輔仁管理評論
中華民國 94 年 1 月，第十二卷第一期，147-166

最小化總流程時間之單階群組排程研究

黃榮華*

(收稿日期：91 年 10 月 17 日；第一次修正：91 年 12 月 30 日；
第二次修正：92 年 4 月 25 日；接受刊登日期：92 年 5 月 22 日)

摘要

群組化生產系統之排程作業簡稱爲群組排程。將個別工件依特定的屬性或製程，透過分類及編碼而形成工件族，因此，生產系統可配合工件族指派到專屬的機器群來設計。本文考慮單階加工中心，可處理若干工件族之加工作業的排程問題，機器若連續處理同一族之工件時，不需再有整備時間，若換族加工時則需要整備時間，這和一般排程將整備時間併入處理時間而不受加工順序影響的假設不同。實務上，若能有效地實施群組排程將可減少整備次數而縮短製程時間。本文以總流程時間爲績效衡量準則進行研究，因爲問題本質爲 NP-hard，故求解工作分兩部分進行，首先發展出一套探索式演算法，能迅速地求得最佳解或近似最佳解，經由資料測試結果顯示，最佳解題率均高達 90%以上，而平均求解時間則不到分枝界限法的百分之一。再者，以探索式所得結果作爲起始解，配合淩越條件及下界值的建構，提出整合式分枝界限法，能夠在合理的時間內，求得最佳解。

關鍵詞彙：群組排程，工件族，整備時間，製程時間，總流程時間

壹・前言

一、問題背景

面對變動快速的營運環境，企業無不致力追求產品多樣化以滿足市場的需求，而群組技術 (group technology, GT) 觀念則是落實少量多樣生產的重要理念。群組技術是將加工件依一定的屬性，例如：尺寸相似、製程相近、或使用相同的加工機具等，加以分類與編碼，成爲數個工件族 (part family)，再將加工特定工件族所需的機器集中形成群 (machine group)，機器的佈置便可根據需要而有不同的排列,而每一機器群即可針對幾個特定的工件族中的各工件進行加工。

* 作者簡介：黃榮華，輔仁大學管理學研究所副教授。

所謂群組排程 (group scheduling)，乃是生產排程作業與群組技術的整合。其採用批量方式，將訂單中的工件組成工件族，然後考慮多群組爲加工處理單元，將工件依序在機器上加工。由於加工件已按特定屬性分群，各群組之間具有不同的加工屬性，加工件的整備時間 (setup time) 必定受到排序的影響。傳統排程作業忽略此一事實，而將整備時間併入個別工件之處理時間。本文針對此一情況，整合群組技術之理念，在考慮加工件之排序作業時，若欲轉換不同群組加工，則會發生主整備時間 (major setup time)，此類整備時間因群組之屬性不同，將有明顯差異，應分開計算而不是合併入工件之處理時間中；但就同一群組內之工件而言，因加工屬性相近，其副整備時間 (minor setup time) 之差異全視個別工件需要，故仍可併入其處理時間中。實務上，若能有效地實施群組排程將可減少主整備時間次數，降低在製品存貨，以及縮短總製程時間 (make span)。

二、文獻探討

有關群組排程作業之研究，一般可分爲單階群組排程與多階群組排程兩類。單階群組排程乃是採用加工中心模式 (GT machine center)，使工件依群組在機器上加工。而多階群組排程乃是以流程型 (GT flow line) 或零工式 (GT cell) 爲主，工件從一機器移動至另一機器有兩種方式。其一爲後序連製法 (successive transfer) 主要是依批量運送 (lot by lot)。另一爲重疊連製法 (over lapping transfer)，主要是以工件爲單元運送 (piece by piece)。然而不論是單階或是多階群組排程，由於各群組之間具有不同的加工屬性，因此各群組於機器上的設置時間與拆卸時間各不相同。如果是同一群組內的工件，因爲加工屬性相似，若能連續作業，可以將工件間的副整備時間併入個別工件之處理時間中，但若允許插件，則會有增加主整備時間次數之情形發生。由於影響總製程時間，最主要因素是處理時間及主整備時間。因此，若能針對主整備次數作有效控制，則可以大量改善批量生產的效率，進而降低生產成本，因爲副整備時間已被併入個別工件之處理時間中，所以本文所提及之「整備時間」實際上爲「主整備時間」。

在單階群組排程方面，根據 Mason & Anderson (1991)，整備時間之變異有二：第一種是轉換時間因群組順序不同而異。此時除了起始群組只有設置時間，而無拆卸時間之外，而其他群組轉換時，必須先拆卸前一群組的設施，並設置此一群組的設施，而此二時間之和必然隨著前後群組的不同而異，對於這類型問題，Yoshida *et al.* (1973) 首先分別以平均流程時間 (mean flow time) 和

總延遲時間 (total tardiness) 爲準則，以分枝界限法求得最佳解。Foo & Wager (1983) 則將問題分割成爲循環性與非循環性問題，前者可視爲銷售員旅行問題，後者可視爲非循環性的銷售員旅行問題。作者針對非循環性問題，以動態規劃方式求解總設置時間最小。Mufit Ozden *et al.* (1985) 提出單向流程規劃以使得流程時間最小。Gupta (1988) 對平均流程時間，提出探索式演算法則。Mason & Anderson (1991) 針對加權流程時間，發展凌越條件 (dominant rules) 與建立下界值 (lower bound)，有效改善分枝界限法的分枝情形。Crauwels *et al.* (1998) 亦針對加權流程時間，比較三種不同凌越條件之分枝界限法，發現求解速率以 Mason & Anderson (1991) 最佳。Dimopoulos & Mort (2001) 亦以群組排程爲題，致力於單階製造單元中的若干項衡量準則，探討其基因演算法之方法論。Gagne *et al.* (2002) 針對總延遲時間，提出數種探索式解法。

