國立臺灣師範大學

資訊工程研究所碩士論文

指導教授：蔣宗哲　教授

參數調整機制於多目標演化式演算法

之效能剖析

A numerical study on parameter control

mechanisms in MOEA/D-AMS

研究生：林裕傑　撰

中華民國　　101　年　7　月

國立臺灣師範大學資訊工程研究所
博碩士學位論文

參數調整機制於多目標演化式演算法之效能剖析

A numerical study on parameter control mechanisms in MOEA/D-AMS

經考試合格特此證明

審查教授：

指導教授：

系主任：

中華民國 101 年 7 月

中文摘要

在現實生活中，我們常常需要解決一些具有多個目標需要考量的問題，並且這些目標通常是互相衝突的，這些問題稱為多目標問題，而多目標最佳化問題的目標便是找出能最佳化這些目標的解集合。演化式演算法 (evolutionary algorithm) 是求解這類問題的常見演算法，其概念為利用族群演化的方式來尋找最佳解集合。MOEA/D 為其中一種知名的演算法，利用將多目標問題拆成單目標來求解的作法可以獲得良好的結果，而 MOEA/D-AMS 與 MOEA/D-APC 便是以該演算法為基礎所改良，其中 MOEA/D-APC 參考了差分演化 (differential evolution) 產生子代的作法，該演算法擁有兩個控制參數 F 與 CR，這兩個參數值是影響子代品質的關鍵，因此 MOEA/D-APC 加入了讓參數隨演化過程調整的機制，經過實驗證明效能有所改善，但仍然在少部分問題上輸給其他的 DE 演算法。

本論文挑出八個具有不同參數調整機制的 DE 演算法，利用 MOEA/D-AMS 為主體分別結合這八種演算法與 MOEA/D-APC 的參數調整機制，藉由對 17 個測試問題進行實驗與分析，討論不同調整機制對效能的影響，並將主要目標放在探討 MOEA/D-APC 的弱項及改進方案上。

關鍵詞：多目標最佳化問題，演化式演算法，差分演化，參數調整機制。

目錄

附圖目錄

附表目錄

第一章 緒論

1.1 多目標最佳化問題

在現實生活中，我們常常需要在多個互相衝突的目標間對一件事物做出決策，例如當商人在生產產品時，必須同時考慮如何提高商品的品質與產量，又能維持低廉的成本，像這類的問題便可稱為多目標最佳化問題 (Multiobjective Optimization Problem, MOP)。

一個多目標最佳化問題可以用下式來表示：

$$\text{Minimize } F(x) = (f_1(x), \cdots, f_m(x)) \quad (1)$$

$$\text{Subject to } \boldsymbol{x} \in \Omega$$

Ω 是決策空間 (decision variable space)，R^m 是目標空間 (objective space)，$F: \Omega \to R^m$ 由 m 個實數目標函式組成，如果 Ω 在 R^m 上是一個封閉且連續區域，而且 x 的所有目標都是連續型的，我們可以稱(1)是一個連續型的多目標問題 (MOP)。圖 1-1 為一個二維決策空間映射到三維目標空間的多目標問題。

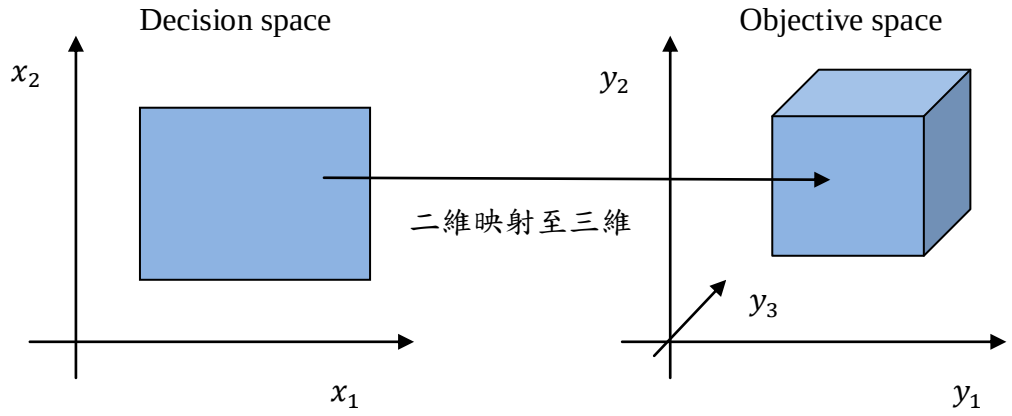

圖 1-1 決策空間(左)及目標空間(右)示意圖

假設 $u = (u_1, \cdots, u_m)$，$v = (v_1, \cdots, v_m)$ 是兩個屬於 R^m 的向量，如果對所有的 $i = 1, \cdots, m$，$u_i \leq v_i$，而且至少存在一個 j 屬於 $\{1, \cdots, m\}$ 使得 $u_j < v_j$，則稱 u 凌越 （dominate ）v，如圖 1-2 所示。

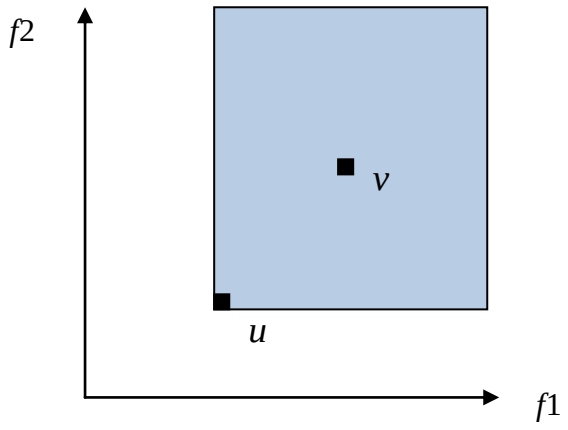

圖 1-2　u 凌越 v 之關係示意圖(最小化問題)

如果不存在一個 $x \in \Omega$ 使得 $F(x)$ 凌越 $F(x^*)$，則稱此點 $x^* \in \Omega$ 為柏拉圖最佳（*Pareto optimal*），或者是非凌越 （non-dominated）解，$F(x^*)$ 稱為柏拉圖目標向量；*Pareto set* 表示所有柏拉圖最佳解的集合，以 *PS* 簡稱之；*Pareto front* 表示所有柏拉圖目標向量的集合，簡稱為 *PF*，$PF = \{F(x) \mid x \in PS\}$。

在多目標最佳化問題裡，找到單一非凌越解並不困難，困難的是如何找到所有的非凌越解，也就是 *PS*。當我們無法找到完整 *PS* 時 (基於問題複雜度或者 *PS* 的大小)，我們希望盡可能地找到足以代表 *PS* 的近似解集合。

1.2 研究範疇

本論文為了比較多種不同參數調整機制的效能，選擇 2011 年陳政南的研究所畢業論文所實作的演算法 MOEA/D-APC（A Multiobjective Evolutionary Algorithm

with Adaptive Parameter Control） [1] 來做修改，該演算法是改良自 MOEA/D-AMS（Multiobjective Optimization Using MOEA/D with a New Mating Selection Mechanism） [2] 演算法。MOEA/D-APC 會針對不同問題來調整演算法的參數設定，改善了原演算法對特定型態問題的求解效率之限制，而本論文的研究目標為加入各種演算法的參數調整機制至 MOEA/D-AMS 內，並統整其實驗結果來比較目前這些較為知名的演算法在調整參數的策略上孰優孰劣，並藉此討論未來在參數調整機制上可考慮的發展方向。

第二章 文獻探討

目前已有許多針對求解多目標問題而生的演算法，在這一章我們將從一些較早期的演算法開始，談到作為我們演算法源頭的 MOEA/D 與 DE，而 DE 的部份因為種類繁多，將依照陳政南[1] 所使用的分類方式來簡單介紹我們將要比較的幾種演算法。

2.1 MOEA/D 與 DE

2.1.1 MOEA/D（A Multiobjective Evolutionary Algorithm Based on Decomposition）[3]

MOEA/D 的重點在於分解與鄰居，分解 (Decomposition) 指的是將多目標問題拆成許多單目標問題來解決，子問題通常會偏重某些目標的演化，藉此讓各個子問題朝不同方向發展來獲得均勻分布的解集合。將問題分解成子問題的方法有許多種，此論文提到了權值整合 （Weighted Sum Approach）、柴比雪夫（Tchebycheff Approach）、邊界交叉（Boundary Intersection Approach）三種方法，其中最重要的是柴比雪夫法，可以下式表示（假設多目標問題為最小化問題）：

$$\text{minimize } [\, g^{te}(x \mid \lambda, z^*) = \max_{1 \le i \le m} \{\lambda_i | f_i(x) - z_i^* |\}] \quad (2)$$

Subject to $x \in \Omega$

$z^* = (z_1^*, \ldots, z_m^*)^T$ 是參考點，由目前各目標上所找到的最佳值所產生，$z_i^* = \min\{f_i(x) | x \in \Omega\}$，$i = 1,..,m$，$\lambda$ 為一組權重值，$\lambda = (\lambda_1, \ldots, \lambda_m)$，$\lambda_i \geqq 0$，並且一組權

重值之和為 1。此式子的意義為，讓每個個體能以不同的方向去靠近參考點。除了分解之外，MOEA/D 還使用了鄰居 (neighborhood) 的觀念來進行交配。每個個體會以互相之間的權重距離來判斷是否為鄰居，距離越遠越不容易被選為鄰居，並且在交配時會從鄰居之中選擇親代，若產生之子代優於親代則可以直接取代。

2.1.2 Differential Evolution（DE） [4]

DE 在突變時會隨機選擇三個不同的個體 x_{r1}、x_{r2}、x_{r3}，並且 $x_{r1}\neq x_{r2}\neq x_{r3}$，利用這三個個體與一個突變參數 F 可以產生一個新的突變個體（$\overline{x_i}$），由下列式子表示:

$$\overline{x_i}=x_{r1}+F(x_{r2}-x_{r3}) \tag{3}$$

新的個體的每個基因位置，會受到交配機率（CR）的影響隨機選擇突變基因或是親代基因，可由下列式子表示:

$$y_{ij}=\begin{cases}\overline{x_{ij}} & , \quad \text{if } rand<CR \text{ or } j=I \\ x_{ij} & , \quad otherwise\end{cases} \tag{4}$$

$rand\in U(0,1)$

i 為個體在族群中之編號，j 為個體內基因之位置，y_{ij} 表示第 i 個子代的第 j 個基因，I 是隨機選出的一個基因位置，I 的設定可以讓新的子代至少有一個基因來自突變個體，目的是確保最後產生的子代跟親代不會完全相同。由式子（3）與（4）可知，影響子代品質之關鍵為 F 與 CR 參數，其意義會在本節末進行說明。除此之外，產生突變個體的突變策略也是變因之一，藉由選擇不同的親

代可造成各種不同的效果，式子（3）即為 DE 的其中一種突變策略，稱為 DE/rand/1/bin，接下來將解說突變策略的格式並介紹其他幾種具有不同特性的策略。

$$\overline{x_i}=\text{base}+F\times\text{difference vector} \tag{5}$$

上列式子表示 DE 的突變基因由基底 (base) 和差異向量 (difference vector) 所組成，基底和差異向量有各種不同的選擇方式，一般 DE 策略可以用這樣的格式表示：DE/*x*/*y*/*z*，*x* 代表基底選擇方式，*y* 代表差異向量的個數，而 *z* 為交配的方法，藉由這種表示法可以快速看出一個 DE 策略的特性。以下將就 *x*、*y*、*z* 這三部份的種類進行介紹。

基底選擇方式

較常見的基底選擇方式有 rand、rand to best、current to best 三種，rand 代表會隨機從族群中挑一個體作為基底，rand to best 則是利用隨機個體與目前最佳個體中的值作為基底，而此值靠近最佳解的多寡是依照 *F* 值來決定。current to best 和 rand to best 相似，但是並非使用隨機個體，而是用將與突變生成子代進行交配的個體。除了這些種類，有些演算法也會另外設計一些變體，如 JADE 的 current to pbest。下面三個例子分別代表基底是以 rand、rand to best、current to best 來選擇時的情形。

rand：$\overline{x_i} = x_{r1} + F \times$差異向量，$x_{r1}$ 為隨機挑選

rand to best：$\overline{x_i} = x_{r1} + F(x_{best} - x_{r1}) + F \times$差異向量，$x_{r1}$ 為隨機挑選

current to best：$\overline{x_i} = x_i + F(x_{best} - x_i) + F \times$差異向量，$x_i$ 為目前個體

差異向量的個數

有些突變策略著重於提升突變後的差異量，以增加演化的多樣性，有些則希望子代能出現在最佳解附近，這些都可以靠差異向量來調整。每個差異向量需要兩個不同的個體之差乘以 F 值來計算，並且每個差異向量所挑選的個體也都必須不同。下面兩個例子分別是差異向量個數為 1 和 2 的情形。

DE/rand/1 ：$\overline{x_i} = x_{r1} + F(x_{r2} - x_{r3})$

DE/rand/2 ：$\overline{x_i} = x_{r1} + F(x_{r2} - x_{r3}) + F(x_{r4} - x_{r5})$

交配方法

交配方法決定了個體內進行交配的基因位置，分為二項式 (binomial) 與冪次 (exponential) 兩種，在突變策略中用 bin 或 exp 表示。二項式代表所有基因都有機會進行交配，而冪次則僅限於其中一段位置。

除突變策略外，F 和 CR 也是兩個對 DE 而言很重要的參數，F 值可以控制差異向量的比重，也就是突變後和親代的差異程度，值越大則差異越大，而 CR 值負責控制交配機率，其大小影響選擇突變基因的多寡。許多 DE 演算法的變型皆是對此兩個控制參數進行修改，這些參數調整機制分為許多種類，見 2.2 節。

2.2 參數調整機制分類

陳政南將各種 DE 的參數調整機制依照數值分布、族群參數個數及參考資訊範圍來分類，藉此我們可以較容易的理解各個演算法的概念與目的。以下將說明其分類方式。

2.2.1 數值的分布方式

數值的分布方式分為連續數值和離散數值，連續數值表示參數值的範圍為一段連續實數，可能是其中的任何一值，具有多樣性較高的特點，而離散數值分布的參數值範圍則為預先設定好的離散數值，通常是先進行實驗再依據其結果來進行設置。

2.2.2 族群參數個數

族群的參數個數分成個別參數（Individual parameter）和單一參數（Single parameter），個別參數表示每個個體都擁有自己的參數，比較能視個體自己的演化狀態決定參數的調整方式，而單一參數指的是整個族群共用一個參數，在應對不同演化狀態時的靈敏度可能就沒有單一參數來得好。

2.2.3 參考資訊的範圍

有些參數調整機制需要使用某些特定的資訊才能對參數進行修正，而依照這些資訊參考的範圍又能分為沒有資訊、個體資訊、親代資訊以及群體資訊四種，沒有資訊表示不需要參考任何東西便能產生新的參數，大部份是使用一定範圍內

隨機值的方式，優點是多樣性較高，但參數可謂不具調整性，而隨機值的範圍通常是經過反覆實驗來做出一個整體效能較好的設定。個體資訊表示需要參考個體演化的情形來調整參數，有些演算法會讓個體擁有自己的參數組，此時便能讓個體在演化時觀察自己過往的演化情形來修正參數，缺點是個體無法利用其他個體的資訊來即時應對當代的演化，通常需要多代累積的資訊才足夠進行調整。親代資訊的參數值多由許多個親代提供的資訊來產生，讓這些資訊跟著個體一起進行演化以達到去蕪存菁的效果，而群體資訊則是使用整個族群的資訊，例如整個族群中產生優良子代的次數，像這類的資訊因樣本多，好的參數值也能擴大影響範圍而不侷限於個體，通常較為可靠。本論文在第二章所要介紹的十種演算法，主要集中於連續數值-個別參數的部份。四種不同的參考資訊範圍可以圖 2-1 表示。

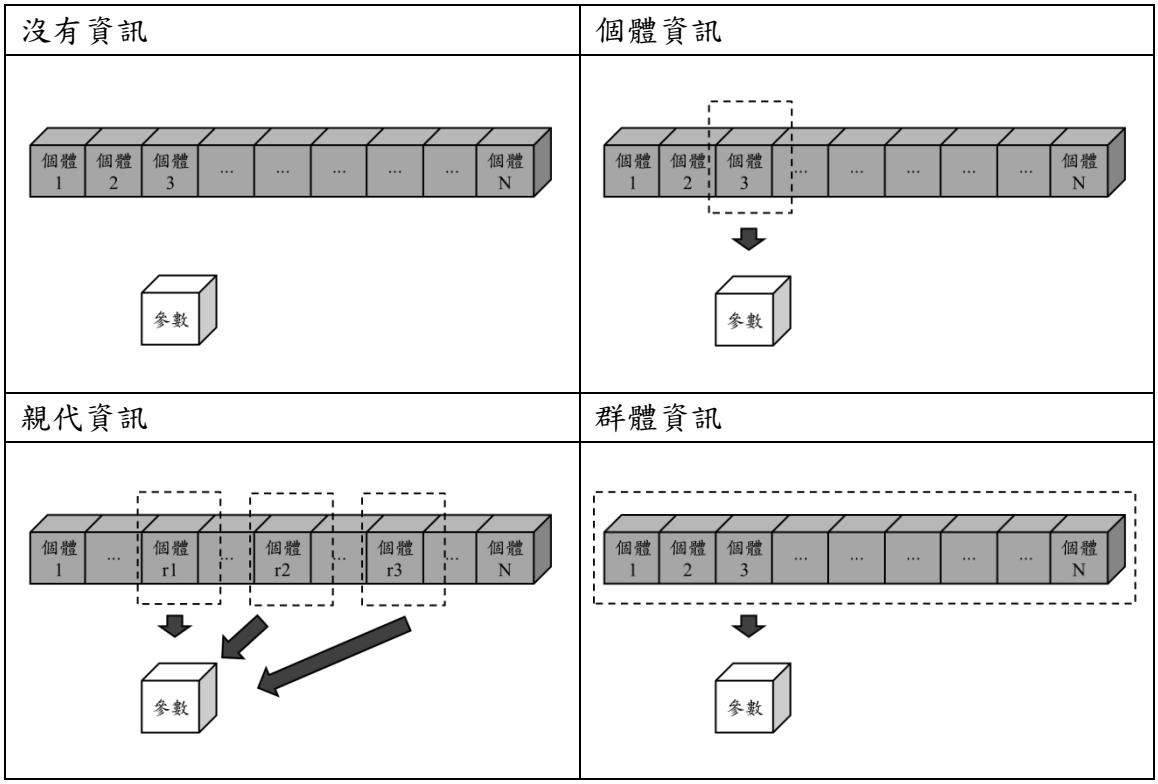

圖 2-1 參考資訊範圍示意圖（陳政南, 2011）

2.3 具參數調整機制之演化式演算法

2.3.1 連續數值-個別參數-沒有資訊

2.3.1.1 Neighborhood Search DE (NSDE) [5]

NSDE 演算法提出了鄰居搜尋 (neighborhood search) 的概念，利用親代作為基底來搜尋附近的區域以達成演化，而對這部份影響最深的便是差異向量的計算，其計算式如下:

$$F_i = \begin{cases} N(0.5,0.5) & , \text{ if } U(0,1)<0.5 \\ C(0,1) & , \text{ otherwise} \end{cases} \tag{6}$$

$$CR_i = U(0,1)$$

i 為個體在族群中編號，$N(0.5,0.5)$ 為常態分布 (normal distribution)，$C(0,1)$ 為柯西分布 (Cauchy distribution)，$U(0,1)$ 則為均勻分布的隨機值，兩種隨機分布的範圍設定為依據實驗結果而產生。在 NSDE 中，每個個體有自己的 F 與 CR 值，並且由式子可知道，NSDE 分別有一半的機會使用常態分布與柯西分布來產生 F 值，常態分布的特點為分布較均勻，因此突變後產生的子代跟親代之距離也會較平均；而使用柯西分布時則會讓F值的差異變大，這可以使個體間的多樣性增強，利用這兩種分布使個體能夠利用不同的差異向量來進行移動。由於 NSDE 沒有使用多餘的資訊來調整參數，所以沒辦法靠著前代的演化來給予優勢，不過也因此其突變可能性較大，較不會被區域最佳解困住。

