

東海大學資訊管理研究所

碩士學位論文

以固定速率法結合蜂群演算法求解在基本週期下
的經濟批量排程問題

Combining the Fixed Rate Approach with the Artificial Bee
Colony Algorithm to Solve the Economic Lot Scheduling
Problem under the Basic Period Approach

指導教授：張育仁 博士

研究生：黃晞臨

中華民國 106 年 02 月

東海大學資訊管理學系碩士學位
考試委員審定書

資訊管理學系研究所 黃晞臨 君所提之論文

以固定速率法結合蜂群演算法求解在基本週期下的經濟批量排程問題

經本考試委員會審查，符合碩士資格標準。

學位考試委員會 召集人：李俊仁 (簽章)

委員：黃有良

鄭辰仰

陳世志

李俊仁

中華民國 106 年 1 月 18 日

致謝

不知不覺間學生生活就要到尾聲了，從論文定下題目到申請口試，途中還真的經歷了不少波折，但在這些艱難的時期有許多人給予我幫助和鼓勵，所以我才能一一排除難關，最後順利通過口試。

首先我要感謝我的指導教授張育仁老師，在研究所期間對我的關心和指導，他教會了我各種做研究和報告的技巧，在我遇到問題的時候一定會在旁邊給予協助跟建議，在準備口試前的一個星期，老師更在百忙中抽空為我進行多次模擬口試，令我能有十足把握的面對口試。除此之外，老師也經常分享一些日常生活經驗和未來出社會時找工作要訣，讓我獲益良多，在研究所這段時間老師對我的種種照顧，真的是非常感謝!!

再來我要感謝和我一起渡過研究所生活中的同學們:爾廷、博名、進皓、侑晨、乃誠、盈秀、陳蘊、靖淳和文樺。全因為有你們在，我的研究所生活才會變得多采多姿，不管一起聊天、吃飯、出遊或是埋頭苦幹趕報告，這些都是我一生難忘的快樂時光!

最後，我要感謝我的家人，雖然遙居海外，但他們是我在求學過程中最大的心靈支柱，沒有他們的支持、付出和鼓勵，我絕對不可能走到今天這一步。爸爸! 媽媽! 感謝你們!

黃晞臨 謹誌

中華民國一百零六年二月

論文名稱：以固定速率法結合蜂群演算法求解在基本週期下的經濟批量排程問題

校所名稱：東海大學資訊管理學系研究所

畢業時間：2017年2月

研究生：黃晞臨

指導教授：張育仁 博士

論文摘要：

經濟批量排程問題是一個 NP-Hard 複雜度的問題，是指在單一生產設備在生產多項產品時，對其產品的生產順序進行排序並調整批量大小，從而降低總生產成本，而得出生產排序必須為可實際執行的。為了能確實的解決現實的生產情況，學者們陸續將經濟批量排程問題分成各種情境來討論，包括生產排程中必須需考慮到產品的保存期以免造成損失、機器有機會生產出瑕疵品等等。

目前已有多位學者提出透過降低生產速率的方法，可以令生產期間的存貨成本明顯降低。因此本研究應用可變動生產速率的方法，求解在基本週期法下產品的閒置時間不一的經濟批量排程問題。

為了求解此問題的最佳化週期乘數以得到較佳的可行解，本研究嘗試將蜂群演算法應用在求解 ELSP 的週期乘數上，數據結果顯示運用固定速率法比傳統的解法求解更有效果，而本研究提出在基本週期法下速率變更的數學模式可以做為其他相關研究的參考。

關鍵詞：經濟批量、基本週期、固定速率、蜂群演算法、閒置時間

Title of Thesis : Combining the Simple Fixed Rate Approach with the Artificial Bee Colony Algorithm to Solve the Economic Lot Scheduling Problem under the Basic Period Approach

Name of Institute : Tunghai University, Institute of Information Management

Graduation Time : (02/2017)

Student Name : Hei-Lam Wong Advisor Name : Dr. Yu-Jen Chang

Abstract :

The economic lot scheduling problem (ELSP) is known as a NP-hard problem. The goal of the ELSP is through optimizing the production schedule and production batch size of all products on a facility to minimize the average total cost. The production schedule must be reasonable and feasible. In order to make the production schedule more suitable for real world production, many researchers started to add different kinds of constraints for solving the ELSP. For example, some production managers must consider the shelf life constraint for all products during generating a feasible schedule in a food production system to avoid lost profits, or a machine may produce defective products.

Several scholars pointed out that applying the fixed rate approach for generating a production schedule of a facility can lower the holding cost of a product in order to reduce the average total cost of a facility. In this study, we apply the fixed rate approach to calculate the idle time of each production cycle and change the production rate of a product in order to solve the ELSP under the basic period approach.

In this study, the key problem is to find out an optimized cyclic multiplier set $\{k_i\}$ for all products. The multiplier k_i means a product i has to be produced for every k_i basic periods. In this research, we apply the Artificial Bee Colony algorithm to search the best cyclic multiplier set $\{k_i\}$ of all products, so we can find out an optimal solution for the ELSP with the lowest average total cost. Our experiment results show that our method is more effective than the traditional basic period approach for solve the ELSP. Our mathematic model can be a reference for other related studies.

Keyword: economic lot, basic period, fixed rate, Artificial Bee Colony algorithm, idle time.

目次

第一章 緒論	1
第一節 研究動機與目的	1
第二節 研究範圍	2
第三節 研究架構與流程	3
第二章 文獻探討	5
第一節 經濟批量排程問題的求解方法	5
第二節 經濟批量排程問題的延伸研究	8
第三節 降低生產速率的的原理和效果	9
第四節 搜尋演算法	11
第三章 研究方法	16
第一節 基本假設	17
第二節 降低生產速率的基本週期法模式建構	17
第三節 人工蜂群演算法求解	20
第四節 人工蜂群演算法的改良	27
第五節 人工蜂群演算法求解 ELSP	28
第四章 數值實驗	33
第一節 數值範例	33
第二節 數據實驗	36
第五章 結論與未來研究	42
第一節 結論	42
第二節 未來研究	43
參考文獻	44
附錄一	53

表次

表 2-1 基本週期法的生產排程組合	7
表 2-2 延伸基本週期法的生產排程組合(1).....	7
表 2-3 延伸基本週期法的生產排程組合(2).....	8
表 3-1 基本週期法下降速產品的選擇	18
表 4-1 實驗環境	33
表 4-2 數值範例產品的參數資料	34
表 4-3 傳統基本週期法求解的結果	34
表 4-4 基本週期法下應用郭育璋(2014)的解法求解的結果.....	35
表 4-5 本研究的解法之求解結果	35
表 4-6 本研究求解範例的生產成本和生產負荷	36
表 4-7 數值實驗產品參數資料範圍	36
表 4-8 隨機實驗的結果	40

圖次

圖 1-1 經濟批量排程問題的運作	1
圖 1-2 本研究的流程圖	4
圖 2-1 傳統生產和固定速率法造成的存貨持有成本之比較	10
圖 3-1 工蜂在一維空間的移動方式	23
圖 3-2 工蜂在一維空間的移動後的新位置	23
圖 3-3 人工蜂群演算法流程圖	26



第一章 緒論

本章節將對研究做概括性的描述，分為以下三個部分：研究目的、研究範圍、研究架構與流程。

第一節 研究動機與目的

經濟批量排程問題(Economic Lot Scheduling Problem, ELSP)是指利用單一機器生產多種產品時，對產品的生產順序進行排序，並決定生產批量的大小以及何時該開始生產，目標為得到最小的總生產成本，且生產排程必須為合理可行。

經濟批量排程問題最早在 1958 年被 Rogers 提出雛形後，一直是許多學者矚目的題目，在各界學者不斷鑽研並提出各種延伸研究議題後，已廣泛應用於許多產業中，如金屬沖件、汽車、電子元件、塑膠工業、紡織品及地毯等產業。在單一機器任意時間內只能生產一種產品的限制下，當工廠要生產多種產品時，需要對各種產品進行生產排序以求得最小的總生產成本，經濟批量排程問題由此而生。圖 1.1 是經濟批量排程問題的基本運作圖，其中 T 代表一次的生產週期，而在這週期內需生產 n 種產品，並重複執行到無限週期。

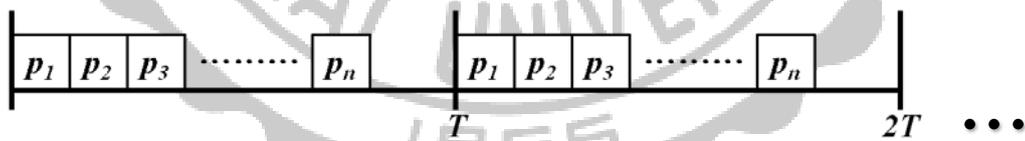


圖 1-1 經濟批量排程問題的運作

降低生產速率利用機器閒置時間過去的論點最早是 Buzacott & Ozkarahan (1983)提出，他們發現以往的 ELSP 問題都是以機器在最大生產速率下進行生產來作為研究假設，這會令機器的產能滿足產品需求量後，機器可能仍有閒置的時間而造成多餘的成本；而如果能利用機器閒置時間來進行生產的話，存貨水準就會大大降低，因而得到更低的總成本。