第二種是轉換時間不因群組間順序不同而異。此時若僅只考慮設置時間而不考慮拆卸時間，則每一群組的設置時間爲一固定值。因此，整備時間即不受前後群組的影響。對於此類問題，可分爲單階與多階兩類型。在單階群組排程方面，Yoshida *et al.* (1973) 針對平均流程時間與總延遲時間，提出探索式演算法求出近似最佳解。Ahn and Hyun (1990) 與 Ghosh (1994) 均以動態規劃法求解。Williams & Wirth (1996), Liao & Liao (1997), 以及 Crauwels *et al.* (1998) 等均設計探索式求解總流程時間問題。而 Herrmann & Lee (1995) 則以基因演算法求解相同問題。Sung *et al.* (2002) 考慮工件隨機來到情形，以動態規劃法求解最小化總製程時間 (make span) 問題。

在多階群組排程方面，Nakamura & Hitomi (1976) 針對總製程時間、平均流程時間與總延遲時間分別以分枝界限法求得最佳解。Hitomi & Ham (1977) 以最小總流程時間爲主要目標，而以機器速度表示的總生產成本爲第二目標進行研究。Ham *et al.* (1979) 以延遲工件數 (the number of tardy jobs) 爲首要目標，總製程時間爲次要目標爲研究主題，並調整機器加工速度，使得生產成本最低。Kim & Bobrowski (1994) 用模擬方法處理一個九部機器之零工式工廠，考慮交期、整備時間及成本的排程問題。Schaller *et al.* (2000) 針對劃分工件族之流程式製造單元排程問題，以總流程時間爲衡量準則，設計分枝界限法，可求得最佳解。Bukchin *et al.* (2002) 針對雙機流程工廠，以平均流程時間爲衡量準則，設計演算法。

三、研究範圍

除了符合少量多樣的市場需求外，存貨成本一直是生產排程作業中相當重要的績效衡量指標，無論是在最終使用者市場，或是在機構市場。尤其在產品壽命週期日益縮短的競爭環境中，存貨不但積壓資金，佔用場地，更可能因爲顧客需求改變，而毫無價值。能維持較低的生產成本，一直是企業競爭優勢中最重要的一環。以最低的「在製品存貨成本」爲目標的排程問題中，主要的衡量準則爲「總流程時間」(total flow time) 或「平均流程時間」(mean flow time)。因此，在本文中，我們致力於探討單階加工中心多工件族之群組排程問題，而整備時間不受群組間順序影響，但群組間允許插件的情形，並針對總流程時間爲衡量準則，進行研究。

傳統上，單階排程以總流程時間爲準則的問題，是透過最短處理時間 (shortest processing time, SPT) 法則來解決。但是，一旦有插入其他群組工件的情況產生，便會發生前後加工件不屬於同一群，而增加整備的次數以致延長製程時間的情形。在這種情況下，根據 SPT 法則是無法獲得最佳解的，因爲此一問題本質上爲 NP-hard，所以本文期望能發展出一套兩階段式解法，階段一：先設計一套探索式演算法，期能迅速地求得最佳解或近似最佳解；階段二：進一步以探索式所得結果作爲起始解，配合凌越條件 (dominant rules) 及下界值 (lower bound, LB) 的建構，提出整合式分枝界限法 (integrated branch and bound procedure)，使能在合理的時間內求取最佳解。

貳・定義與符號說明

一、符號部份

g = 群數，即所有工件共分成 g 群。

n_k = 第k群的工件數，$(k = 1, 2, \dots, g)$。

n = 總工件數，故$n_1 + n_2 + \dots + n_g = n$。

J_i = 編號i的工件，$(i = 1, 2, \dots, n)$。

p_i = 第i個工件的處理時間，$(i = 1, 2, \dots, n)$。

$[i]$ = 某排序的第 i 個作業順位。

$J_{[i]}$ = 排在第i個順位之工件。

$p_{[i]}$ = 排在第i個順位之工件的處理時間。

g_i = J_i所屬的群組。例如：$g_2 = 1$ 表示J_2是第一個群組的工件。

$g_{[i]}$ = $J_{[i]}$所屬的群組。例如：$g_{[2]}$ = 1 表示排在第 2 順位的工件屬於第一個群組。

$C_{[i]}$ = 排在第i個順位之工件的完工時間。

F = 總流程時間 ($F = \sum_{i=1}^{n} C_{[i]}$)。

s = 整備時間 (setup time)。

S = 一組可行排序 (feasible sequence)。

二、定義部份

定義一：問題 ($n_1, n_2, \ldots, n_g$)/ 1/ G/ F

對一個單階加工中心，考慮具有g個群組，各群組分別含有$n_1, n_2, \ldots, n_g$個不同加工件，而整備時間不受群組間順序影響，但群組間允許插件的情形；個別工件上機後必須加工完畢，才能改加工下一個工件；並且以總流程時間 (total flow time, F) 爲績效衡量準則的群組排程問題。

定義二：A 集合

此爲演算法求解過程中，已排序的工件之集合。A 集合中工件的作業順位乃依工件排入集合之先後，由前向後 (forward) 排序。

定義三：B 集合

此爲演算法求解過程中，尚未排序的工件之集合。

所以一開始進行排程作業時，A 集合爲空集合，B 集合中有 n 個工件，待所有工件均已排序，A 集合中將有 n 個工件，而 B 集合變爲空集合。

參・探索式演算法

因爲 ($n_1, n_2, \ldots, n_g$) / 1/ G/ F問題本質上爲NP-hard，所以求解工作分兩部分進行，首先發展一套探索式演算法，目的在於能夠迅速地求得最佳解或近似最佳解；其次，將以探索式結果作爲起始解，來建構分枝界限法，使能快速地求得最佳解。

