

# 穩健多目標基因演算法應用於流程型工廠之排程研究

廖麗滿 黃敬仁 林志諭

國立勤益科技大學工業工程與管理系

## 摘要

本研究以基因演算法為基礎發展啟發式演算法，求解最大完工時間、總流程時間、總延遲時間為目標之流程型工廠排程問題，分析交配機制、強化策略，以及分散策略於多目標基因演算法之效果。首先，運用 OPX、2PX、SJOX 三種交配機制，分別分析其演算品質。然後，求解時間與品質的權衡下，執行強化策略，其中強化策略包含選擇優良解進行局部搜尋、使用簡易啟發式演算法加入優良解，以及利用人造解演算機制加入優良解。並於解群體的分散度較低時，運用分散策略產生部份新解。為得到搜尋策略與參數的最佳組合，應用變異數分析法，且以綜合相對誤差為指標，獲得較佳之柏拉圖最佳解。實驗結果顯示，本演算法可求得更有效的柏拉圖最佳解。

**關鍵詞：**基因演算法，流程型工廠，排程，多目標，柏拉圖最佳解。

## ROBUST MULTI-OBJECTIVE GENETIC ALGORITHM FOR FLOWSHOP SCHEDULING PROBLEMS

Li-Man Liao Ching-Jen Huang Zhi-Yu Lin

*Department of Industrial Engineering and Management  
National Chin-Yi University of Technology  
Taichung, Taiwan 411, R.O.C.*

**Key Words:** genetic algorithm, flowshop, scheduling, multi-objective, Pareto optimal solutions.

## ABSTRACT

This paper proposes a GA-based algorithm for flowshop scheduling problems (FSP) with multiple objectives which are makespan, total tardiness and total flow time. The algorithm analyzes the effects of crossover, intensification and diversification strategies in multi-objective genetic algorithms (MOGA). Firstly, OPX, 2PX, and SJOX crossover mechanisms are applied and their performance analyzed. Then, considering the tradeoffs of run time and solution quality, the GA-based heuristic applies three intensification strategies to rapidly search for good solutions. The strategies include local search, simple heuristics, and an artificial solution production mechanism. Additionally, if the diversity value falls below a given threshold value, a diversification strategy is applied where part of the population is regenerated. In order to obtain a good search strategy and calibrate the parameters of GA-based algorithms, analysis of variances (ANOVA) is adopted. The optimal combination of GA parameters is found and the better Pareto optimal solution set is obtained. Computational results show that the heuristic can find more effective Pareto optimal solutions.



## 一、緒論

組裝生產線常見於流程型工廠之生產環境，而生產排程爲了兼顧交貨能力，以及生產成本最小化的目標，故排程準則經常是多元的，例如機器利用率（utilization）、交貨（delivery）準時性、在製品存貨（work in process）水準等。生產排程指標中，最大完工時間（makespan）是所有工作的總完成時間，所有工作的完成時間愈小，表示機器利用率愈大；延遲時間（tardiness）是交貨準時性重要指標，工作總延遲時間愈小表示企業的準時交貨的能力佳；流程時間（flow time）是工作在生產線上的時間，工作的流程時間愈小，表示在製品存貨水準愈低。至於指標的重要程度，實務上則由決策者對生產系統認知與企業目標決定之。

2001年，T'kindt and Billaut [1] 針對多目標排程做廣泛性的調查研究，強調多目標排程的重要性。多目標最佳化起源於柏拉圖最佳解（Pareto optimal solutions），柏拉圖最佳解中的任意解是無法凌駕的（non-dominated）。本研究是以基因演算法（genetic algorithm, GA）爲基礎發展演算法，求解最大完工時間（makespan,  $C_{max}$ ）、總流程時間（total flow time,  $F$ ）與總延遲時間（total tardiness,  $T$ ）三目標的流程型工廠排程問題。本演算法試圖加入強化（intensification）與分散（diversification）策略，且利用變異數分析法（analysis of variances, ANOVA），找出適合的GA參數。以個別與綜合相對誤差爲衡量所求得之柏拉圖最佳解，作爲演算法之績效衡量指標。