在探討 ELSP 的問題時，為了能有效的運用機台閒置產能並減少存貨持有成本，會以變更生產速率的方法來進行生產，常見的變更速率法有兩種：固定速率法與彈性速率法。以過往的 ELSP 研究的結果顯示，固定速率法能比得到傳統方法更低的成本；而在應用固定速率法的 ELSP 問題研究中，其解決方法主要為基本週期法和共同週期法。在共同週期法下，所有產品在每個生產週期中都要進行生產。而在基本週期法下所有產品只需要在第一個週期要生產，之後的生產排程則按照產品的週期乘數安排，因此基本週期法在排程安排上比共同週期法來得更有彈性，進而可得到比共同週期法更好的可行解。綜合以上述論，本研究將提出一個透過固定速率法進行降速，以基本週期法求解 ELSP 的問題模式。

第二節 研究範圍

本研究基於各項產品生產週期不一樣的情況下，運用基本週期法搭配變更生產速率，探討會出現的問題和解決的方法。而為了求得最佳的產品週期乘數組合以得出最低的總成本，本研究利用蜂群演算法(Artificial Bee Colony algorithm)來對 ELSP 問題進行求解，它的主要特點是不需要了解問題的特殊信息，只需要對問題進行優劣的比較，通過各人工蜂個體的局部尋優行為，最終在群體中使全局最優值突現出來，有著較快的收斂速度。為了使實驗更貼近現實情況，本研究加入每個生產週期的閒置時間都不一樣的限制條件。

第三節 研究架構與流程

本研究的架構和流程如圖 1-2 所示，並詳述如下：

1. 說明研究主題與目的

本研究在基本週期法下 ELSP 問題，加入每個週期的閒置時間都不一樣的前提來進行探討，透過降低生產速率以有效降低存貨成本，並利用蜂群演算法求解此經濟批量排程問題。

2. 文獻回顧與探討

為了使本研究更加完善，將對經濟批量排程問題中各種常見的求解方法，和以往經濟批量排程問題的延伸研究進行介紹，接著針對以往經濟批量排程問題中的相關研究和降低生產速率的生產方法進行深入的探討，最後對各種演算法和其相關的 ELSP 應用例子作分析。

3. 問題假設和演算法的建構

描述此 ELSP 問題的內容及其基本假設，並為問題中的數學符號作定義和說明，再為本問題建構出的目標函式與限制式。最後說明蜂群演算法的概念和其在本研究的應用方法。

4. 模式驗證與求解

經由上述所建構出的模式，利用數值範例來進行實際操作，運用大量隨機實驗測試成本改善幅度和演算法的效率。以求得最佳解。

5. 數值分析與結論

探討本研究的實驗結果，分析變更生產速率對成本的影響，最後作出具有歸納性的結論，並提出建議和未來研究方向。

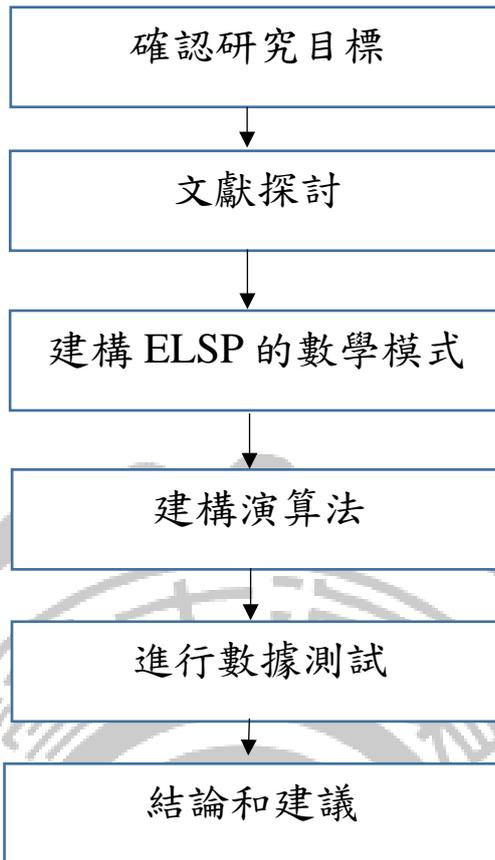


圖 1-2 本研究的流程圖

第二章 文獻探討

本章節主要分為四個部份來進行探討，首先經濟批量排程問題的起源、定義、求解方法和其延伸研究，第二部份探討對生產速率進行降速的原理和效果，第三部份為探討各種搜尋演算法的特質和其在經濟批量排程問題上的應用，最後第四部份將介紹本研究所使用的蜂群演算法的由來及運作原理。

經濟批量排程問題最早是由 Rogers(1958)所提出，目的是要在一台機器生產多種產品時，如何透過去調整產品生產批量大小和順序來獲得最小的生產總成本。Hsu 於 1983 年證明經濟批量排程問題是一個 NP-Hard 問題，基本假設經由 Bomberger (1966)整合之後，包括以下幾項：

1. 在同一時間之內一台機器只能生產一種產品
2. 顧客對其產品有不斷的需求
3. 產品之存貨數量與其存貨持有成本的關係呈線性
4. 產品的生產率、需求率和存貨持有成本皆是已知數而且不會隨著時間變動
5. 產品的種類會影響其整備時間和整備成本
6. 產品的批量大小和生產順序並不影響其整備時間和整備成本

第一節 經濟批量排程問題的求解方法

由於 ELSP 已被確認是一個 NP-hard 問題，不易確保所求的解為合理可行，因此眾多學者開始對此問題投入研究，並提出各種解法。Elmaghraby 於 1978 將求解 ELSP 的解法分類為兩種：

1. 啟發式解法(heuristic approaches)
2. 分析式解法(analytic approaches)

啟發式解法的原理是利用反覆式(iterative)搜尋的程序，在可接受的成本(求解時間、佔用空間等)下給出一個可行解，優點是執行時間較短，但通常不一定能

保證得到解是全域最佳解。

分析式解法的原理是基於要求解的問題上加上各種限制式，藉此得出在該情況下的最佳解，這類解法在求解問題的複雜度上升時，求解時間也會大幅增加，其常見的類型有：

1. 獨立解法
2. 共同週期法
3. 基本週期法
4. 延伸基本週期法

獨立解法

各項產品的生產成本個別計算其最佳解之後，將得到的解加總以得到生產總成本。由於沒考量到生產工具的產能問題，得到的最終解大多不可行，但也因為無視產能的限制，如果得出的解為可行，則該解為全域最佳解。

共同週期法

共同週期法於 1962 年由 Hanssmann 所提出，其假設為每項產品在每一個生產週期都要生產，所以所有產品都有一個共同的生產週期時間，在生產週期內各項產品的生產順序可以自由排列並且只會生產一次，之後以此生產週期為基準重覆生產，最後得出生產總成本，共同週期法和獨立解法不同的地方在於得到的解必定為可行解，但通常得到的總成本都會比其他求解法方來得高。

基本週期法

由 Bomberger(1966)所提出，其假設為所有產品在第一個生產週期都要生產一次，之後為每項產品定下一個週期乘數 k_i ，週期乘數必須是整數，產品的產量必須等於該週期的需求，各項產品接下來在各個週期是否需要生產依照其週期乘數而定；例如：

1. 產品 A 的週期乘數為 2，表示每 2 個週期都需要生產一次。
2. 產品 B 的週期乘數為 4，則表示每 4 個週期生產一次。

表 2-1 基本週期法的生產排程組合

週期 產品	1	2	3	4	5	6	7	8	9	10	11	12
A($k_i=2$)	O		O		O		O		O		O	
B($k_i=4$)	O				O				O			

跟共同週期法比起來，基本週期法因為各項產品不需每個週期都生產，所以得到的生產總成本會比較低，但相對地設備會因為有空閒時間，而造成浪費而影響最終解的品質，而且得到解並不一定可行。

延伸基本週期法

延伸基本週期法的原理和基本週期法極為相似，唯一不同的地方在於去除了第一個週期所有產品都必需生產的限制，各項產品的第一次生產週期可以任意排序在該週期乘數 k_i 之內，例如產品 A 週期乘數為 2，那這項產品的第一次生產週期可以在第 1 或 2 週期，產品 C 週期乘數為 4，那這項產品的第一次生產週期可以在第 1、2、3 或 4 週期選擇其一，之後依照其週期乘數和基本週期法一樣在特定週期生產，表 2-2 和 2-3 是說明該例子的示意圖。

表 2-2 延伸基本週期法的生產排程組合(1)

週期 產品	1	2	3	4	5	6	7	8	9	10	11	12
A($k_i=2$)		O		O		O		O		O		O
C($k_i=4$)	O				O				O			

表 2-3 延伸基本週期法的生產排程組合(2)

週期 產品	1	2	3	4	5	6	7	8	9	10	11	12
$A(k_{i=2})$	0		0		0		0		0		0	
$C(k_{i=4})$			0				0				0	

相比起基本週期法，延伸基本週期法避免了每一項產品在第一週期都要生產而造成生產設備之後會有過多的閒置時間的缺點，根據過去的相關文獻，延伸基本週期法普遍可以找到比基本週期法更好的最佳解，但同時也使得求解過程變得更為複雜，因為在得到各項產品的週期乘數後需要對排程的可行性進行檢查。

綜合以上四種求解方法，獨立解法可以視為 ELSP 問題求解時成本的下界，共同週期法則可為可行解的上界。

第二節 經濟批量排程問題的延伸研究

過去幾十年來，學者們為了使 ELSP 問題更貼近現實情況，做出了各種不同的延伸研究。Maxwell(1964)、Dobson(1992)提出整備時間與生產順序相依關係的 ELSP 模式問題，一般的 ELSP 問題都會假設各項產品的整備時間與生產順序都是獨立的，其變動不會對彼此影響，但在現實的生產情況是是有可能會相互影響的。

