

應用蟻群分群系統解決多途程單元形成問題

Ant clustering systems for solving cell formation problems with alternative processes

高有成 Yu-cheng Kao¹
葉翔宇 Shung-yu Yeh²

摘要

群組技術(Group Technology, GT)在單元形成問題中是很重要的的解題工具，群組技術的目標是將生產的設備(機器和零件)形成若干的區塊，目的是要將相似的機器和零件集中在一起，其優點是減少零件在單元之間的搬運距離，降低搬運成本，提升機器的使用率。本研究主要在解決多途程單元形成的問題，應用蟻群分群系統，藉由每隻螞蟻的互相碰撞，螞蟻彼此的參數不斷學習更新，將相似的螞蟻歸類為同一群，進而找出最佳的零件分群結果，接著再進行機器分群和途程選擇，以完成單元形成。

關鍵字：單元形成問題、蟻群分群系統、群體智慧。

Abstract

This paper proposes an ant clustering system (ACS) approach to solve cell formation problems with alternative process routings. The objective is to minimize the number of exceptional elements outside machine-cell blocks. The ant clustering system which carries out part clustering contains four main steps: parameter initialization, random meetings, free ant assignment, and cluster merge. According to the part clustering result, the algorithm performs machine assignment and then select a best route for each machine cell block.

Keywords : Ant clustering system, cell formation problems, alternative process routings.

壹、簡介

由於商業模式及客戶需求的不斷改變，商品從批次量的生產，到達少量多樣化的型態，產生了巨大的改變，工廠為了因應此特性，特別發展出了單元製造系統(Cell Manufacturing System, CMS)來針對此問題做解決，群組技術是單元製造系統最有效的處理方法之一。

群組技術依據零件在設計或製造上相似度，進行零件族群與機器單元的形成，使零

¹大同大學資訊經營所副教授(聯絡地址：台北市中山北路三段 40 號，聯絡電話：(02)25925252# 3604，E-mail：ykao@ttu.edu.tw)

²大同大學資訊經營研究所研究生

件家族儘量在所屬的機器單元內加工，通常設定零件家族至其他單元加工搬運次數越小越好。而求解單元形成，是設計單元製造系統時最重要的第一個步驟，統稱為單元形成問題(Cell formation problem, CFP)。在單元形成問題中依零件加工性質資料可分為單途程與多途程問題，單途程單元形成問題為每個零件只有一種加工途程，多途程比一般單途程問題更複雜，每個途程會有多種加工途程，也比較符合工廠真實應用情形。

單元形成問題可以視為資料分群問題，而隨著群體智慧的興起，相關學者紛紛將群體智慧技術應用在資料分群研究上，也獲得良好的分群效果。蟻群分群系統是屬於群體智慧演算法中的一種，本研究使用蟻群分群系統搭配相似係數，解決零件分群的問題，能將相似度高的零件合成零件家族。本演算法利用蟻群分群系統的特性：群體性與隨機性，使零件分群結果不會在分群過程中過早決定，降低資料本身特性之影響。

另外，蟻群分群系統在過去的文獻中，是解決單途程問題，但在本研究中，是應用在多途程單元形成問題。而本文所應用的蟻群分群系統，能有效的處理上述的複雜問題，經過實驗測試，蟻群分群系統能在大型處理問題上，能降低例外元素的數量，並且處理問題的時間也都能縮短，因此能證明，本文所提出的方法，能在多途程的單元形成問題上，有明顯的改善效果。

貳、文獻探討

2.1 單元形成問題

典型的單元形成問題中，零件加工只有單一固定途程，稱為簡單群組技術問題(Simple GT problem)，若單元形成問題中零件具有多個加工途程考量，則稱此類問題為廣義群組技術問題(Generalized GT problem)，Kusiak (1987)是第一個定義此問題，並且提出 p-median 模組解決此問題。Nagi et al. (1990)針對多途程單元形成問題發展了一種啟發式演算法，主要的目的是做途程的挑選和形成製造單元。Sankaran and Kasilingam (1990)使用數學規劃法解決多途程問題。Moon and Chi (1992)使用類神經網路的 constraint satisfaction model 解決多途程單元形成問題。Won and Kim (1994, 1997) and Won (2000)針對於多途程單元形成問題提出了不同的相似係數方法。Hwang and Ree (1996)發表兩階段程序，包含途程的選擇和零件家族的形成，用以解決多途程單元形成問題。Spiliopoulos and Sofianopoulou (2007)設計一個 bounding scheme 審查所有多途程的組合，以便於減少搜索的空間。多途程單元形成問題必須從多個途程中挑選唯一途程，因此，比單途程單元形成問題更複雜。

2.2 求解單元形成方法介紹：

文獻上解決單元形成至今已發展出很多種的方法，根據 Papaioannou and Wilson (2009)年提到單元形成的設計方法主要可分成以下五大類：

1. 數學規劃法(Mathematical programming)、

2. 啟發式演算法(Heuristics)、
3. 萬用啟發式演算法(Metaheuristics)、
4. 混合式萬用啟發法(Hybrid Metaheuristics)、
5. 人工智慧法則(AI approaches)。