本研究其餘章節安排如下，第二節爲文獻探討，針對多目標之排程問題的搜尋技術，以及GA應用強化策略與分散策略之搜尋機制，加以探討；第三節建構混合型基因演算法（hybrid genetic algorithm, HGA）之演算架構，包含整體的演算流程，以及參數說明；第四節則以ANOVA選擇HGA演算法的重要參數；第五節則進行HGA之績效分析；第六節是爲本研究之成果。

## 二、文獻探討

### 1. MOGA之相關文獻

Schaffer [2] 提出以向量評估基因演算法（vector evaluated genetic algorithms, VEGA）求解多目標規劃問題。VEGA將配對池分成數個子配對池，每一個子配對池對應到一個目標函數，從母代中選取單一目標優良解，複製解到相對應目標的子配對池中，直到所有的子配對池的解滿足爲止，此複製方式可使個別目標表現優良的解，具有較高之選取機率。

Ishibuchi and Murata [3] 發展GA與局部搜尋法（local search）結合的混合型演算程序，求解多目標流程型工廠排程問題，稱之爲genetic local search（GLS）演算法。該

演算法以隨機方式產生各目標權重，計算解的適合度值。以輪盤法爲選取機制，進行交配（crossover）與突變（mutation），將交配、突變後的解與非凌駕的解合併，然後進行局部搜尋。該演算法之執行時間與VEGA演算法相差不多，但在搜尋柏拉圖最佳解的數量與品質上，都比VEGA優異。

Deb等人[4]評估多目標問題時，運用非凌駕解排序方法，並結合GA架構，提出非凌駕解排序之基因演算法（non-dominated sorting genetic algorithm II, NSGA-II）。該研究將母體的解依照凌駕解個數，以及解被凌駕的次數，進行分群，然後，計算群內每一個解的群集距離（crowding distance），作爲衡量此解的適合度值。複製程序中，將母代與子代的解予以合併，進行非凌駕解排序，選取與群體大小相同之較優解，作爲下一代解。該研究顯示NSGA-II具有較佳擴散性的解，且快速收斂在柏拉圖最佳解邊界上。

Arroyo and Armentano [5] 應用NSGA-II演算程序，在多目標流程型工廠環境下，建構GLS演算法，其目的在於提供近似柏拉圖最佳解（approximate Pareto optimal solutions）。透過NSGA-II演算程序計算群體內每一個解的群聚距離，依此距離值計算適合度值。另外，Chang等人[6]提出了漸進式權重（gradual-priority weighting, GPW）之演算架構，以漸進式權重計算適合度值，權重計算方式是以演化代數來決定目標權重，在預先決定的目標順序下，以GPW引導GA搜尋解空間的方向，進行全域解的搜尋。GPW在搜尋柏拉圖最佳解個數上，比變動權重法來得好。

### 2. 應用強化策略之相關文獻

GA應用菁英政策在複製程序[3-5]，合併母代與子代解，進行非凌駕解排序，選取群體大小之較優解，作爲下一代解。Chang等人[7]在GA架構上加入兩個強化策略：完全複製（duplication）與人造解的建構（fabrication）機制。完全複製方法是將母代中優良解直接複製到子代裡，可以加速收斂過程，但不致陷入局部最佳解的情況；人造解的建構是從菁英解集合裡，透過統計方式取得每一個解的染色體結構，再以機率方式產生新的人造解，加速搜尋較佳解的速度。該研究證實這兩個策略在單目標搜尋上，都有加速收斂效果，能夠獲得品質較佳的柏拉圖最佳解。

Chang等人[8]儲存菁英解的染色體結構，以快速收斂尋找更佳之人造染色體（artificial chromosome），作爲人造解的產生機制。人造解機制分別建構在簡易基因演算法（simple genetic algorithm, SGA）與NSGA-II演算法上，可於流程型工廠排程環境下，求解單一目標或多目標的問題。在SGA與NSGA-II加入人造染色體機制，皆可加速基因演算法收斂速度，並改善解的品質。Chang等人[9]也結合子群體基因演算法（sub-population genetic algorithm, SPGA）與探勘基因結構（mining gene structure,



MGS) 技術，發展 MGSSPGA (sub-population genetic algorithm with mining gene structures) 演算法。2009 年，Zhang 等人 [10] 不但考量菁英解的染色體結構，並且將菁英解的品質加以排序，列入求解人造解的演算機制，稱之為 weighted simple mining gene structure (WSMGS)。以上各研究的主要目的，皆在快速產生新的優良解，減少搜尋時間。

