$  在分子包含了時延項，圖 4-1 為  $G(s)$  的方塊流程圖：

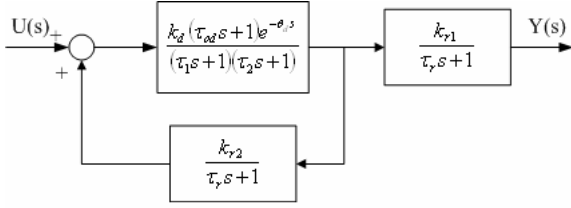


圖 4-1 原系統  $G(s)$  之方塊流程圖

其轉移函數如下：

$$G(s) = \frac{Y(s)}{U(s)} = \frac{k_d k_{r1} (\tau_{od} s + 1) e^{-\theta_d s}}{(\tau_r s + 1)(\tau_1 s + 1)(\tau_2 s + 1) - k_{r2} k_d (\tau_{od} s + 1) e^{-\theta_d s}} \quad (4-1)$$

其中 (4-1) 式的參數值與 [9] 相同，分別為  $k_{r1} = 0.258$ 、 $k_{r2} = 0.281$ 、 $k_d = 1.4494$ 、 $\theta_d = 0.2912$ 、 $\tau_r = 1.3498$ 、 $\tau_{od} = 0.3684$ 、 $\tau_1 = 1.9624$  及  $\tau_2 = 0.43256$ 。在本研究中，我們將  $G(s)$  降階至二階時延系統  $H_2(s)$ ：

$$H_2(s) = \frac{k_{2,p}(s + \tau_{2,z})e^{-\tau_{2,d}s}}{s^2 + a_{2,1}s + a_{2,0}} \quad (4-2)$$

其中性能指標依據 (2-2)，有 51 個  $w_i$  的頻率點值， $w_i = 10^{-2+0.1i} \in [10^{-2}, 10^3]$ ， $i = 0, 1, \dots, N = 50$ ，我們最小化其性能指標，並依據 (2-3) 的限制條件  $H_2(0) = G(0)$ ，這個限制條件代入 (4-2) 式，希望原系統  $G(s)$  與降階後系統在輸入為步階響應時能夠相近。我們藉由此限制條件，我們可以求得參數  $a_{2,0}$  如下：

$$a_{2,0} = \frac{k_{2,p}(1 - k_{r2}k_d)}{k_d k_{r1}} \tau_{2,z} \quad (4-3)$$

因此，針對二階時延系統  $H_2(s)$ ，我們要尋找的最佳參數分別為  $a_{2,1}$ 、 $k_{2,p}$ 、 $\tau_{2,z}$  以及  $\tau_{2,d}$ 。由於原系統  $G(s)$  是穩定的，故每個參數的範圍將限制在  $[0, \infty)$ 。

為了能公平比較，我們使用 DEA、TDEA 及 ATDEA 之參數設定如 [9]：染色體數目為 60、最大世代為 300、交配率為 0.6、量化因子為 1.2、搜尋空間檢查週期為 10 代、搜尋空間擴大係數為 1.2。針對系統  $H_2(s)$ ，在模擬過程中使用 DEA1、DEA2、TDEA1、TDEA2 執行五次，並分別在四個分別為  $[0, 1]_4$ 、 $[0, 10]_4$ 、 $[0, 50]_4$  及  $[0, 100]_4$  的搜尋範圍空間去尋找最佳參數值  $X_4 = [a_{2,1}, k_{2,p}, \tau_{2,z}, \tau_{2,d}]$ ，其模擬的結果為圖 4-2 至圖 4-17。接著 ATDEA1 及 ATDEA2 也執行五次，其

初始搜尋範圍為  $[0, 0.1]_4$ ，分別為圖 4-18 至圖 4-19。最後比較 DEA1、DEA2、TDEA1、TDEA2 在四個空間的最佳結果值和 ATDEA1 及 ATDEA2 之最佳結果值，如圖 4-20 至圖 4-24。