本研究所使用的方式是屬於萬用啟發式法則，運用螞蟻的氣味辨識方式，根據螞蟻碰撞時，互相的資訊交換，並同時更新自己的氣味，用以衡量對方是否屬於同群的螞蟻，依此判斷彼此的相似程度，將巢中的螞蟻分類為數群，形成單元區塊。

2.3 螞蟻演算法

過去十幾年來，已經有許多研究學者投入群體智慧(Swarm Intelligence)這個領域，他們藉由觀察自然界生物、昆蟲的生活習性，獲得了很多的啟發。像是昆蟲社會中他們所擁有集體行為(Collective Behavior)、自我組織(Self-Organization)的能力等，皆成為群體智慧研究者爭相研究的主題。其中人工螞蟻演算法算是較為成功的主題，可分為螞蟻族群最佳化(Ant Colony Optimization, ACO)和螞蟻分群模式(Ant Colony Clustering)兩種模式。前者為一種的Meta-heuristic 方法，最常見的是用於解決組合最佳化問題而後者為本研究採用之模式，介紹如下。

Deneubourg et al. (1991) 是最早利用螞蟻撿起和放下的兩種簡單行為提出螞蟻分群模式。在該模式中，所有的物件與螞蟻先被隨機的放在2D 的棋盤內，每一個棋盤格裡最多只能有一個物件。之後，每一隻螞蟻隨機的在棋盤上移動，空手螞蟻會撿起異於週遭環境的物件，而持有物件的螞蟻會放下與週遭環境相似的物件，來達到物件分群的效果。另外Deneubourg 也賦予人工螞蟻記憶的功能，紀錄著螞蟻最近遇到物件的種類與數量。所以螞蟻記憶中的相同物件數量比例，決定了其目前週遭環境的物件相似程度，並被利用來決定撿起與丟下物件的機率值大小。Deneubourg 模式只能針對兩類物件分群，因此Lumer and Faieta (1994)改進其模式，能夠對多類別物件進行分群，稱之為LF Model。後來Monmarche (1999)結合LF Model 和K-means方法進行資料分群，並且證明了該混合式分群法比單獨使用螞蟻分群法或K-mean分群方法之結果為佳。

而Labroche (2003)提出異於LF Model 的螞蟻分群演算法，叫作AntClus模式。他觀察自然界中的螞蟻都擁有一種氣味散佈在表皮，其氣味會根據螞蟻本身以及其所在的環境而不斷改變。當螞蟻彼此遭遇時，假如雙方是相似的，牠們將會分享相同的氣味，亦即將彼此視為相同巢穴的成員。其演算法一開始將要分群的物件資料指派給螞蟻，亦即每一隻螞蟻都代表一個物件。再來模擬自然界中的螞蟻兩兩相互接觸，演算法中稱這個過程叫”碰撞”。作者觀察真實螞蟻相互接觸導致氣味改變的行為，提出六種遭遇行為法則，作為兩隻人工螞蟻遭遇時，改變螞蟻各自狀態的依據。在連續隨機遭遇過程中，人工螞蟻呈現自我組織(Self-Organization)的能力：藉由隨機遭遇與彼此交互作用，使得相似的螞蟻最後會

擁有同樣氣味，進而達到物件分群的效果。以上兩種人工螞蟻分群模式皆擁有彈性、穩定性、非中央式控制和自我組織等特性，而這些特性皆可解決複雜的資料分群問題。

2.4 蟻群分群系統

在Kao and Li (2008) 的文獻中是以Labroche (2003) 所提出的AntClust概念為基礎，並首次應用在解決單途程單元形成問題上，雖然本研究與Kao and Li 所使用的解題工具同為蟻群辨識系統，但差異是本研究的目的是在解決多途程單元形成問題，期望能將蟻群分群系統更能夠應用在工廠的實際情況。

本研究以Kao and Li 的概念為基礎，嵌入零件相似係數進而提出新的螞蟻零件分群演算法來有效的解決多途程零件分群。本演算法分群概念主要分為四個階段：初始階段，遭遇階段，雜群合併階段以及家族形成階段。在零件家族形成之後，接下來便進行多途程的選擇，這裡會挑出產生最少例外元素的途程，並將其他的途程捨去。在途程選擇完成後，則開始做區域搜索(Local Search, LS)，將零件和機器做對調與置換的動作，並且再度考慮已經捨去的途程，若區域搜索到達預設參數的次數時，則程式終止，並回傳最佳值。

參、數學模式

以下公式定義參考自 Kao and Lin (2011)：

符號定義：

C：單元的總數。

M：機器的總數。

$M_{\max}$ ：機器單元內，機器數的限制上限。

P：零件的總數。

R_p ：零件 P 的途程總數。

$$b_{mpr} = \begin{cases} 1 : \text{假設零件 } p \text{ 在途程 } r \text{ 需要機器 } m \text{ 生產} \\ 0 : \text{其他} \end{cases}$$

$$W_{km} = \begin{cases} 1 : \text{假設第 } m \text{ 台機器指派給第 } k \text{ 個單元} \\ 0 : \text{其他} \end{cases}$$

$$X_{kp} = \begin{cases} 1 : \text{假設第 } p \text{ 個零件指派給第 } k \text{ 個單元} \\ 0 : \text{其他} \end{cases}$$

$$Y_{pr} = \begin{cases} 1 : \text{假設零件 } p \text{ 所選擇的途程是 } r \\ 0 : \text{其他} \end{cases}$$