### 3. 應用分散策略之相關文獻

2005 年，Ruiz 等人 [11] 研究具有相依設置時間之流程型工廠的問題，提出有效的 GA 演算程序。2006 年，Ruiz 等人 [12] 也在流程型工廠，發展一個穩健 (robust) 的 GA 演算架構，該研究的起始解是採用修正 NEH 演算法，交配方式是以 similar block order crossover (SBOX)、similar job order crossover (SJOX)、similar block 2-point order crossover (SB2OX)、similar job 2-point order crossover (SJ2OX)、partial-mapped crossover (PMX) 五種機制，結果顯示 SB2OX 為最優，而 SJ2OX 次之。而突變方式是以交換 (swap)、相鄰交換 (adjacent swap)、移動 (shift) 三種機制，結果是以移動方式為最佳。另外，該研究為了避免多代演化後，陷入局部最佳解的情況，提出了一項重新開始機制 (restart scheme)，當連續數代的子代解無法比母代解為佳時，即進行重新開始搜尋機制，以跳脫區域最佳解的窘境。

Pasupathy 等人 [13] 提出 Pareto genetic algorithm with an archive of non-dominated solutions subjected to a local search (PGA-ALS)，求解流程型工廠排程問題。該演算法先運用非凌駕解排序方法，將群體分群，做為解的第一級衡量準則，然後，再計算每一群中解的群集距離，將解的群集距離值遞減排序，作為解的第二級衡量準則。該研究採用競賽法為選取方式，隨機由群體選取兩個解進行第一級衡量準則比較，當兩個解的第一級衡量準則相同時，再進行第二級衡量準則的比較。在每一代演算後，進行 LS，以改善與更新非凌駕解的品質。

多目標的研究在建立柏拉圖最佳解時，若有效解集合的相似解過多，將造成更新程序耗時，導致 GA 的演算效率降低，故更新柏拉圖最佳解時，應兼顧效率問題，故 Vallada and Ruiz [14] 在 GA 演算程序中，利用解群體的離散值 (diversity value)，判斷菁英解的相似度，作為啟動重新開始機制之依據。

## 三、混合型演算法建構

### 1. 演算流程

本研究在 GA 程序中，試圖加入強化策略與分散策略，稱之為混合型基因演算法 (hybrid genetic algorithm, HGA)。HGA 起始解參考 NEH 與修正 NEH 演算法，產生

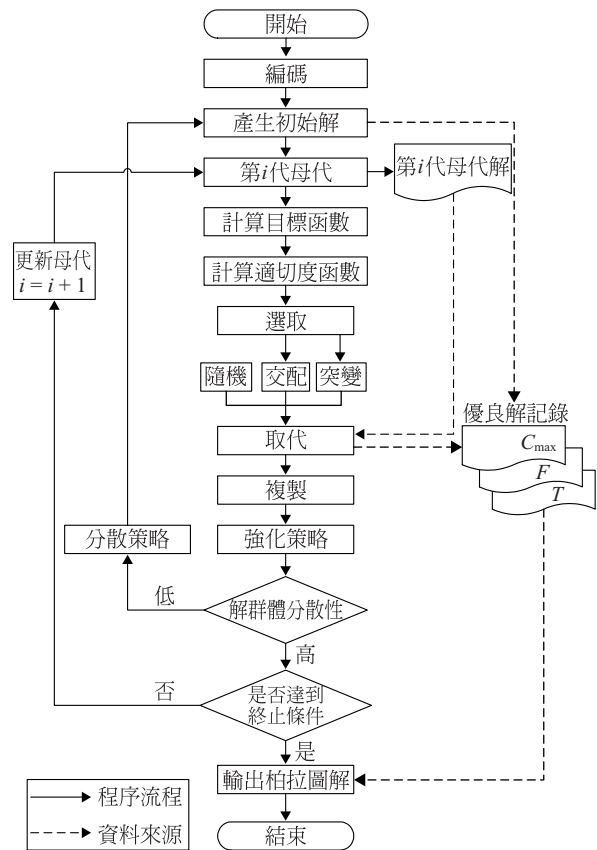


圖 1 HGA 演算流程圖

起始解群體；然後，應用 GPW 程序決定各目標之權重，計算解的適合度值，有效地搜尋解空間。求解的過程中，加入局部搜尋程序、加入優良解與加入人造解等強化策略，期望在搜尋解空間時，能更為快速找出較佳的解。