一、相關定理

定理一：

如果 B 集合中，有數個工件其處理時間均爲未排序工件中最小者，而其

中存在某一工件與 A 集合中最後一個已排序工件同群，此時優先將其排入 A 集合爲最佳解。

證明：

設有兩工件其處理時間均爲未排序工件中的最小處理時間 (p^*)，其中一個與最後一個 (設已排入q個) 已排序工件(J_i)同群，另一個不同群，假設集合A中，最後一個工件的完工時間爲$C_{[q]}$，排入與J_i同群工件則$C_{[q+1]} = C_{[q]} + p^*$，若排入與$J_i$不同群之工件，則$C_{[q+1]} = C_{[q]} + p^* + s > C_{[q]} + p^*$，故以與$J_i$同群者爲優先。

定理二：

設 B 集合所有工件中，最小處理時間爲 p^*，但不存在任一工件之處理時間爲 p^*，且與 A 集合中最後一個已排序工件同群時，若未排序工件中有一與 A 集合中最後一個已排序工件同群，且其處理時間與 p^*之差小於整備時間(s)，則優先將其排入 A 集合爲最佳解。

證明：

假設有兩工件爲J_i, J_j，其中J_i爲B集合中具有最小處理時間的工件，即$p_i = p^*$，假設A集合已有q個工件，但J_i與$J_{[q]}$不同群，而另一工件J_j與$J_{[q]}$同群，且$p_j - p_i \leq s$，此時，若先排入J_i，則$C_{[q+1]} = C_{[q]} + p^* + s$；若先排入$J_j$，則$C_{[q+1]} = C_{[q]} + p_j \leq C_{[q]} + p^* + s$，所以應該優先將$J_j$排入A集合。

定理三：

若 B 集合之未排序工件並無符合定理一、二情況，而數個具有最小處理時間之工件，又分屬不同群組時，在同群工件數最多者中，任取其一排入 A 集合爲最佳解。

證明：

設k個$p = p^*$的工件中分屬於h ($h \geqq 2$) 個不同的群組，各有$r_1, r_2, \ldots, r_g$個工件 ($r_1 + r_2 + \ldots + r_g = k$)，且$r_1 \geqq r_2 \geqq \ldots \geqq r_g$，若先排入$r_1$個工件中的任一工件，則之後將因符合定理一的情況，而連續將此r_1個工件排入。反之，若先排入其他群組之任一工件 (設其含有r'個工件，且r'爲$r_2, r_3, \ldots, r_g$中之一項。) 則之後將因符合定理一的情況，而連續將此r'個工件排入，這會因爲$r_1 > r'$，使得在先排入r_1個工件的情況下，跨群發生在[$q + r_1$]與[$q + r_1 + 1$]作業順位之間，如此，增加的完工時間，會低於跨群發生在[$q + r'$]與[$q + r' + 1$]作業順位之間的排序。

定理四：

若 B 集合中，未排序之工件並無符合定理一、二情況，而數個具有最小處理時間之工件，分屬不同群組時，若件數最多者不只一個群組，則比較這些群組在 B 集合中其他工件的處理時間，取最小者，將該群具有 p*之任一工件排入 A 集合為最佳解。

證明：

將符合定理四情況的工件優先排入集合 A，將使得集合 B 中除了具有 p*之工件外，具備次小處理時間的工件能在符合定理二的情況下排入 A 集合。

二、探索式演算法之求解步驟

利用上述定理，發展出探索式演算法，能夠迅速的取得一組 $(n_1, n_2, \ldots, n_g) / 1/ G/ F$問題的近似最佳解。

步驟 1.若 B 集合為空集合，則到步驟 8。反之，在 B 集合中，找出所有處理時間最小的工件，假設有 k 個，其處理時間均為 p*。

步驟 2.如果 k=1 且 A 集合為空集合，到步驟 3。如果 k=1 但 A 集合並非空集合，檢視此一工件與 A 集合中最後已排序工件是否同群，是則到步驟 3，否則到步驟 4。如果 k>1 且 A 集合為空集合到步驟 5，若 A 集合不為空集合則到步驟 6。

步驟 3.將此一工件依序排入 A 集合中，回到步驟 1。

步驟 4.檢視 B 集合中是否存在與 A 集合最後已排序工件同一群組，且其處理時間與 p*相差不大於整備時間的工件，有則將此一工件依序排入 A 集合中，回到步驟 1。否則到步驟 3。

步驟 5.由 k 個工件屬於相同群組件數最多的工件中，任選一個排入 A 集合中所有已排序工件的下一個順位，到步驟 1。若件數最多者不只一個群組時，則比較這些群組在 B 集合中之其他工件的處理時間，取最小者，將該群具有 p*之工件依序排入 A 集合中，回到步驟 1。若這些群組在 B 集合中均無工件存在，則任選一工件排入 A 集合，到步驟 1。

步驟 6.檢視 k 個工件是否有與最後一個排入 A 集合的工件同群者，有則都排入 A 集合，回到步驟 1。否則，到步驟 7。

步驟 7.檢視 B 集合中是否存在與最後一個進入 A 集合之工件同群，且其處

理時間與 p*相差不大於整備時間的工件，若有，則將此工件排入 A 集合中，回到步驟 1。否則到步驟 5。

步驟 8.停止。

三、釋例

茲以 $(n_1= 2, n_2= 2, n_3= 1) / 1/ G/ F$ 問題為例，假設更換加工群組之整備時間s=1，其他資料如表一所示，說明如下：

表一 $(n_1= 2, n_2= 2, n_3= 1) / 1/ G/ F$ 問題

J_i	J_1	J_2	J_3	J_4	J_5
g_i	1	1	2	2	3
p_i	7	8	7	9	5

步驟 1.$B= \{J_1, J_2, J_3, J_4, J_5\}$，B集合中處理時間最小者為$J_5$，$p^*= p_5= 5$，$k= 1$。

步驟 2.$k=1$ 且 A 集合為空集合，到步驟 3。

步驟 3.將J_5排入A集合，$A= \{J_{[1]}= J_5 \}$，$B=\{J_1, J_2, J_3, J_4 \}$，到步驟 1。

步驟 1.$B=\{J_1, J_2, J_3, J_4\}$，B集合中處理時間最小者為J_1及J_3，$p^*= 7$，$k= 2$。

步驟 2.$k=2$ 且 A 集合不為空集合，到步驟 6。

步驟 6.$g_{[1]}= 3$，$g_1= 1$，$g_3= 2$，沒有與$J_{[1]}$同群者，到步驟 7。

步驟 7.第三群中沒有其他工件了，到步驟 5。

步驟 5.J_1與J_3不同群，B集合中，$g_2 = g_1= 1$，$g_4= g_3= 2$，而$p_2= 8 < p_4= 9$，故將J_1排入A集合，$A=\{J_{[1]}= J_5, J_{[2]}= J_1\}$，$B=\{J_2, J_3, J_4\}$，到步驟 1。