目標函數：最小化例外元素數量

數學模型：

$$\text{Min } E_x = \sum_{k=1}^C \sum_{m=1}^M \sum_{p=1}^P \sum_{r=1}^{R_p} b_{mpr}(1 - W_{km})X_{kp}Y_{pr} \quad (1)$$

限制式：

$$\sum_{k=1}^C W_{km} = 1 \text{ for } m \in \{1, 2, \dots, M\} \quad (2)$$

$$\sum_{k=1}^C X_{kp} = 1 \text{ for } p \in \{1, 2, \dots, P\} \quad (3)$$

$$\sum_{r=1}^{R_p} Y_{pr} = 1 \text{ for } p \in \{1, 2, \dots, P\} \quad (4)$$

$$2 \leq \sum_{m=1}^M W_{km} \leq M_{\max} \text{ for } k \in \{1, 2, \dots, C\} \quad (5)$$

$$2 \leq \sum_{k=1}^C X_{kp} \text{ for } p \in \{1, 2, \dots, C\} \quad (6)$$

目標函數(1)是計算單元外的加工次數(例外元素數)越少越好。限制式(2)為每台機器只能屬於一個機器單元；(3) 每個零件只能屬於一個零件單元；(4) 每個零件只能挑選一個途程；(5) 每個機器單元至少要包含 2 台以上的機器，但小於等於機器最大上限數。(6) 每個零件家族至少要包含兩個或兩個以上的零件。

肆、方法論

本研究解決多途程單元形成問題是先形成零件家族，接著進行機器指派，最後挑選途程，完成單元形成，另外再使用區域搜尋來改善解的品質。零件家族的形成是使用蟻群分群系統，並使用零件的相似係數進行零件分群。蟻群分群系統對零件的分群概念主要分為以下階段：初始階段，碰撞階段，雜群合併階段以及家族形成階段，形成零件家族後，再依照產生最少的例外元素數為原則，將機器指派給已分群的零件家族，接著做多途程的選擇(將產生最低例外數的途程挑出，其他的途程則刪除)，然後做區域搜索，當區域搜索做到預設的次數時，程式則會停止，並且回報所找到的最佳解。

4.1 整體流程：

蟻群分群系統的演化流程如下所示：

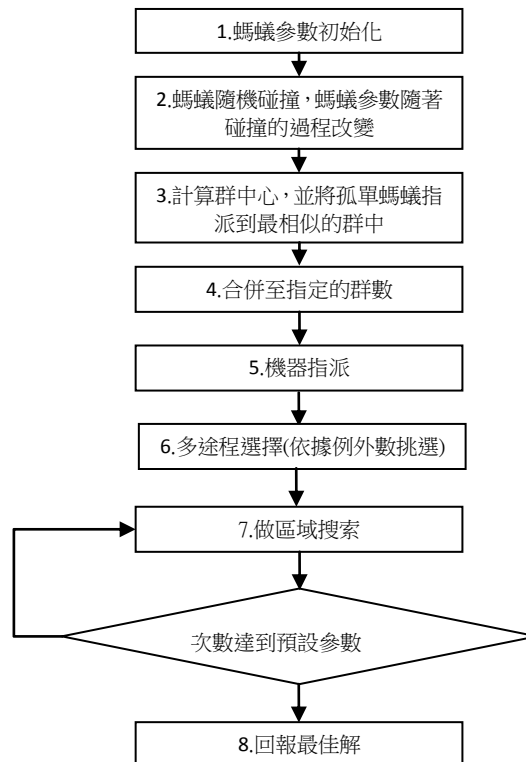


圖 1 蟻群分群系統流程圖

4.2 初始階段：

初始的階段有三個步驟如下所示：

步驟一 將要分群的每一個零件的每個途程指派給一隻螞蟻。

步驟二 將每一隻螞蟻的四個參數歸零，包括Label、M、M+和Template。

步驟三 進行 Template 值初始化。初始化過程中，每一隻螞蟻會與其他螞蟻進行 L 次的隨機遭遇，每遭遇一次，就利用公式(3)改變其 Template 值。所以其 Template 值會從零開始不斷調整，調整 L 次後的值就是該隻螞蟻 Template 的初始值。此處 Labroche (2003) 建議 L = 75 次即可。在本步驟中，Label、M、M+仍保持為 0。