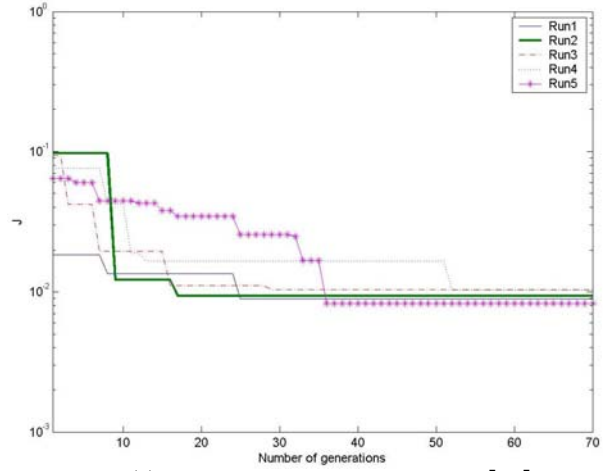


圖 4-2  $H_2(s)$  之 DEA1 在搜尋範圍空間為  $[0, 1]_4$  之模擬結果圖

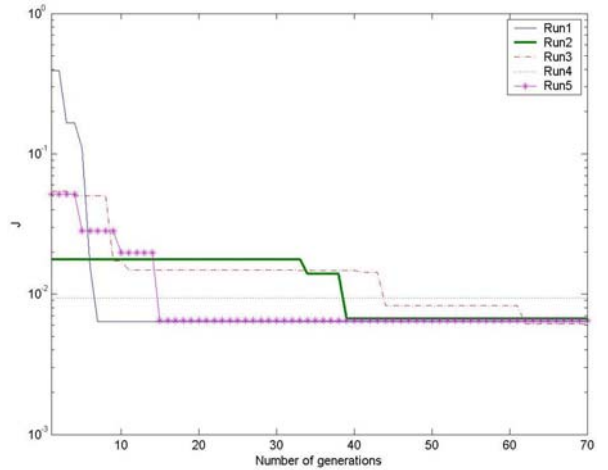


圖 4-3  $H_2(s)$  之 TDEA1 在搜尋範圍空間為  $[0, 1]_4$  之模擬結果圖

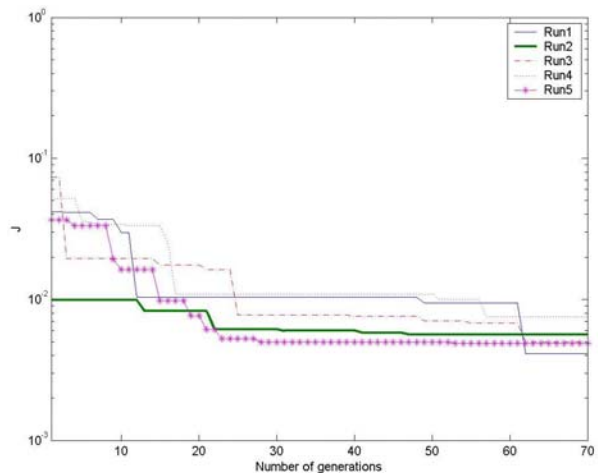


圖 4-4  $H_2(s)$  之 DEA2 在搜尋範圍空間為  $[0, 1]_4$  之模擬結果圖

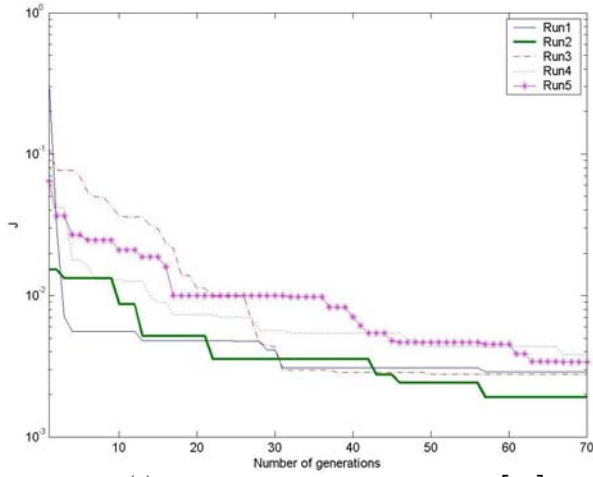


圖 4-5  $H_2(s)$ 之 TDEA2 在搜尋範圍空間為  $[0,1]_4$  之模擬結果圖

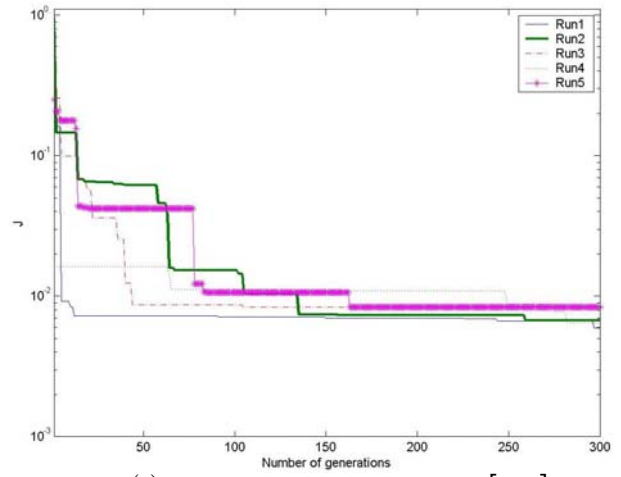


圖 4-8  $H_2(s)$ 之 DEA2 在搜尋範圍空間為  $[0,10]_4$  之模擬結果圖

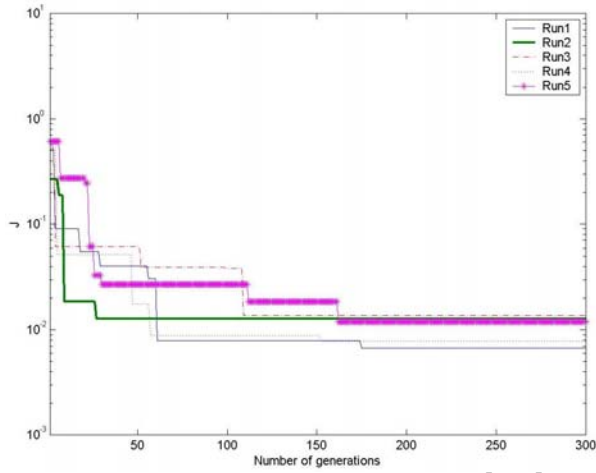


圖 4-6  $H_2(s)$ 之 DEA1 在搜尋範圍空間為  $[0,10]_4$  之模擬結果圖

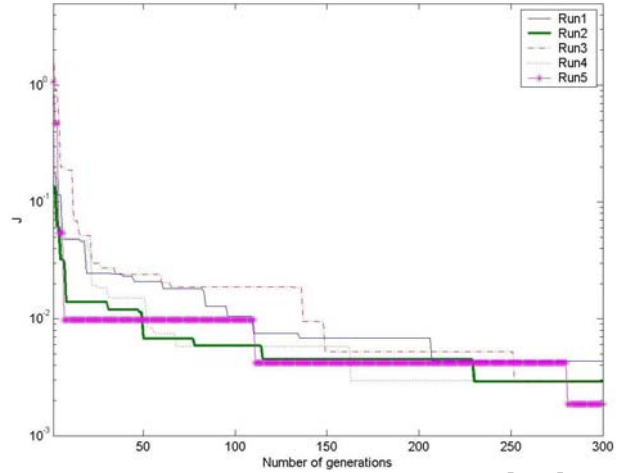