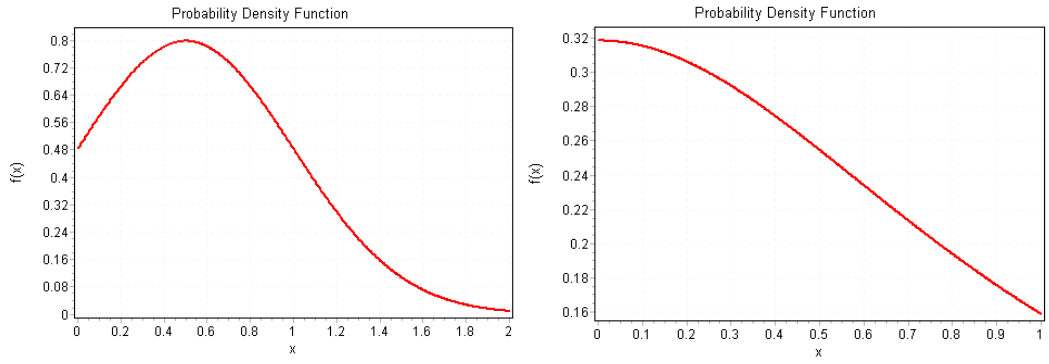

圖 2-2 常態分布與柯西分布的機率密度函式圖，左圖為 N(0.5,0.5)，右圖為 C(0,1)。

2.3.2 連續數值-個別參數-個體資訊

2.3.2.1 Self-Adapting Control Parameters in DE（jDE）[6]

jDE 的概念為，好的參數值應該能產生更多好的個體，因此讓每個個體有自己的一組參數值，產生優良子代的參數值便有機會跟著個體演化下去，否則就隨著個體一起淘汰掉。其參數調整方法見下式：

$$F_{i,G+1}=\begin{cases} F_l + rand_1 \times (F_u - F_l) & , \text{ if } rand_2<\tau_1 \\ F_{i,G} & , \text{ otherwise} \end{cases} \qquad (7)$$

$$CR_{i,G+1}=\begin{cases} rand_3 & , \text{ if} rand_4<\tau_2 \\ CR_{i,G} & , \text{ otherwise} \end{cases}$$

G 表示目前的演化代數，G+1 則為下一代，$F_{i,G}$ 表示第 G 代的第 i 個個體的 F 參數。$rand_1 \sim rand_4$ 為範圍 [0,1] 的均勻分布隨機值，並且其值不重覆。τ_1 與 τ_2 分別為決定 F 與 CR 參數是否沿用上一代的門檻值，若產生的隨機值小於該門檻值，則 F 與 CR 參數會在其合理範圍內產生一隨機數作為新的值。jDE 藉著 τ_1 與 τ_2 兩個門檻值來控制 F 與 CR，其意義為我們有多少機率使用舊的參數值來進行突變。利用這樣的做法，既能嘗試有效利用優良的參數值，也能確保一定程度的多樣性。

2.3.2.2 DE algorithmwith Self-adaptive Strategy and Control Parameters (SspDE)[7]

SspDE 跟 jDE 類似，都是使用成功經驗來加強後代的演化，希望能讓優良參數的影響力繼續延伸，主要是著重在突變策略的調整。以下為其參數的調整方式：

$F_i \in [0.1, 1]$

$CR_i \in [0, 1]$

$Strategy_i \in$ {rand/1/bin, rand to best/2/bin, rand/2/bin, current to best/1/bin}

上面的式子表示了 F、CR 與突變策略三個參數的範圍定義，較特別的是它一次考慮了四種突變策略。有別於一代一代的進行調整，SspDE 會使用三組長度為 50 的串列來存放 F、CR 與突變策略三種參數值，初始化時會分別依照設定範圍產生隨機值填滿串列。在演化過程中，會按順序使用串列中的參數組來演化，並記錄產生之子代與親代的優劣情形，每 50 代會更新一次串列，此時曾經產生優良子代的參數組便會保留，不足的部份再以跟初始化相同的方式填入隨機值，之後打亂順序並進行下 50 次的演化。

2.3.3 連續數值-個別參數-群體資訊

2.3.3.1 Self-adaptive DE (SaDE)[8]

SaDE 也是使用多突變策略的演算法，分別為 DE/rand/1/bin 與 DE/current to best/2/bin，前者的隨機性可提供族群探索空間的可能性，後者則對增加收斂度有所幫助，而使用哪種策略則是由是否產生優良子代的次數來決定，其算法見下式：

$$\text{突變策略} = \begin{cases} \text{DE/rand/1/bin} & , \quad \text{if } rand_1 < p_1 \\ \text{DE/current to best/2/bin} & , \quad \text{otherwise} \end{cases} \tag{8}$$

$$p_1 = \frac{ns1 \cdot (ns2 + nf2)}{ns2 \cdot (ns1 + nf1) + ns1 \cdot (ns2 + nf2)}$$

p_1 代表使用 DE/rand/1/bin 的機率，$ns1$ 與 $ns2$ 分別代表 DE/rand/1/bin 與 DE/current to best/2/bin 兩個策略在學習時間（learning period，在此設定為50代）內成功產生優良子代的次數，$nf1$ 與 $nf2$ 則是失敗的次數。一開始 p_1 會設為0.5，在經過學習時間，也就是每50代演化後，會依上式重新計算並將 $ns1$、$ns2$、$nf1$ 與 $nf2$ 歸零，因此兩種策略的使用機率會從一開始相同的 50% 隨著演化而增減。

除了突變策略，SaDE 的 CR 值也有其調整機制，其算法見下式：

$$F = N(0.5, 0.3)$$

$$CR = N(CRm, 0.1)$$

F 值為平均數 0.5，標準差 0.3 的常態分布隨機值，其範圍為 (0, 2]，在一開始不需要特別設定。CR 的算法較特別，為一個以 CRm 為平均值，標準差 0.1 的常態分布隨機值，而 CRm 代表所有產生過優良個體的 CR 值之平均，如同突變

策略的調整，在經過固定代數的演化後會進行 *CRm* 的統計（此處設定為 25 代），而 *CR* 則是每 5 代便重新計算一次，藉此讓 *CR* 值依過往的優良經驗移動來提升效能。

2.3.3.2 Self-Adaptive DE Algorithm in Constrained Real-Parameter Optimization（jDE-2）[9]

jDE-2 對 jDE 進行了改良，從原本只使用一個突變策略改為兩個，分別為 DE/rand/1/bin 和 DE/current to best/1/bin，因此每個個體所擁有的參數組也變為兩組，如下圖所示：

	突變策略1		突變策略2	
個體1	F_1^1	CR_1^1	F_1^2	CR_1^2
個體2	F_2^1	CR_2^1	F_2^2	CR_2^2
⋮				
個體n	F_n^1	CR_n^1	F_n^2	CR_n^2

圖 2-3 jDE-2 個體與參數關係示意圖。

至於如何選擇突變策略，jDE-2 使用和 SaDE 一樣的作法，利用成功產生優良子代的次數來決定是否提高其中一種策略的使用機率。

在 *F* 和 *CR* 值的調整方面，跟 jDE 的作法相同，只是多了另一組參數組需考慮而已，見下式：

$$F^{s}_{i,G+1}=\begin{cases}F_l+rand_1\times(F_u-F_l) & , \text{ if } rand_2<\tau_1\\ F^{s}_{i,G} & , \text{ otherwise}\end{cases} \quad (9)$$

$$CR^{s}_{i,G+1}=\begin{cases}rand_3 & , \text{ if } rand_4<\tau_2\\ CR^{s}_{i,G} & , \text{ otherwise}\end{cases}$$

s 即代表哪一種策略，F^1_i 表示第i個個體的第一種策略 （DE/rand/1/bin）所使用的 F 參數值。在演化次數達到 l（一般設為1000）代後，jDE-2 會將族群中最差的前 k（一般設為70）個個體在合理範圍內以均勻分布之隨機值初始化，目的是將那些對求解問題效能不太有貢獻的個體去除，嘗試以加入新成員的方式推動演化的狀態。

2.3.3.3 Self-adaptive Neighborhood Search DE(SaNSDE) [10]

此演算法融合了 NSDE 與 SaDE，藉由 SaDE 對 CR 與突變策略的調整能力，來改善 NSDE 對於參數只有隨機性而沒有調整機制的缺點。在突變策略方面，SaNSDE 的作法與 SaDE 相同。對於 F 值的處理，NSDE 原本有各半的機率使用常態分布與科西分布，而 SaNSDE 則將該機率以類似 SaDE 的方式調整之，也就是說必須記錄突變出優良子代的 F 值是由常態分布還是科西分布產生的，才能以其成功及失敗的次數來對之後的F值進行調整，見下式：

$$F_i=\begin{cases}N(0.5,0.3) & , \text{ if } U(0,1)<fp\\ C(0,1) & , \text{ otherwise}\end{cases} \quad (10)$$

CR 值的算法也從 SaDE 處經過了修改，除了 CR_m，還需要 CR_{rec} 來記錄在 25 代之內產生過優良子代的 CR 值，具體算法見下式：

$$CR_i = N(CR_m, 0.1) \tag{11}$$

$$CR_m = \sum_{k=1}^{|CR_{rec}|} w_k \times CR_{rec}(k)$$

$$w_k = \frac{\Delta f_{rec}(k)}{(\sum_{k=1}^{|\Delta f_{rec}|} \Delta f_{rec}(k))}$$

$$\Delta f_{rec}(k) = f(k) - f_{new}(k)$$

$\Delta f_{rec}(k)$ 代表新值與原值的適應度之差，w_k 則是算出當前個體的適應度差在整體裡佔多少比重，將 CR_{rec} 內存放的每個 CR 值乘上自己的比重，即求出 CR_m，其概念為即使同樣產生優良子代，這些 CR 值可能還是有優劣之分，故捨棄原本單純算出平均值的方式，用比重來提高一些效果較好的 CR 值之影響力。

2.3.3.4 Adaptive DE with Optional External Archive （JADE） [11]

JADE 使用了算術平均 (arithmetic mean) 與勒莫平均 (Lehmer mean) 來保留優良的參數值，並且會和上一代所使用的參數做一定比例的結合來達到參數調整的目的，其 CR 與 F 值算法如下述。

$$CR_i = randn_i(\mu_{CR}, 0.1) \tag{12}$$

$$\mu_{CR} = (1 - c) \times \mu_{CR} + c \times \text{mean}_A(S_{CR})$$

S_{CR} 代表成功產生過優良子代（優於親代）的 CR 值之集合，$\text{mean}_A(S_{CR})$ 為其算術平均，c 之範圍介於0與1，可決定參考過往 CR 值的比例，μ_{CR} 之初始值設為0.5，之後每代演化結束後會依上式進行更新，再藉著常態分布來算出 CR_i。因為記錄了 S_{CR}，即使演化結束了，優良的CR值依然能繼續發揮一定的影響力來對之後的 CR 進行調整。

$$F_i=randc_i(\mu_F, 0.1) \tag{13}$$

$$\mu_F=(1-c)\times\mu_F + c\times \text{mean}_\text{L}(S_F)$$

$$\text{mean}_\text{L}(S_F)=\frac{\sum_{F\in S_F}F^2}{\sum_{F\in S_F}F}$$

如同 CR 值，F 值的計算也參考了前代演化的結果。S_F 為成功產生優良子代的 F 值之集合，$\text{mean}_\text{L}(S_F)$ 為其勒莫平均，算出 μ_F 後可由科西分布得出 F 值。因為越大的 F 值越能提高演化率，而太小的 μ_F 值可能產生過早收斂之情形，並且勒莫平均產生之數值平均來說大於算術平均，所以在計算 F 值的時候使用的是勒莫平均。

2.3.3.5 Self-adaptive Multi-objective DE with Direction Information Provided by Archived Inferior Solutions （JADE2） [12]

為求解多目標最佳化問題，Zhang 與 Anderson 之後又發表了 JADE2 演算法，主要是對個體的選擇與交配行為進行修改。如同之前提到的凌越概念，在問題目標數大於等於二的時候，個體間是有可能產生互不凌越的關係（例如全部的個體都在非凌越解集合內），此時我們需要別的機制來比較其優劣以進行選擇的步驟，JADE2 認為在同一個階級上，擁有較低密集度的個體有利於在之後的演化確保整個族群的分散度，所以使用了擁擠距離（crowding distance）來評估個體間的密度關係，其計算方式見下式：

$d_i = \prod_{j=1}^{k} d_{ij}$ (14)

$k = 2(M-1)$

d_{ij} 為個體 i 至最近的第 j 個個體間的歐基里德距離，M 為問題的目標數，d_i 為個體 i 之擁擠距離，值越小代表此個體附近的密集度較高，而我們會挑選密度較低者來進行演化。

至於突變的部份，因為族群在演化初期雖然能朝柏拉圖前緣逼近，但在一定代數演化後經常呈現收斂速度減緩而往其他方向發展的情形，此時我們以選擇機制挑出的優良個體可能變得不具有太大的影響力去帶動整個族群演化，因此 JADE2 另外將之前選擇時被淘汰的個體儲存起來，有機會再被用來計算差異向量，其算法見下式：

$$v_{i,g} = x_{i,g} + F\left[(x^{p}_{best,g} - x_{i,g}) + (\tilde{x}_{r1,g} - x_{r2,g})\right] \quad (15)$$

$x^{p}_{best,g}$ 代表從前 $p \times 100\%$ 個優良個體中隨機挑出的個體 （$p \epsilon (0,1]$），$\tilde{x}_{r1,g}$ 是從族群與被淘汰個體中隨機挑出之親代，$x_{r2,g}$ 則單純由族群中挑出，藉由考慮那些曾被淘汰的個體，來嘗試再次增強整個族群往柏拉圖前緣演化的動力。

2.3.3.6 Self-adaptive JADE (SaJADE) [13]

JADE 演算法只對 F 與 CR 參數進行了調整，而 SaJADE 與 JADE 的不同是加入了多種突變策略來使用，其選擇方式可由下式表示：

$S_i = \lfloor \eta_i \times k \rfloor + 1$

S_i 表示我們要使用的突變策略，k 表示策略的個數。關於 η_i 值的調整，

SaJADE 參考了 JADE，見下式：

$$\eta_i = \begin{cases} rndn_i(\mu_s, 1/6) & , \quad \text{if } g = 1 \\ rndn_i(\mu_s, 0.1) & , \quad \text{otherwise} \end{cases} \tag{16}$$

$$\mu_s = (1-c) \times \mu_s + c \times \text{mean}_A(H_s)$$

η_i 為一常態分布的隨機值，在第一代演化時會以 1/6 作為標準差，而 η_i 之平均值 μ_s 設為 0.5，使常態分布範圍剛好在 0 與 1 之間。$\text{mean}_A(H_s)$ 為產生優良個體的 η_i 值之算術平均，藉由 c 值，我們可控制這些優良參數的影響力，以此產生新的 η_i 值來決定該代用何種突變策略。

2.3.4 連續數值-個別參數-親代資訊

2.3.4.1 Self-adaptive DE（SDE）[14]

此演算法對參數調整機制做了一些簡單的修改，每個個體有自己的參數值，其初始值為隨機產生，主要是利用常態分布來調整 F 與 CR，見下式：

$$F_i = F_{i1} + N(0, 0.5) \times (F_{i2} - F_{i3}) \tag{17}$$

$$CR_i = N(0.5, 0.15)$$

$i1$、$i2$、$i3$ 為從族群中挑出的三個個體，利用其中兩者的 F 值之差乘上一個常態分布值，與第三個個體的 F 值相加即得到一個新的 F 值，希望藉此讓好的參數能彼此影響，使參數隨著個體朝最佳化發展，而 CR 則是一個常態分布值。

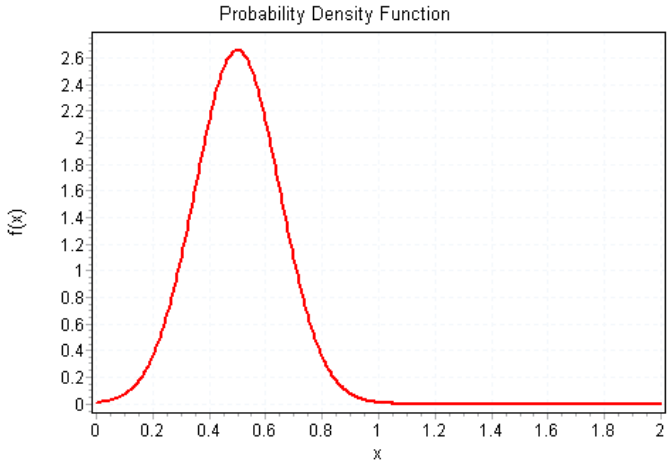

圖 2-4 常態分布 $N(0.5,0.15)$ 的機率密度函式圖。

2.3.5 連續數值-單一參數-群體資訊

2.3.5.1 Adaptive DE Algorithm for Multiobjective Optimization Problems

（ADEA） [15]