符號定義：

Label：螞蟻身上所代表的費洛蒙。當兩隻螞蟻身上的 Label 相同時，代表這兩隻螞蟻是同群螞蟻。

M：螞蟻隨機碰撞時，若有同樣 Label 的兩隻螞蟻碰撞時的 M 值會增加。

M+：螞蟻隨機碰撞時，當兩隻「同群」的螞蟻碰撞，若兩隻螞蟻的相似度高，則 M+值會增高，若兩隻螞蟻的相似度低，則 M 和 M+會減少。

M 和 M+ 的值在 0~1 之間，若以 X 代表 M 或 M+ 值，則增加或減少 X 值的公式分別為公式(1)和公式(2)，計算方式如公式(1)及公式(2)所示：

$$X_{t+1} = (1 - \alpha) \times X_t + \alpha \quad (1)$$

$$X_{t+1} = (1 - \alpha) \times X_t \quad (2)$$

α ：為一個常數值，此處由 Labroche (2003) 建議 $\alpha=0.2$ 。

t 為一個代數，從 0 開始算。

Template 值：是兩隻螞蟻是否決定互相接受對方的門檻值，計算方式如公式(3)：

$$T_i = \frac{(\overline{\text{Sim}(i, \dots)}) + \text{Max}(\text{Sim}(i, \dots))}{2} - k \quad (3)$$

$(\overline{\text{Sim}(i, \dots)})$ 是第 i 隻螞蟻和其他螞蟻碰撞的平均相似度，而 $\text{Max}(\text{Sim}(i, \dots))$ 是第 i 隻螞蟻和其他螞蟻碰撞的最大相似度。在碰撞的過程中，兩隻螞蟻分別以公式(4)進行接受判斷。當公式(4)為真實，螞蟻 i 和螞蟻 j 是屬於互相接受的狀態，表示螞蟻是屬於同一群。

$$(\text{Sim}(i, j) > T_i) \wedge (\text{Sim}(i, j) > T_j) \quad (4)$$

在此， $(\text{Sim}(i, j))$ 所指的是螞蟻 i 和螞蟻 j 的相似度，而相似係數公式採用 Islam and Sarker (2000) 所提出之公式(見公式(5))

相似係數的計算：

$$\text{Sim}(i, j) = \frac{a + \sqrt{axd}}{a + b + c + d + \sqrt{axd}} \quad (5)$$

a：零件 i 與零件 j 同時所需的機器數。

b：零件 i 所需的機器數。

c：零件 j 所需的機器數。

d：零件 i 與零件 j 所不需要的機器數。

4.3 碰撞階段：

在此階段裡，會從全部的螞蟻中，任意的挑選兩隻螞蟻使其互相碰撞，當兩隻相似的螞蟻遇到時，便會互相交換並且同時更新自己本身的費洛蒙，而兩隻螞蟻身上都會標上同樣的標籤，就如同一張識別證，在同群的螞蟻中，每隻螞蟻都有相同的識別證。

我們這邊將螞蟻的費洛蒙以 Label 作表示，而螞蟻在相遇的過程中，會產生學習的結果(Template 值)，作為接受對方的標準。

每隻螞蟻身上具有四個參數：分別是 Template、M、M+、Label 值，在碰撞學習的過程中，四個值會不斷的做改變。本實驗中，每隻螞蟻會用下列六個規則，來判定是否接受對方(若互相接受則代表屬於同群，反之則屬不同群)。

規則 1：兩隻相似的螞蟻在尚未被分群，碰撞後將產生一個新的群組。(此法大多用在初始尚未分群的階段，而雙方的 Label 會相同)。

規則 2：當未分群的螞蟻和已分群的螞蟻碰撞後，如果兩隻螞蟻相似，這隻未分群的螞蟻會指派到已分群的蟻群中。(Label 會改成相同)

規則 3：當兩隻螞蟻有相同的 Label 時，在相互碰撞後，會合併成更大的蟻群(而雙方的 M、M+都會增加)。

規則 4：在同群的螞蟻中，兩隻相同 Label 值的螞蟻碰撞時，但互相不接受對方的情況下(代表兩隻螞蟻不相似)：具較小的 M+值的螞蟻，會被趕出原來的巢穴之外，且牠的三個參數值(Label, M, and M+)會被重新設定為 0，而未被趕出群的螞蟻，M 值會增加，M+會減少。

規則 5：當相似的螞蟻從不同的群集相遇時，擁有較小 M 值的螞蟻會被屬於另一群 M 值較大的螞蟻所合併。

規則 6：在以上的規則都未使用的條件下，Label, M, and M+不做任何的改變。

演算法在不斷經由螞蟻的互相碰撞，讓法則三、四、五產生交互作用，將可逐漸形成一群一群穩固且適當的較大群組。最後判斷是否符合迭代次數 NIter ($NIter = 250 * N$)。公式中的 250 是由實驗結果建議的係數，N 為螞蟻數目。當符合迭代次數，則進入下一階段。

4.4 合併雜群階段：

透過六個規則的分群之後，大部分的螞蟻在此階段已經分群完成，但仍有些螞蟻因為和同伴的相似度太低，或者再碰撞的過程沒有遇到相似的同伴，而成為孤單的個體，此時，就要將這些孤單螞蟻合併到最相似的蟻群之中。