圖 4-9  $H_2(s)$ 之 TDEA2 在搜尋範圍空間為  $[0,10]_4$  之模擬結果圖

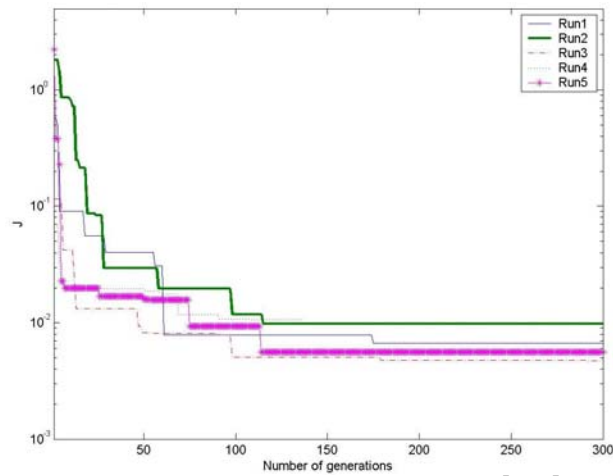


圖 4-7  $H_2(s)$ 之 TDEA1 在搜尋範圍空間為  $[0,10]_4$  之模擬結果圖

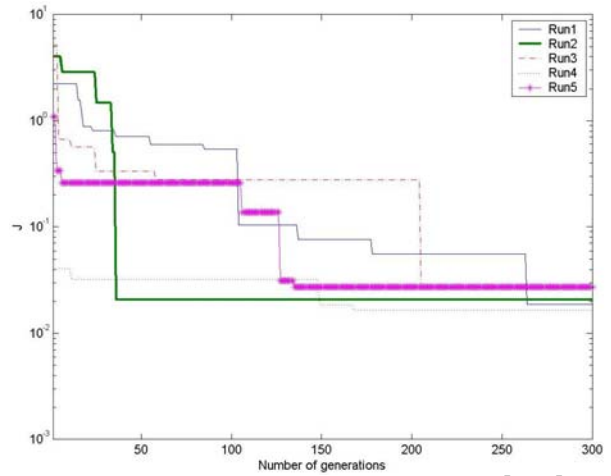


圖 4-10  $H_2(s)$ 之 DEA1 在搜尋範圍空間為  $[0,50]_4$  之模擬結果圖



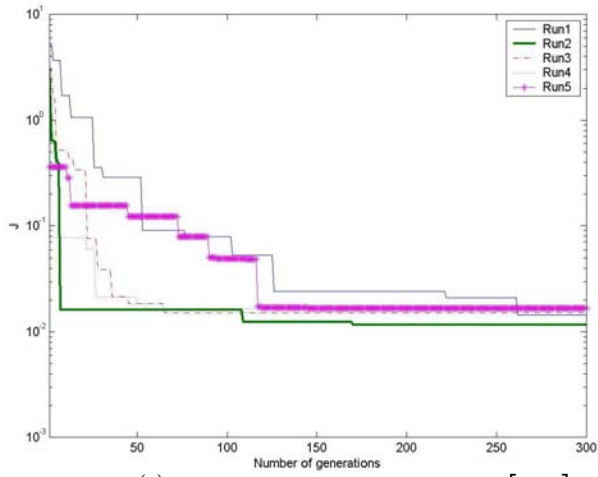


圖 4-11  $H_2(s)$  之 TDEA1 在搜尋範圍空間為  $[0,50]_4$  之模擬結果圖

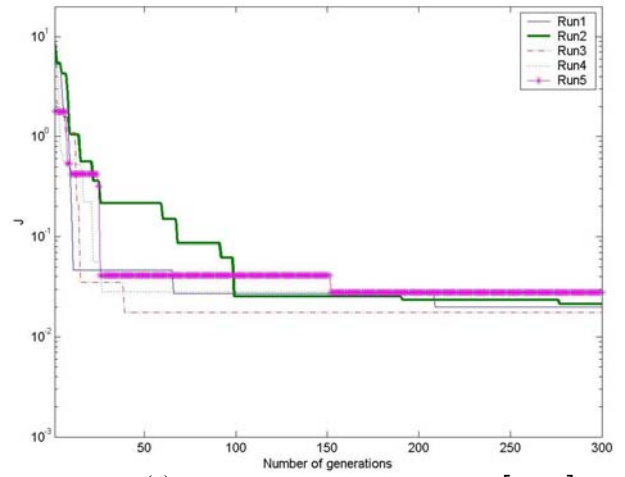


圖 4-14  $H_2(s)$  之 DEA1 在搜尋範圍空間為  $[0,100]_4$  之模擬結果圖

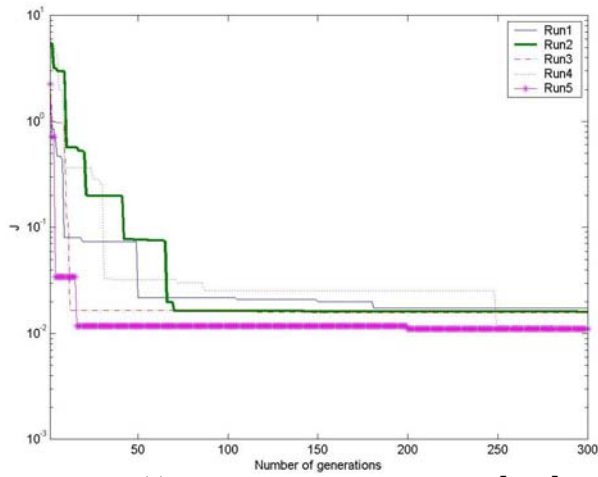


圖 4-12  $H_2(s)$  之 DEA2 在搜尋範圍空間為  $[0,50]_4$  之模擬結果圖

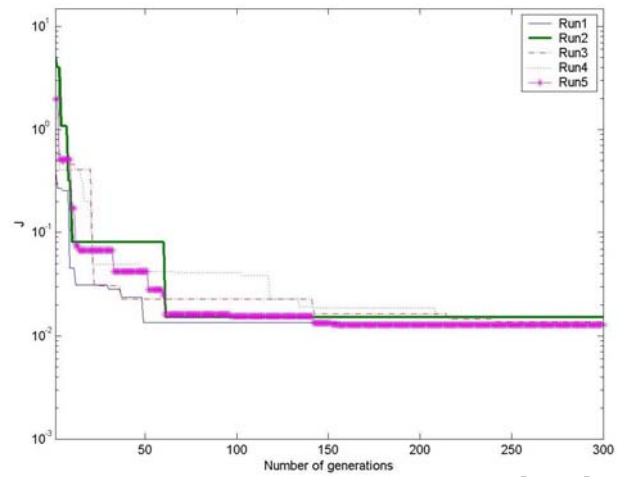


圖 4-15  $H_2(s)$  之 TDEA1 在搜尋範圍空間為  $[0,100]_4$  之模擬結果圖