ADEA 對 F 參數進行調整的依據與其他機制較為不同，其調整方式可以下式表示：

$$F=\max\left(\frac{\sum_{j=1}^{k}\sum_{i=1}^{m}|d_{ij}-d_j|+df}{\sum|Q|\cdot\bar{d}+df},1-\frac{2\,|P|}{|Q|},\ l_{min}\right) \tag{18}$$

第一個參數代表族群的分布情形，值越大則代表個體在階級中的分布越不平均。在計算前需先對族群做分階（ranking）[16]與擁擠距離[16]計算，$|Q|$ 為族群個數，$\bar{d}$ 為平均擁擠距離，df 為邊界解的距離，d_{ij} 為第 j 階第 i 個個體的擁擠距離，$\bar{d}_j$ 為第 j 階的個體的平均擁擠距離，分子代表個體在階級中的分布情形，分母則為族群之擁擠距離。第二個參數代表目前族群中非凌越解的比例，$|P|$ 代表非凌越解之數目。第三個參數為 F 的最小值，用以確保 F 值不會過於逼近至

0。由於在初期演化時，我們尚未進行過夠多次的選擇來留下好的個體，族群的分布情形會較不平均，因此第一個參數的值會先呈現較高的狀態，隨著演化再慢慢降低。當演化代數增加，通常找到的非凌越解也會越來越多，所以第二個參數的值也會跟著減少，故整體來說 ADEA 對 F 值的調整也是屬於前高後低的趨勢，藉此強化前期族群演化的廣闊度。

第三章 方法與步驟

本章先介紹 MOEA/D-AMS 及加入參數調整機制的 MOEA/D-APC 演算法，再說明本論文計劃如何比較多種 DE 演算法的參數調整機制。

3.1 MOEA/D-AMS

以 MOEA/D 為基礎，MOEA/D-AMS 加入了新式的親代選擇機制，其中包括收斂評估機制、密集度評估機制以及交配池選擇機制，以下將簡述之。

3.1.1 收斂評估機制

演化中已非常接近柏拉圖前緣之個體，繼續佔用計算資源也很難對整體效能的提升有所貢獻，因此 MOEA/D-AMS 會記錄個體沒有更新的代數，達到 α 次則視為已收斂並且將資源分配給其他未收斂的個體使用。

3.1.2 密集度評估機制

為了使我們在求解問題時，最後得到的近似最佳解除了接近還能均勻的分散於柏拉圖前緣，MOEA/D-AMS 在每代開始繁殖前會計算個體間的擁擠距離來代表密集度，再確保密集度較分散的前 β 個個體必能分配到計算資源，以此來增加其演化次數，而使用擁擠距離的缺點為當許多密集度高的個體集中在一起，或是個體在各維目標上與相鄰個體之差值差異過大時，容易造成誤判而降低密集度。

3.1.3 交配池選擇機制

在 MOEA/D 演算法中，我們會選擇在目標空間中相似的個體來形成交配池，但這些個體在決策空間中未必相似，而決策空間中不相似的個體在交配後易產生與親代差異較大之子代，因為在演化後期與決策空間中相似之個體交配比較有機會產生可以更新效能的個體，所以在經過一定的演化代數後，啟動交配池選擇機制，將交配池成員置換成決策空間相似的解進而增加演算法效率。

3.1.4 MOEA/D-AMS 基本流程

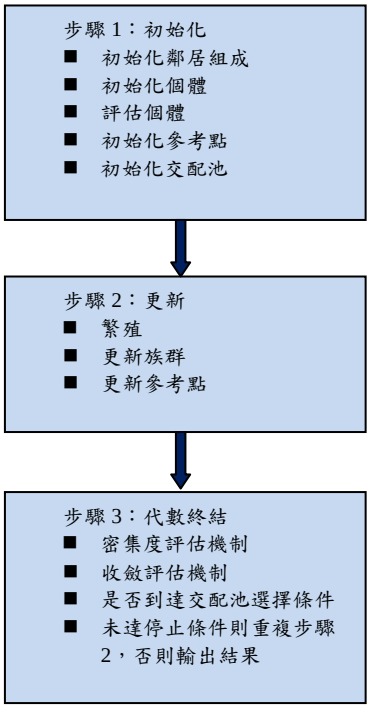

圖 3-1 MOEA/D-AMS 演算法流程。

3.2 MOEA/D-APC

MOEA/D-APC 演算法改良自 MOEA/D-AMS 演 算法。通常一個多目標演化式演算法不會只有一個參數，而且每個參數的功能都不相同，要調整所有的參數是複雜且龐大的工作，所以 MOEA/D-APC 挑選了對 MOEA/D-AMS 效能影響

最大的 DE 繁殖方法，對其參數進行調整。其中的參數調整機制包括演化過程的參數調整、參數值選擇以及族群收斂調整機制，以下將說明演算法流程與各調整機制的運作方式。

3.2.1 演化過程的參數調整

此處參考 SspDE 的參數調整機制，利用上升值 （U）、中間值（M）、下降值（L） 來對參數進行調整，每個值之間的差距相同，在 MOEAD-APC 中，初始的 *F* 值範圍為 {0.3,0.5,0.7}，0.5 便是中間值，0.3 與 0.7 分別為下降值與上升值，依靠這三個值來確立一個小範圍的搜尋區域，以此來找尋更好的參數值。建立一長度為族群大小之串列並填入相等個數的上升值、中間值、下降值，其餘則以中間值填滿，然後進行亂序排列（見圖 3-2 與圖 3-3）。每次的突變依序使用該串列中的值作為參數值，當突變的子代取代族群中的個體時，會將這個參數值記錄下來，經過一個代數的演化後再統計這三個參數值產生良好子代的次數，由產生最多良好子代的值作為下一代演化的中間值，上升值與下降值也跟著調整之。利用這樣的方式，參數會自行移動到效能較佳的值，而達到演化過程的參數調整之目的。

L	L	L	M	M	M	U	U	U	L

圖 3-2 參數串列示意圖。

U	L	M	M	L	L	U	M	L	U

圖 3-3 串列亂序示意圖。

3.2.2 參數值選擇

不同參數值產生優良子代的次數有可能會相同，但其效果會有差距，所以希望即使產生相同次數的優良子代，在參數值之間也能夠有判斷優劣的機制。許多文獻都提出不同的參數選擇方式，如 JADE 對於族群中產生優良子代的參數值計算莫勒平均，而莫勒平均所計算出的平均值比算術平均值還要大，所以 JADE 的參數調整機制會偏好於數值較大的 F 值，而 FADE [17] 則依據兩代之間在決策空間和目標空間的差距來判斷演化速度，兩代之間的差距大表示在演化初期，增加參數 F 的數值，差距小則表示演化已經接近收斂，適合將參數值降低。ADEA [15] 依據凌越關係將族群做階級區分，依照每個階級的個體擁擠距離來判斷演化狀態，擁擠距離不平均判斷為演化初期，適合較大的 F 值，而每個階級有較平均的擁擠距離，表示演化經過了多次的環境選擇，保留下來的個體分布較平均，則調整參數值為較小的數值。統整這些文獻的觀點，可知在演化初期以及效能較差的個體上，適合使用數值較大的參數值，而越接近收斂的狀態，適合將參數值降低，因此我們將三種參數值設定不同的比重，上升值設定較高的比重，下降值設定較低的比重，而中間值的比重不變，判斷參數值的好壞則由產生良好子代的次數乘以參數值比重為參考依據。

3.2.3 MOEA/D-APC 基本流程

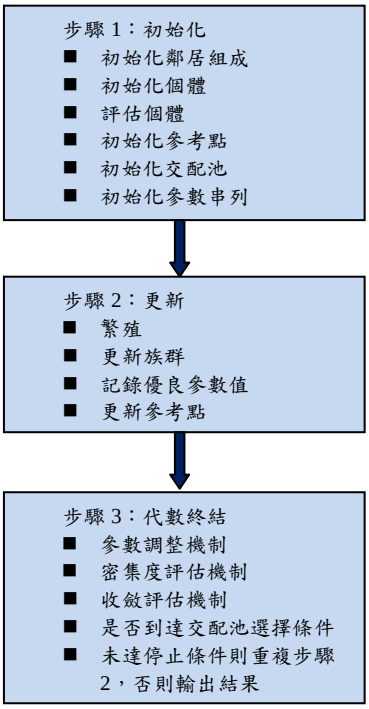

圖 3-4 MOEA/D-APC 演算法流程。

3.3 各演算法之參數調整機制比較

為了解在某些情況下表現優於 MOEA/D-APC 之演算法的特性，我們撰寫一個以 MOEA/D-AMS 為主體的實驗用程式，並挑選了八個不同演算法的參數調整機制與其結合並進行實驗，分別為 JADE、NSDE、SDE、SspDE、JDE、SaDE、SaNSDE 與 ADEA，藉由 17 個測試問題來比較這些機制的效能差異，並依結果來討論未來的研究方向。

此實驗用程式之流程與 MOEA/D-APC 相同，見圖 3-4。

步驟 1 初始化：

1.1 計算權重向量之間的歐基里德距離，再對每個權重向量 λ^I 找 T 個歐基里德距離最短的權重向量為其鄰居，記為 $B(i) = \{ i_1,\ldots, i_T\}$，而$\lambda^{i1},\ldots,\lambda^{iT}$ 即為 λ^i 的鄰居。

1.2 在決策空間產生均勻分布的初始族群 $\{x^1,\ldots, x^N\}$。

1.3 計算每個個體在目標空間的目標值 $\{F(x^1),\ldots, F(x^N)\}$。

1.4 依照族群在目標空間的目標值計算參考點 $z = (z_1,\ldots, z_m)$，而每個 z_i 為族群在第 i 個維度的最佳目標數值。

1.5 初始化個體交配池。

1.6 **視使用的參數調整機制不同，所需要初始化的參數資料也不一樣，例如 SspDE 就和 MOEA/D-APC 類似，需產生一組相同長度的 *F*、*CR*、DE 參數串列，而 NSDE 只需要為每個個體產生自己的 *F* 與 *CR* 值。**

步驟 2 更新：

循環地 (round robin) 挑選未收斂的個體執行下列工作，直到挑選 N 個個體為止:

2.1 選擇交配池：均勻分布的方式產生一個隨機數 $rand$，若 $rand<\delta$ 以 $B(i)$ 作為交配池，若 $rand>=\delta$ 以整個族群為交配池。

2.2 設定參數，此時視演算法而有不同的作法，如 SspDE 會依演化代數來使用個體自己擁有的參數串列上某個位置的參數值。

2.3 繁殖：由交配池中選擇親代進行突變，而突變策略依照各演算法指定的種類執行，有些演算法使用了一個以上的突變策略，像是 SaNSDE 就有機率使用 DE/rand/1/bin 或 DE/current to best/1/bin 兩種策略，此時若沒有在個體內預先指定好使用何種策略，便需依靠隨機值來決定。

2.4 更新族群：隨機由交配池中選擇一個個體，而當繁殖出來的子代優於這個個體時，則由子代取代之，而子代可以取代交配池中的個體最多 n_r 次，並將更新的狀況記錄給收斂評估機制使用。

2.5 有些演算法在調整參數時會使用到前代演化過程中的資訊，通常是依照有無產生比親代更優秀的子代來決定記錄與否，視使用的演算法不同，在這個步驟需記錄之後演化所需要的資訊。像 SaDE 便會記下產生良好子代的 *CR* 值，並且其平均值也會影響往後產生的 *CR* 值。

2.6 更新參考點：判斷更新族群後是否要更新參考點。

步驟 3 代數終結：

3.1 執行參數調整機制，此處是實驗程式最重要的部份，我們希望藉由在這個步驟實行不同演算法的參數調整機制，來觀察其產生的結果。有些預先設定好參數串列的演算法在固定代數後才會在此調整參數，而調整的方式則依各演算法的公式而定。

3.2 執行收斂評估機制。

3.3 執行密集度評估機制。

3.4 若演化代數已達到終止代數的 $\gamma\%$ ，將交配池 $M(i)$ 置換為與個體在決策空間中歐基里的距離小的為鄰居。

3.5 若達到終止條件則將 $\{F(x^1),\ldots,F(x^N)\}$ 輸出，即為近似最佳解，若未達終止條件，再執行步驟 2，直到達成中止條件為止。

第四章 實驗設計

4.1 測試問題

測試的 17 個問題為 DTLZ1-DTLZ2 [3]（2 個問題）、F1-F9 [18]（9 個問題）以及 UF4-UF7（4 個問題）、UF9-UF10 [19]（2 個問題），表為 17 個問題的定義:

表 4-1 多目標最佳化問題。

問題	多目標函式	數值範圍
F1	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-x_1^{0.5(1.0+\frac{3(j-2)}{n-2})})^2$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-x_1^{0.5(1.0+\frac{3(j-2)}{n-2})})^2$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$	$[0,1]^n$ n=30
F2 (UF1)	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-\sin(6\pi x_1+\frac{j\pi}{n}))^2$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-\sin(6\pi x_1+\frac{j\pi}{n}))^2$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ n=30
F3	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-0.8x_1\cos(6\pi x_1+\frac{j\pi}{n}))^2$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-0.8x_1\cos(6\pi x_1+\frac{j\pi}{n}))^2$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ n=30

表 4-1 多目標最佳化問題（續）。

問題	多目標函式	數值範圍
F4	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-0.8x_1\cos(\frac{6\pi x_1+\frac{j\pi}{n}}{3}))^2$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-0.8x_1\cos(6\pi x_1+\frac{j\pi}{n}))^2$ where $J_1=\{j\mid j$ is odd and $2\leq j\leq n\}$ and $J_2=\{j\mid j$ is even and $2\leq j\leq n\}$	$[0,1]\times[-1,1]^{n-1}$ n=30
F5 (UF2)	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}\{x_j-[0.3x_1\cos\left(24\pi x_1+\frac{4j\pi}{n}\right)+0.6x_1]\cos(6\pi x_1+\frac{j\pi}{n})\}^2$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}\{x_j-[0.3x_1\cos\left(24\pi x_1+\frac{4j\pi}{n}\right)+0.6x_1]\cos(6\pi x_1+\frac{j\pi}{n})\}^2$ where $J_1=\{j\mid j$ is odd and $2\leq j\leq n\}$ and $J_2=\{j\mid j$ is even and $2\leq j\leq n\}$	$[0,1]\times[-1,1]^{n-1}$ n=30
F6 (UF8)	$f_1=\cos(0.5x_1\pi)\cos(0.5x_2\pi)+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-2x_2\sin(2\pi x_1+\frac{j\pi}{n}))^2$ $f_2=\cos(0.5x_1\pi)\cos(0.5x_2\pi)+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-2x_2\sin(2\pi x_1+\frac{j\pi}{n}))^2$ $f_3=\sin(0.5x_1\pi)+\frac{2}{\lvert J_3\rvert}\sum_{j\in J_3}(x_j-2x_2\sin(2\pi x_1+\frac{j\pi}{n}))^2$ where $J_1=\{j\mid 3\leq j\leq n$,and $j-1$ is a multiplication of 3$\}$ and $J_2=\{j\mid 3\leq j\leq n$,and $j-2$ is a multiplication of 3$\}$ and $J_3=\{j\mid 3\leq j\leq$ n,and j is a multiplication of 3$\}$	$[0,1]^2\times[-2,2]^{n-2}$ n=10

表 4-1 多目標最佳化問題（續）。

問題	多目標函式	數值範圍
F7	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(4y_i^2-\cos(8y_i\pi)+1.0)$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(4y_i^2-\cos(8y_i\pi)+1.0)$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$ and $y_i=x_j-x_1^{0.5\left(1.0+\frac{3(j-2)}{n-2}\right)}$, $j=2,\ldots,n$	$[0,1]^n$ $n=10$
F8 (UF3)	$f_1=x_1+\frac{2}{\lvert J_1\rvert}(4\sum_{j\in J_1}y_i^2-2\prod_{j\in J_1}\cos\left(\frac{20y_j\pi}{\sqrt{j}}\right)+2)$ $f_2=1-\sqrt{x_1}+\frac{2}{\lvert J_2\rvert}(4\sum_{j\in J_2}y_i^2-2\prod_{j\in J_2}\cos\left(\frac{20y_j\pi}{\sqrt{j}}\right)+2)$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$ and $y_i=x_j-x_1^{0.5\left(1.0+\frac{3(j-2)}{n-2}\right)}$, $j=2,\ldots,n$	$[0,1]^n$ $n=10$
F9	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j-\sin(6\pi x_1+\frac{j\pi}{n}))^2$ $f_2=1-x_1^2+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j-\sin(6\pi x_1+\frac{j\pi}{n}))^2$ where $J_1=\{j\mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j\mid j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ $n=30$

表 4-1 多目標最佳化問題（續）。

問題	多目標函式	數值範圍
DTLZ1	$f_1=(1+g(x))x_1x_2$ $f_2=\big(1+g(x)\big)x_1(1-x_2)$ $f_3=(1+g(x))(1-x_1)$ where $g(x)=100(n-2)+100\sum_{i=3}^{n}\{(x_i-0.5)^2$ $-\cos(20\pi(x_i-0.5))\}$	$[0,1]^n$ $n=10$
DTLZ2	$f_1=(1+g(x))\cos(\frac{x_1\pi}{2})\cos(\frac{x_2\pi}{2})$ $f_2=\big(1+g(x)\big)\cos(\frac{x_1\pi}{2})\sin(\frac{x_2\pi}{2})$ $f_3=(1+g(x))\sin(\frac{x_1\pi}{2})$ where $g(x)=\sum_{i=3}^{n}x_i^2$	$[0,1]^2\times[-2,2]^{n-2}$ $n=10$
UF4	$f_1=x_1+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}h(y_j)$ $f_2=1-x_1^2+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}h(y_j)$ $y_i=x_j-\sin\left(6\pi x_1+\frac{j\pi}{n}\right),j=2,\ldots,n$ $h(t)=\frac{\lvert t\rvert}{1+e^{2\lvert t\rvert}}$ where $J_1=\{j\lvert j$ is odd and $2\leq j\leq n\}$ and $J_2=\{j\lvert j$ is even and $2\leq j\leq n\}$	$[0,1]\times[-2,2]^{n-1}$ $n=30$