步驟 1.$B=\{J_2, J_3, J_4\}$，B集合中處理時間最小者為J_3，$p^*= 7$，$k= 1$。

步驟 2.$k=1$ 且A集合不為空集合，但$g_{[2]}=1$，$g_3= 2$，到步驟 4。

步驟 4.B集合中與$J_{[2]}$同群者尚有J_2， $p_2- p^*= 8- 7= 1 \le s=1$，將$J_2$排入A集合，$A=\{J_{[1]}= J_5, J_{[2]}= J_1,J_{[3]}= J_2\}$，$B= \{J_3, J_4\}$，到步驟 1.。

步驟 1.$B=\{J_3,J_4\}$，B集合中處理時間最小者為J_3，$p^*=7$，$k=1$。

步驟 2.$k=1$ 且A集合不為空集合，但$g_{[3]}=1$，$g_3=2$，到步驟 4。

步驟 4.B集合中無與$J_{[3]}$同群者，到步驟 3。

步驟3.將J_3排入A集合中，A= $\{J_{[1]}= J_5, J_{[2]}= J_1, J_{[3]}= J_2, J_{[4]}= J_3\}$，B=$\{J_4\}$，到步驟1。

步驟 1.B=$\{J_4\}$，B集合中處理時間最小者爲J_4，p*=9，k=1。

步驟 2.k=1，A集合不爲空集合，$g_{[4]}= 2= g_4$，步驟 3。

步驟3.將J_4排入A集合，A= $\{J_{[1]}= J_5, J_{[2]}= J_1, J_{[3]}= J_2, J_{[4]}= J_3, J_{[5]}= J_4\}$，到步驟1。

步驟 1.B 集合爲空集合，到步驟 8。

步驟 8.停止。

求得一組排序：$\{J_5, J_1, J_2, J_3, J_4\}$，其總流程時間F=111。

肆・整合式分枝界限法

爲能求得最佳解，並驗證本文所設計之探索式解法的有效性，我們將建構整合式分枝界限法，分枝結點的選擇方式是以最新界限法則 (the newest bound rule) 來進行，而探索式解法所得的結果將被作爲第一個上界值 (upper bound)，配合有效的凌越條件及下界值之設計，能夠快速地刪除不必要的分枝，省去不必要的結點及其試算，使能在合理的時間內求取最佳解。

一、相關定理

定理五：

假設A集合中，已經排序的工件共有h個，若$J_{[h]}$爲$\{J_{[1]}, J_{[2]}, \ldots, J_{[h]}\}$中，屬於該群的最後一個工件，則將排在其後的所有工件之完工時間，均需增加一個整備時間。

證明：

設$i > h$，$C_{[i]} = p_{[1]}+ p_{[2]}+ \ldots+ p_{[i]}$，若$J_{[h]}$爲由 1 到h中，所屬之群的最後一個工件，則剩餘工件無論怎麼排序，$g_{[h+1]}$必不等於$g_{[h]}$，因此$p_{[h+1]} = p_{[h+1]}+ s$，則$C_{[i]} = p_{[1]}+ p_{[2]}+ \ldots+ p_{[h+1]}+ s+ \ldots+ p_{[i]}$，故在h之後的工件的完工時間都會增加一個整備時間。

定理六：

在特定結點下，未排序的n-h個工件分屬r個群組，各群組所含工件數依序 (由大到小) 有$n_1, n_2, \cdots, n_r$個工件，若依SPT法則排序之後，相同群組的工件均能相連，則該排序爲分枝的最佳可行解，該分枝至此已被洞悉。

證明：

SPT 法則適用於 n/ 1/ G/ F 排程問題，當各工件以 SPT 法則排序後，各相同群組的工件均已相連時，即表跨越群組的次數已是最少的情況，此時所加上的整備時間當然最少，因此，再向下分枝的最佳解必爲此一排序，故此分枝至此已被洞悉。

定理七：

在特定結點下，未排序的n-h個工件分屬r個群，各群工件數依序 (由大到小) 有$n_1, n_2, \cdots, n_r$個工件，該分枝的最佳狀況將爲：依SPT法則排序之後，相同群組的工件均能相連，且群組間之順序，正好是工件數較多的群組在前，此理想解值可作爲往下分枝的下界值。

證明：

假設有兩組排序，S與S'，其中均有兩群相連的工件 (分別有n_v個及n_u個工件，且$n_v > n_u$) 相鄰排列，而其他排在這兩群以外，各順位上的工件均相同。在排序S中，有n_v個工件群組在有n_u個工件群組之前，另一組排序S'中，則有n_u個工件群組有n_v個工件群組之前。S與S'的各工件的完工時間如下表所示：