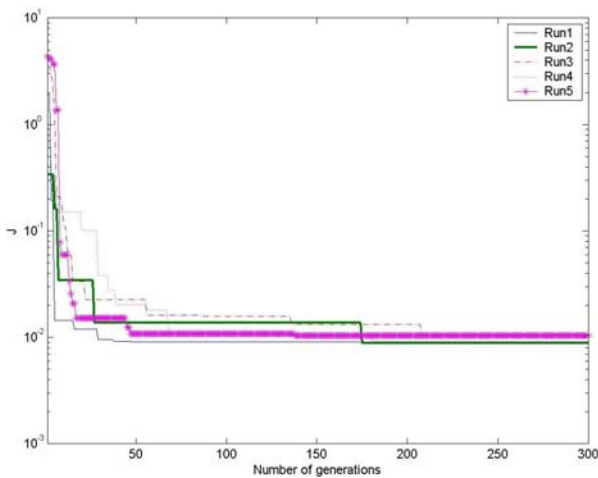


圖 4-13  $H_2(s)$  之 TDEA2 在搜尋範圍空間為  $[0,50]_4$  之模擬結果圖

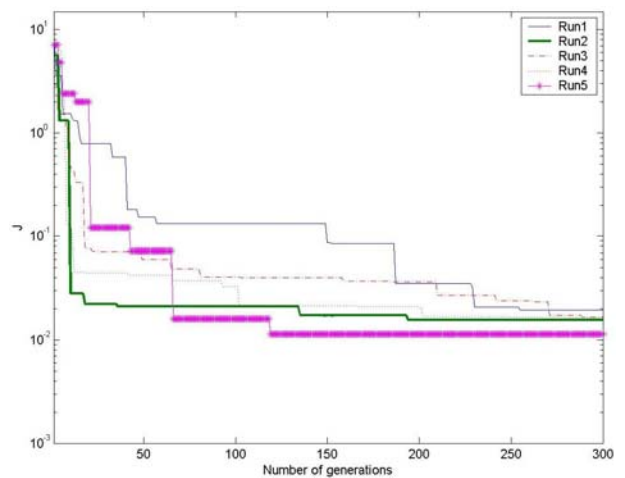


圖 4-16  $H_2(s)$  之 DEA2 在搜尋範圍空間為  $[0,100]_4$  之模擬結果圖

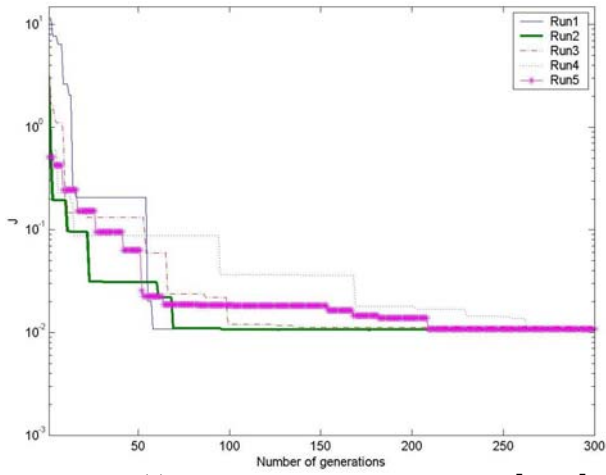


圖 4-17  $H_2(s)$  之 TDEA2 在搜尋範圍空間為  $[0,100]_4$  之模擬結果圖

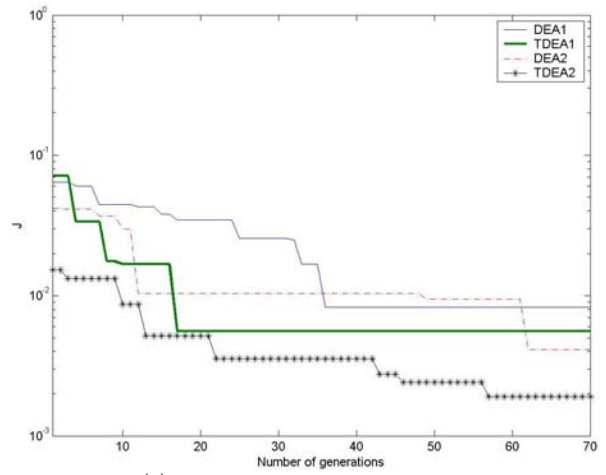


圖 4-20  $H_2(s)$  之兩個 DEA 及 TDEA 在搜尋範圍空間為  $[0,1]_4$  的最佳模擬結果圖

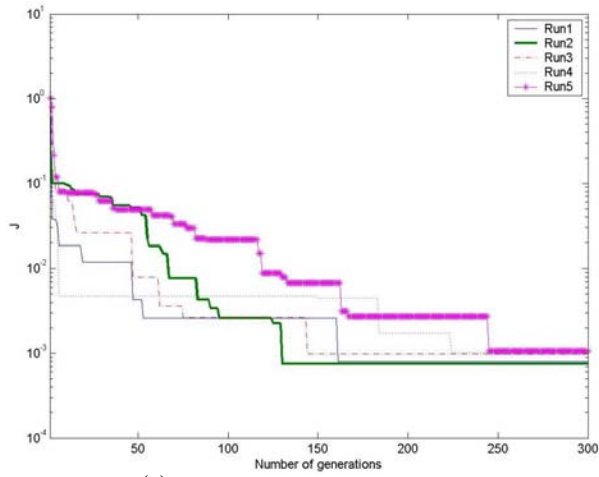


圖 4-18  $H_2(s)$  之 ATDEA1 在初始搜尋範圍空間為  $[0,0.1]_4$  之模擬結果圖

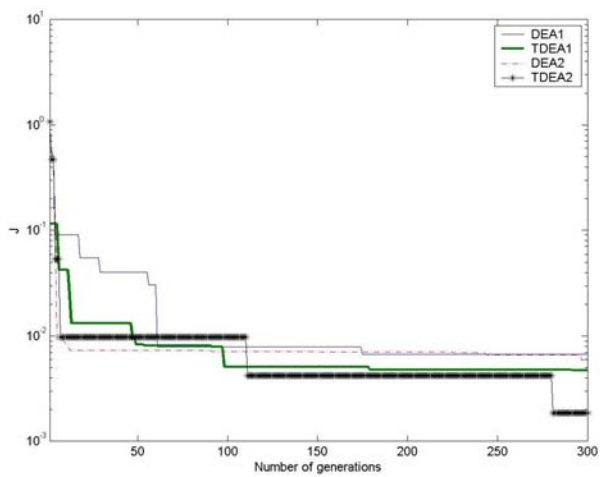


圖 4-21  $H_2(s)$  之兩個 DEA 及 TDEA 在搜尋範圍空間為  $[0,10]_4$  的最佳模擬結果圖