表 4-1 多目標最佳化問題（續）。

問題	多目標函式	數值範圍
UF5	$f_1=x_1+(\frac{1}{2N}+\varepsilon)\lvert\sin(2N\pi x_1)\rvert+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}h(y_j)$ $f_2=1-x_1^2+(\frac{1}{2N}+\varepsilon)\lvert\sin(2N\pi x_1)\rvert+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}h(y_j)$ $y_i=x_j-\sin\left(6\pi x_1+\frac{j\pi}{n}\right),j=2,\ldots,n$ $h(t)=2t^2-\cos(4\pi t)+1$ where $J_1=\{j\lvert j$ is odd and $2\le j\le n\}$ and $J_2=\{j\lvert j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ $n=30$
UF6	$f_1=x_1+\max\{0,2(\frac{1}{2N}+\varepsilon)\sin(2N\pi x_1)\}+$ $\frac{2}{\lvert J_1\rvert}(4\sum_{j\in J_1}y_i^2-2\prod_{j\in J_1}\cos\left(\frac{20y_j\pi}{\sqrt{j}}\right)+2)$ $f_2=1-x_1^2+\max\{0,2(\frac{1}{2N}+\varepsilon)\sin(2N\pi x_1)\}+$ $\frac{2}{\lvert J_2\rvert}(4\sum_{j\in J_1}y_i^2-2\prod_{j\in J_1}\cos\left(\frac{20y_j\pi}{\sqrt{j}}\right)+2)$ $y_i=x_j-\sin\left(6\pi x_1+\frac{j\pi}{n}\right),j=2,\ldots,n$ where $J_1=\{j\lvert j$ is odd and $2\le j\le n\}$ and $J_2=\{j\lvert j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ $n=30$

表 4-1 多目標最佳化問題（續）。

問題	多目標函式	數值範圍
UF7	$f_1=\sqrt[5]{x_1}+\frac{2}{\lvert J_1\rvert}\sum_{j\in J_1} y_j^2$ $f_2=1-\sqrt[5]{x_1}+\frac{2}{\lvert J_2\rvert}\sum_{j\in J_2} y_j^2$ $y_i=x_j-\sin\left(6\pi x_1+\frac{j\pi}{n}\right), j=2,\ldots,n$ where $J_1=\{j \mid j$ is odd and $2\le j\le n\}$ and $J_2=\{j \mid j$ is even and $2\le j\le n\}$	$[0,1]\times[-1,1]^{n-1}$ $n=30$

4.2效能指標

此處使用 inverted generational distance (IGD) 作為評估指標，其計算方式如下:

$$\mathrm{IGD}(P^*,P)=\frac{\sum_{v\in P^*} d(v,P)}{\lvert P^*\rvert} \tag{19}$$

P 為演算法求得的近似最佳解，P^* 為目前已知在柏拉圖前緣均勻分布的最佳解，v 代表 P 中的其中一個個體，$d(v,P)$ 用來計算每個柏拉圖前緣最佳解與距離該解最近的個體距離，將所有算出之距離總和除以柏拉圖前緣最佳解個數，來求出每個柏拉圖前緣最佳解到最近之近似解的平均距離，數值越小表示近似最佳解的效能越好，而要使其值變小必須同時逼近柏拉圖前緣並且分布均勻，是一種常見的評估指標。

4.3參數設定

4.3.1 基礎參數設定(MOEAD-AMS)

1. DE 控制變數 $CR = 1.0$、$F = 0.5$，polynomial mutation 分布變數 $\eta = 20$、突變機率 $p_m = 1/n$，n 為決策空間的維度。

2. 在 MOEA/D [3] 內所使用到的權重向量（均勻分布於目標空間的向量），其個數設定為和族群個數相同，而在族群大小的部份參照 MOEA/D-AMS [2] 文獻內的設定，目標空間維度為 2 時族群個數為 300，目標空間維度為 3 時族群個數為 595。

3. 依照 MOEA/D [3] 文獻內的設定，個體的鄰居個數 $T = 20$，選擇鄰居為交配池的機率 $\delta = 0.9$，每個突變產生的子代最多可以取代族群個體的數量 $n_r = 2$。

4. 演算法終止代數為 500 代。

5. 收斂評估機制參數 $\alpha = 5$，密集度評估機制參數 $\beta = 2\%$，交配池選擇機制啟動參數 $\gamma = 50\%$，$\varepsilon = 5$。

4.3.2 各演算法之調整機制所需參數設定

本論文所挑選之問題雖與各演算法文獻內之測試問題不盡相同，仍然將文獻內提及過之參數設定做為優先參考，若無設定則與 4.3.1 之 DE 控制變數預設值相同。

1. NSDE [5] 依照文獻內的參數設定，參數 F 由兩種不同的隨機分布方式所產生，而參數值使用這兩種隨機分布的機率均為 0.5。

2. jDE [6] 依照文獻內的參數設定，τ_1、τ_2 兩個門檻值設為 0.1，即是有 0.9 的機率使用之前的參數值，F 的初始值為 0.5，CR 的初始值為 0.9。

3. SspDE [7] 依照文獻內的參數設定，每個個體的參考串列長度為 50，也就是每 50 代會重新安排串列內容，而串列中每個位置的內容有 0.8 的機率使用之前有更新親代的參數值，有 0.2 的機率使用隨機數值。

4. SaDE [8] 依照文獻內的參數設定，CRm 的初始值為 0.5。

5. SaNSDE[10] 依照文獻內的參數設定，CRm 的初始值為 0.5。

6. JADE [11] 依照文獻內的參數設定，參數 $c = 0.1$，也就是要產生新的參數值時參考族群中產生良好子代的參數值平均的比例為 0.1，參考個體之前所使用的參數值比例為 0.9，μ_{CR} 初始值為 0.5，μ_F 初始值為 0.5。

7. SDE [14] 依照文獻內的參數設定，參數調整機制為依照 DE 演算法的方式演化，參數值演化的 F 值為 $N(0,0.5)$，F 的初始值為 0 到 1 的隨機數。

4.4效能評比

將我們在第三章所選擇的幾種參數調整機制分別與 MOEA/D-AMS 結合後，依照 4.3 的參數設定對 17 個問題各執行 20 次，利用這 20 次的 IGD 值之平均來比較各參數調整機制的表現，其結果如表 4-2 所示，以藍色為底的值代表此參數調整機制在該問題上效能最好，紅色則為效能最差。

表 4-2　IGD 效能評比。

問題	NSDE	jDE	SspDE	SaDE	SaNSDE	JADE	SDE	ADEA	MOEA/D-APC
F1	0.001456	0.00147735	0.001462	0.001568	0.001541	0.001429	0.0015388	0.001426	0.001337
F2	0.009333	0.0108698	0.010486	0.0113261	0.013528	0.005182	0.0133502	0.011926	0.003194
F3	0.004491	0.00452189	0.004573	0.0048783	0.004595	0.003433	0.0059871	0.007297	0.002558
F4	0.004022	0.00384995	0.003335	0.003826	0.004342	0.002761	0.0056511	0.005227	0.002209
F5	0.006282	0.0055597	0.006593	0.0055181	0.005774	0.006128	0.006304	0.008417	0.005192
F6	0.031077	0.0313711	0.0316	0.0323304	0.031574	0.029878	0.034262	0.031171	0.028287
F7	0.012886	0.0149444	0.006411	0.0236424	0.020234	0.00201	0.0512567	0.001978	0.001401
F8	0.098828	0.100873	0.105963	0.103217	0.099592	0.026898	0.116168	0.002229	0.001474
F9	0.001814	0.00184761	0.001809	0.0024165	0.00247	0.001597	0.0024794	0.001623	0.001434
UF4	0.048575	0.0462382	0.045943	0.0472871	0.047326	0.042938	0.0449001	0.075495	0.052755
UF5	0.297543	0.322024	0.275772	0.243698	0.304045	0.248509	0.234308	0.309991	0.326128
UF6	0.30707	0.303238	0.271237	0.230776	0.285954	0.2631	0.29508	0.182378	0.188392
UF7	0.006096	0.0806481	0.02291	0.0164098	0.046167	0.014777	0.0438712	0.017993	0.002455
UF9	0.055896	0.0822007	0.081989	0.0567218	0.058533	0.059413	0.0514234	0.051849	0.058765
UF10	0.335431	0.377197	0.289038	0.254958	0.246902	0.37327	0.281883	0.362821	0.351256
DTLZ1	0.022082	0.0220666	0.022043	0.0220668	0.022084	0.022098	0.0220866	0.023634	0.022567
DTLZ2	0.026882	0.0268368	0.026877	0.0268684	0.026831	0.026831	0.026872	0.027058	0.026728

表 4-3　IGD 效能標準差，以粗體標示者為該問題上的最小值。

問題	NSDE	jDE	SspDE	SaDE	SaNSDE	JADE	SDE	ADEA	MOEA/D-APC
F1	6.16E-05	7.13E-05	5.98E-05	0.0001055	0.000131	5.46E-05	6.36E-05	4.83E-05	**1.63E-05**
F2	0.004985	0.00578465	0.00492	0.0048685	0.005948	0.000857	0.0063516	0.007883	**0.000655**
F3	0.000964	0.000729552	0.001426	0.0010648	0.000706	0.000537	0.0011551	0.0092	**0.00029**
F4	0.001306	0.00159109	0.001574	0.0010178	0.001686	**0.000369**	0.0028766	0.004946	0.000763
F5	0.002057	**0.00106112**	0.001948	0.0011526	0.001723	0.001837	0.0011839	0.006805	0.001505
F6	0.001879	0.00224062	0.001594	0.0026889	0.002633	0.001303	0.0018449	0.001938	**0.001061**
F7	0.014027	0.017684	0.005932	0.0184338	0.024632	0.000316	0.0232592	0.000283	**4.28E-05**
F8	0.027986	0.0279327	0.025106	0.0155643	0.033696	0.032404	0.0176208	0.000328	**6.18E-05**
F9	0.00014	0.000103651	0.000212	0.0006468	0.000515	5.70E-05	0.0003252	6.61E-05	**5.64E-05**
UF4	0.001986	0.00198043	0.001996	0.0020968	0.002022	**0.001301**	0.0021704	0.003184	0.004163
UF5	0.112159	0.140443	0.081905	0.0774527	0.144101	0.063456	0.0522425	0.049862	**0.043267**
UF6	0.243255	0.244371	0.243413	0.199097	0.174452	0.193677	0.247219	0.198546	**0.153972**
UF7	0.001862	0.135181	0.07752	0.045828	0.097057	0.039015	0.112535	0.046332	**0.000497**
UF9	0.040247	0.0527873	0.053799	0.0391577	0.039667	0.044655	**0.0233785**	0.035485	0.051024
UF10	0.120901	0.13717	0.116354	0.0956905	0.104127	0.125213	0.126711	0.091072	**0.073999**
DTLZ1	8.67E-05	7.47E-05	0.000101	6.91E-05	0.000121	**5.31E-05**	5.35E-05	0.001653	0.000257
DTLZ2	0.00011	**0.000103315**	0.000122	0.0001753	0.000144	0.00013	0.0001478	0.000203	0.000137

整理出各問題上表現最佳的 IGD 值後，我們希望了解其他調整機制的結果與其差距，藉此觀察哪些調整機制與最佳 IGD 值較接近，並且這些機制是否有共同的特質。表 4-3 所列之數值代表表 4-2 中各調整機制的 IGD 值與該問題效能最好的 IGD 值之差除以最佳 IGD 值，以百分比表示，將其分為三個群組並以不同顏色區別，藍色代表最接近最佳 IGD 值（10% 以內），綠色次之（200% 以內），紅色則為差距最多（大於 200%）。計算出表 4-3 後，依照表 4-2 與表 4-3 統整出表 4-4，近最佳 IGD 數為表 4-3 內該調整機制在多少問題上的 IGD 與最佳 IGD 差距在 10% 以內（含 0% 與 10%），最差 IGD 數則為表 4-2 內該調整機制在多少問題上的 IGD 值為最差。

表 4-4　IGD 效能差距。

問題	NSDE	jDE	SspDE	SaDE	SaNSDE	JADE	SDE	ADEA	MOEA/D-APC
F1	9%	11%	9%	17%	15%	7%	15%	7%	0%
F2	192%	240%	228%	255%	324%	62%	318%	273%	0%
F3	76%	77%	79%	91%	80%	34%	134%	185%	0%
F4	82%	74%	51%	73%	97%	25%	156%	137%	0%
F5	21%	7%	27%	6%	11%	18%	21%	62%	0%
F6	10%	11%	12%	14%	12%	6%	21%	10%	0%
F7	820%	967%	358%	1588%	1344%	43%	3559%	41%	0%
F8	6606%	6745%	7090%	6904%	6658%	1725%	7783%	51%	0%
F9	27%	29%	26%	69%	72%	11%	73%	13%	0%
UF4	13%	8%	7%	10%	10%	0%	5%	76%	23%
UF5	27%	37%	18%	4%	30%	6%	0%	32%	39%
UF6	68%	66%	49%	27%	57%	44%	62%	0%	3%
UF7	148%	3184%	833%	568%	1780%	502%	1687%	633%	0%
UF9	9%	60%	59%	10%	14%	16%	0%	1%	14%
UF10	36%	53%	17%	3%	0%	51%	14%	47%	42%
DTLZ1	0%	0%	0%	0%	0%	0%	0%	7%	2%
DTLZ2	1%	0%	1%	1%	0%	0%	1%	1%	0%

表 4-5　IGD 結果統整。

	近最佳 IGD 數	最差 IGD 數	差距
NSDE	5	2	3
jDE	4	3	1
SspDE	4	0	4
SaDE	7	1	6
SaNSDE	4	1	3
JADE	6	0	6
SDE	5	5	0
ADEA	6	4	2
MOEAD-APC	13	1	12

由表 4-2 與 4-3，可看出 MOEA/D-APC 在大部份 UF 問題上與其他調整機制的效能相近甚至被超越，並且 F2、F7、F8 與 UF7 問題幾乎全部的調整機制效能都與最佳 IGD 值有明顯的差距，而表 4-4 則讓我們能從 MOEA/D-APC 外觀察到另外幾種表現特別突出的調整機制。因為圖表呈現方式的受限，之後主要對二維問題進行討論。

依照參數調整的方式，我們將這幾種機制對 *F* 與 *CR* 參數的調整方式分別分為未調整與調整兩類，未調整代表該參數可能為定值或是在某個範圍內隨機產生，而是否視為調整則依照產生新的參數時是否有使用到目前個體以外的資訊。

表 4-6 參數調整方式分類。

參數		*F*	
	調整方式	未調整	調整
CR	未調整	NSDE jDE	SDE ADEA
	調整	SaDE SaNSDE	SspDE JADE

由表 4-5 可發現，有三種調整機制的效能特別優秀，分別為 SspDE、SaDE 與 JADE，其共通點為皆是個別參數類，其中除了 SspDE 為個體資訊類外，其餘兩種機制都是群體資訊類。搭配表 4-6 來看，可觀察到 SspDE 與 JADE 同樣屬於同時對 *F* 與 *CR* 進行調整的類別，而 SaDE 與 SaNSDE 僅對 *CR* 進行調整，這四種機制對 *CR* 的調整可能是使效能比其他機制更好的關鍵。

藉著表 4-2、表 4-3、表 4-4、表 4-5 與表 4-6，我們可以發現兩個較明顯的問題，在之後的小節中將進行討論。

1. MOEA/D-APC 在 UF5 的效能最差，可能該問題對 *F* 參數的變動範圍需求較廣，或是需要對 *CR* 的調整（MOEA/D-APC 固定為 1）。
2. F2、F7、F8 與 UF7 問題，大部份調整機制效能不佳，而 MOEA/D-APC 的效能卻可以超越其他機制，其原因在於？

4.4.1 UF5 問題效能探討

UF5 是 MOEA/D-APC 表現最差的問題 ，為觀察其參數變化與其他機制的差別，我們每五代紀錄一次參數值，執行 20 次後取其平均，將所有機制於 UF5 問題的參數變動曲線繪出，結果如下圖。

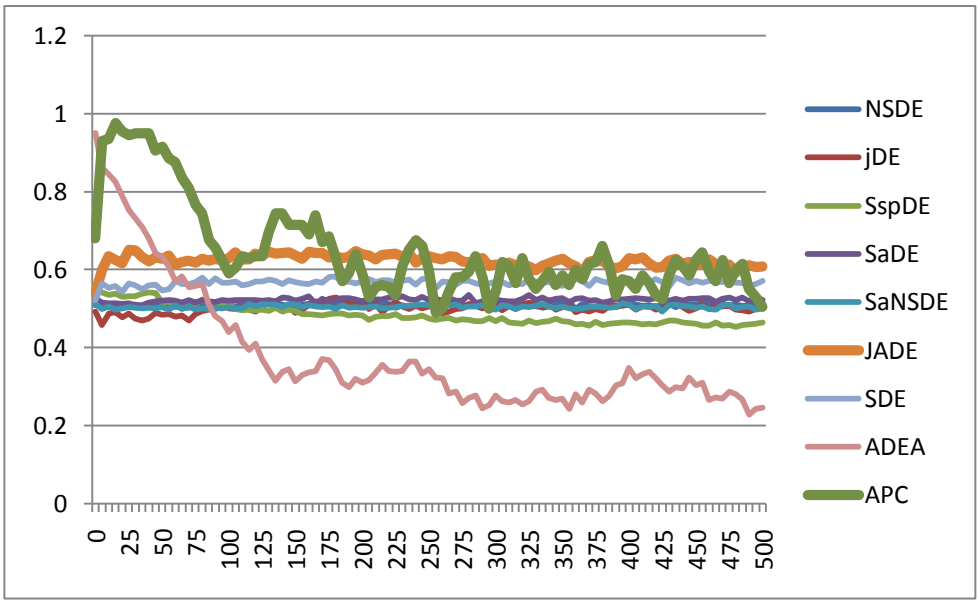

圖 4-1　各機制在求解 UF5 問題時的 *F* 參數平均值變動圖，縱軸為 *F* 值，橫軸為代數。

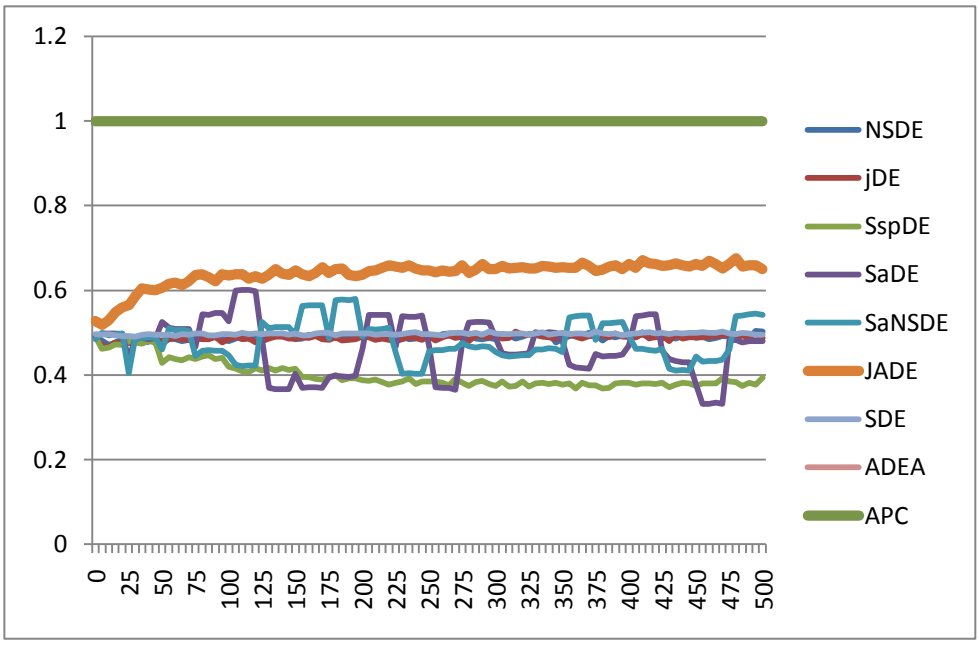

圖 4-2　各機制在求解 UF5 問題時的 *CR* 參數平均值變動圖，縱軸為 *CR* 值，橫軸為代數。

由於 MOEA/D-APC 的 *CR* 曲線與其他機制明顯的不同，我們推測該特點是影響效能差異的主因，故將 *F* 值固定為 0.5（因大部份機制的 *F* 值曲線分布在 0.5 附近），將 *CR* 值分別設為 0.4、0.5、0.6、0.7、0.8、0.9 及 1，求解 UF5 問題 20 次再計算平均 IGD 值，並以表 4-4 的算法計算與最佳 IGD 值的差距，結果如下。