有三個規則將孤單螞蟻合併到相似的蟻群中：

規則 1：計算群的群中心向量。

規則 2：計算群集的相似度。

規則 3：將孤單螞蟻合併到最相似的群中。

群中心向量 C 第 j 個元素計算的公式：

$$c_j = \frac{\sum_{i=1}^{ncm} a_{ij}}{ncm} \quad \text{for } j = 1, \dots, m \quad (6)$$

ncm ：第 i 群的成員數。

Example：

$$P_1 = [1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1]$$

$$P_2 = [1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0]$$

$$P_3 = [1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1]$$

$$C = [3/3, 2/3, 2/3, 1/3, 0/3, 0/3, 0/3, 2/3]$$

四捨五入後

$$C = [1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1]$$

4.5 零件家族形成階段：

在孤單螞蟻指派完畢，合併所得出的群數，可能遠比設計者所期望的群數多，因此在本階段，必須將多餘的群數進行合併，至設計者所期望的群數為止。

在 Kao and Li(2008)所採取的做法：首先計算每群零件家族的群中心，再剩餘群中找出兩群群中心最相似的零件家族，合併此兩群成一群。

在本論文中，我們改良Kao and Li(2008)的做法，會找尋零件家族成員數最少的群，將此群的成員皆視為個別的孤單螞蟻，依照孤單螞蟻合併的模式，將此群成員個別合併到其餘的群，因為，經由碰撞規則所合併成較大的群，螞蟻成員間通常會具有較強的相似度，故在此會優先挑選成員數較低的群做個別的合併，而非整群合併至另一群中。

4.6 機器指派：

將每台機器指派給零件家族，以產生最少的例外元素為原則做機器分配。

4.7 途程的選擇：

機器指派完之後，接者做途程的挑選，以產生最少例外元素為原則，將產生較多例外元素的途程剔除，留下唯一的途程，以限制式(4)為依據，一個零件只挑選唯一途程。

4.8 區域搜索：

區域搜索分成三個步驟：步驟1：機器的區域搜索部分會隨機選擇兩台不同群中的機器做隨機的交換。首先隨機抽取某兩個機器單元，再從隨機抽取到的兩個機器單元中，分別選擇任意一台機器，而交換的次數為 $P \times$ 機器數，如果機器再對調之後，例外元素有減少，那麼就會將此兩台機器做對調，另外一種區域搜索是將機器置換到其他的機器單

元中，則會從第一個機器單元中的第一台機器，分別置換到其他的機器單元(如第一個機器單元的第一台機器調換到第二個機器單元，再調換到的三個機器單元，第一個機器單元的第二台機器調換到第二個機器單元，再調換到第三個機器單元...依序調換)，每台機器依此類推，而置換的次數為機器數 $\times$ (家族數-1)，如果例外元素有下降，則將機器調至其他的機器單元中。

步驟 2：零件也是做相同的區域搜索，隨機選擇某兩個屬於不同群的零件做調換，首先隨機抽取某兩個零件家族，再從隨機抽取到的兩個零件家族中，分別選擇任意一個零件，如果零件互相調換後，能讓例外元素下降，則零件就對調，交換的次數為 $P \times$ 螞蟻數，另外的則是將零件分別置換至其他的零件家族，則會從第一個零件家族中的第一個零件，分別置換到其他零件家族，每個零件依此類推，(如第一個零件家族的第一個零件調換到第二個零件家族，再調換到的三個零件家族，第一個零件家族的第二個零件調換到第二個零件家族，再調換到第三個零件家族...依序調換)，若例外元素數下降，則將零件調至另一個家族，置換的次數為螞蟻數 $\times$ (家族數-1)次。

步驟 3：當做完機器與零件的交換與搬動後，會重新再度考慮已經消去的途程，若將已經消去的途程重新考慮後，例外元素有減少，或是搬動次數有降低，則會重新置換已消去的途程，置換的次數為總途程數-零件數。

P：為一個參數值，建議 $P=75$ 。

停止條件：若上述區域搜索的三個步驟執行到預設的次數時，則會回報所找尋到的最佳解，整個程式就執行完畢。

伍、範例說明

本節以 4 個機器和 5 個零件為範例，如圖 2 所示，參考 Kao and Lin (2011)，依照圖 2 來說明本論文所提出的方法。

5.1 螞蟻參數初始化

設定參數：零件數為 5 個，但途程共有 11 個(螞蟻數將每個零件的每個途程都設定為一隻螞蟻)：11 個、機器數：4 台、單元數 $C=2$ 、每群機器最大數 $M_{\max} = 2$ 、 $K=0$ 、 $\alpha=0.2$ 、螞蟻初始化次數： $N_{\text{Iter}} = 75 \times n$ (n 為螞蟻隻數，本例總共有 11 隻螞蟻)。

	P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
	R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
M1	0	0	1	0	1	1	0	1	1	0	1
M2	0	1	1	1	0	0	1	0	0	0	0
M3	1	0	0	1	1	0	0	0	1	1	0
M4	1	1	0	0	0	1	1	1	0	1	0