表二　S 的完工時間表

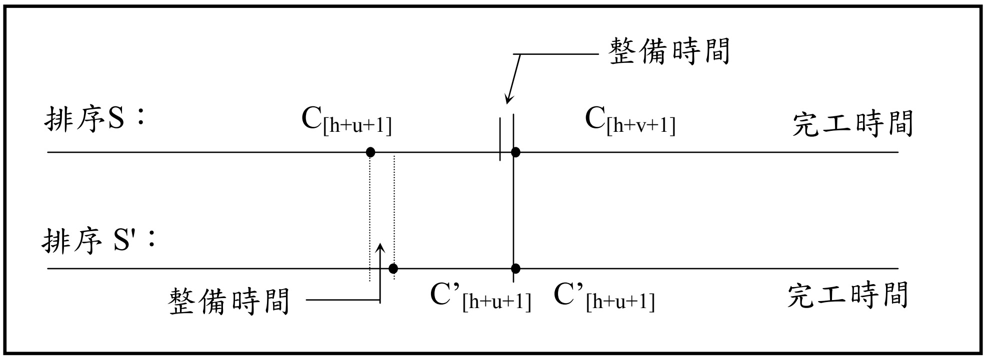
工件	$J_{[h+1]}$	$J_{[h+2]}$	…	$J_{[h+v]}$	$J_{[h+v+1]}$	…	$J_{[h+u+v]}$
處理時間	$p_{[h+1]}$	$p_{[h+2]}$	…	$p_{[h+v]}$	$p_{[h+v+1]}$	…	$p_{[h+u+v]}$
完工時間	$C_{[h+1]}$	$C_{[h+2]}$	…	$C_{[h+v]}$	$C_{[h+v+1]}$	…	$C_{[h+u+v-1]} + p_{[h+u+v]}$

表三　S'的完工時間表

工件	$J_{[h+1]}$	$J_{[h+2]}$	…	$J_{[h+u]}$	$J_{[h+u+1]}$	…	$J_{[h+v+u]}$
處理時間	$p_{[h+1]}$	$p_{[h+2]}$	…	$p_{[h+u]}$	$p_{[h+u+1]}$	…	$p_{[h+v+u]}$
完工時間	$C_{[h+1]}$	$C_{[h+2]}$	…	$C_{[h+u]}$	$C_{[h+u+1]}$	…	$C_{[h+u+v-1]} + p_{[h+u+v]}$

詳細說明如下圖一所示。由於$v > u$，$h+u+1 < h+v+1$，故S在[h+ v+ 1]順位，工件才會有跨越群組的情況產生，而S'在[h+ u+ 1]順位，便產生跨越群組的現象，因爲$C'_{[h+u+1]} = C_{[h+u+1]} + s > C_{[h+u+1]}$，因此，在S'中，[h+ u+ 1] 順位上的

工件因爲不同群組工件相連，必需加上一個整備時間，而在S中，從[h+ u+ 1]到[h+ v- u]的工件並未跨越群組，不需要加上整備時間，因而會有較小的完工時間。



圖一　定理七之示意圖

二、整合式分枝界限法之求解步驟

整合探索式演算法所得，以之爲起始解之分枝界限法，使能夠運用以下步驟，迅速的取得一組 $(n_1, n_2, \ldots, n_g)$ / 1/ G/ F問題的最佳解。

步驟 1.起始 (initialization)

使用探索式演算法得到一組可行解，並以其F值作爲上界值 (upper bound, UB)。

步驟 2.分枝 (branching)

由未洞悉 (unfathomed) 的子集中，選一具有最小下界值 (lower bound, LB)者分枝。

步驟 3.定界 (bounding)

對所產生的新結點，根據下列步驟計算其下界值 (LB)。

步驟 3-1.計算所有已排序的工件 (設有 h 個) 的完工時間。

步驟 3-2.將所有未排序的工件 (有 n-h 個) 依 SPT 法則排序。

步驟 3-3.計算未排序工件所屬的群數，設所有工件分屬 r 群。

步驟 3-4.設第i群的未排序工件有n_i件，將所有n_i由大到小排列並計算累計值A_i。

即$A_1= n_1, A_2= n_1+ n_2, \ldots, A_r= n_1+ n_2 + \ldots+ n_r$

步驟 3-5.對已排序的工件考慮整備時間，但未排序的工件暫不考慮整備時間的情況下，先按 SPT 排序後，接在已排序的工件之後，計算完工時間。

步驟 3-6.依定理五，若已排序的最後一個工件已是該群的最後一個工件，則所有未排序工件的完工時間均加上一個整備時間。

步驟 3-7.依定理六，若剩餘工件依 SPT 法則排列後同群工件均已相連，則此分枝已洞悉，計算此排序的實際 F 值與 UB 比較，以較低者作爲新的 UB。

步驟 3-8.依定理七，在第$h+ A_1+ 1$ ($= h+ n_1+ 1$) 之後 (含) 所有工件的完工時間均加上一個整備時間，再在第$h+ A_2+ 1$ ($= h+ n_1+ n_2+ 1$) 之後所有工件的完工時間再加上一個整備時間，……，在第$h+ A_{r-1}+ 1$ ($= h+ n_1+ n_2+ \ldots+ n_{r-1}+ 1$) 之後的所有工作加上整備時間。

步驟 3-9.計算預估的 F 值，作爲下界值 (LB)。

步驟 4.洞悉 (fathoming)