Giri *et al.*(2003)、蔣其軒(2007)、Filemon(2015)等人探討設備在有可能失控的情況下生產的 ELSP 問題。現實設備會因為長時間的使用而出現問題，這 ELSP 問題假設當設備在生產一定週期的產品後，便會進入失控的狀態，會產出有問題的瑕疵品而造成損失。

Soman *et al.*(2004)探討了產品在貨架上有時效問題(shelf life)的 ELSP 問題。V.Kayvanfar & M.Zandieh(2012) 探討會劣化的生產和存貨成本的 ELSP 問題。很多產品都有其保存期限，其中以食品業最為明顯，若是存放過久，超過了保存期將

會造成產品損壞不能再銷售，必須報廢或銷毀而造成損失，因此如果應用基本週期法，並對產品的生產週期進行降速，則可以降低存貨成本，並延後保存期限的到來。

Tang & Teunter(2006)提出產品有被退回情況的 ELSP 模式，也被稱為有回收情形的 ELSP 模式(ELSP with Returns, ELSPR)；銷售出去的瑕疵品經由外部回收之後並不一定要報廢，可以在修復之後再銷售出去，從而減輕部份的生產需求。Teunter *et al.* (2008)之後進一步提出多項產品下有售後退回情況的 ELSP，他研究假設定為生產系統分成兩條生產線，分別負責瑕疵品的回收修復和製造產品。Simone Zanoni(2011)探討了在基本週期法下有多項產品的製造和回收再製(remanufacturing)的問題。Angelo(2015)提出運用鄰域搜索求解有售後退回和產品修復的 ELSP 問題。

Yao *et al.*(2012)提出在共同週期法下探討ELSP檢驗與排程問題，在產品檢驗採用全數檢驗(consensus inspection)的條件下，加入生產產能和檢驗產能等限制，並為此問題設計出一個搜尋演算法來求出週期長度跟各項產品的檢驗次數的組合，以得到最低的生產總成本。

Nils(2014)提出透過模擬最佳化求解在整備上順序相依的隨機性ELSP問題。Onder (2011)提出運用人工蜂群演算法求解延伸週期法下的ELSP問題。並繼續於(2014)提出使用DABC在基本週期法下求解有退回情況的ELSP問題。

第三節 降低生產速率的原理和效果

在傳統的 ELSP 問題上，一般都是假設機器以使用最大的生產速率來進行生產，後來學者們發現，如果設備有閒置時間的話，降低生產速率並利用閒置時間來進行生產的話，可有效降低存貨持有成本，使得生產總成本得以下降。目前降低生產速率的方法有兩種：固定速率法(fixed rate approach)和彈性速率法(flexible rate approach)。

在固定速率法下，一個產品在生產週期時間中能對生產速率作出變更一次，新的速率比最大速率要低，並以此生產速率來進行循環生產。最早將固定速率法把應用在 ELSP 問題的學者是 Buzacott and Ozkarahan (1983)，在圖 2.1 說明固定速率法的實際應用，在傳統的生產方式下，機器將會以最大速率生產(黑色實線)，這時機器的閒置時間和存貨持有成本都處於最高位，如果能降低其生產速率至黑色虛線並利用多餘的閒置時間來進行生產，那麼生產成本就會明顯的下降(著色三角區域)。

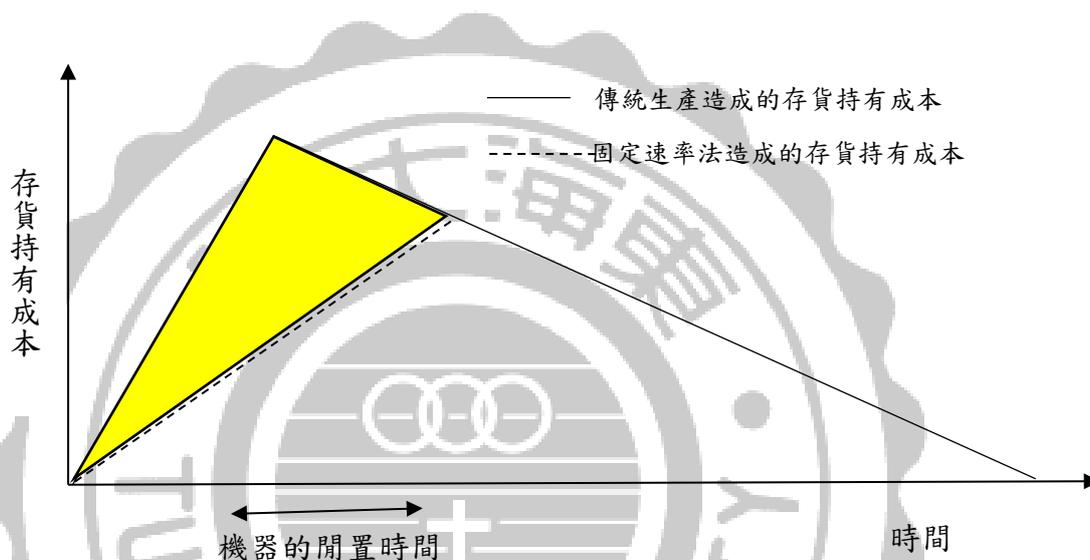


圖 2-1 傳統生產和固定速率法造成的存貨持有成本之比較

Silver (1990)提出在共同週期法下利用固定速率法求解生產排程問題時，只要對其中一項產品的生產速率進行降速，其他產品維持以最大速率來進行生產，就可以達到降低成本的目的。Moon & Christy (1998)提出在固定速率法下生產速率是有上限和下限，生產速率的下限為產品的需求率，而上限則為原有的最大生產速率，當生產速率增加時，平均總成本也會跟著上升。

Khouja (1999)以 Silver (1990)的研究作為基礎，提出以固定速率法求解不可靠生產系統的經濟批量排程問題模式，所謂的不可靠系統是指隨著生產系統的生產批量和生產速率增加時，生產出來的產品將會有品質下降的情形發生。結果顯示在不可靠生產系統的假說下，品質下降會使得批量大小和生產速率的提高。

Yang *et al.* (2000) 探討在一機器上生產兩種產品時的生產排程問題，透過應用固定速率法來改變生產速率，藉此對生產和整備排程進行最佳化，並運用動態規劃(Dynamic Programming)與Hamilton–Jacobi–Bellman方程式來確認解答的品質。

郭育璋(2014)探討了在基本週期法下，應用固定速率法求解產品具有保存期限限制的ELSP問題。並透過粒子群演算法協助搜找週期乘數。其假設所有週期的閒置時間都跟第一個週期的閒置時間一樣，以此作為降速的依據。

第四節 搜尋演算法

除了啟發式解法和分析式解法之外，由於決定產品週期乘數的組合對求解有關鍵性的影響，因此學者嘗試加入各種方法來尋找產品週期乘數組合，其中最為廣泛使用的就是透過各種搜尋演算法來求解經濟批量排程問題。

模擬退火法

由 N. Metropolis(1953)提出，Kirkpatrick(1983)將其應用於最佳化問題，原理源自冶金學中的退火，退火是一種物理過程，當固體金屬被加熱至臨界點的高溫時，便會從固態轉成液態，這時金屬中所有的分子都會在液態物質中重新隨機的自由排列，隨著溫度的緩慢下降，這些分子就會因為能量變少而重新排列並回到固體狀態。模擬退火法的優點在於所使用的搜尋方法簡單易懂，並具有跳離區域最佳值的能力。但在執行模擬退火時，當中參數跟最佳化問題本身有相當的關聯，因此在求解問題時大多需要需針對不同的問題而做出各種改寫。李世炳(2002)將模擬退火法和遺傳演算法結合，提出導引模擬退火法來改良局部及全局搜尋能力，S. Afshin(2006)提出透過模擬退火法求解在雙目標兩階段供應鏈的排序問題，Pempera(2013)探討應用模擬退火法來求解雙目標的流程型工廠排程問題。

遺傳演算法

遺傳演算法的核心概念將事物依其屬性進行如基因 DNA 一樣的編碼 (coding and representation)，並在群體之間藉由編碼的運算，透過交配 (Cross-over)、反轉 (inversion) 與突變 (Mutation) 三項基本操作繁衍出下一代。透過函數設計可

以選擇適應值較高的「下一代」繼續參與繁衍，藉此去求得最適合環境的物種（最佳解）。Khouja *et al.* (1998) 使用遺傳演算法和基本週期法來求解 ELSP，先以遺傳演算法求解出產品的週期乘數，並使用 Bomberger (1966) 的產品資料作數據實驗。Moon *et al.* (2002) 探討生產批量可以隨著時間變動大小的 ELSP，並應用混合式遺傳演算法來求解。Sarker & Newton (2002) 使用遺傳演算法求解經濟批量大小排程問題，以決定產品的採購政策和最佳批量大小。Soman *et al.* (2004) 使用一個類似分枝界限法的搜尋演算法求解 ELSP，數值實驗結果顯示能求得優於共同週期法的結果。Lin *et al.* (2003) 提出單一產品生產系統下，考量原物料庫存問題的 ELSP 模式，應用漸進搜尋局部近似最佳解的演算法，解決原物料及完成品的庫存問題。