另外，在解群體的相似度偏高時，應用重新開始之分散策略，避免搜尋解陷入局部最佳解。而在記錄與更新柏拉圖最佳解的程序中，分別採用個別目標與綜合目標加以記錄。詳細的演算流程如圖 1 所示。

以下針對 HGA 演算步驟，以及重要的參數說明之。

- (一) 編碼 (coding)：染色體表示工作序列，每一基因表示一個工作。
- (二) 起始解：應用 NEH 與修正 NEH 演算法建構起始解，其中 NEH、NEH\_EDD、NEH\_WPT 演算法，各產生一解；而修正 NEH 演算法：NEH\_R(r)<sub>Cmax</sub>、NEH\_R(r)<sub>T</sub>、NEH\_R(r)<sub>F</sub>，各產生 35 個解，故起始解個數共有 108 個。
- (三) 目標函數：因為  $C_{max}$ 、 $F$ 、 $T$  三個值的具有差異性，以其平均值調整目標值，故三個目標函數分別為  $f_1(x) = 5 \times C_{max}$ 、 $f_2(x) = 1 \times F$ 、 $f_3(x) = 2 \times T$ 。
- (四) 適合度函數 (fitness function)：透過 GPW [6] 計算每一代各目標的權重值。
- (五) 選取 (selection)：採用競賽法為選取機制。



- (六) 交配 (crossover)：採用單點交配 (OPX)、兩點交配 (2PX)、相似工作順序交配 (SJOX) 三種機制。
- (七) 突變 (mutation)：採用移動 (shift) 突變機制。
- (八) 複製 (reproduction)：將交配與突變所產生的解 (稱為子代) 與原母代解合併，進行非凌駕解排序，選取較優解作為下一代解群體。
- (九) 強化策略：強化策略有三：(1) 執行部份較優解之局部搜尋；(2) 每一代由修正 NEH 演算法產生數個優良解，加入下一代之複製程序；(3) 以 WSMGS 人造解演算機制求出入造解，加入解群體中。
- (十) 分散策略：判斷解群體的差異程度，是否必須執行重新開始之搜尋機制。若是，則以相對誤差百分比為指標，保留群體大小的 20%；以移動突變機制產生群體大小的 40%；其他由隨機產生。
- (十一) 停止條件：達到最大演化代數則停止。

#### 四、HGA 參數實驗分析

以 Visual Basic 撰寫 HGA 演算法，在 PC Core 2 Quad 2.66 GHz 下執行。實驗例子中的工作數  $n$  設定為 60；機器數  $m$  分為 5 和 7；工作處理時間為  $U(1, 99)$ ；工作延遲因子為 0.4，而到期日離散因子為 0.6、1.2 兩種，故本研究設定實驗例子組合共有 4 種，分別為 M5-T04R06、M5-T04R12、M7-T04R06、M7-T04R12。而各組合問題分別進行 20 個例子實驗，每個例子重覆執行 5 次。績效的指標分為個別與綜合相對誤差百分比 (relative difference percentage, RDP)，公式如下：

##### 1. 個別績效指標

$$RDP_k = \min_i \left\{ \frac{O_{k,i} - \min_l O_{k,l}}{\max_l O_{k,l} - \min_l O_{k,l}} \right\} \times 100\%, k = 1, 2, 3 \quad (1)$$

##### 2. 綜合績效指標

$$RDP = \min_i \left\{ \sum_{k=1}^3 \frac{O_{k,i} - \min_l O_{k,l}}{\max_l O_{k,l} - \min_l O_{k,l}} \right\} \times 100\% \quad (2)$$

$O_{1,l}$ 、 $O_{2,l}$ 、 $O_{3,l}$  分別為第  $l$  個有效解之  $C_{\max}$ 、 $F$ 、 $T$  值。

GA 參數實驗分為五項，首先，針對交配方式與交配率進行實驗；選擇較佳的交配方式後，再執行交配率與突變率組合實驗；考量時間與有效解的品質的前提下，決定停止準則，以建立基本 GA 架構；再試圖加入強化策略，以及分散策略等機制，分析其演算績效。以下依序進行實驗與 ANOVA 分析。