每一新節點遇到下列情況之一即爲洞悉。

(1)$LB \geq UB$。若一子集滿足此一條件，則將該子集刪除，不再考慮。

(2)此節點已分枝完畢，即在此分枝上已排入所有的工件，故爲一組可行解。

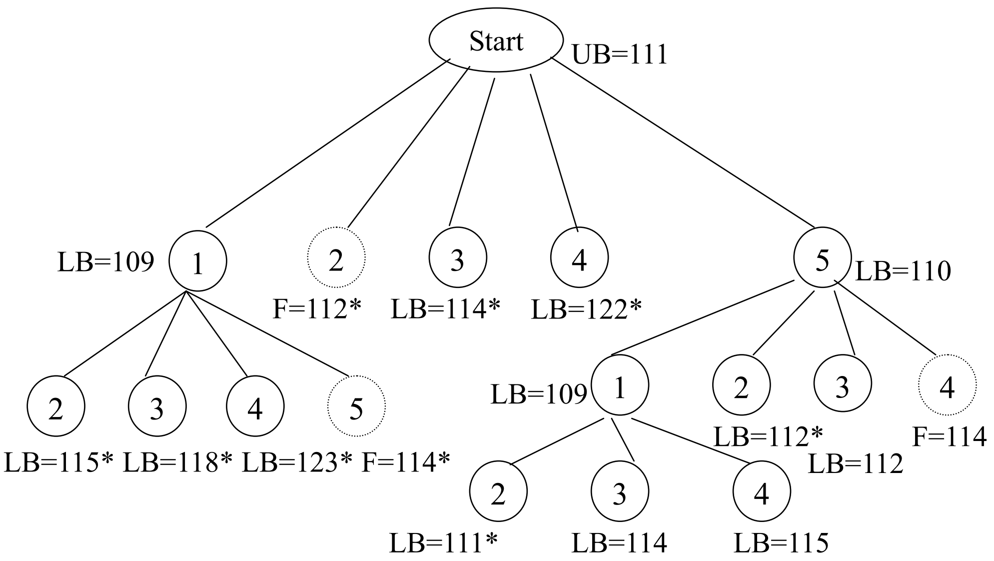
此時若 UB 大於此分枝之 F 值，則以此分枝之 F 值作爲新的 UB。

步驟 5.停止 (stop)

當所有分枝都已洞悉。

三、釋例

前面 ($n_1= 2, n_2= 2, n_3= 1$) / 1/ G/ F 問題，以探索式演算法解題後，求得一組排序$\{J_5, J_1, J_2, J_3, J_4\}$, F=111。以此作爲起始解，且UB=111，繼續以分枝界限法求解，其完整過程如圖一所示，結果得到排序$\{J_5, J_1, J_2, J_3, J_4\}$即爲最佳解。



註：○已可確定結點以下之最佳 F 值
* 已洞悉

圖二 $(n_1, n_2, \ldots, n_g)$ / 1/ G/ F分枝界限法示意圖

伍・資料測試

爲了驗證本文所提各項演算法之有效性，我們將以模擬資料來進行資料測試。根據 Crauwels *et al.* (1998) 針對以總流程時間爲準則之單階群組排程問題，比較三種不同凌越條件之分枝界限法，發現求解速率以 Mason & Anderson (1991) 所建構的解法爲最佳。因此，在驗證過程中，將與 Mason & Anderson (1991) 所建構的分枝界限法作比較。

一、建立資料

本文所發展的各項演算式，均以 Quick BASIC 語言撰寫程式，經編譯成執行檔之後，再以配有 Intel Pentium-4 CPU 之 PC 進行資料測試。測試所用的處理時間資料是以 Fisher (1976) 提出的方法建立：

1.處理時間 (p_i)：隨機取自範圍爲[1, 10]，且呈均等分配 (uniform distribution) 之整數。

2.整備時間：本研究假設其爲一定值。

二、測試資料

茲以不同的整備時間 (s= 1, s= 2)、群組數 (g= 2, g= 3, g= 4)、以及工件數 (n= 5, n= 10, n= 15, n= 20, n= 25, n= 30) 加以組合，各測試 30 次，共計 1080 組資料，所得結果如表四所示。表中，OPT %爲探索式演算法之最佳解題率，當然，整合式分枝界限法及 Mason & Anderson (1991) 分枝界限法之最佳解題率都是 100%，而各項求解時間之單位爲秒。

由資料測試的結果，可以發現本文所發展的探索式演算法，能快速地求解問題，時間絕不是一般列舉法所能相比，尤其當工件數爲 30 時，相對於整合式分枝界限法，平均求解時間更是不到其百分之一。重要的是，本探索式演算法有相當高的最佳解題率，最低的情形仍有 90%的正確率，且絕大部分都能達到 100%，當排程作業必須迅速完成，或者生產活動遇有突發狀況，例如：待料、緊急插單等，得進行重排程 (reschedule) 時，本演算法可以在分秒必爭的情境中，以極短的時間作出很好的管理決策。當然，如果最佳解爲絕對必要時，依循此探索式演算法所計算出來的起始解，因爲已經是或相當地接近最佳解，以其 F 值作爲上界值，將之併入分枝界限法中，如此可以有效地刪除不必要的分枝，提升分枝界限法的解題效率，因而整合式分枝界限法對於各式問題，都能在合理的時間內求得最佳解。此一作法，相較於現有文獻 Mason & Anderson (1991) 的解法，只是以傳統的分枝界限法處理方式，在求解效率上，會有明顯的改善，但是，在探索式解法之最佳解比率降低時，本文之整合式分枝界限法反而比 Mason & Anderson (1991) 的解法稍差，足見該解法之凌越條件具有很高的刪除效能。另外，在 n= 5 的某些情況中，本文之整合式分枝界限法也會因爲須要先使用探索式求起始解，使得求解時間較長。過去對這類問題之研究，大多著重於最佳解的取得，實際上，具有時效性的次佳解有時候可能比最佳解更符合管理需求。因而，基於實務考量，本文所提出的演算法正可彌補此一不足。此外，由資料測試結果顯示：當群組數增加、工件數增加、或是整備時間增加，都會使解題時間變長。而探索式演算法獲取最佳解的解題率，可能隨著群組數增加或工件數增加而略爲降低，但和整備時間的增減，並沒有明顯的關係。