Sai & Hing (2011) 提出了運用兩層遺傳演算法求解 ELSP 中產品的生產頻率，並於 2012 提出混合遺傳演算法與整數規劃(integer programming) 來求解多部機器生產的 ELSP 問題。Yao *et al.* (2012) 設計一混合式遺傳演算法 (Hybrid Genetic Algorithm, HGA) 來求解 ELSP，並搭配搜尋演算法產生多組可行的候選解 (candidate solutions)，不同於以往只產出一最佳解的模式，提供多組解讓管理者參考。Capa (2014) 提出運用多目標遺傳演算法求解有不確定性的 ELSP 問題。賴閔揚 (2014) 將多目標遺傳演算法應用在混合流程型的生產排程問題上。

螞蟻演算法

螞蟻最佳化演算法是由 Colormi、Dorigo 與 Maniezzo (1991) 年所發表，最初是為了解決旅行業務員 (Traveling Salesman Problem, TSP) 的問題所發展出來的演算法。演算法的原理在於模仿螞蟻尋找食物的過程，螞蟻在尋找食物的時候，會沿途分泌一種被稱為費洛蒙 (pheromone) 的化學物質；這種化學物質對螞蟻來說是一種訊號，可以為其他螞蟻作為尋找食物和自己回到窩巢的引導。費洛蒙會隨著時間而揮發，所以當路徑愈長，費洛蒙就會揮發的愈多，而下一次當螞蟻在選擇路徑的時候，是以費洛蒙的濃度作為選擇的標準，因此路徑愈短就愈容易螞蟻被選上，長路徑的費洛蒙濃度較低，所以選擇此路徑的螞蟻就會比較少。Heng & Hong (2013) 探討在可再生和非再生資源限制下利用螞蟻最佳化演算法求解具有多項資源組態的專案計劃排程問題，Ariyasingha (2015) 提出改良的 PSACO (pareto

strength ant colony optimization)，以此求解多目標效能最佳化的螞蟻最佳化演算法模式來求解 TSP 問題，並與傳統的多目標螞蟻演算法作比較。

粒子群(鳥群)演算法

粒子群(鳥群)演算法(Particle Swarm Optimization, PSO)也是常被學者使用的搜尋演算法，主要運用在最佳化問題上，該演算法是由 Kenned and Eberhart 於 1995 所提出。它的原理是透過模擬鳥類的群體生活行為所建構的群體智能模式，當一群鳥(粒子)在一解空間內隨機搜索食物(可行解)，而這個空間中只有一個地方有最好的食物(最佳解)，且所有的鳥都不知道最好的食物位在何處。但他們知道當前的位置距離最好的食物還有多遠，以及目前距離食物最近的鳥的位置，那麼找到最好的食物的最佳策略就是「搜尋距離食物最近的鳥之周遭區域」，因此所有的鳥便會慢慢往該方向聚集以求得更好的食物(較好的可行解)。

由於原始 PSO 中鳥群會向目前最佳解的方向前進，容易導致收斂過快而只求區域最佳解，Eberhart and Shi(1998)提出在 PSO 加入慣性權重(Inertia Weight)，強化其全域搜尋能力，成為了日後 PSO 的標準。Merwe and Engelbrecht(2003)應用 PSO 來最佳化資料分群的問題。林子元(2009)在半導體晶圓廠設施的佈置問題上以 PSO 求解。Ai and Kachitvichyanukul(2009)提出透過 PSO 求解車輛途程問題。Lima *et al.* (2010)結合 PSO 與 K-means 建立網路異常偵測模型，並以真實網路流量做實證，結果顯示此模型能有效分辨出異常行為。賴阮明 (2009)和 Lian *et al.* (2011)把 PSO 運用在零工工廠排程問題(Job-Shop Scheduling Problem, JSP)的求解上，林信宏(2010)應用 PSO 於多目標排程問題研究，並和遺傳演算法求解的結果作出比較。Shun(2014)提出運用 TPSO(twin particle swarm optimization)求解具有具相依整備時間的多工流程型工廠排程問題。Fuqing(2014)改良 PSO，並提出 DDPSO(particle swarm optimization with decline disturbance index)來求解多目標零工式工廠排程問題。Milica (2016)透過混沌粒子群演算法求解整合製程規劃和排程問題。

蜂群演算法

蜂群演算法是由Karaboga於2005年所提出，起源於觀察蜜蜂在採食花蜜時的行為，是種具有群體智慧的仿生演算法，最初是用於求解多維度之最佳化問題。蜂群的群體智慧模型分別由三個部份組成：食物源、被雇傭的蜜蜂（employed foragers）和未被雇傭的蜜蜂（unemployed foragers）。採蜜的蜂群有招集蜜蜂到食物源以及放棄食物源兩種行為模式。蜂群一共分為三種：工蜂、觀察蜂和偵察蜂，工蜂負責探索花蜜，並提供食物源的品質訊息給於蜂巢等待的觀察蜂。觀察蜂依照工蜂所提供的訊息前往食物源協助搜索花蜜。偵察蜂負責找出初始食物源，並會在食物源收益度無法改善時，於搜索空間隨機性地探索，藉以求得新的食物源。

蜂群演算法的概念是把可行解當成食物源，而花蜜則是該可行解的適應度，先隨機設定一個解當成最初食物源，然後派出工蜂到該食物源附近探索，把花蜜（適應值）的品質訊息傳給於蜂巢等待的觀察蜂，並和之前的食物源作出比較，當新的食物來源之花蜜量比之前的食物源來得要多的時候，工蜂會記憶新的食物源並且放棄之前的食物源。當工蜂都完成工作時，觀察蜂將根據各工蜂來提供的訊息來判斷各食物源花蜜的高低，並依照花蜜較高的食物源為優先的方法到各工蜂所在的位置協助搜尋。當同一個食物源的花蜜在一定次數後還沒有改善，工蜂將放棄該食物源，並要求偵察蜂去搜索新的食物源。

蜂群演算法是一種近年才被提出的演算法，而他創新的搜尋方法馬上吸引了學者的興趣，並研究出各種延伸版本。Pei (2009)提出了Interactive Artificial Bee Colony (IABC)，以原來ABC的架構為基礎，在最後套用萬有引力定理來改良觀察蜂的選擇機制，Guopu & Sam (2010)提出了Gbest-guided Artificial Bee Colony algorithm (GABC)，利用目前所找到的最佳解作為導引，來引導工蜂與觀察蜂往此方向進行搜索，以縮短搜索的時間。Harikrishna(2009)提出Parallel Artificial Bee Colony (PABC)，以平行處理的概念改良人工蜂群演算法，藉此增加運算速度與解的品質。Reza等學者(2011)提出多目標的人工蜂群演算法(Multi-Objective Artificial Bee Colony, MOABC)以求解網格環境中工作流量排程的最佳化問題。Quan(2011)提出一個名為離散型工蜂群演算法(Discrete Artificial Bee Colony, DABC)用於求解彈性流程式生產批量分割的問題，Onder & M.Faith(2012)提出在延伸基本週期法下應用蜂群演算法求解，李維平(2014)等人提出以群中心策略改良人工蜂群演算

法，透過相對位置策略對偵查蜂的搜尋路線作出修改，並混合Guopu & Sam所提出的GABC，發展出名為CSGABC(Centroid-Strategy and Gbest-guided Artificial Bee Colony algorithm)的改良型蜂群演算法。Jian(2017)利用蜂群演算法求解多目標最佳化問題。

小結

在回顧文獻的過程中可以發現，以往學者研究應用固定速率法變更生產速率的 ELSP 模式時，大多都是共同週期法，郭育璋(2014)是第一個提出在基本週期法下應用固定速率法變更生產速率，但其假設是每個週期的閒置時間都跟第一個生產的閒置時間一樣，這和現實情況並不相符。蜂群演算法是一個近年興起的新演算法，擁有跳脫區域最佳解和群體資訊交換的特性，而且幾乎沒有被採用在求解變更生產速率的 ELSP 的問題上。本研究嘗試利用蜂群演算法來求解在基本週期法下應用固定速率法的 ELSP 問題，並在每個生產週期重新計算其閒置時間，以求得出的解能更貼近現實情況。

第三章 研究方法

本章節將建立起本研究的數學模式，首先描述本研究的基本假設，然後定義各個數學符號並建構出本研究的數學模式，最後說明蜂群演算法的原理並對本問題的模式進行求解。

本研究的問題是在基本週期法下以固定速率法更動產品生產速率，以求得最小的生產總成本；所使用的數學符號如下：

1. n ：產品的總項目
2. a_i ：產品 i 的生產整備成本
3. h_i ：產品 i 的存貨持有成本
4. s_i ：產品 i 的生產整備時間
5. d_i ：產品 i 的需求率
6. p_i ：產品 i 的生產率
7. h_i ：產品 i 的存貨持有成本
8. k_i ：產品 i 的基本週期乘數
9. B ：生產的基本週期
10. T ：產品的生產週期， $T=k_i * B$
11. $TC_{i,y}$ ：產品 i 在第 y 週期的單位時間平均成本
12. $TC(B,K)$ ：所有產品的總成本
13. K ：所有週期乘數的最小公倍數
14. Z_i ：產品 i 在 K 週期內的生產次數