##### 實驗一：交配方式之選擇

GA 參數與水準設定如表一所示，兩點交配 (2PX)、相似工作順序交配 (SJOX)、交配方式有單點交配 (OPX)，

表一 交配方式選擇之 GA 參數表

參數	設定方式與水準
群體大小	108
總演化代數	421 (切割數 20)
突變率 (Pm)	0.2
交配方式	2PX、SJOX、OPX
交配率 (Pc)	0.0、0.1、0.2、0.3、0.4、0.5、0.6、0.7、0.8、0.9、1.0

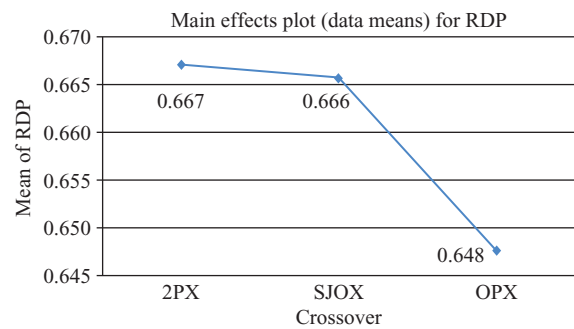


圖 2 交配方式之主效果圖

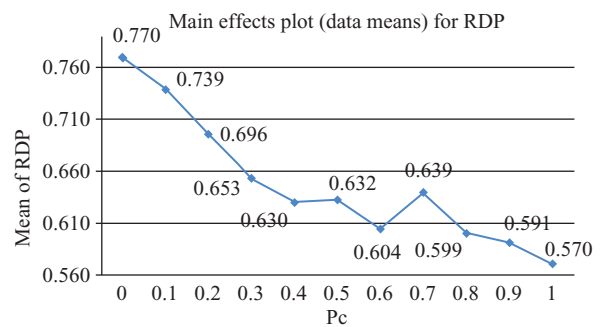


圖 3 OPX 交配率之主效果圖

搭配 0~1.0 之交配機率。

實驗結果顯示：交配方式與交配率之檢定 P 值分別為 0.001 與近似 0，表示兩者有顯著差異。從交配方式的主效果圖 (圖 2)，可看出 OPX 是最佳的交配機制。

執行 OPX 交配機制時，交配率越大時，相對誤差值越小，如圖 3 所示。在求解時間與品質的權衡下，較適的交配率應於 0.6~0.8 之間。

##### 實驗二：交配率與突變率組合之決定

以 OPX 交配方式與移動突變方式，執行實驗二，決定較佳之交配率 (Pc) 與突變率 (Pm)，兩者機率和設為 1。實驗結果顯示各水準有顯著差異。從圖 4 看出，交配率與突變率較佳組合為 0.75、0.25，所求得之綜合相對誤差最小。



表二 不同演化代數（切割數）之執行時間（單位：秒）

問題型式	演化代數（切割數）				
	157(12)	273(16)	421(20)	601(24)	813(28)
M5-T04R06	2.64	3.49	4.59	5.94	7.52
M5-T04R12	2.62	3.49	4.62	5.98	7.58
M7-T04R06	3.07	3.96	5.12	6.53	8.17
M7-T04R12	3.14	4.07	5.18	6.58	8.21
平均	2.87	3.75	4.88	6.26	7.87

表三 不同 LS 參數之執行時間（單位：秒）

問題型式	$(g_1, x, m)$				
	None*	(10, 1, 10)	(30, 3, 10)	(50, 5, 10)	(100, 10, 10)
M5-T04R06	4.59	28.84	40.36	43.97	56.91
M5-T04R12	4.62	48.77	67.60	69.23	91.81
M7-T04R06	5.12	32.63	56.28	54.09	74.83
M7-T04R12	5.18	32.93	45.26	54.17	68.47
平均	4.88	35.79	52.37	55.36	73.01