表 4-7　UF5 問題調整 *CR* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表						
CR 值	0.4	0.5	0.6	**0.7**	0.8	0.9	1
IGD 值	0.30902	0.31399	0.32163	**0.30275**	0.32434	0.36038	0.40606

表 4-8　UF5 問題調整 *CR* 值測試結果與表 4-7 最佳 IGD 值之差距。

	與最佳 IGD 值之差						
CR 值	0.4	0.5	0.6	0.7	0.8	0.9	1
差距	2%	4%	6%	0%	7%	19%	34%

表 4-9　MOEA/D-APC 使用不同 *CR* 值解 UF5 問題的效能差異。

CR 值	1	0.7
IGD	0.326128	0.207865

由表 4-7 可以發現，在 *CR* 值為 0.7 的時候效能最佳，為了了解這個結果是否適用於 MOEA/D-APC，將 MOEA/D-APC 的預設 *CR* 值改為 0.7 解 UF5 問題 20 次，計算平均 IGD 值，結果如表 4-9 所示，效能明顯比 *CR* 值為 1 時較佳，因此我們認為這個問題對 MOEA/D-APC 來說需要較低的 *CR* 值來提供近親繁殖（即子代較接近親代）的能力，由於目前 MOEA/D-APC 尚無對 *CR* 進行調整的機制，故可以說只對單一的參數值進行調整有其極限，有些問題需要對兩個參數都進行調整才能求得好的結果。為了了解 *CR* 值的改變對 MOEA/D-APC 造成的實際影響，我們將其中一次求解的過程紀錄下來，繪出其族群分布圖並以不同顏色區別參數所在的區間，並分別挑選在該問題上表現最好 （SDE 與 SaDE）與最差（MOEA/D-APC 與 jDE）的前兩種機制來比較，結果如下。

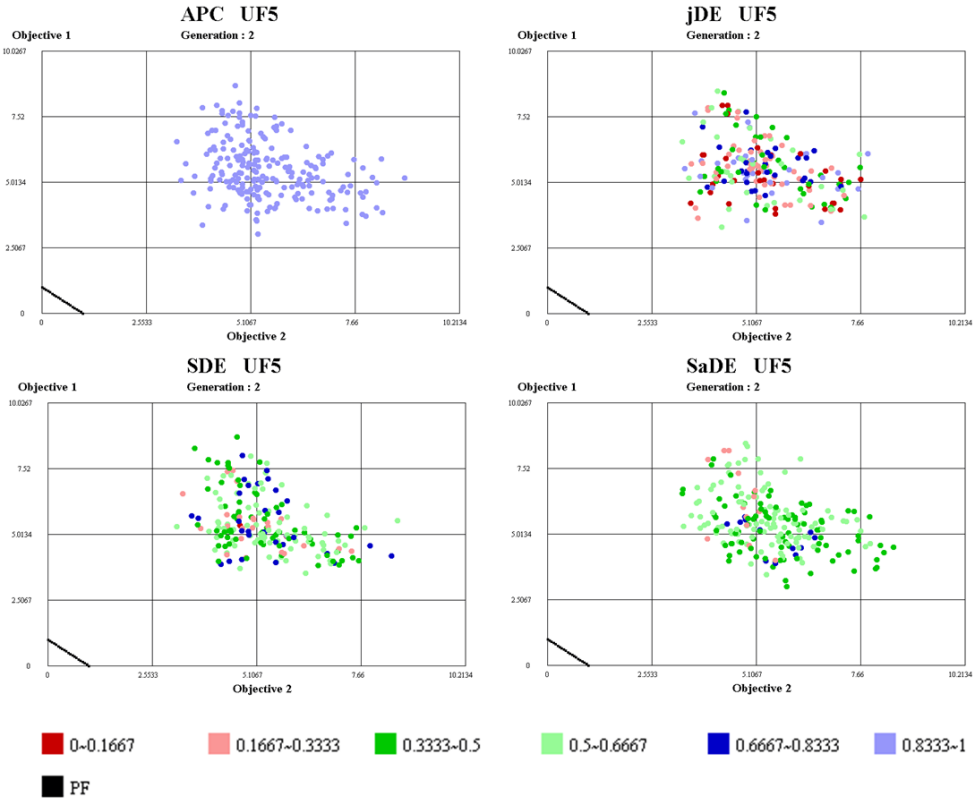

圖 4-3　*CR* 參數於第 2 代時在族群中的分布情形。

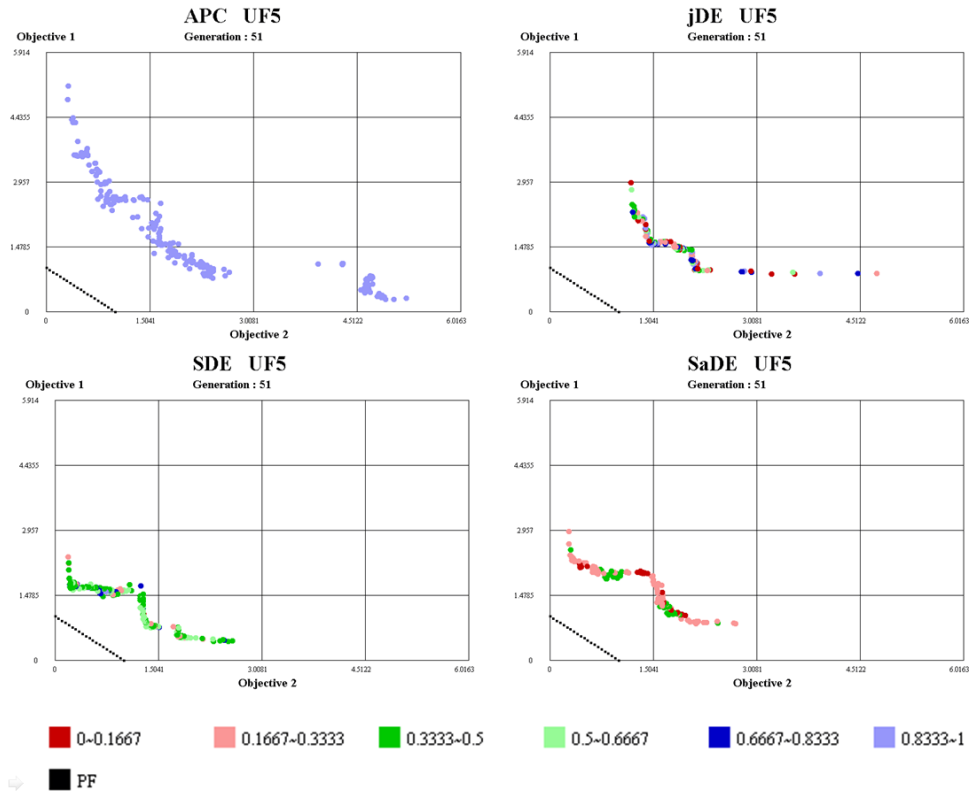

圖 4-4　*CR* 參數於第 51 代時在族群中的分布情形。

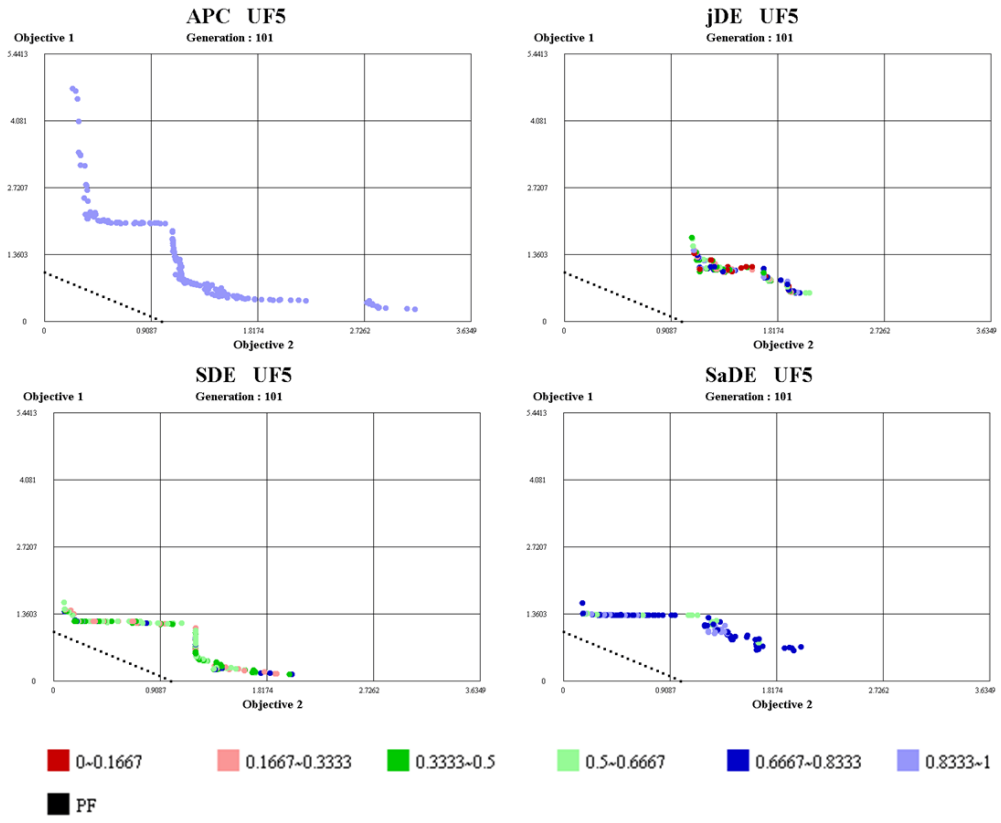

圖 4-5　*CR* 參數於第 101 代時在族群中的分布情形。

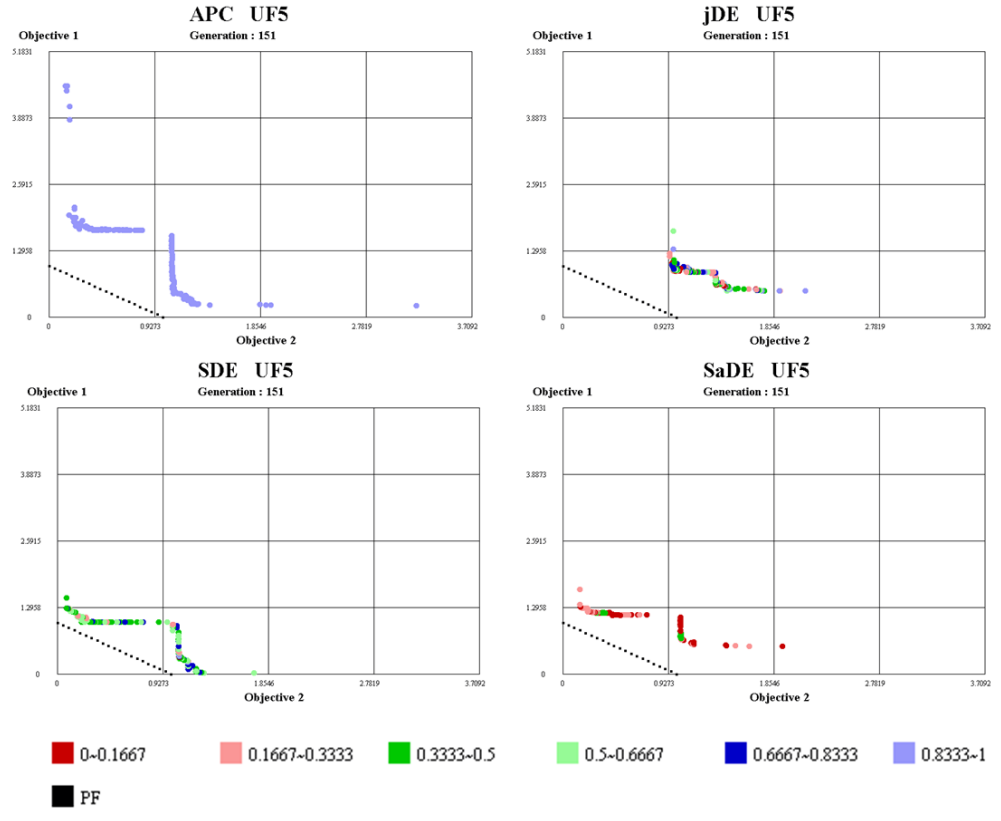

圖 4-6　*CR* 參數於第 151 代時在族群中的分布情形。

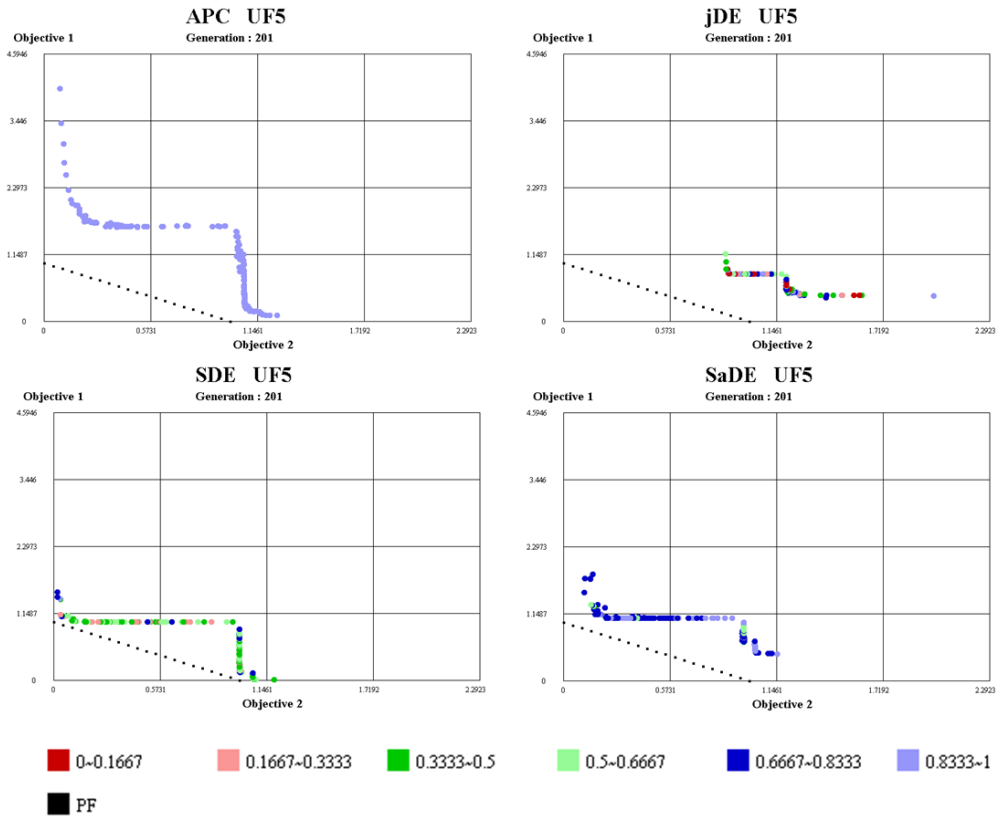

圖 4-7　*CR* 參數於第 201 代時在族群中的分布情形。

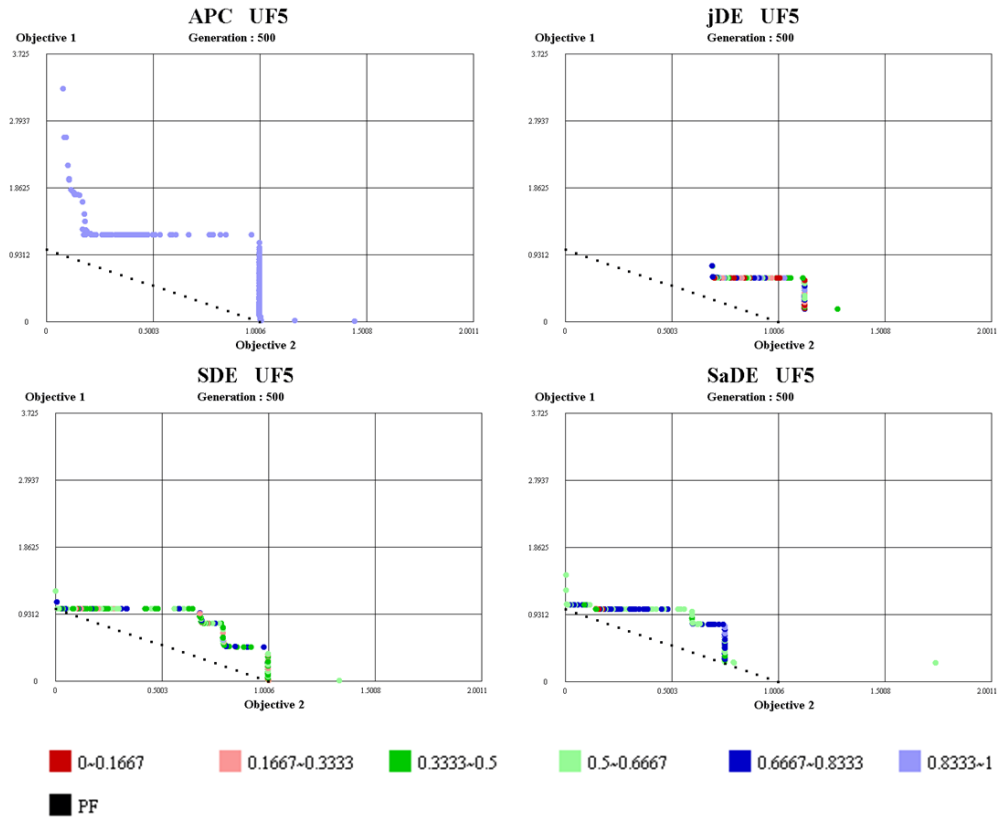

圖 4-8　*CR* 參數於第 500 代時在族群中的分布情形。

由圖 4-3 至圖 4-8 可發現，該問題有一塊較難收斂的區域，若 *CR* 為 1 時難以繼續往柏拉圖前緣收斂，而幾乎沒有進行任何調整的 jDE 機制則是隨著演化被困在一小塊區域內無法擴張族群的分布。SDE 與 SaDE 都能對難收斂區域進行深入的動作，但兩機制主要分布的 *CR* 參數不太相同，這可能是因為該問題的優良 *CR* 值區段不只有一個，由表 4-6 便可知道 *CR* 值在 0.7 與 0.4 的時候效能都是良好的，由此可證該問題不適合過度的進行遠親繁殖（即 *CR* 值較高，使子代與親代相似度低）。