圖 2. 初始矩陣

在零件家族形成模組中，首先輸入初始矩陣至螞蟻零件分群系統，欄數表示零件編號，列數表示機器編號，所有參數Label=M=M+=Template=0。

我們將計算螞蟻間的相似度及每隻螞蟻的Template值，以圖2的初始矩陣為例，在本次的相似度的計算，總共11隻螞蟻，以第1隻螞蟻為例，將第1隻螞蟻和其他的第2~10隻螞蟻計算相似度(相似度是同時有2隻螞蟻才能計算)，而每隻螞蟻計算相似度的次數為75次(隨機抽取)。例如：在75次中，第1次第1隻螞蟻和第3隻螞蟻計算，第2次和第5隻螞蟻計算，第3次和第6隻螞蟻做計算，一直到做完75次，每次是隨機選取對象，我們隨機抽取一個例子做實際計算。

首先，以第1隻螞蟻的第一個途程[0 0 1 1]為例，隨機抽取到了第5隻螞蟻的第1個途程[0 0 1 1]，接下來計算這兩隻螞蟻的相似度，也就是 $\text{Sim}(P1, P5)$ ，以下列公式為計算方式， $a=2$ 、 $b=0$ 、 $c=0$ 、 $d=2$ ，帶入公式(5)後：

(a 、 b 、 c 、 d 的定義請參考4.2 初始階段的內容)。

$\text{Sim}(P1, P5)$ 所得到的相似度是0.66，經過以上的程序做75次，會得到75個相似度，才能根據公式(3)計算第1隻螞蟻的Template值。

$\overline{(\text{Sim}(1, \dots))}$ ：75次 $\text{Sim}(1, \dots)$ 的平均值， $\text{Max}(\text{Sim}(1, \dots))$ ：75次中， $\text{Sim}(1, \dots)$ 的最大值。

以第1隻螞蟻計算，Ant 1 產生了如下的Sim結果總共有75筆= 0 、 0.66 、 0.4 、 0.4 、 0 、 0 、 0.4 、 0.4 、 $\dots$ 、 0.4 。

根據公式 T_i 所算出第1隻螞蟻的Template值會得到0.5057。

以上面的結果以此類推會計算出總共11隻螞蟻的Template值如下：Ant 1：0.5057、Ant 2：0.4871、Ant 3：0.3937、Ant 4：0.3279、Ant 5：0.5244、Ant 6：0.5304、Ant 7：0.4853、Ant 8：0.5209、Ant 9：0.4972、Ant 10：0.5119、Ant 11：0.3210，而每隻螞蟻的Template值產生後，就會將值記錄起來。

5.2 螞蟻隨機碰撞，螞蟻參數隨著碰撞的過程改變

我們會將Label相同的螞蟻聚在一起，形成了初步的零件分群結果圖3。由圖3的第一列可以看到，在螞蟻碰撞過程完成後，大致上的分群已經完成，目前碰撞完的結果總共分成6群，以上藍底的數字中，相同的數字代表屬於同群的零件。

3	4	2	2	5	1	4	1	5	3	6
P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
0	0	1	0	1	1	0	1	1	0	1
0	1	1	1	0	0	1	0	0	0	0
1	0	0	1	1	0	0	0	1	1	0
1	1	0	0	0	1	1	1	0	1	0

圖3 碰撞完成的結果

碰撞階段中隨機挑選兩隻螞蟻進行碰撞，依碰撞公式計算共需隨機碰撞總共 $(250 \times 11 = 2750)$ 次。在碰撞期間每隻螞蟻的Label、M及M+值都會根據6個規則不斷的學習調整。M和M+使用公式(1)和公式(2)學習，公式(1)是的使用在M和M+在增加狀況，而公式(2)使用在M和M+減少狀況。

α ：為一個常數值，此處由Labroche (2003)建議 $\alpha = 0.2$ 。

5.3 孤單螞蟻的合併

從圖1可看到，此階段是進行孤單螞蟻合併，將未分到群的螞蟻作群中心計算，分類到最相似的群中，經過指派後，群數從原本的6群合併成5群。

註：第5個零件的第2個途程原本是編號6的孤單螞蟻，在此階段中，合併到第1群。

本階段首先計算每一群的群中心表1，計算完之後，將孤單螞蟻和每個群中心做相似度的計算表2，最終，將孤單螞蟻指派到最為相似的零件家族中。

以第1群為例：

表1 計算群中心

P3	P4	總和	總和平均	群中心
R1	R1			
1	1	2	1	1
0	0	0	0	0
0	0	0	0	0

表2 計算相似度

第1群群中心	第6群群中心	
1	1	a=1
0	0	b=2
0	0	c=1

將 a 、 b 、 c 、 d 值(a 、 b 、 c 、 d 定義請參考本文4.2節)帶入公式(5)，可得出 $\text{Sim}(G1, G6)=0.32$ 。

以上述步驟，將第6群的群中心和每群的群中心做計算，可得到下列5種相似度：

$\text{Sim}(G1, G6)=0.32$ 、 $\text{Sim}(G2, G6)=0.28$ 、 $\text{Sim}(G3, G6)=0$ 、 $\text{Sim}(G4, G6)=0$ 、 $\text{Sim}(G5, G6)=0.32$ ，將上述的數值皆計算出來後，就要將孤單螞蟻作指派(指派到相似度最大的群中)，因此我們將第6群指派給第1群，就完成了孤單螞蟻的指派(雖然 $\text{Sim}(G1, G6)=\text{Sim}(G5, G6)$)，在此時，我們會任意的選擇其中一群做指派，在這裡我們選擇了第1群，見圖4。