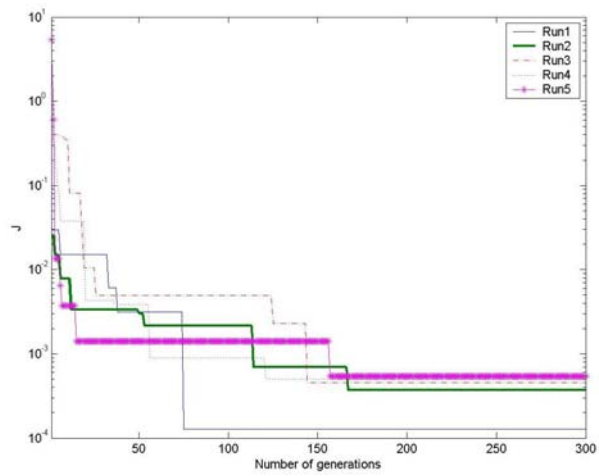


圖 4-19  $H_2(s)$  之 ATDEA2 在初始搜尋範圍空間為  $[0,0.1]_4$  之模擬結果圖

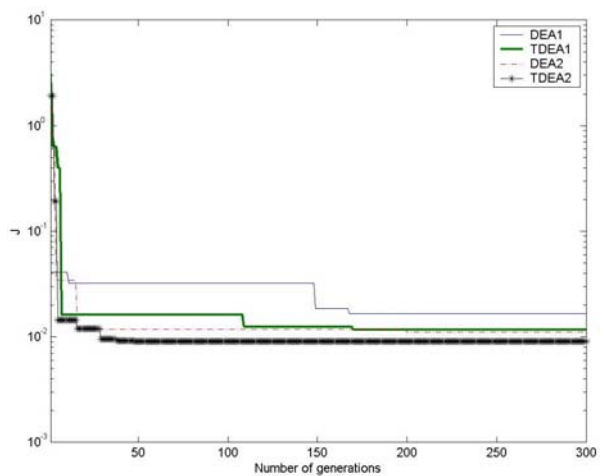


圖 4-22  $H_2(s)$  之兩個 DEA 及 TDEA 在搜尋範圍空間為  $[0,50]_4$  的最佳模擬結果圖



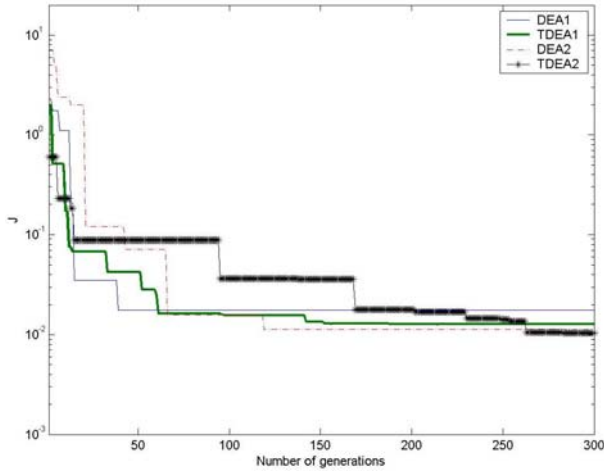


圖 4-23  $H_2(s)$  之兩個 DEA 及 TDEA 在搜尋範圍空間為  $[0,100]_4$  的最佳模擬結果圖

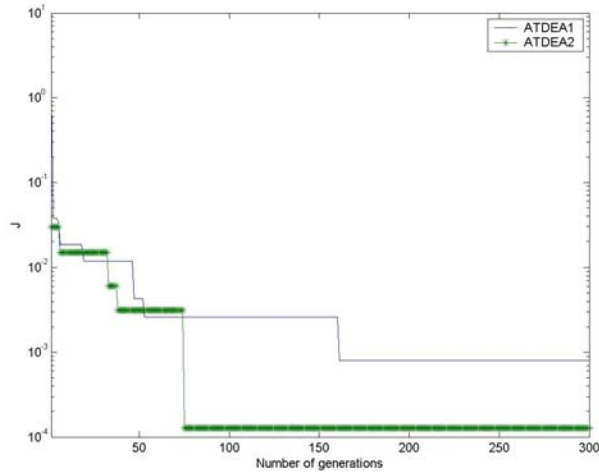


圖 4-24  $H_2(s)$  之兩個 ATDEA 的最佳模擬結果圖

表 4-1 為 TDEA 及 ATDEA 與[9]之比較，我們取性能指標之平均值、最佳值及標準差來比較，由表中我們可以發現，各個搜尋範圍空間的性能指標 J 值均比[9]之 DEA 還要好，證明了我們提出來的 TDEA 及 ATDEA 較優。

$H_2(s)$  分別表示如下：

$$H_2(s) = \frac{0.0795276(s + 1.0000000)e^{-0.70105502s}}{s^2 + 0.88362603s + 0.12605465} \quad (4-4)$$