表四　各項解法之資料測試結果

整備時間 (s)	工件數 (n)	群組數 (g)	求解時間 (單位：秒)			
			本文探索式解法	OPT %	整合式分枝界限法	Mason& Anderson
1	5	2	0.011	100	0.043	0.040
		3	0.008	100	0.041	0.040
		4	0.012	100	0.047	0.064
	10	2	0.019	100	0.193	0.567
		3	0.029	100	0.386	0.902
		4	0.030	100	0.631	1.127
	15	2	0.043	100	3.283	3.637
		3	0.050	100	4.862	5.710
		4	0.058	97	10.894	9.874
	20	2	0.063	100	6.672	7.524
		3	0.109	100	7.535	9.873
		4	0.059	93	14.469	11.769
	25	2	0.143	100	13.543	17.581
		3	0.157	100	17.476	19.877
		4	0.190	93	38.433	20.454
	30	2	0.193	100	26.113	34.067
		3	0.287	97	68.753	36.751
		4	0.276	90	93.787	37.473
2	5	2	0.067	100	0.099	0.039
		3	0.013	100	0.049	0.043
		4	0.018	100	0.037	0.037
	10	2	0.026	100	0.371	0.964
		3	0.033	100	0.115	0.847
		4	0.019	100	0.453	1.233
	15	2	0.039	100	3.756	3.533
		3	0.049	100	4.886	4.782
		4	0.047	97	9.733	8.721
	20	2	0.057	100	7.078	7.545
		3	0.110	100	7.963	8.784
		4	0.060	93	13.271	10.737
	25	2	0.155	100	13.493	19.933
		3	0.167	93	29.767	21.003
		4	0.201	90	39.431	22.467
	30	2	0.197	100	27.757	34.113
		3	0.291	93	87.633	38.667
		4	0.302	90	96.453	41.737

陸・結論

多樣化的市場需求帶動營運環境快速變遷，企業無不致力提供差異化的產品來滿足消費者的需求，而群組技術 (group technology, GT) 正是落實少量多樣生產的重要理念。所謂群組排程 (group scheduling)，乃是生產排程作業與群組技術的整合。將傳統整備時間合併於處理時間，而不受排序影響的不合理假設條件，作了一番修改，使之更符合實務情形。有關群組排程之研究，可分為單階與多階兩大類，但都會因為工件所屬群組之不同，在考慮加工件之排序作業時，若欲轉換不同群組加工，則會發生主整備時間 (major setup time)，此類整備時間因群組之屬性不同，將有明顯差異；但就同一群組內之工件而言，因加工屬性相近，其副整備時間 (minor setup time) 之差異全視個別工件需要，故仍可併入其處理時間。實務上，若能有效地控制主整備時間次數，便能可縮短總製程時間 (make span)，而改善生產效率及降低生產成本。

由於存貨成本一直是生產排程作業中相當重要的績效衡量指標，尤其在產品壽命週期日益縮短的競爭環境中，存貨不但積壓資金，降低空間的使用效益，更糟的是可能會因為顧客需求改變，而變得毫無價值。以最低的「在製品存貨成本」為目標的排程問題中，主要的衡量準則為「總流程時間」(total flow time) 或「平均流程時間」(mean flow time)。因此，本文致力於探討單階加工中心多工件族之群組排程問題，假設整備時間不受群組間順序影響，但群組間允許插件的情形，並以總流程時間為衡量準則。因為問題本質上是 NP-hard，所以求解工作分兩階段進行，首先我們發展一套探索式演算法，目的在於能夠迅速地求得最佳解或近似最佳解。經資料測試結果顯示其求解效率甚高，且得到最佳解的比率都在 90%以上。據此，再進一步發展整合式分枝界限法 (integrated branch and bound procedure)，以探索式所得結果作為起始解，配合有效的凌越條件 (dominant rules) 及下界值 (lower bound) 的建構，能快速刪除不必要的分枝，省去多餘的結點及繁複的試算，使能在合理的時間內，取得最佳解。

參考文獻

Ahn, B. H. and Hyun, J. H., "Single Facility Multi-class Job Scheduling", C*omputers and Operations Research*, (17), 1990, pp.265-272.

Bruno, J. and Sethi, R., "Task Sequencing in a Batch Environment with Setup Times", *Foundations of Control Engineering*, (3), 1978, pp.105-117.

Bukchin, J., Tzur, M., and Jaffe, M., "Lot Splitting to Minimize Average Flow-time in a Two-machine Flow-shop", IIE Transactions, (34), 2002, pp.953-966.

Buzacott, J. A. and Duton, S. K., "Sequencing Many Jobs on a Multi-purpose Facility", *Naval Research Logistics*, (18), 1971, pp.75-82.

Crauwels, H. A. J., Potts, C. N., and Van Wassenhove, L. N., "Local Search Heuristic for Single Machine Scheduling with Batch Set-up Times to Minimize Total Weighted Completion Time", *Annals of Operations Research*, (70), 1997, pp.261-279.

Crauwels, H. A. J., Hariri, A. M. A., and Van Wassenhove, L. N., "Branch and Bound Algorithms for Single-machine Scheduling with Batch Set-up Times to Minimize Total Weighted Completion Time", *Annals of Operations Research*, (83), 1998, pp.59-76.

Dimopoulos, C., and Mort, N., "A Hierarchical Clustering Methodology Based on Genetic Programming for the Solution of Simple Cell-formation Problems", *International Journal of Production Research*, (39), 2001, pp.1-19.

Dobson, G., Karmarkar, U. S., and Rummel, J. L., "Batching to Minimize Flow Times on One Machine", *Management Science*, (33), 1987, pp.784-799.

Foo, F. C., and Wager, J. G., "Set-up Times in Cyclic and a Cyclic Group Technology Scheduling Systems", *International Journal of Production Research*, (21), 1983, pp.63-73.

Gange, C. Price, W. L., and Gravel, M., "Comparing an ACO Algorithm with Other Heuristics for the Single Machine Scheduling Problem with Sequence-dependent Setup Times", *Journal of the Operational Research Society,* (53), 2002, pp.895-906.

Ghosh, J. B., "Batch Scheduling to Minimize Total Completion Time", *Operations Research Letters,* (16), 1994, pp.271-275.

Gupta, J. N. D., "A Search Algorithm for the Traveling Salesman Problem", *Computers and Operations Research*, (5), 1978, pp.243-248.