3	4	2	2	5	1	4	1	5	3	1
P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
0	0	1	0	1	1	0	1	1	0	1
0	1	1	1	0	0	1	0	0	0	0
1	0	0	1	1	0	0	0	1	1	0
1	1	0	0	0	1	1	1	0	1	0

圖4 孤單螞蟻合併

5.4 合併成指定的群數

從圖5可得知，此階段是將原本的既有族群，合併到所指定的群數，在此原本有5群，在這階段會合併成2群，分別形成了第1群和第4群。

1	4	1	4	1	1	4	1	1	1	1
P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
0	0	1	0	1	1	0	1	1	0	1
0	1	1	1	0	0	1	0	0	0	0
1	0	0	1	1	0	0	0	1	1	0
1	1	0	0	0	1	1	1	0	1	0

圖5 合併成指定群數

5.5 機器指派

見圖6，由於機器指派彼此間是一個獨立的步驟，在此階段會依照每一台機器做個別指派，我們考慮的指派原則為，在不違反最大機器數的限制式下，指派該台機器至產生最低例外數的家族。

		1	4	1	4	1	1	4	1	1	1	1
		P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
		R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
1	M1	0	0	1	0	1	1	0	1	1	0	1
4	M2	0	1	1	1	0	0	1	0	0	0	0
1	M3	1	0	0	1	1	0	0	0	1	1	0
4	M4	1	1	0	0	0	1	1	1	0	1	0

圖6 進行機器指派

5.6 多途程選擇

從圖7中，這個階段進行了多途程的選擇，根據零件途程所產生的例外數進行挑選，從中選取產生例外元素數最少的途程，其中黃色為底的途程，是代表所挑選的途程，其他白底(標示-1)的途程則是代表未選取。

		1	4	4	1	1	1	4	1	1	1	1
		P1	P1	P1	P2	P2	P3	P3	P4	P4	P5	P5
		R1	R2	R3	R1	R2	R1	R2	R1	R2	R1	R2
		-1	2	-1	-1	2	-1	2	-1	2	-1	2
1	M1	0	0	1	0	1	1	0	1	1	0	1
4	M2	0	1	1	1	0	0	1	0	0	0	0
1	M3	1	0	0	1	1	0	0	0	1	1	0
4	M4	1	1	0	0	0	1	1	1	0	1	0

圖7 多途程的選擇

5.7 區域搜索

機器的區域搜索部分會隨機選擇兩台不同群的機器做隨機的交換，以本範例為例，機器隨機交換的次數為 $75 \times 4 = 300$ 次，另外一個方式是將每台機器置換到其他的機器單元中，如果例外元素有減少，那麼就做置換，以本範例為例，置換的次數為 $4 \times (2-1) = 4$ 次，而零件的區域搜索是隨機選擇兩個不同群的零件做交換，以本範例為例，零件隨機交換的次數為 $75 \times 5 = 375$ 次，另一個方式是將每個零件置換到其他零件家族中如果例外元素有減少，那麼就置換，以本範例為例，置換次數為 $5 \times (2-1) = 5$ 次，而最終我們會再度考慮已經消去的途程，若將消去的途程做置換，而例外元數有下降，則將已消去的途程置換，以本範例為例，置換的次數為總途程數-零件數為 $11-5=6$ 次。

5.8 回報最佳解

在圖8中，會將所選取的途程和對應的機器家族做組合，並將零件和機器的家族編號做重排(從1開始)，形成最後的分群結果。

		1	1	1	2	2
		P2	P4	P5	P1	P3
		R2	R2	R2	R2	R2
1	M1	1	1	1	0	0
1	M3	1	1	0	0	0
2	M2	0	0	0	1	1
2	M4	0	0	0	1	1

圖8 回報最佳解

陸、比較研究

為了驗證本系統對多途程單元形成問題的成效，我們與 Kao and Lin (2011)於年提出的 CFPSO 進行比較，本文的程式是由 Java 所撰寫，執行於 Intel(R)Core(TM)2Duo CPU T6400@2.00GHz 3GB RAM 環境。在參數設定方面 $K=0$ 、 $\alpha=0.2$ ，是依據實驗所得最佳值所決定。本研究從文獻挑選 5 個題庫進行測試，每個例題的分群組數設定、機器最大數 Mmax：是參考其來源文獻。矩陣大小從 12 台機器與 20 個零件(12X20)至 26 台機器與 28 個零件(26X28)不等。表 3 為本研究所提出蟻群分群系統和文獻 CFPSO 的實驗結果比較，可以明顯的看到，在第 3、4、5 較大型的問題中，使用蟻群分群系統的方法，找尋到最佳解的時間有顯著的降低，而在第 5 題裡，蟻群分群系統更可以找到有較低例外元素的解。因此可以本演算法對於較大型的問題，處理的速度能比過去文獻的方法更快，也能夠找到最佳的解。