註：None\*表示沒有執行局部搜尋（LS）程序

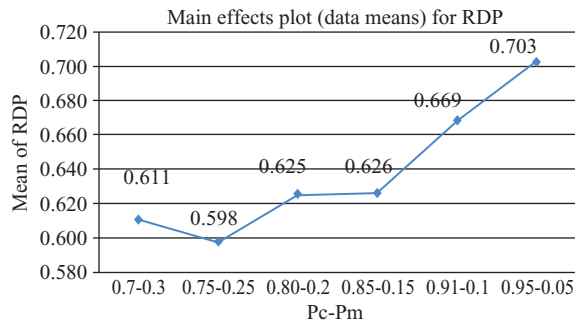


圖 4 交配率與突變率組合之主效果圖

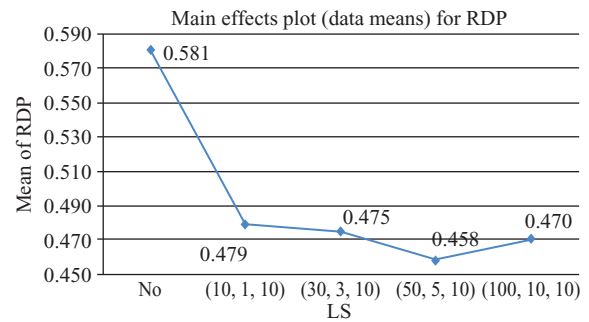


圖 6 LS 之主效果圖

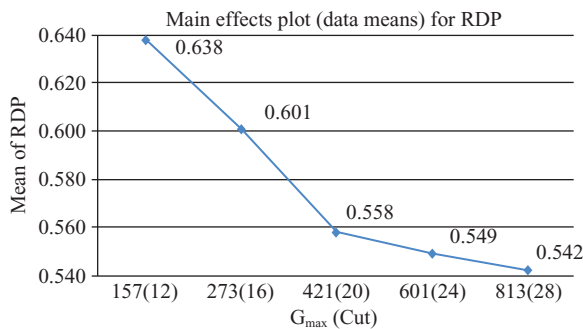


圖 5 演化代數（切割數）之主效果圖

實驗三：停止準則

應用 GPW 計算每一代各目標的權重值，每一演化代數 ( $G_{max}$ ) 相對於特定切割數 (Cut)。停止準則有五個水準，以演化代數（切割數）表之，為 157(12)、273(16)、421(20)、601(24)、813(28)。執行的結果如圖 5，而執行時間列示於表二。

由圖 5 演化代數之主效果圖看出，演化代數  $G_{max}$  為 421 時，綜合相對誤差為 0.558%，若再增加演化代數，綜合相對誤差降低幅度相當緩慢，故在求解時間與品質的權衡下，本研究將演化代數設為 421，即切割數為 20。

實驗四：強化策略

本研究探討三種強化策略對求解品質的影響，參數水準與實驗結果說明如下。

1. 局部搜尋法 (local search, LS)

局部搜尋法的移步方式，直接影響執行時間，在時間與品質考量下，初步實驗所得之工作的最大移步距離 ( $d$ ) 約為 10。LS 另外兩個參數為  $g_1$  與  $x$ ，表示每經過  $g_1$  代，則選取母代中最佳的  $x$  個解，進行 LS。最大移步為 10 的前題下，LS 搜尋的時機與次數以  $(g_1, x, d)$  表示，共有四個水準：(10, 1, 10)、(30, 3, 10)、(50, 5, 10) 與 (100, 10, 10)。實驗結果顯示，執行 LS 有明顯的改善，如圖 6 所示。但四個水準沒有顯著差異，故選擇時間較短者，如表三所示。故每 10 代選擇一個最佳解進行 LS，即為 (10, 1, 10) 組合。

2. 每一代使用修正 NEH 演算法產生數個解

以當時的目標權重值選擇適當之修正 NEH 演算法，每一代所加入數個解，其水準有 0、5、10、15、20 五個水準，實驗結果經 ANOVA 分析，顯示 P 值為 0.810，表示沒有顯著差異，故本研究放棄此強化機制。

3. WSMGS 人造解演算機制加入優良解

應用 Zhang 等人 [10] 提出的 WSMGS 演算程序，先排序菁英解，再以加權方式求得人造解，加入演化的群體。其主要參數為每隔  $g_2$  代，執行加入人造解， $g_2$  的水準有 0、1、2、3、4、5，實驗結果經 ANOVA 分析，顯示 P 值為 0.924，沒有顯著差異，故本研究放棄此強化機制。