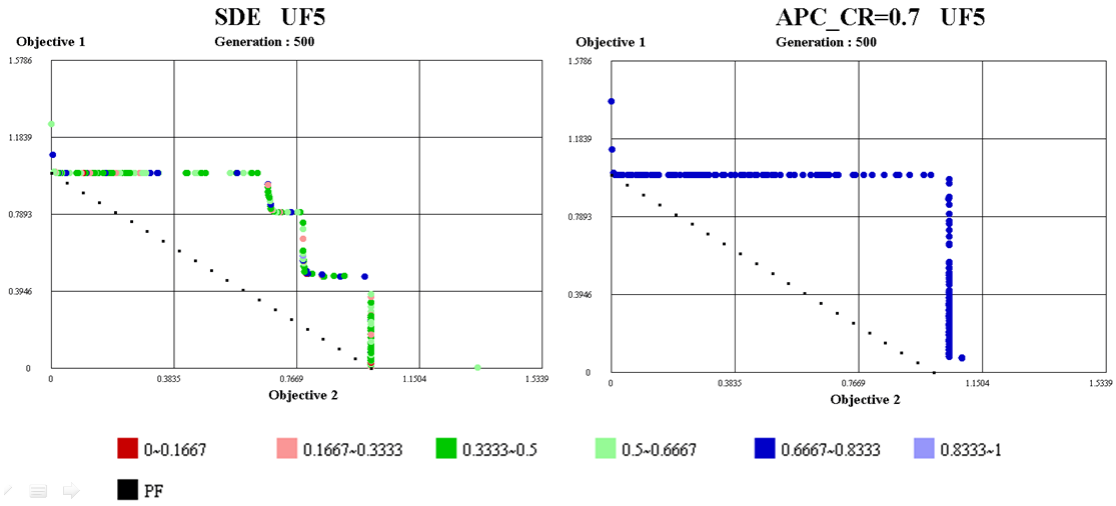

圖 4-9 第 500 代時 *CR* 參數在族群中的分布情形。

將 MOEA/D-APC 的 *CR* 值設為 0.7 後，觀察其最後一代的族群分布（見圖 4-9），可以發現其形狀有接近 SDE 的趨勢，最明顯的便是能找到原本找不到的邊緣解，在難收斂區域邊界上的個體也比原本密集許多，因此我們認為對 MOEA/D-APC 來說較低的 *CR* 值有助於解 UF5 問題，可見將 *CR* 值固定為 1 並非適用於所有問題，若想提高 MOEA/D-APC 的效能，加入 *CR* 值的調整機制是勢在必行的。

4.4.2 F2 問題效能探討

為了觀察參數變動的走向是否跟大部份機制在此問題上效能不佳有關，我們每 5 代紀錄一次 *F* 與 *CR* 參數值，並且每個機制各執行 20 次再計算參數在該代數上的平均值，結果如圖 4-10 所示。

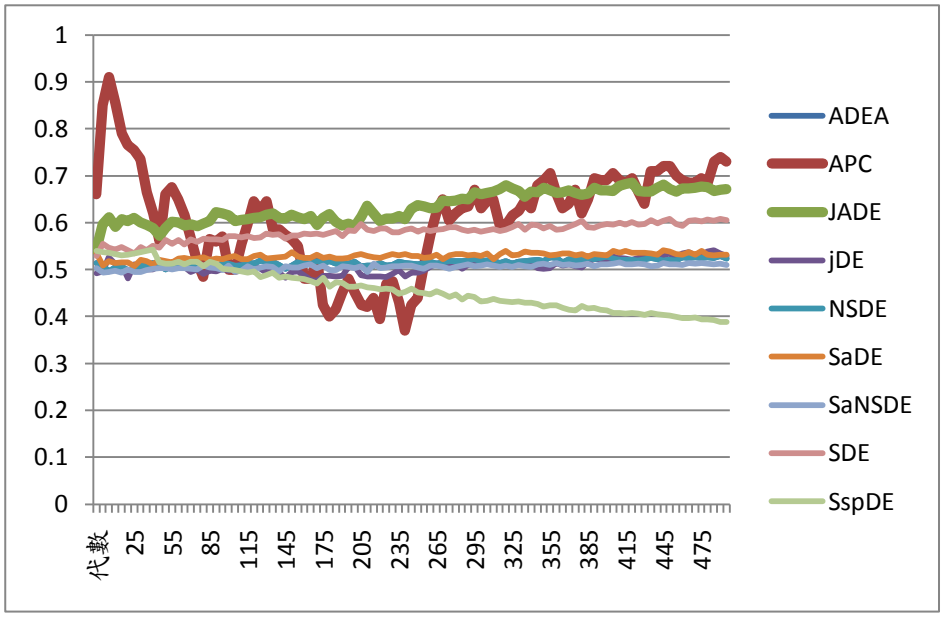

圖 4-10 各機制在求解 F2 問題時的 *F* 參數平均值變動圖，縱軸為 *F* 值，橫軸為代數。

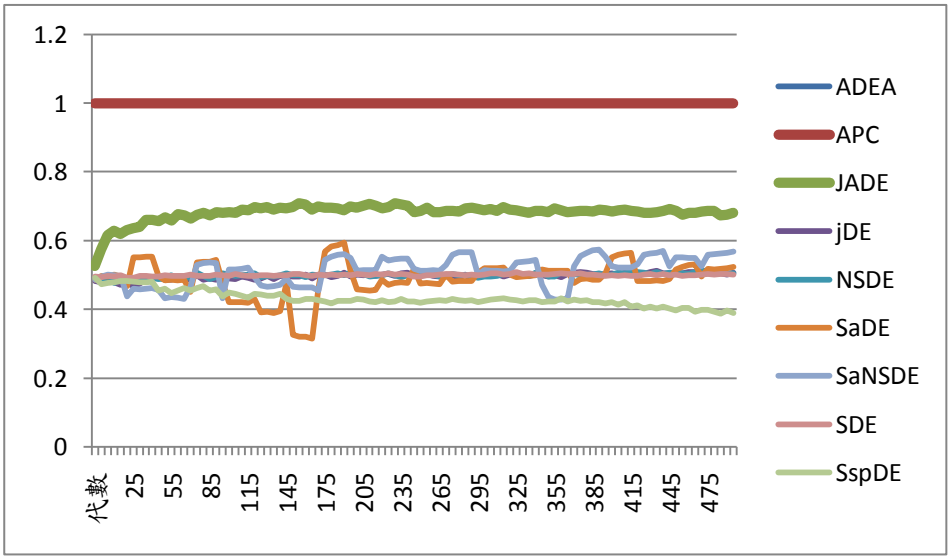

圖 4-11 各機制在求解 F2 問題時的 *CR* 參數平均值變動圖，縱軸為 *CR* 值，橫軸為代數。

由圖 4-10 及圖 4-11 可知，JADE 與 MOAE/D-APC 兩個在該問題上效能較好的機制，其 *F* 與 *CR* 值的變動與其他機制有較大的不同，*F* 值在中後期的演化上變動曲線皆漸漸由 0.6 上升至 0.7，而 *CR* 參數的部份 JADE 也很明顯的比其他機制為高（ADEA 與 MOEAD-APC 固定為 1），整體來說 JADE 與 MOAED-APC 的兩個參數在該問題上都有著較大的值，所以我們認為 F2 問題需要較高的 *F* 與 *CR* 參數才能求得好的結果。

由於 MOEAD-APC 與 JADE 在 *CR* 值的變動上與其他機制相較起來有顯著的差距且變動情形更為穩定，因此我們嘗試驗證高 *CR* 值是否有助於求解 F2 問題。將 *F* 值固定為 0.5（因大部份機制之 *F* 值變動曲線皆聚集在 0.5 附近），再將 *CR* 值分別設為 0.5、0.6、0.7、0.8、0.9 與 1，執行 20 次後計算平均 IGD 值，求得最佳 IGD 後再以表 4-3 之算法求出與各機制效能的差距。

表 4-10　F2 問題調整 *CR* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表					
CR 值	0.5	0.6	0.7	0.8	0.9	**1**
IGD 值	0.017577	0.0141067	0.006977	0.0051621	0.003484	**0.002335**

表 4-11　F2 問題調整 *CR* 值測試結果與表 4-10 最佳 IGD 值之差距。

	與最佳 IGD 值之差					
CR 值	0.5	0.6	0.7	0.8	0.9	1
IGD 值	653%	504%	199%	121%	49%	0%

由表 4-10 與表 4-11 可看出，高 *CR* 值確實對解 F2 問題有所幫助，在 *CR* 值為 1 時的效能明顯比其他值來得好。該問題可能需要利用遠親繁殖來增加個體多樣性。在知道這樣的結果後，便又好奇對 MOEA/D-APC 來說最佳的 *F* 值

為何，於是將 *CR* 固定為 1 對 *F* 值來進行調整，因為由圖 4-10 觀察出各機制的 *F* 值變動曲線大部份分布在 0.4 至 0.8 間，於是把 *F* 值的範圍設為 0.4 至 0.8，一樣計算 20 次結果的平均 IGD 值，結果如下。

表 4-12　F2 問題調整 *F* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表								
F 值	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8
IGD 值	0.0053	0.0044	0.0023	0.00404	0.004	0.0039	0.0043	0.00475	0.00567

表 4-13　F2 問題調整 *F* 值測試結果與表 4-12 最佳 IGD 值之差距。

	與最佳 IGD 值之差								
F 值	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8
IGD 值	126%	88%	0%	73%	72%	69%	86%	104%	143%

可以發現在 *CR* 值為 1 時，*F* 值為 0.5 的整體效能明顯比其他結果好，但是 MOEA/D-APC 對 *F* 值的調整卻沒有穩定的落在 0.5 附近。同樣的，對其他有對 *CR* 值進行調整的機制來說，也無法將 *CR* 調整到 1，即使最後所計算出的效能是最佳的。因為 F2、F7、F8 與 UF7 問題各機制表現相似，我們認為所發生的問題可能也相同，所以先對剩下的三個問題進行與 F2 問題一樣的實驗後再探討其原因。

4.4.3 F7 問題效能探討

與 F2 問題相同，先繪製兩個參數於該問題上的變動曲線圖。

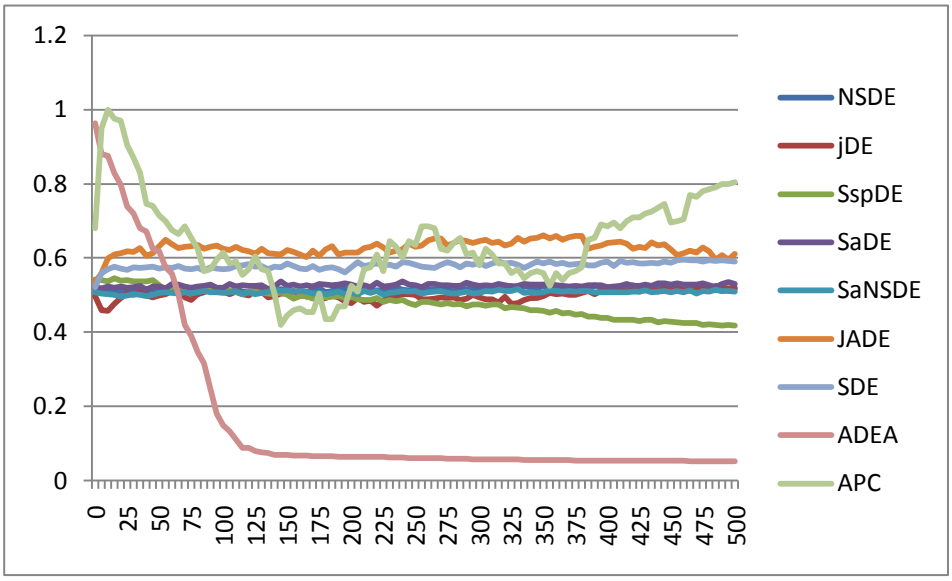

圖 4-12 各機制在求解 F7 問題時的 *F* 參數平均值變動圖，縱軸為 *F* 值，橫軸為代數。

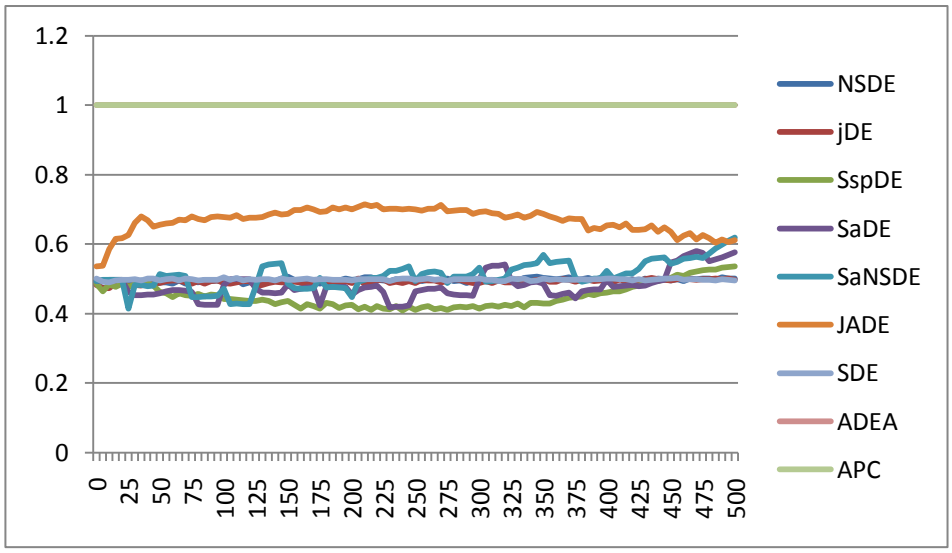

圖 4-13 各機制在求解 F7 問題時的 *CR* 參數平均值變動圖，縱軸為 *CR* 值，橫軸為代數。

由於 MOEA/D-APC 的 *CR* 參數與其他機制相差較大，故先固定 *F* 值為 0.5 並對 *CR* 進行調整來進行實驗，範圍參照圖 4-13 設為 0.4 至 1。

表 4-14　F7 問題調整 *CR* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表						
CR 值	0.4	0.5	0.6	0.7	0.8	0.9	**1**
IGD 值	0.09505	0.069956	0.0245	0.00609	0.00281	0.00166	**0.00143**

表 4-15　F7 問題調整 *CR* 值測試結果與表 4-14 最佳 IGD 值之差距。

	與最佳 IGD 值之差						
CR 值	0.4	0.5	0.6	0.7	0.8	0.9	1
IGD 值	6519%	4771%	1607%	324%	96%	16%	0%

由表 4-14 可知，最佳 *CR* 值為 1，將其固定再對 *F* 進行調整，範圍參照圖 4-12 設為 0.4 至 0.8。

表 4-16　F7 問題調整 F 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表								
F 值	0.4	0.45	**0.5**	0.55	0.6	0.65	0.7	0.75	0.8
IGD 值	0.00143	0.00139	**0.00136**	0.001394	0.00139	0.00143	0.001418	0.0014	0.00147

表 4-17　F7 問題調整 *F* 值測試結果與表 4-16 最佳 IGD 值之差距。

	與最佳 IGD 值之差								
F 值	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8
IGD 值	5%	2%	0%	2%	2%	5%	4%	4%	8%