$$H_2(s) = \frac{0.0266114(s + 6.10646821)e^{-0.4894422s}}{s^2 + 1.65105264s + 0.25757166} \quad (4-5)$$

$$H_2(s) = \frac{0.00495562(s + 35.0505313)e^{-0.39077937s}}{s^2 + 1.7606862s + 0.27531759} \quad (4-6)$$

$$H_2(s) = \frac{0.00415886(s + 65.9114728)e^{-0.5907097s}}{s^2 + 2.7140333s + 0.43448690} \quad (4-7)$$

$$H_2(s) = \frac{0.0025538(s + 68.1706405)e^{-0.3787929s}}{s^2 + 1.7606862s + 0.2759465} \quad (4-8)$$

$$H_2(s) = \frac{0.07691638(s + 1.0000000)e^{-0.72751985s}}{s^2 + 0.78112629s + 0.12191563} \quad (4-9)$$

$$H_2(s) = \frac{0.02550000(s + 6.50900000)e^{-0.63850000s}}{s^2 + 1.48000000s + 0.26308437} \quad (4-10)$$

$$H_2(s) = \frac{0.07200000(s + 11.01876566)e^{-0.95597987s}}{s^2 + 6.95911268s + 1.25749433} \quad (4-11)$$

$$H_2(s) = \frac{0.36000000(s + 30.14500000)e^{-1.04934638s}}{s^2 + 93.13892480s + 17.20118560} \quad (4-12)$$

$$H_2(s) = \frac{0.02710000(s + 4.70065000)e^{-0.47905000s}}{s^2 + 1.18575000s + 0.20191464} \quad (4-13)$$

由表 4-1 比較後可得知，本文所提出的 ATDEA 中的 ATDEA2 所得到的性能指標 J 值，不管在平均值或是最佳值來講，均比[9]還要好。

## 五、結論

本文所提出的 TDEA 及 ATDEA 方法，改善了傳統 DEA 的缺點，藉由田口直交表的方式挑選出突變的最佳操作方式，並且使用了搜尋空間自動擴大機制，讓搜尋能力的速度以及尋找最佳值皆能提升。在第四部份中，利用 TDEA 及 ATDEA 的方法應用在模式簡化，使得降階後系統的特徵能夠與原來系統相似不遠，或許傳統 DEA 也可以做到此結果，但在機率方面顯得比 TDEA 小很多。