其中本研究的決策變數是 k_i (產品 i 的基本週期乘數) 為決策變數，當 $\{k_i\}$ 決定後， B 就跟著決定了。

第一節 基本假設

本研究基本假設為以下：

1. 產品需求是持續不斷的。
2. 同一時間內只能生產一種產品。
3. 在一生產週期時間 $k_i * B$ 內，產品 i 僅會生產一次。
4. 在任何時間點內，產品 i 的需求率 d_i 、生產速率 p_i 、整備時間 s_i 、整備成本 A_i 和存貨持有成本 h_i 都是已知且不會隨時間改變。
5. 所有產品皆由同一機台生產，且機台的產能可以滿足所有產品的需求量。
6. 整備成本與整備時間只與被生產出來的產品有關，與生產的順序及批量大小無關。

第二節 降低生產速率的基本週期法模式建構

Buzacott and Ozkarahan(1983)認為只要能有效地利用機器的閒置時間進行生產，在降低其中一種產品的生產速率下，其餘產品仍以最大生產速率進行生產，就可以有效降低成本。而在套用基本週期法之後，由於每個週期生產的產品組合不同，設備的閒置時間也會不同，因此首先需掌握各個週期的閒置時間，才能決定要如何降低生產速率，接著依各週期的產品組合選擇出適合各週期降速的產品。以此方式決定各週期閒置時間，則單一產品在不同的生產週期降速時，會因不同的閒置時間而有不同的新生產速率。

接著，在基本週期法下適合降低生產速率的產品不一定在每個週期都會生產，由於已經確定了各週期的閒置時間，那麼要利用降速取得最低成本的話，需選擇該週期單位時間存貨持有成本($d_i h_i$)最高的產品來降速(Buzacott & Ozkarahan, 1983)，以得到最低的總成本。表 3-1 說明此概念，此範例共有三個產品以基本週期法進行生產，產品一的週期乘數 k_i 為 1；產品二的 k_i 為 3；產品三的 k_i 為 2，單位時間存貨持有成本($d_i h_i$)為 $d_3 h_3 > d_2 h_2 > d_1 h_1$ ，而選擇 $d_i h_i$ 最大的產品來做降速生產，將

可省下最多的成本，因此對於降速的產品選擇優先權為產品三 > 產品二 > 產品一，即依照產品的 $d_i h_i$ 大小來做為優先權的排序依據。

表 3-1 基本週期法下降速產品的選擇

週期 \ 產品	1	2	3	4	5	6
產品一($k_i=1$)	0	0	0	0	0	0
產品二($k_i=3$)	0			0		
產品三($k_i=2$)	0		0		0	
降速的產品	3	1	3	2	3	1
(O 表示產品在該週期會進行生產)						

最後，每當生產週期達到各產品週期乘數的最小公倍數時，便會成為一個重複生產的循環週期，因此對單一產品而言，可以平均一循環週期內的各週期之成本，用於計算平均單位時間成本。

本研究的數學模式如下：

$$TC_{i,y}(p_{i,y}^{new}) = \frac{k_i h_i d_i B \left(1 - \frac{d_i}{p_{i,y}^{new}}\right)}{2} + \frac{a_i}{k_i B} \quad (3-1)$$

$$TC_i = \frac{\sum_{y=1}^K TC_{i,y}(p_{i,y})}{Z_i} \quad (3-2)$$

$$\min TC = \sum_{i=1}^n \frac{\sum_{y=1}^K TC_{i,y}(p_{i,y}^{new})}{Z_i} \quad (3-3)$$

$$TC_{i,y} = \begin{cases} \frac{k_i h_i d_i B \left(1 - \frac{d_i}{p_{i,y}^{new}}\right)}{2} + \frac{a_i}{k_i B} & \text{若產品 } i \text{ 在第 } y \text{ 個週期有生產} \\ 0 & \text{若產品 } i \text{ 在第 } y \text{ 個週期沒有生產} \end{cases} \quad (3-4)$$

$$I_y = B - \sum_{i=1}^n Y_{i,y} \left(s_i + \frac{d_i k_i B}{p_{i,y}} \right) \quad y \in \{1, 2, \dots, K\} \quad (3-5)$$

$$B \geq \sum_i Y_{i,y} \left(s_i + \frac{d_i k_i B}{p_{i,y}} \right) \quad (3-6)$$

$$Y_{i,y} = \begin{cases} 1 & \text{若產品 } i \text{ 在第 } y \text{ 個週期進行生產} \\ 0 & \text{若產品 } i \text{ 在第 } y \text{ 個週期沒有進行生產} \end{cases} \quad (3-7)$$

$$p_{i,y}^{new} = \begin{cases} 0 & \text{若產品 } i \text{ 在第 } y \text{ 個週期沒有進行生產} \\ p_i & \text{若產品 } i \text{ 在第 } y \text{ 個週期進行生產, 但沒有降速生產} \\ d_i k_i B \times \frac{1}{\frac{d_i k_i B}{p_i} + I_y} & \text{產品 } i \text{ 有在第 } y \text{ 個週期內生產, 且有降速生產} \end{cases} \quad (3-8)$$

$$k_i = 2^{u_i}, u_i \in N \quad (3-9)$$

公式(3-1)表示產品 i 在第 y 週期的單位時間平均成本。

公式(3-2)為一項產品的平均單位時間成本，其中 Z_i 為所有產品在 K 週期內的生產次數，而 K 則是所有週期乘數的最小公倍數。

公式(3-3)是用於計算本研究的平均總成本。由於產品在不同週期可能會因為速率改變，而有不同的平均單位時間成本，這會導致不能比較解答品質的好壞。本研究將所有產品的週期乘數的最小公倍數設為 K ，當一產品以 K 個週期循環生產時， K 個週期內的所有產品之平均單位時間成本的總和來作解答。

公式(3-4)用來計算產品 i 在第 y 個週期的生產成本 $TC_{i,y}$ ；若產品 i 在第 y 個週期沒有生產，則 $TC_{i,y}=0$ 。

公式(3-5)是用於計算各個週期的閒置時間，令 I_y 為第 y 週期的閒置時間

公式(3-6)用於判斷每個週期的時間長度 B 是否足夠讓所有產品生產完成。

公式(3-7)用於判斷產品 i 在第 y 個週期是否有進行生產；若有則 $Y_{i,y}=1$ ，沒有話則 $Y_{i,y}=0$ 。

公式(3-8)用於計算要進行降速的產品在第 y 個週期的生產速率；產品的生產速率根據該產品在該週期的狀況而有三種可能：

1. 產品 i 沒有在第 y 個週期內生產，其生產速率是為 0。
2. 產品 i 有在第 y 個週期內生產，但沒有降速。
3. 產品 i 有在第 y 個週期內生產，且有降速生產。

公式(3-9)說明 k_i 的定義，基本週期法下，求解週期乘數 k_i 為關鍵問題，其求解策略大致分為兩種：一般整數策略(general integer policy)和二冪策略(power of two policy)。一般整數策略規定 k_i 必須是大於 0 的正整數，二冪策略由 McClain, Thomas and Mazzola (1992)提出，規定 k_i 必須是 2 的次方數 ($2^{u_i}, u_i \geq 0$)。因為二冪策略可以有效縮減搜尋空間，並且能得到良好的解答，所以以往探討 ELSP 的研究在求解以基本週期為基礎的解法時，如 BP 法和 EBP 法，均使用二冪策略來對週期乘數 k_i 進行求解。

第三節 人工蜂群演算法求解

基本週期法中，產品週期乘數 k_i 組合將直接影響解答的品質，根據公式(3-9)的二冪策略，只要求得 u_i 便可得到 k_i 。因此本研究嘗試利用人工蜂群演算法(Artificial Bee Colony algorithm, ABC)來求解 u_i 組合，藉由其資訊交換和更換食物源的特性，使得演算法不容易陷入區域最佳解，可期望能夠得到全域最佳解。

人工蜂群演算法由 Karaboga(2005)所提出，是一種模仿蜜蜂行為的最佳化方法。在自然環境中，蜜蜂會於蜂巢的資訊分享區透過跳搖擺舞(waggle dance)來與其他蜜蜂分享花蜜的多寡、距離和方向。當蜜蜂朝食物源的方向跳肚皮舞的時候，搖擺肚皮的次數越多，則代表該食物源距離蜂巢越遠。也有不一樣種類的蜜蜂，是利用繞搖擺舞圈子的速度，來告知其他蜜蜂同伴食物源距離蜂巢有多遠，距離越遠的時候就會繞得越慢，距離越近的時候繞得越快；若食物的數量越多，則跳舞時間會越久。另外，蜜蜂們會隨著太陽的方向與花蜜源還有蜂巢的相對位置而改變跳舞的角度。在 ABC 中，蜜蜂的搜索空間為參數最佳化問題的解答空間，工蜂所在的位置對應到一個隨機解。工蜂在搜尋過程中會記憶採過的花蜜(走過的位置)，並將新位置的花蜜(適應值)和舊的花蜜做比較，藉此不斷更新最佳值，工蜂在搜尋過程中也不斷與其他的工蜂進行資訊交換並把相關訊息給所有蜜蜂，獲得目前全域資訊的最佳位置與其對應的適應值，稱為全域最佳值(global best, *gbest*)。