經實驗後，我們認為該問題在 *CR* 值為 1，*F* 值為 0.5 時效能最佳。

4.4.4 F8 問題效能探討

重覆與前兩個問題一樣的步驟，先繪製兩個參數於該問題上的變動曲線圖。

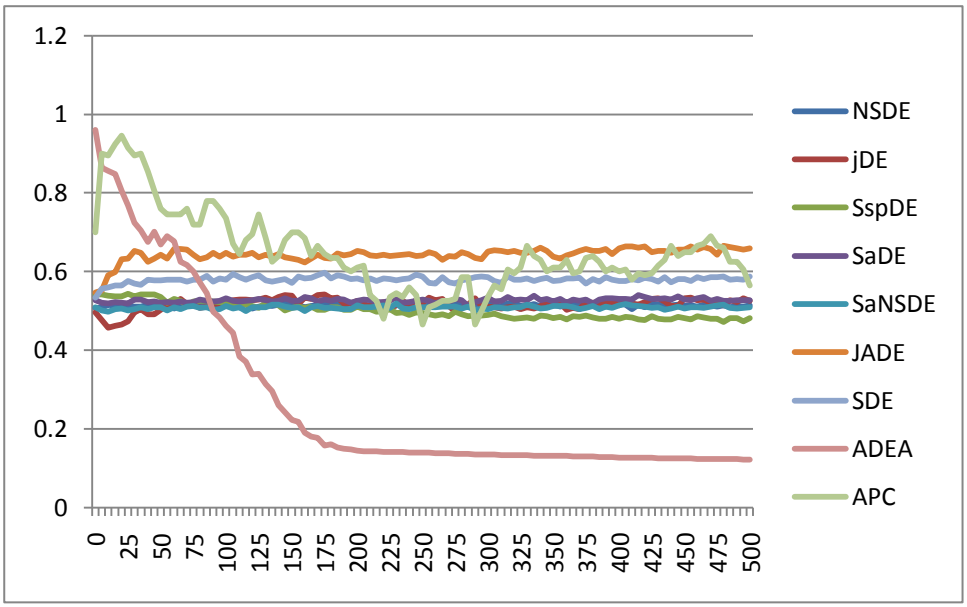

圖 4-14 各機制在求解 F8 問題時的 *F* 參數平均值變動圖，縱軸為 *F* 值，橫軸為代數。

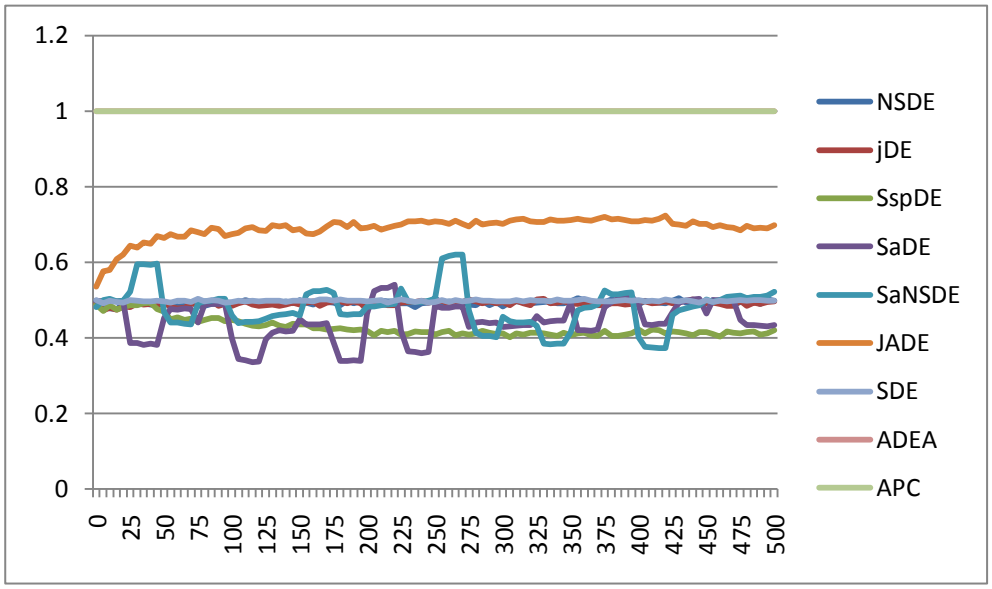

圖 4-15 各機制在求解 F8 問題時的 *CR* 參數平均值變動圖，縱軸為 *CR* 值，橫軸為代數。

將 *F* 固定為 0.5 對 *CR* 值進行調整，範圍參照圖 4-15 設為 0.4 至 1。

表 4-18　F8 問題調整 *CR* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表						
CR 值	0.4	0.5	0.6	0.7	0.8	0.9	**1**
IGD 值	0.14118	0.144777	0.117574	0.0779712	0.025523	0.004906	**0.0013871**

表 4-19　F8 問題調整 *CR* 值測試結果與表 4-18 最佳 IGD 值之差距。

	與最佳 IGD 值之差						
CR 值	0.4	0.5	0.6	0.7	0.8	0.9	1
IGD 值	10078%	10337%	8376%	5521%	1740%	254%	0%

求得最佳 *CR* 值為 1，再對 *F* 進行調整，範圍參照圖 4-14 設為 0.4 至 1。

表 4-20　F8 問題調整 *F* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表						
F 值	0.4	**0.5**	0.6	0.7	0.8	0.9	1
IGD 值	0.013716	**0.001387**	0.001449	0.00149	0.00155	0.00172	0.0061788

表 4-21　F8 問題調整 *F* 值測試結果與表 4-20 最佳 IGD 值之差距。

	與最佳 IGD 值之差						
F 值	0.4	0.5	0.6	0.7	0.8	0.9	1
IGD 值	889%	0%	4%	8%	12%	24%	345%

求得最佳 *F* 值為 0.5。

4.4.5 UF7 問題效能探討

重覆與前三個問題相同的步驟，先繪製兩個參數於該問題上的變動曲線圖。

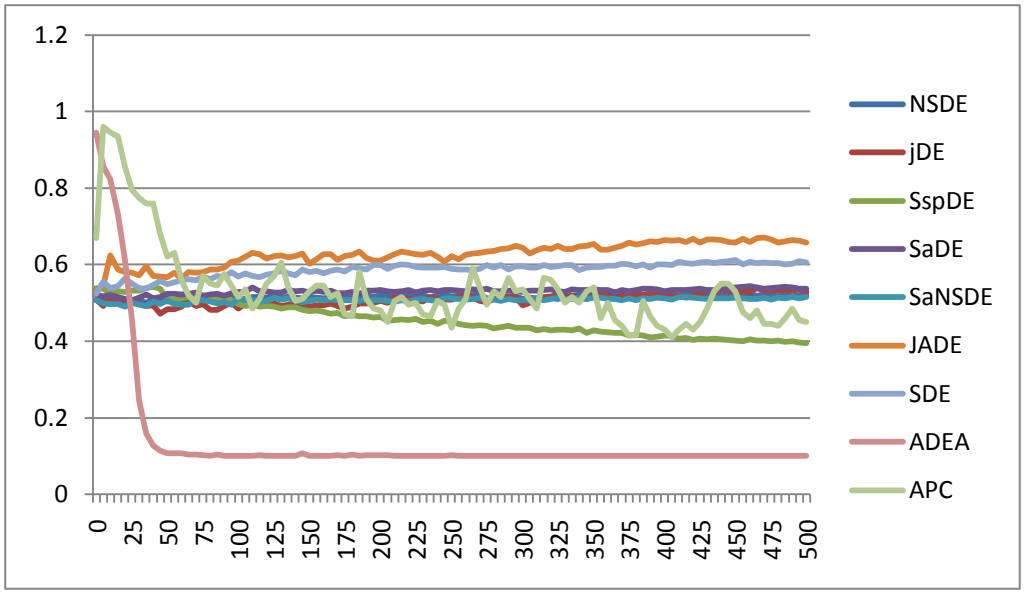

圖 4-16 各機制在求解 UF7 問題時的 *F* 參數平均值變動圖，縱軸為 *F* 值，橫軸為代數。

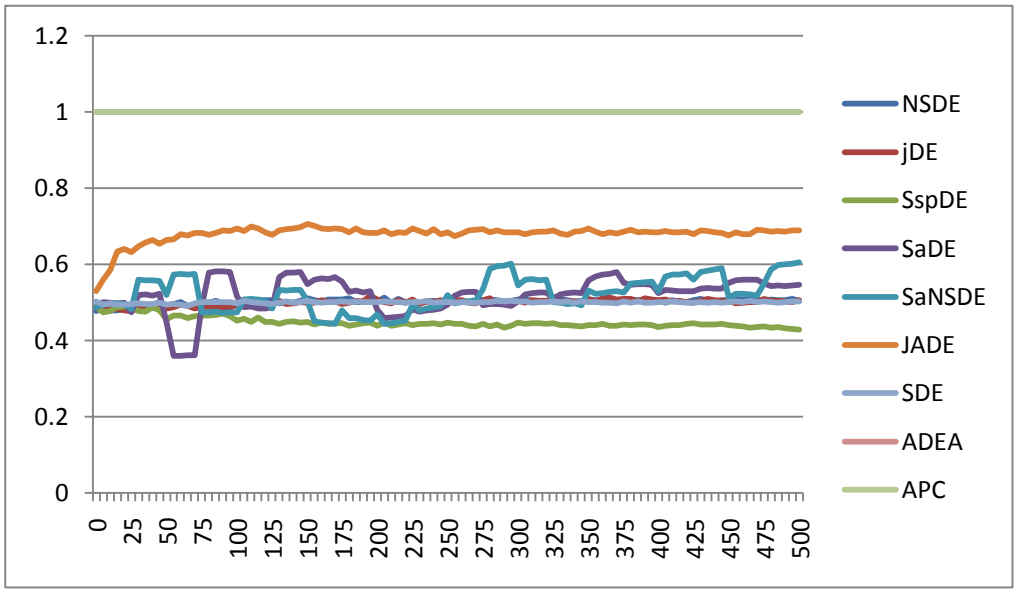

圖 4-17 各機制在求解 UF7 問題時的 *CR* 參數平均值變動圖，縱軸為 *CR* 值，橫軸為代數。

由前三個問題的實驗結果，可發現 *CR* 值對效能的影響較為劇烈，故該問題只對 *CR* 進行調整，*F* 值一樣設定為 0.5。

表 4-22　UF7 問題調整 *CR* 值測試結果，最佳 IGD 值以粗體表示。

	IGD 值表				
CR 值	0.4	0.5	0.6	0.8	**1**
IGD 值	0.062481	0.0506329	0.044541	0.0175326	**0.00227**

表 4-23　F8 問題調整 *CR* 值測試結果與表 4-22 最佳 IGD 值之差距。

	與最佳 IGD 值之差				
CR 值	0.4	0.5	0.6	0.8	1
IGD 值	2653%	2131%	1862%	672%	0%

經實驗後可知最佳 *CR* 值為 1。

由四個問題的實驗結果可知，我們所找到的整體效能最佳的參數組合為 $F = 0.5$，$CR = 1$，但大部份機制都無法調整到最佳的參數值，我們推測可能有兩個原因，一個是資訊深度的不足，一個是只考慮更新子代次數不足以對參數做正確的調整。

資訊深度不足的問題可以圖 4-18 表示，若我們每一代都以該參數值產生良好子代的次數決定調整方向，則我們分別會在 1 到 4 代時往 f_4、f_2、f_5 和 f_1 進行調整，但是以累積的產生良好子代次數來說，最後我們會發現 f_3 才是整體來說最好的參數值，因此參數才會一直無法調整到實驗求得的最佳參數。

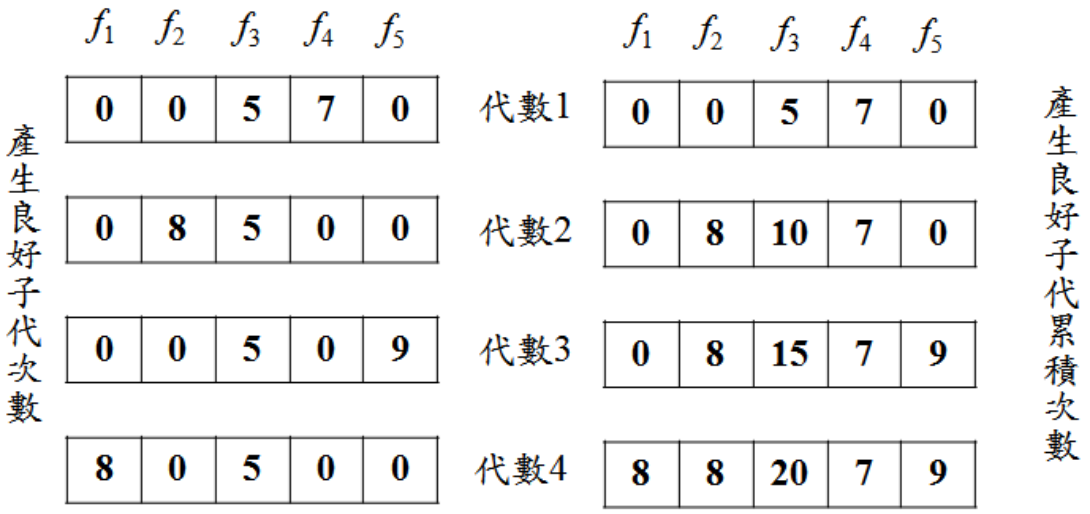

圖 4-18 資訊深度問題示意圖。

為了確認是否是資訊深度造成的問題，我們將 *F* 值固定為 0.5，*CR* 值分別設為 0.1 至 1，各執行 F2、F7、F8 與 UF7 問題 20 次，計算平均一次所產生的良好子代次數與所更新的適應度差值之總和，若 *CR* 值為 1 時所更新的子代次數統計下來較 *CR* 值為 0.5 來得高，那麼參數調整的失常就極有可能是資訊深度不足所造成的。實驗結果如下。

表 4-24 F2 問題之各參數值產生良好子代次數紀錄表。

CR 值	產生良好子代次數	更新適應度差值總和	差值總和/次數
0.4	58146.1	284.696	0.004896
0.5	49822.1	284.96	0.00572
0.6	46340	283.923	0.006127
0.7	43716.7	284.352	0.006504
0.8	33088.8	285.628	0.008632
0.9	31532	287.337	0.009113
1	35521.7	284.584	0.008012

表 4-25　F7 問題之各參數值產生良好子代次數紀錄表 。

CR 值	產生良好子代次數	更新適應度差值總和	差值總和/次數
0.4	46550.3	564.3	0.012122
0.5	45030.8	563.893	0.012522
0.6	42153.3	566.943	0.01345
0.7	34702.2	575.356	0.01658
0.8	22189.9	577.418	0.026022
0.9	18778.8	577.481	0.030752
1	18840.7	573.352	0.030432

表 4-26　F8 問題之各參數值產生良好子代次數紀錄表 。

CR 值	產生良好子代次數	更新適應度差值總和	差值總和/次數
0.4	44724.9	388.334	0.008683
0.5	42932.8	396.57	0.009237
0.6	40699.4	400.19	0.009833
0.7	39471.1	408.122	0.01034
0.8	34631.1	410.052	0.011841
0.9	26947.8	417.548	0.015495
1	25927.8	414.792	0.015998

表 4-27　UF7 問題之各參數值產生良好子代次數紀錄表 。

CR 值	產生良好子代次數	更新適應度差值總和	差值總和/次數
0.4	55767.9	285.234	0.005115
0.5	44404.1	294.102	0.006623
0.6	40428.2	292.814	0.007243
0.7	36874.2	290.352	0.007874
0.8	36841.3	289.378	0.007855
0.9	31004.7	288.834	0.009316
1	34245.6	292.868	0.008552

由表 4-24 至表 4-27 可看出，其共通點為越高的 *CR* 值更新子代次數通常越少，但每次更新的適應度相差越多，也就是說解的質比低 *CR* 值來得好。雖然

如此，各參數設定所更新的適應度差值總和其實是差不多的，那麼當解的質不同時究竟會造成什麼樣的影響便是我們關心的重點，為了易於觀察我們將表 4-24 至表 4-27 的實驗結果畫出最後一代的族群分布圖來進行比較。

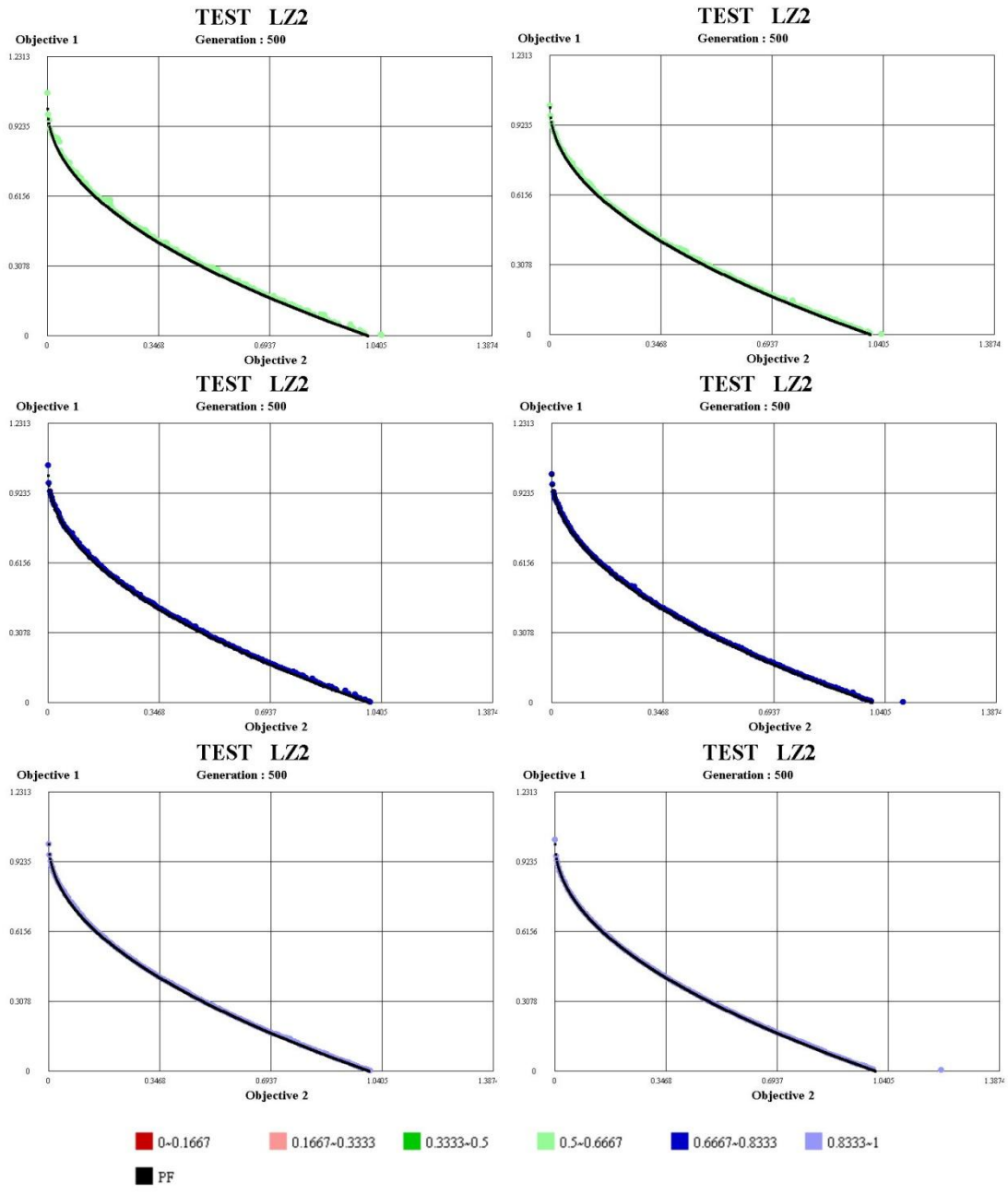

圖 4-19　以不同 *CR* 值解 F2（LZ2）問題，第 500 代的族群分布圖，由上至下，由左至右的 *CR* 值分別為 0.5、0.6、0.7、0.8、0.9、1。