總體而言，TDEA 的效率確實非常卓越，因此對於傳統 DEA 相關的應用，使用 TDEA 或 ATDEA 預期將會有實際提升效果的可能性。

表 4-1 二階時延系統  $H_2(s)$  之 TDEA 及 ATDEA 與[9]比較表

搜尋空間 $x^4$	Cheng[9]		TDEA & ATDEA			
	簡化模型 $H_2(s)$	性能指標 J (最佳值)	最佳簡化 模型 $H_2(s)$	性能指標 J (平均值)	性能指標 J (最佳值)	性能指標 J (標準差)
$[0,1]_4$	(4-4) 式	$1.1057 \times 10^{-2}$	(4-9) 式	$2.9610 \times 10^{-3}$	$1.9004 \times 10^{-3}$	$6.450 \times 10^{-4}$
$[0,10]_4$	(4-5) 式	$9.1112 \times 10^{-3}$	(4-10) 式	$2.9824 \times 10^{-3}$	$1.6937 \times 10^{-3}$	$8.0268 \times 10^{-4}$
$[0,50]_4$	(4-6) 式	$9.5698 \times 10^{-3}$	(4-11) 式	$9.4299 \times 10^{-3}$	$8.8695 \times 10^{-3}$	$5.4469 \times 10^{-4}$
$[0,100]_4$	(4-7) 式	$1.1318 \times 10^{-2}$	(4-12) 式	$1.0790 \times 10^{-2}$	$1.0346 \times 10^{-2}$	$2.7844 \times 10^{-4}$
adjustable	(4-8) 式	$9.1953 \times 10^{-3}$	(4-13) 式	$3.9720 \times 10^{-4}$	$1.2781 \times 10^{-4}$	$1.4641 \times 10^{-4}$

## 參考文獻

- [1] A. J. Hugo, P.A. Taylor and J. D, Wright, 1996, "Approximate dynamic models for recycle systems", *Ind. End. Chem. Res.*, vol. 35, no. 2, pp.485-487.
- [2] Y. S. Zhong, 2001, "Extreme stability margins of interval systems", *IEE Proceedings on Control Theory and Applications*, vol.148, no. 1, pp.77-80.
- [3] C. S. Hsieh and C. Hwang, 1989, "Model reduction of continuous-time systems using a modified Routh approximation method", *IEE Proceedings on Control Theory and Applications*, vol.136, no. 4, pp.151-156.
- [4] X. Tai, H. Zhang and Y. Sun, 2006, "A minimum output information loss method for stochastic model reduction", *IEEE Proceedings of 6<sup>th</sup> World Congress on Intelligent Control and Automation*, pp. 1186-1190.
- [5] P. Houlis and V. Sreeram, 2006, "A parametrized controller reduction technique", *IEEE Proceedings of 45<sup>th</sup> on Decision and Control*, pp. 3430-3435.
- [6] X. X. Huang, W. Y. Yan and K. L. Teo, 2001, " $H_2$  near-optimal model reduction", *IEEE Trans. Automat. Contr.*, vol. 46, no. 8, pp. 1279-1284.
- [7] A. Ferrante, W. Krajewski, A. Lepschy and U. Viaro, 1999, "Convergent algorithms for  $L_2$  model reduction", *Automatica*, vol. 35, pp. 75-79.
- [8] S. L. Cheng and C. Hwang, 2001, "Optimal approximation of linear systems by a differential evolution algorithm", *IEEE Trans. Sys., Man and Cyber.*, vol. 31, no.6, pp. 698-707
- [9] A. Megretski, 2006, "H-infinity model reduction with guaranteed suboptimality bound", *IEEE Proceedings of the 2006 American Control Conference*, pp. 448-453.
- [10] K. M. Grigoriadis, 1995, "Optimal  $H^\infty$  model reduction via linear matrix inequalities: Continuous- and discrete-time cases", *Syst. Contr. Lett.*, vol. 26, no. 5, pp. 321-333.
- [11] J. F. Leu and C. Hwang, 2001, "Optimal  $L^\infty$  model reduction using genetic algorithms", *J. of the Chinese Institute of Engineers*, vol. 24, no. 5, pp. 607-618.
- [12] D. Kavranoglu and M. Bettayeb, 1994, "Characterization and computation of the solution to the optimal  $L^\infty$  approximation problem," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1899-1904.
- [13] R. Storn, 1997, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces", *J. Global. vol. 11*, pp. 341-359.
- [14] R. Storn and K. Price, 1995, "Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces", *Technical Report TR-95-012*, International Computer Science Institute, Berkeley, USA.
- [15] D. E. Goldberg, 1989, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Massachusetts.