實驗五：分散機制

應用 Vallada and Ruiz [14] 所提出判斷解群體的分散



表四 HGA 與 CH 之  $C_{max}$  相對誤差

機器數	工作數	$C_{max}$ 較佳個數		$RPD_1$ (%)	
		HGA	CH	HGA	CH
5	20	40	1	0	0.269
	40	40	1	0	0.232
	60	40	1	0	0.226
	80	40	3	0	0.216
7	20	40	1	0	0.410
	40	40	0	0	0.356
	60	40	0	0	0.312
	80	40	0	0	0.306

表六 HGA 與 CH 之  $T$  相對誤差

機器數	工作數	$T$ 較佳個數		$RPD_3$ (%)	
		HGA	CH	HGA	CH
5	20	40	0	0	0.621
	40	40	0	0	0.742
	60	40	0	0	0.733
	80	40	0	0	0.775
7	20	40	0	0	0.707
	40	40	0	0	0.741
	60	40	0	0	0.755
	80	40	0	0	0.825

表五 HGA 與 CH 之  $F$  相對誤差

機器數	工作數	$F$ 較佳個數		$RPD_2$ (%)	
		HGA	CH	HGA	CH
5	20	40	0	0	0.505
	40	40	0	0	0.500
	60	40	0	0	0.427
	80	40	0	0	0.380
7	20	40	0	0	0.664
	40	40	0	0	0.598
	60	40	0	0	0.541
	80	40	0	0	0.550

表七 HGA 與 CH 之三個目標綜合相對誤差

機器數	工作數	執行時間 (秒)	$RPD$ 較佳個數		$RPD$ (%)	
			HGA	CH	HGA	CH
5	20	2.69	40	0	0.415	1.814
	40	13.30	40	0	0.387	1.932
	60	33.60	40	0	0.397	1.889
	80	39.42	40	0	0.317	1.885
7	20	2.69	40	0	0.487	2.065
	40	13.17	40	0	0.453	2.084
	60	27.22	40	0	0.447	2.039
	80	46.23	40	0	0.408	2.107

值 (Div) 的方法, 作為啟動重新開始機制之依據。每隔  $g_3$  代解群體的分散值若低於 Div 值, 則啟動重新開始機制。其中  $g_3$  的水準有 0、5、10、15、20、25, 而 Div 值的水準有 0.2、0.3、0.4, 經 ANOVA 分析, 顯示 P 值為 0.655, 表示沒有顯著差異, 故本研究放棄重新開始之分散機制。

### 五、HGA 績效分析

本研究 HGA 與 Chang 等人 [6] 提出的 GA 演算法 (簡稱為 CH) 比較。兩者在執行時間相同的前提下, 進行演算績效比較, 故將 CH 演算法中 LS 的搜尋次數予以放大, 使得 CH 可執行更多次的搜尋。

在演算績效方面, 機器數為 5、7, 工作數為 20、40、60、80 組合下, 進行績效比較。首先比較將兩演算的有效解加以凌駕, 再將其所剩下的有效解的個數加以比較; 然後, 再進行個別相對誤差與綜合相對誤差之比較, 從表四、五、六可看出, HGA 個別目標衡量下, 三個目標皆明顯比 CH 為佳。綜合目標衡量下, HGA 的有效解個數都比 CH 多, 而且 HGA 綜合相對誤差皆小於 0.5%, CH 的綜合相對誤差平均約為 2%, 明顯得知 HGA 具有較佳之演算績效。

### 六、結 論

本研究提出的 HGA 演算法, 以變異數分析法訂定 GA 的參數, 結果顯示: 交配機制則以單點交配為最佳; 強化策略則以局部搜尋策略最佳, 明顯改善了演算績效, 為了加強演算效率, 故採用限制區間之工作交換方式; 而重新開始機制的分散策略, 應用在多目標流程型工廠排程, 並沒有顯著提升績效, 主要原因是三目標之間具有衝突性, 且本演算法在搜尋過程中, 已將搜尋空間之分散性納入考量, 故重新開始機制並未明顯改善柏拉圖最佳解。HGA 演算法的整體品質而言, 不論是柏拉圖最佳解的個數上, 或是個別目標、綜合目標的品質上, 都比 CH 演算法為佳, 顯示 HGA 為一個有效的演算法。