人工蜂群演算法包含四個組成要素：食物源、工蜂(employed bees)、觀察蜂

(Onlookers)和偵察蜂(Scouts)；兩種行為：蜜蜂搜尋食物源和放棄某個食物源。四個階段：初始化階段、工蜂階段、觀察蜂階段、偵查蜂階段。

食物源：一個解空間裡的一個位置，意味著一個解，而食物源的花蜜，即其適應值。

偵察蜂：負責找出初始食物源和在放棄舊食物源後尋找新的食物源，它在工蜂到該食物源的位置後便回到蜂巢。

工蜂：負責在偵察蜂找到食物源後，計算其花蜜(適應值)後，將花蜜的訊息傳回給蜂巢中的觀察蜂，接著繼續用隨機移動的方式到附近的位置尋找花蜜，記錄並更新最好的花蜜，當經過一定次數之後得到的花蜜(適應值)都沒有改善後，工蜂會放棄該食物源，並呼叫偵察蜂去找新的食物源。

觀察蜂：負責協助工蜂，在工蜂將花蜜的訊息傳給蜂巢後，觀察蜂會根據花蜜量的高低來決定要到哪一隻工蜂的位置協助搜尋。

蜂巢出動的順序為偵察蜂、工蜂、觀察蜂。偵察蜂首先以公式(3-11)的隨機方式到解空間的隨機位置，作為初始食物源，接著工蜂取代偵察蜂的位置，進行搜索。

$$x_{bi}^j = x_{mini} + rand[0,1](X_{maxi} - X_{mini}) \quad (3-11)$$

x_{bi}^j 是第 b 隻工蜂在第 i 維度的第 j 次位置， X_{maxi} 為求解空間第 i 維度的上限， X_{mini} 為求解空間第 i 維度的下限。接著工蜂計算花蜜(適應值)後，得到目前的個體最佳解，然後跟所有工蜂比較以得到全域最佳值(global best, **gbest**)，並將花蜜的資訊傳回蜂巢。在工蜂不斷移動時，觀察蜂也會根據工蜂們傳回蜂巢的花蜜資訊決定到哪一隻工蜂的位置協助搜尋。

工蜂在解空間裡移動時，是以公式(3-12)隨機的方式選另一隻工蜂的位置並

向其所在方向前進， x_{bi}^{j+1} 為第 b 工蜂在第 i 維度移動後的新位置，每移動一次便會重新選另一隻工蜂作為移動目標。

$$v_{bi}^j = rand[-1,1](x_{bi}^j - x_{obi}^j) \quad ob = rand[1,SN] \quad ob \neq b \quad (3-12)$$

$$x_{bi}^{j+1} = x_{bi}^j + v_{bi}^j \quad (3-13)$$

工蜂在搜尋空間所處的新位置以 x_{bi}^{j+1} 表示， ob 族群中第 ob 隻工蜂， x_{obi}^j 為隨機選擇另一工蜂 ob 第 i 維度的位置， v_{bi}^j 為其移動方式，工蜂到達新位置後，計算該位置的適應值。



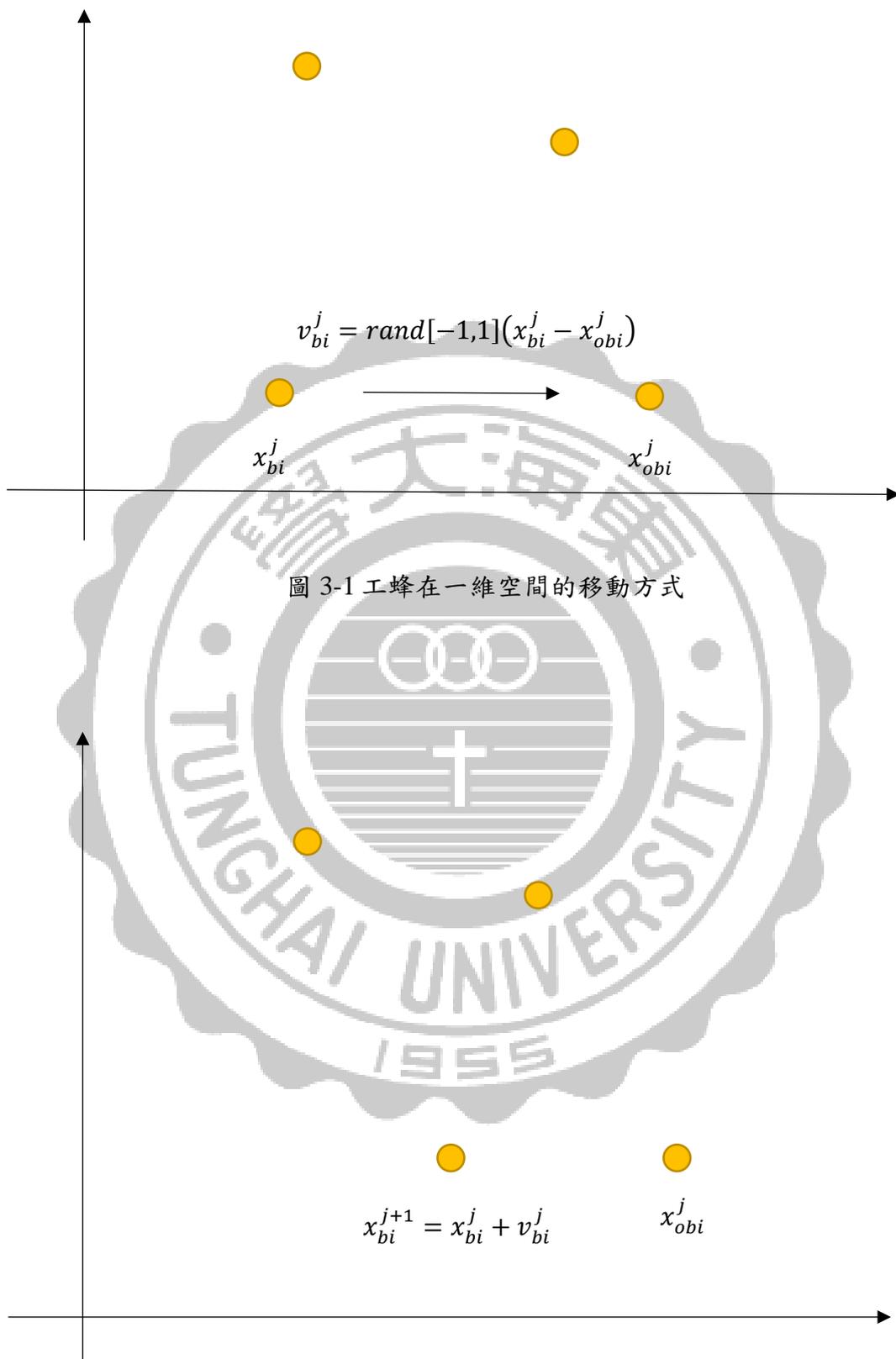


圖 3-1 工蜂在一維空間的移動方式

圖 3-2 工蜂在一維空間的移動後的新位置

觀察蜂在工蜂每一次於新位置計算完花蜜(適應值)後，會根據工蜂傳回蜂巢的花蜜訊息，依轉盤法(公式 3-14)決定要到哪一隻工蜂的所在的食物源位置。到達工蜂所在的位置後，觀察蜂用跟工蜂一樣的方式(公式 3-12)移動到該食物源的周圍，並計算該位置的花蜜，如果新位置的花蜜比工蜂位置的花蜜要好，那麼工蜂便會移動到該觀察蜂的位置，觀察蜂則回到蜂巢等候下次的分配。

$$p_w = \frac{fit_w}{\sum_w fit_w} \quad (3-14)$$

p_w 為第 w 個食物源被選上的機率， fit_w 為全部食物源數量中第 w 個食物源的適應值， $\sum_w fit_w$ 為總適應值，因此當食物源適應值愈高，被觀察蜂選上的機率也跟著提高。觀察蜂在輔助搜尋一次後，便回到蜂巢等候下一次分配。

蜂群演算法的執行

初始化階段

Step 1.1 首先設定工蜂數量(SN)，這是決定整體蜂群的大小，因為蜂群中工蜂跟觀察蜂的數量為 1:1，接著設定迭代數上限(MCN)、食物源 w 未改善次數的上限(limit_w)。