Gupta, J. N. D., "Optimal Schedules for Single Facility with Two Job Classes", *Computers and Operations Research*, (11), 1984, pp.409-413.

Gupta, J. N. D., "Flow Shop Schedules with Sequence Dependent Setup Times", *Journal of the Operations Research Society of Japan*, (29), 1986, pp.206-219.

Gupta, J. N. D., "Single Facility Scheduling with Multiple Job Classes", *European Journal of Operational Research*, (33), 1988, pp.42-45.

Ham, I., Hitomi, K., Nakamura, N., and Yoshida, T., "Optimal Group Scheduling and Machining-speed Decision under Due-date Constrains", *Transactions of the American Society of Mechanical Engineers*, (101), 1979, pp.128-134.

Herrmann, J. W. and Lee, C. Y., "Solving a Class Scheduling Problem with a Genetic Algorithm", *ORSA Journal on Computing,* (7), 1995, pp.443-452.

Hitomi, K., and Ham, I., "Group Scheduling Techniques for Multi-product Multi-stage Manufacturing Systems", *Transactions of the American Society of mechanical Engineers*, (99), 1977, p.759.

Jordan, C. and Drexl, A., "Discrete Lot Sizing and Scheduling by Batch Sequencing", *Management Science*, (44), 1998, pp.698-713.

Kim, S. C. and Bobrowski, P. M., "Impact of Sequence-dependent Setup Time on Job Shop Scheduling Performance", *International Journal of Production Research*, (32), 1994, pp.1503-1520.

Liao, C. Y. and Liao, L. M., "Single Machine Scheduling with Major and Minor Setup Times", *Computers and Operations Research,* (42), 1997, pp.169-178.

Mason, A. J. and Anderson, E. J., "Minimizing Flow Time on a Single Machine with Job Classes and Setup Times", *Naval Research Logistics*, (18), 1991, pp.333-350.

Monma, C. L. and Potts, C. N., "On the Complexity of Scheduling with Batch Setup Times", *Operations Research*, (37), 1989, pp.798-804.

Mufit Ozden, Egbelu, P. J., and Iyer, A. V., "Job Scheduling in a Group Technology Environment for a Single Facility", *Computer and Industrial Engineering*, (9), 1985, pp.67-72.

Nakamura, N. and Hitomi, K., "Optimization of Group Scheduling for the Multiple Production Stages", *Transactions of Japan Society of Mechanical Engineers,* (42), 1976, pp.2964-2973.(in Japanese).

O'Grady, P. J., "Search Based Job Scheduling and Sequencing with SetupTtimes", *Omega, International Journal of Management science*, (16), 1988, pp.547-552.

Olsen, T. L., "A Practical Scheduling Method for Multi-class Production Systems with Setups", *Management Science*, (45), 1999, pp.116-130.

Potts, C. N. and Kovalyov, M. Y., "Scheduling with Batching: A Review", *European Journal of Operational Research*, (120), 2000, pp.228-249.

Radharamanan, R., "Group Scheduling for Batch Production", *Proceedings of International Conference on Robotics and Factories of the Future, Springer- Verlag*, 1984, pp.16-35.

Schaller, J. E., Gupta, J. N. D., and Vakharia, A. J., "Scheduling a Flow Line Manufacturing Cell with Sequence Dependent Family Setup Times", *European Journal of Operational Research*, (125), 2000, pp.324-339.

Sung, C. S., Choung, Y. I., Hong, J. M., and Kim, Y. H., "Minimizing Makespan on a Single Burn-in Oven with Job families and Dynamic Job Arrivals", *Computers and Operations Research,* (29), 2002, pp.995-1007.

Szwarc, W. and Gupta, J. N. D., "A Flow Shop Problem with Sequence-dependent Additive Setup Times", *Naval Research Logistics*, (34), 1987, pp.619-627.

White, C. H. and Wilson, R. C., "Sequence Dependent Setup Times and Job Sequencing", *International Journal of Production Research*, (15), 1977, pp.191-202.

Wilbrecht, J. L. and Prescott, W. B., "The Influence of Setup Times on Job Shop Performance", *Management Science*, (16), 1969, pp.274-280.

Williams, D. and Wirth, A., "A New Heuristic for a Single Machine Scheduling Problem with Set-up Times", *Journal of the Operational Research Society,* (47), 1996, pp.175-180.

Yoshida, T., Nakamura, N., and Hitomi, K., "Optimization of Group Scheduling for a Single Stage Production", *Transactions of Japan Society of Mechanical Engineers*, (39), 1973, pp.1993-2003. (in Japanese)

Minimizing Total Flow Time of Group Scheduling for the Single Production Stage

RONG-HWA HUANG *

ABSTRACT

The integration of production scheduling with group technology has come to be called group scheduling. In this system, the classification and coding procedure for each so called part family is arranged according to the similarity of the attributes or manufacturing processes of the individual components. Under this system, therefore, a production system relating each part family to a specific machine group can be designed. The present study limits itself to the scheduling problems associated with single stage GT processing centers. In this context, we find that although a new setup is required when moving between jobs of different families, no new setup is required when moving between jobs of the same family. This is different from the traditional research of scheduling which assumes the setup time for a job is independent of job sequencing and, therefore, could be included in the processing time for each job. Our study found that by effectively implementing group scheduling the number of setups was reduced and consequently the make span was diminished. Our study addressed most specifically performance measures of total flow time. Since this problem is NP-hard, we developed a heuristic approach to derive a sub-optimal solution. The computational results indicate the heuristic approach we developed had a very high rate to get the optimal solution, above 90%, and its running time is really short. The solution was then used as the initial point of branch and bound procedure. Further, several dominate rules and a lower bound of the optimal flow time were developed. The integrated branch and bound procedure we developed can get the optimal solution within reasonable time.

Keywords: group scheduling, part family, setup time, make span, flow time

* Rong-Hwa HUANG, Associate Professor, Graduate Institute of Management, Fu Jen Catholic University.