### 符號索引

$C_{max}$	最大完工時間
Cut	切割數
d	最大移步距離
Div	解群體的分散值





$F$	總流程時間
$G_{\max}$	演化代數
$O_{1,l}$	第 $l$ 個有效解之 $C_{\max}$ 值
$O_{2,l}$	第 $l$ 個有效解之 $F$ 值
$O_{3,l}$	第 $l$ 個有效解之 $T$ 值
$P_c$	交配率
$P_m$	突變率
$RDP$	相對誤差百分比
$T$	總延遲時間

### 參考文獻

1. T'kindt, V. and Billaut, J. C., "Multicriteria Scheduling Problems: A Survey," *RAIRO Operations Research*, Vol. 35, No. 2, pp. 143-163 (2001).
2. Schaffer, J. D., "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms," *International Conference on Genetic Algorithms*, Morgan Kaufmann, NY, USA, pp. 93-100 (1985).
3. Ishibuchi, H. and Murata, T., "A Multi-Objective Genetic Local Search Algorithm and Its Application to Flowshop Scheduling," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, Vol. 28, No. 3, pp. 392-403 (1998).
4. Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp. 182-197 (2002).
5. Arroyo, J. E. C. and Armentano, V. A., "Genetic Local Search for Multi-Objective Flowshop Scheduling Problems," *European Journal of Operational Research*, Vol. 167, No. 3, pp. 717-738 (2005).
6. Chang, P. C., Hsieh, J. C., and Lin, S. G., "The Development of Gradual-priority Weighting Approach for the Multi-objective Flowshop Scheduling Problem," *International Journal of Production Economics*, Vol. 79, No. 3, pp. 171-183 (2002).
7. Chang, P. C., Wang, Y. W., and Liu, C. H., "New Operators for Faster Convergence and Better Solution Quality in Modified Genetic Algorithm," *Lecture Notes in Computer Science, Advances in Natural Computation, First International Conference, ICNC 2005*, Changsha, China, Vol. 3611, pp. 983-991 (2005).
8. Chang, P. C., Chen, S. H., Fan, C. Y., and Chan, C. -L., "Genetic Algorithm Integrated with Artificial Chromosomes for Multi-Objective Flowshop Scheduling Problems," *Applied Mathematics and Computation*, Vol. 205, No. 2, pp. 550-561 (2008).
9. Chang, P. C., Chen, S. H., and Liu, C. H., "Sub-Population Genetic Algorithm with Mining Gene Structures for Multiobjective Flowshop Scheduling Problems," *Expert Systems with Applications*, Vol. 33, No. 3, pp. 762-771 (2007).
10. Zhang, Y., Li, X., and Wang, Q., "Hybrid Genetic Algorithm for Permutation Flowshop Scheduling Problems with Total Flowtime Minimization," *European Journal of Operational Research*, Vol. 196, No. 3, pp. 869-876 (2009).
11. Ruiz, R., Maroto, C., and Alcaraz, J., "Solving the Flowshop Scheduling Problem with Sequence Dependent Setup Times Using Advanced Metaheuristics," *European Journal of Operational Research*, Vol. 165, No. 1, pp. 34-54 (2005).
12. Ruiz, R., Maroto, C., and Alcaraz, J., "Two New Robust Genetic Algorithms for the Flowshop Scheduling Problem," *Omega*, Vol. 34, No. 5, pp. 461-476 (2006).
13. Pasupathy, A., Rahendran, C., and Suresh, R. K., "A Multi-Objective Genetic Algorithm for Scheduling in Flow Shops to Minimize the Makespan and Total Flow Time of Jobs," *The International Journal of Advanced Manufacturing Technology*, Vol. 27, No. 7-8, pp. 804-815 (2006).
14. Vallada, E. and Ruiz, R., "Genetic Algorithms with Path Relinking for the Minimum Tardiness Permutation Flowshop Problem," *Omega*, Vol. 38, No. 1-2, pp. 57-67 (2010).

2010年01月13日 收稿

2010年01月18日 初審

2010年05月03日 複審

2010年06月09日 接受