表 3 多途程實驗的比較數據

No. Source(s)	MXP	單元數	Mmax	CFPSO		蟻群分群系統	
				Ex	Time(s)	Ex	Time(s)
1. Sofianopoulou(1999)	12X20	3	5	24	0.201	24	0.917
2. Sofianopoulou(1999)	14X20	3	7	25	1.665	25	1.83
3. Nagi(1990)	20X20	5	5	1	1.731	1	1.521
4. Won and Kim(1997)	26X28	5	7	13	9.877	13	3.977
5. Won and Kim(1997)	26X28	6	7	16	16.169	15	7.312

柒、結論

本研究以蟻群分群系統為基礎，透過螞蟻碰撞的過程，每隻螞蟻的氣味會不斷的更新，並且會將相似的螞蟻歸類到同一群內，而未分群的螞蟻，則以計算相似度的方式，將螞蟻歸類到最相似的群中，最後將群合併至設定的群數，再進行多途程的選擇，經過區域搜索的過程後，找出最佳解。

在未來的研究方向，本研究可以朝向解決多目標的單元形成問題，例如機器加工順序、各機器產能的平衡、機器稼動率、機器費用等等，可以針對實際工廠在生產線上的動態環境配置議題來研究。

參考文獻

- Deneubourg, J. L., S. Goss, N., A. Sendova-Franks, C. Detrain & L. Chretien (1991), "The dynamics of collective sorting robot-like ants and ant-like robots," *Proc. Of the 1st Conf. on Sim. of Adaptive Behavior*, pp.356-363.
- Hwang, H. and Ree, P. (1996), "Routes selection for the cell formation problem with alternative part process plans," *Computers and Industrial Engineering*, 30 (3), 423 - 431.
- Islam, K., Md, S. and Sarker, B.R. (2000), "A similarity coefficient measure and machine-partsgrouping in cellular manufacturing systems," *Int. J. Prod. Res.*, 38, 699 - 720.
- Kusiak, A. (1987), "The generalised group technology concept," *International Journal of Production Research*, 25 (4), 561-569.
- Labroche N., N. Monmarche, G. Venturini (2002), "A new clustering algorithm based on the chemical recognition system of ants," *Proceedings of ECAI 2002*, pp.345-349.
- Labroche N., N. Monmarche & G. Venturini (2003), "AntClust: Ant Clustering and Web Usage Mining," *GECCO 2003*, pp.25-36, Springer-Verlag Berlin Heidelberg.
- Labroche N., N. Monmarche, G. Venturini (2003), "Web sessions clustering with artificial ants colonies," *WWW2003*, May 20-24, Budapest, Hungary. ACM xxx.
- Lumer, E. & B. Faieta (1994), "Diversity and adaptation in populations of clustering ants," *Proceeding of the Third International Conference on Simulation of Adaptive Behavior : From Animals to Animats*, MIT Press/ Bradford Books.
- Monmarche, N., M. Slimane & G. Venturini (1999), "AntClass: discovery of clusters in numeric data by a hybridization of an ant colony with the Kmeans algorithm," *Internal Report*, No.213, E3i, pp.1-21.
- Moon, Y.B. and Chi, S.C., (1992), "Generalised part family formation using neural network techniques," *Journal of Manufacturing Systems*, 11 (3), 149 - 159.
- Nagi, R., Harhalakis, G., and Proth, J.M., (1990), "Multiple routings and capacity considerations in group technology applications," *International Journal of Production Research*, 28 (12), 2243-2257.
- Papaiounou, G.. and Wilson, J.M., (2009), "The evolution of cell formation problem methodologies based on recent studies (1997-2008): Review and directions for future research," *European Journal of Operational Research*, vol. 206, no. 3, pp.509-521.
- S. Sofianopoulou (1999), "Manufacturing cells design with alternative process plans and/or replicate machines," *International Journal of Production Research*, 37:3, 707-720

- Sankaran, S. and Kasilingam, R.G., (1990), "An integrated approach to cell formation and part routing in group technology manufacturing systems," *Engineering Optimization*, 16 (3), 235 – 245.
- Spiliopoulos, K. and Sofianopoulou, S., (2007), "Manufacturing cell design with alternative routings in generalised group technology: Reducing the complexity of the solution space," *International Journal of Production Research*, 45 (6), 1355 – 1367.
- Won, Y. and Kim, S., (1994), "An assignment method for the part-machine cell formation problem in the presence of multiple process routes," *Engineering Optimization*, 22 (3), 231 – 240.
- Won, Y. and Kim, S., (1997), "Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings," *Computers and Industrial Engineering*, 32 (1), 207 – 220.
- Won, Y., (2000), "New p-median approach to cell formation with alternative process plans," *International Journal of Production Research*, 38 (1), 229 – 240.
- Kao, Y. C. and Lin, C. H. (2011), "A PSO-based approach to cell formation problems with alternative process routings," *International Journal of Production Research*,
- Kao, Y. C. and LI, Y. L. (2008), "Ant colony recognition systems for part clustering problems," *International Journal of Production Research* , Vol. 46, No. 15, 1 August 2008, 4237 – 4258