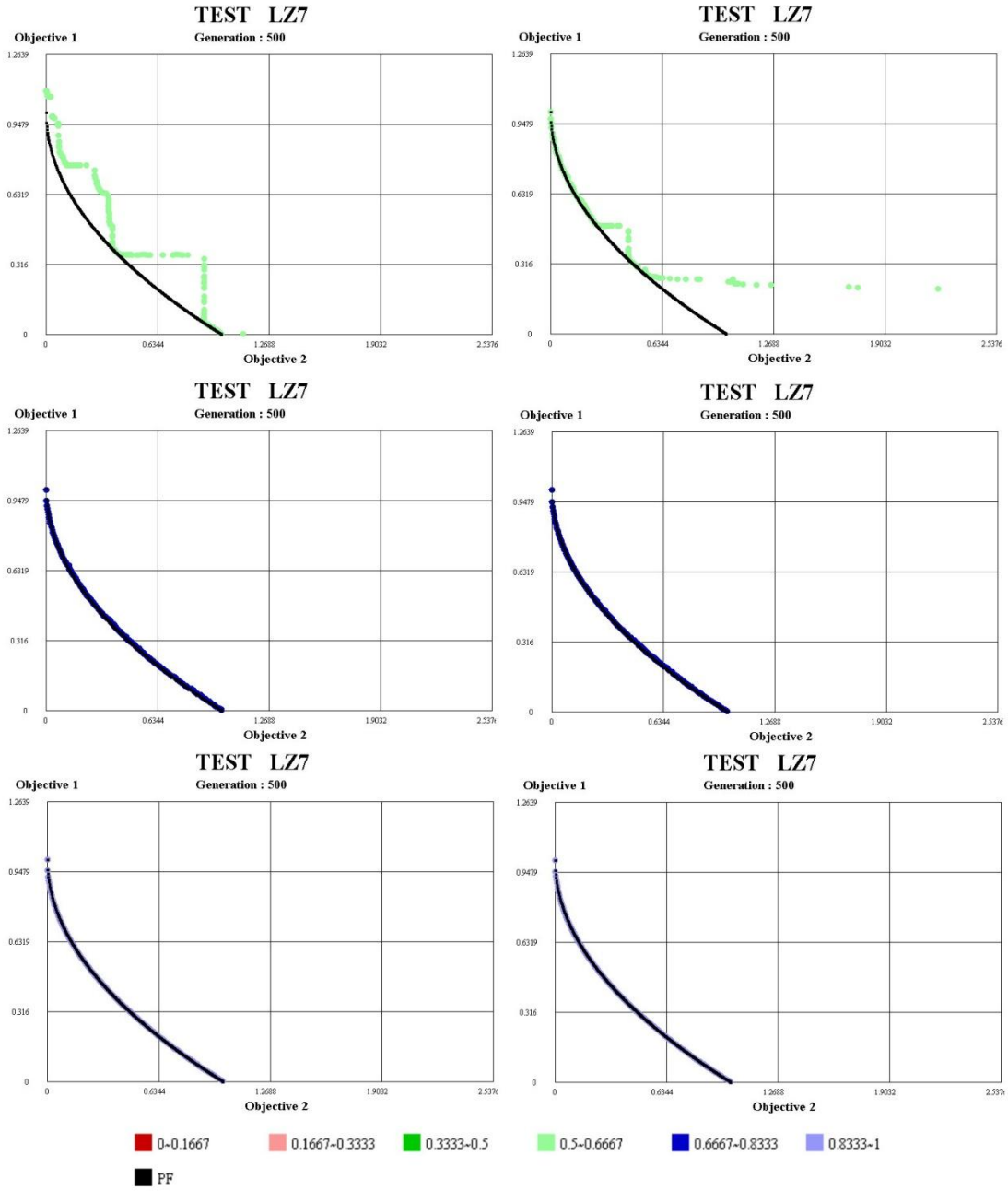

圖 4-20　以不同 *CR* 值解 F7（LZ7）問題，第 500 代的族群分布圖，由上至下，由左至右的 *CR* 值分別為 0.5、0.6、0.7、0.8、0.9、1。

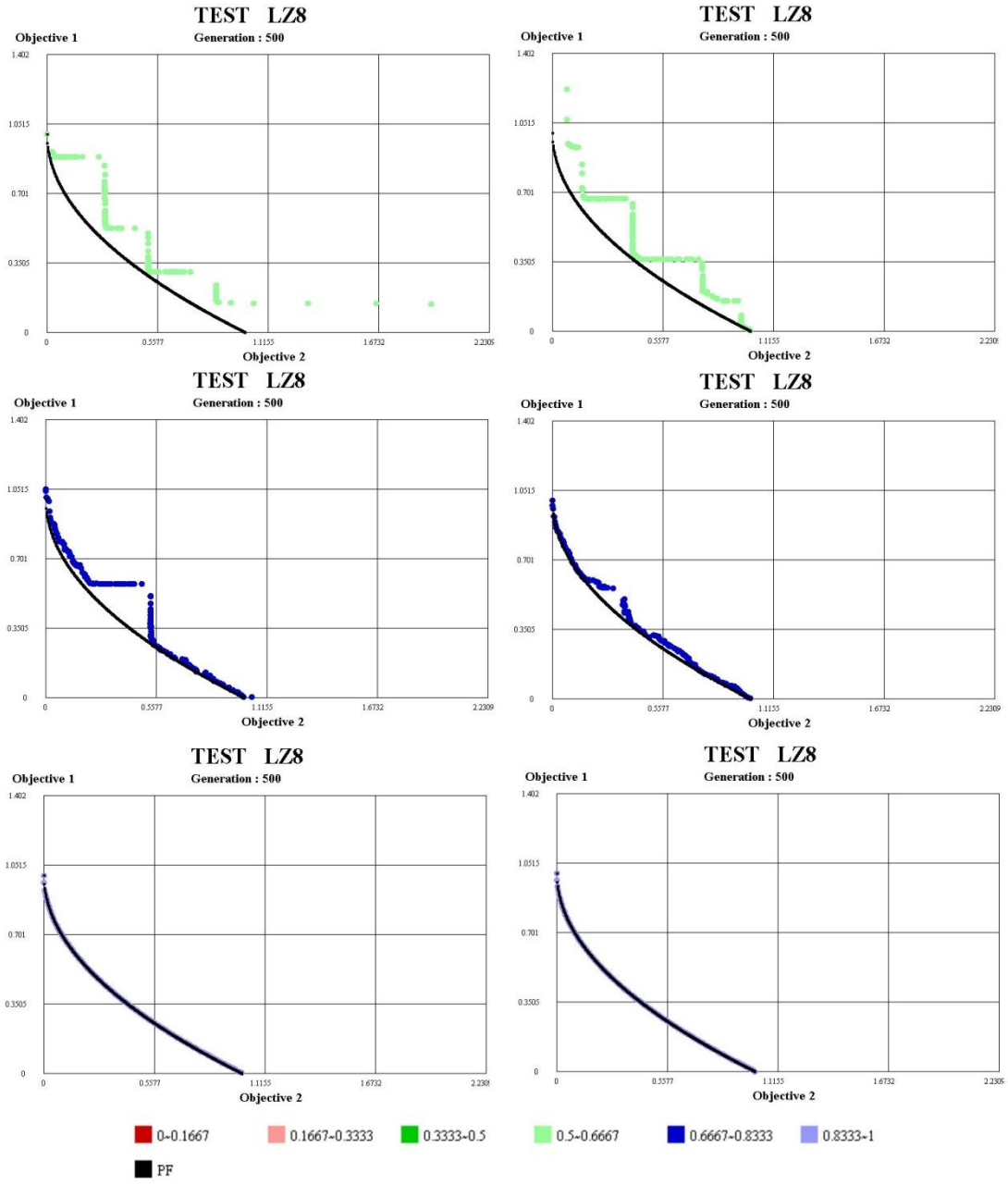

圖 4-21　以不同 *CR* 值解 F8（LZ8）問題，第 500 代的族群分布圖，由上至下，由左至右的 *CR* 值分別為 0.5、0.6、0.7、0.8、0.9、1。

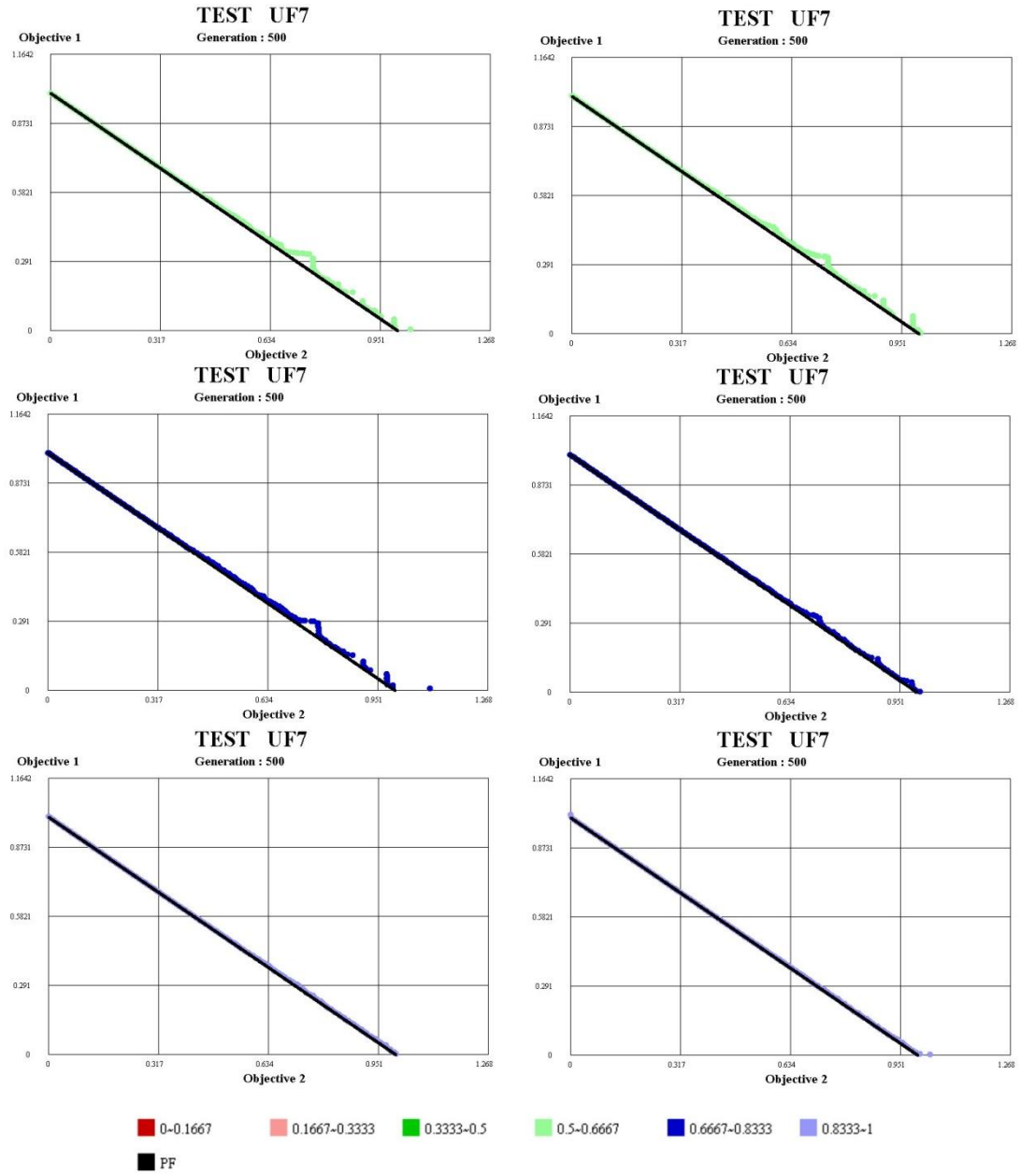

圖 4-22　以不同 *CR* 值解 UF7 問題，第 500 代的族群分布圖，由上至下，由左至右的 *CR* 值分別為 0.5、0.6、0.7、0.8、0.9、1。

由圖 4-19 至圖 4-22 可知，*CR* 值影響了族群收斂的能力，尤其是 F7 與 F8 問題最為明顯，當 *CR* 值接近 0.5 時族群較難靠近柏拉圖前緣，但是若以更新子代次數來評估的話，便很容易因為經常更新而往 0.5 來調整，事實上族群可能只是在難收斂區域的邊界上移動而沒有真的往柏拉圖前緣收斂，這便是我們所

推測的第二個原因，評估參數是否良好的標準太少，若要考慮適應度的話，不同 *CR* 值的更新適應度差值和其實相差不明顯，需要除以更新子代次數後才較易看出差別，如果要使用此項資訊來調整參數勢必要增加資訊深度來提供足夠可靠的紀錄。

由 UF5、F2、F7、F8 與 UF7 的實驗，我們做出了幾個結論，一個是如果要提高 MOEA/D-APC 在 UF5 問題上的效能，必須從 *CR* 值的調整著手，然而經過資訊深度問題的測試後，我們知道大部份機制所賴以調整的資訊有所不足，只使用所更新的良好子代次數難以判斷一個參數是否真的對族群來說有良好的影響，可考慮的改良方式有二，一是使用與 SaNSDE 相似的做法，利用參數所更新的適應度差做為權重值來控制參數的影響力，另一種方法則是從族群的形狀與演化方向著手，當我們經過固定代數發現族群的演化方向不平均，那麼或許就可以考慮提高 *CR* 值以突破僵局。

第五章 結論與未來發展

本論文以 MOEA/D-AMS 為主體，加入 NSDE、jDE、JADE、SDE、SaDE、SaNSDE、SspDE 與 ADEA 八種演算法對 DE 中的 *F* 與 *CR* 參數調整之機制，利用所挑選的 17 個測試問題與自行開發的參數區間族群分布圖顯示軟體，對這八個機制與 MOEA/D-APC 做一次比較性的實驗，藉著其中五個問題的探討來提出研究者往後對於參數調整機制可考慮的改進方向。

實驗中所討論的五個問題，主要是基於效能的差異而挑選，其中 UF5 為 MOEA/D-APC 效能最差的問題，而 F2、F7、F8 與 UF7 則是其他八種機制普遍效能較差，本論文嘗試找出各機制於這五個問題上效能不理想的原因，整理出兩個要點，一個是必須對 *F* 與 *CR* 參數一起進行調整，另一個是大部份機制只利用了更新子代次數與適應度差來調整參數，然而實驗結果說明只使用這兩個條件是不太足夠的，因此我們在製出族群分布圖後，認為調整標準可加入族群的方向性來考慮，而這也是未來可以對 MOEA/D-APC 進行改良的目標

參考文獻

[1] C. N.Chen,“A Multiobjective Evolutionary Algorithm with Adaptive Parameter Control,”M. S. thesis, Department of Computer Science and Information, National Taiwan Normal University, Taipei, Taiwan, 2011.

[2] Y. P. Lai,“Multiobjective Optimization Using MOEA/D with a New Mating Selection Mechanism,” M. S. thesis, Department of Computer Science and Information, National Taiwan Normal University, Taipei, Taiwan, 2010.

[3] Q. Zhang, H. Li,“MOEA/D: A Multiobjective Evolutionary AlgorithmBased on Decomposition”*IEEE Transactions on Evolutionary Computation*, Vol. 11, No. 6, 2007.

[4] K.Price, R. M.Storn, J. A.Lampinen,*Differential Evolution*: *A Practical Approach to Global Optimization*,(Natural Computing Series)Springer–Verlag New York, Inc., Secaucus, NJ,2005.

[5] Z. Yang, X. Yao, J. He, “Making a Difference to Differential Evolution,” *Advance in Metaheuristics for Hard Optimization*, Vol. 1, pp. 397– 414, 2008.

[6] J. Brest, S. Greiner, B. Boskovic, M. Mernik, V. Zumer, “Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems,” *IEEE Transactions on Evolutionary Computation*, Vol.

10, No. 6, pp. 646–657, 2006.

[7] Q. Pan, P.N. Suganthan, L. Wang, L. Gao, R. Mallipeddi, "A Differential Evolution Algorithm with Self-adapting Strategy and Control Parameters," *Computers & Operations Research*, Vol. 38, No. 1, pp. 394–408, 2011.

[8] A.K. Qin, P.N. Suganthan, "Self-adaptive Differential Evolution Algorithm for Numerical Optimization," *IEEE Congress on Evolutionary Computation*, Vol. 2, pp. 1785–1791, 2005.

[9] J. Brest, V. Zumer, M.S. Maucec, "Self-Adaptive Differential Evolution Algorithm in Constrained Real-Parameter Optimization," *IEEE Congress on Evolutionary Computation*, Vol. 1, pp. 215–222, 2006.

[10] Z. Yang, K. Tang, X. Yao, "Self-adaptive Differential Evolution with Neighborhood Search," *IEEE Congress on Evolutionary Computation*, Vol. 1, pp. 1110–1116, 2008.

[11] J. Zhang, A. C. Sanderson, "JADE: Adaptive Differential Evolution with Optional External Archive," *IEEE Transactions on Evolutionary Computation*, Vol. 13, No. 5, pp. 945–958, 2009.

[12] J.Zhang, A.C. Sanderson, "Self-adaptive Multi-objective Differential Evolution with Direction Information Provided by Archived Inferior Solutions,"*IEEE*

Congress on Evolutionary Computation, Vol. 1, pp. 2801–2810, 2008.

[13] W. Gong, Z. Cai, C. X. Ling, H. Li "Enhanced Differential Evolution With Adaptive Strategies for Numerical Optimization," *IEEE Transactions on Systems, man, and Cybernetics, Part B*, Vol. 41, No. 2, pp. 397–413, 2011.

[14] M. G.H. Omran, A. Salman, A. P. Engelbrecht, "Self-adaptive Differential Evolution," *Computational Intelligence and Security*, Vol. 3801, pp.192–199, 2005.

[15] W. Qian, A. Li, "Adaptive Differential Evolution Algorithm for Multiobjective Optimization Problems," *Applied Mathematics and Computation*, Vol. 201, No. 1–2, pp. 431–440, 2008.

[16] K.Deb,S. Agrawal,A. Pratap,T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II,"*IEEE Transactionson Evolutionary Computation*, Vol. 6,No.2, pp. 182–197, 2002.

[17] J. Liu, J. Lampinen, "A Fuzzy Adaptive Differential Evolution Algorithm," *Soft Computing*, Vol. 9, No. 6, pp. 448–462, 2005.

[18] H.Li, Q. Zhang,"Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II,"*IEEE Transactionson Evolutionary Computation*,Vol. 13,No.2, pp.284–302,2009.

[19] Q.Zhang, A.Zhou,S. Zhao,P. N. Suganthan, W. Liu,S. Tiwari,"Multiobjective Optimization Test Instances for the CEC 2009 Special Session and Competition,"*The School of Computer Science and Electronic Engineering*, *University of Essex* (*Technical Report CES-487*), 2008.