Step 1.2 蜂群中的偵察蜂們以公式 (3-11) 找到在搜索空間的隨機位置，接著工蜂便會跟隨到偵察蜂所在的位置。

工蜂階段

Step 2.1 工蜂根據公式(3-2)移動到新位置。

Step 2.2 接著，工蜂計算完新位置的適應值後，比較新舊地點的適應值，如果新的比較好，那新位置將會取代舊的食物源；若否，則維持不動。接著將適應值跟

g_{best} 比較，如果適應值比 g_{best} 高則更新 g_{best} 。

Step 2.3 當所有工蜂都計算完適應值後，將食物源適應值的資訊傳回給蜂巢中的觀察蜂。

觀察蜂階段

Step 3.1 觀察蜂將透過公式(3-14)的輪盤法決定應該到哪一隻工蜂的位置協助搜尋。

Step 3.2 觀察蜂依照公式(3-2)移動到新位置，計算該位置的適應值。

Step 3.3 觀察蜂計算完新位置的適應值後，比較新地點和工蜂位置的適應值，如果新的比較好，那工蜂將移動到新位置並取代舊的食物源；若否，工蜂則維持不動。

Step 3.4 將工蜂的適應值跟 g_{best} 比較，如果適應值比 g_{best} 高，則更新 g_{best} 。

Step 3.5 判定迭取代是否已達到最大值(MCN)；若是，結束演算法並輸出最佳解，若否，工蜂開始判定是否放棄食物源。

Step 3.6 工蜂判斷食物源 w 的未改善次數是否已達到 $limit_w$ ；若是，進入偵察蜂階段，若否，則回到工蜂階段。

偵察蜂階段

Step 4 偵察蜂以公式(3-11)找到在搜索空間的隨機位置作為新食物源，接著工蜂便會跟隨到偵察蜂所在的位置，回到工蜂階段。

人工蜂群演算法的流程如下圖：

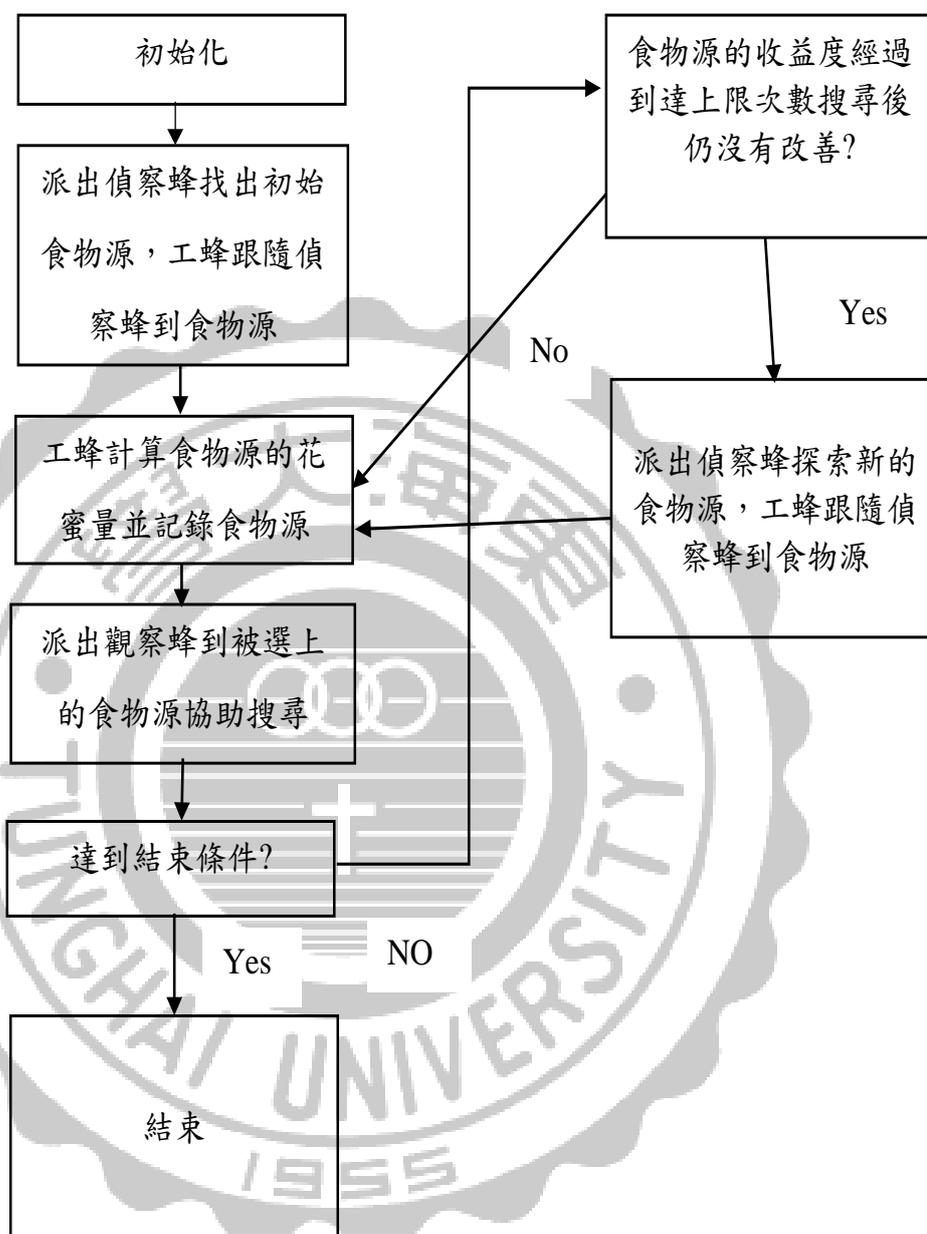


圖 3-3 人工蜂群演算法流程圖

第四節 人工蜂群演算法的改良

在蜂群演算法被提出之後，其放棄和搜尋新食物源的機制使得在求解過程中不容易落入區域最佳解而受到不少學者所注目。但因為蜜蜂的移動方式太過於隨機，因此不少學者嘗試對工蜂和觀察蜂的移動方法作出改良，以求縮短演算的搜尋時間並加強求解能力。

Pei & Jeng(2008)提出了Interactive Artificial Bee Colony algorithm (IABC)，主要是改良於數值最佳化問題的求解，透過在加入萬有引力公式來改變觀察蜂的移動方式。

Reza等學者(2012)提出多目標的人工蜂群演算法(Multi-Objective Artificial Bee Colony, MOABC)以尋求網格環境中最佳的工作流量排程，並且希望將執行時間與成本最小化。而實驗方面則是利用網格模擬器所得到的例題進行測試，並且與網格模擬器中的排程演算法分析比較；其結果說明MOABC不僅在求解上提供良好的範圍給予決策面，且所得到的解皆優於模擬器中的排程演算法。

Zhu & Kwong(2010)提出 Gbest-guided Artificial Bee Colony algorithm (GABC)，以目前所找到的最佳解作為導引，修改工蜂與觀察蜂的移動方式使他們往最佳解的方向進行搜索，其公式如下面所表示：

$$v_{bi}^j = x_{bi}^j + rand[-1,1](x_{bi}^j - x_{obi}^j) + rand[0,C](y_i - x_{bi}^j) \quad ob \in \{1,2,\dots,SN\}$$

$ob \neq b, C > 0$

y_i 為 g-best 在第 i 維度的位置，這引導方法和 PSO 以向量牽引將粒子導向最佳解方向的原理非常類似，雖然演算法除了在移動方式作修改之外其他部份幾乎沒有改變，但是帶來的效果卻非常明顯，能有效的提升演算法搜尋最佳解時的求解速度，因此，本研究將參考 GABC 中工蜂的移動方式來改良蜂群的移動方式。

第五節 人工蜂群演算法求解 ELSP

在基本週期法中，求解的品質好壞取決於所有產品的週期乘數 k_i 組合，因此在選定改良方式後，本研究便使用改良後的 ABC 來求解所有產品的 u_i 組合，以獲得最低總成本。透過 Soman *et al.*(2004) 提出的方式(附錄一)我們可以求出週期乘數 k_i 的上下限。原始的蜂群演算法只提供一維空間的移動方法，而在一個多維空間的情況，工蜂 b 在 n 維空間的位置可以用位置向量 $p_v = [x_{b1}^j, x_{b2}^j, x_{b3}^j, \dots, x_{bn}^j]$ 來表示。以本研究的目標問題而言，工蜂所處的位置向量為其所有產品的週期乘數組合。而工蜂回傳到的花蜜值，即其位置適應值(fitness value)可以 fit_b^j 表示， b 代表哪隻工蜂， j 代表蜜蜂 b 的第幾次移動，而 fit_b^j 即為此目標函數計算出的平均總成本。所有蜜蜂的每一個位置向量適應值，都會分別對應到一個由此最佳化問題目標函數計算得出的平均總成本。為了使蜂群演算法能配合本研究的 ELSP 問題，需要在公式上有所修改，首先因為本研究的目標為最低總成本，若適應值愈小，則表示平均總成本愈小，但這跟原始輪盤法相反，所以需要把公式(3-14)變更成公式(3-15)。

$$p_w = \frac{\frac{1}{fit_b^j}}{\sum_w \frac{1}{fit_b^j}} \quad (3-15)$$

其次，為了令演算法能更有效的進行搜尋，本研究將參考 Zhu & Kwong(2010) 的 Gbest-guided Artificial Bee Colony algorithm (GABC) 中工蜂和觀察蜂的移動方式，將公式(3-12)改寫成公式(3-16)

$$v_{bi}^j = rand[-1,1](x_{bi}^j - x_{obi}^j) + rand[0,1](y_i - x_{bi}^j) \quad ob = rand[1,SN] \quad ob \neq b \quad (3-16)$$

本研究在基本週期法下，應用固定速率法建構經濟批量排程問題模式。當給定一組週期乘數時，模式的求解流程如下：

Step 1. 找出一組所有產品的週期乘數組合。

Step 2. 以公式(3-6)確認週期時間 B 的產能是否足夠滿足各週期所有產品的需求；若能滿足則以公式(3-5)計算各週期的閒置時間，並前往 Step 5，若無法滿足則結束此程式。

Step 3. 以公式(3-8)計算各週期被選擇降速的產品之新生產速率。

Step 4. 以公式(3-4) 計算出各產品在第 y 週期的總成本。

Step 5. 以公式(3-3)計算平均總成本。

以上為給定一組週期乘數時的求解流程，而本研究問題模式的求解品質關鍵為所有產品週期乘數的組合，不同的組合會產生不同的成本，因此，需找出多組週期乘數組合並比較其適應值，即該週期乘數組合的生產總成本。本研究藉 ABC 來搜尋多組產品週期乘數組合，並求解出具有最低總成本的產品週期乘數組合，詳細運用 ABC 來求解的運作流程描述如下：

Step 1. 初始化階段

- 1.1. 設定要生產的產品數量為 n ，而產品數量即為所需的維度，因此即在 n 維空間中求解所有產品的週期乘數的次方數。
- 1.2. 設定工蜂數量(SN)、最大迭代數 (MCN)、食物源未改善次數的極限 ($maxlimit$)，工蜂總數=觀察蜂總數(OL)。
- 1.3. 設定 $b=1$ 。
- 1.4. 選擇第 b 隻工蜂，以公式 (3-11) 找出第 b 隻工蜂在所有維度 n 上的位置 X_b ， $X_b = (x_{b1}, x_{b2}, \dots)$ ，工蜂的位置以一陣列 pv 存取，每隻工蜂的位置陣列 pv 即為其產品週期乘數的次方數的組合。
- 1.5. 當所有維度上的位置都找到後，計算此位置陣列的適應值，意即 ELSP 的平均總成本，若是第一隻工蜂則其適應值也同時做為全體最佳解 $gbest$ 。
- 1.6. 將第 b 隻工蜂的適應值和 $gbest$ 做比較，因目標為最低總成本，若適應值小於 $gbest$ ，則以此工蜂的適應值取代 $gbest$ 。

- 1.7. 若 b 不大於 SN ，則 $b = b + 1$ 並回到 1.5 再次執行。若 b 等於 SN ，進入工蜂階段。
- 1.8. 設定 $t=1$ ， t 為迭代數。
- 1.9. 設定 $limit_w=0$ ，為食物源 w 的未改善次數。

Step 2. 工蜂階段

- 2.1 設定 $b=1$ 。
- 2.2 選擇第 b 隻工蜂。
- 2.3 透過公式(3-16)計算工蜂的移動方式，並使用公式(3-13)更新工蜂的位置，再找出第 b 隻工蜂在 n 維度空間上的位置 x_{bN} ，位置以一陣列 pv 存取。
- 2.4 計算第 b 隻工蜂新位置的適應值，若新位置的適應值小於適應值，則以新的適應值取代舊適應值。若大於適應值，則 $limit_w = limit_w + 1$ ，舊適應值維持不變。若 $limit_w$ 等於 $maxlimit$ (未改善次數的上限)，移動到偵查蜂階段，若否，繼續進行 Step2.5。
- 2.5 比較第 b 隻工蜂的適應值是否小於 $gbest$ ；若是，則以該工蜂的適應值取代 $gbest$ 。
- 2.6 若 b 不大於 SN ，則 $b = b + 1$ ，回到 Step2.2；若 b 大於 SN ，進入 Step 3 觀察蜂階段。

Step 3. 觀察蜂階段

- 3.1 設定 $b=1$ 。
- 3.2 設定 $r=1$ 。
- 3.3 $OINum=OL$ ， $OINum$ 是剩餘觀察蜂的數量， OL 為觀察蜂的總數。
- 3.4 觀察蜂透過公式(3-15)分發到各工蜂的位置，每次送出一隻觀察蜂時， $OINum = OINum - 1$ ，重覆此步驟直到 $OINum = 0$ 。
- 3.5 第 b 隻工蜂所分發到的觀察蜂數量為 Num_b^r 。
- 3.6 選擇第 b 隻工蜂。
- 3.7 設定 $b_r = 1$ 。

- 3.8 選擇屬於第 b 隻工蜂的第 r 隻觀察蜂，透過公式(3-16)計算觀察蜂的移動方式，並使用公式(3-13)更新觀察蜂的位置。再找出第 r 隻工蜂在 n 維度空間上的位置 b_r ，位置以一陣列 b_{pv}^r 存取。
- 3.9 若 r 不大於 Num_b^r ，則 $r = r + 1$ 並回到Step 3.7再次執行。若 $r = Num_b^r$ ，進入Step 3.10。
- 3.10 計算第 r 隻觀察蜂新位置的適應值，若新位置的適應值小於第 b 隻工蜂的適應值，則以此新適應值取代第 b 隻工蜂的適應值，同時工蜂的位置更新到觀察蜂的位置。
- 3.11 若 r 不大於 Num_b^r ，則 $r = r + 1$ 並回到3.8再次執行。若 $r = Num_b^r$ ，進入3.12。
- 3.12 比較第 b 隻工蜂的適應值是否小於 $gbest$ ，若是則以該工蜂的適應值取代 $gbest$ 。
- 3.13 若 b 不大於 SN ，則 $b = b + 1$ ， $r=1$ ，並回到Step 3.6再次執行。若 b 大於 SN ，回到Step 2.2。

Step 4. 偵查蜂階段(當 $limit_w$ 大於 $maxlimit$)

- 4.1 偵察蜂依公式(3-11)發現新的食物源。
- 4.2 第 b 隻工蜂移動到該位置後，以公式(3-11)找出第 b 隻工蜂在 n 維空間上的位置 X_b ， $X_b = (x_{b1}, x_{b2}, \dots)$ ，工蜂的位置以一陣列 p_v 存取。
- 4.3 計算第 b 隻工蜂的適應值並和 $gbest$ 做比較，若適應值小於 $gbest$ ，則以此適應值取代 $gbest$ 。
- 4.4 設定 $limit_b=0$ ，因為到達新的食物源，所以未改善次數重置。
- 4.5 回到 Step 2.6。

Step 5. 判斷是否結束

- 5.1 若 t 不大於 MCN ， $t=t+1$ ，回到 2.2，若 t 大於 MCN ，進入 Step 5.2。
- 5.2 輸出 $gbest$ 的適應值做為最佳解，即本 ELSP 模式的最低總成本。

小結

本研究為了能更精確的掌握設備的閒置時間，在每個生產週期重新計算其閒置時間，這樣可以使產品的降速時間有更大的調整空間。此外本論文首次應用蜂群演算法於降速的 ELSP 問題上，並透過修改其移動方式來改善工蜂和觀察蜂的移動方法，希望藉此提高演算法的求解效率。



第四章 數值實驗

本章節將討論在基本週期法下運用固定速率法的求解 ELSP 的情形，說明在不同的產品組合下，變更生產速率會對成本造成怎樣的變化。本研究先透過一組數值範例說明應用固定速率法時對成本所造成影響，再以隨機實驗檢驗 ELSP 模式的改善程度和演算法所帶來影響。

第一節 數值範例

本研究之實驗環境如表 4-1 所示，所使用的軟體為 NetBeans IDE，是一套由昇陽電腦（Sun Microsystems）建立的開放原始碼的軟體開發工具，可以用於 Java，C/C++，PHP，HTML5 等程式的開發，本身是一個開發平台，可以通過擴展插件來擴展功能。

表 4-1 實驗環境

作業*系統	Microsoft Windows 10
中央處理器	Intel(R) Core™ i7-4710MQ CPU@ 2.50GHz
記憶體	20 GB
開發軟體	NetBeans IDE 8.2
軟體版本	JDK 1.8.

本範例的設定為 1 台機台及 5 種產品，表 4-2 說明 5 種產品的參數資料，當中整備成本 A_i 和存貨持有成本 h_i 單位為「千元」， s_i 是整備時間以「天」為單位； d_i 為產品需求率、 p_i 為產品生產速率是以「個/天」作為單位。

表 4-2 數值範例產品的參數資料

產品號碼	A_i (千元)	h_i (千元)	s_i (天)	d_i (個/天)	p_i (個/天)
1	20	0.175	0.005	9	150
2	20	0.263	0.007	15	150
3	20	0.350	0.004	9	150
4	20	0.638	0.006	40	150
5	20	0.525	0.003	5	150

表 4-3 為傳統基本週期法所得出的結果，產品的週期乘數組合為(1,2,2,1,2)，週期時間為 1.931 天，而平均總成本為 73.223 千元。表 4-4 是應用郭育璋(2014)所提出的固定速率法降低生產速率的基本週期法所得出的結果，其特性為所有週期的閒置時間均和第一週期的閒置時間相同。產品的週期乘數組合為(1,2,2,1,1)，週期時間為 2.115 天，而平均總成本為 67.319 千元。因為利用了閒置時間進行降速生產，成本比傳統的基本週期法得到了 8%的改善，因此可得證應用固定速率法的確可以有效降低平均總成本。

表 4-3 傳統基本週期法求解的結果

產品號碼	1	2	3	4	5
週期乘數 k_i	1	2	2	1	2
週期時間 B (天)	1.912				
平均總成本(千元)	73.223				

表 4-4 基本週期法下應用郭育瑋(2014)的解法求解的結果

產品號碼	1	2	3	4	5
週期乘數 k_i	1	2	2	1	1
週期時間 B (天)	2.115				
平均總成本(千元)	67.319				

表 4-5 是本研究應用固定速率法降低生產速率，並每個生產週期開始時都會重新計算閒置時間，在基本週期法下所得出的結果。其產品的週期乘數組合為 (1,2,2,1,2)，週期時間為 1.912 天，而平均總成本為 61.713 千元。和傳統的基本週期法相比，本研究提出的方法將平均總成本降低了 15.7%，而和郭育瑋(2014)提出的解法比較，則平均總成本下降了 8.3%。

表 4-5 本研究的解法之求解結果

產品號碼	1	2	3	4	5
週期乘數 k_i	1	2	2	1	2
週期時間 B (天)	1.912				
平均總成本(千元)	61.713				

表 4-6 說明數值範例中應用本研究的解法，得到各週期的的生產負荷和單位時間生產成本。一產品在某週期生產時，此產品在該週期的生產負荷的計算方式如公式 (4-1)。

$$S_i + \frac{d_i B}{p_i^{new}} \quad (4-1)$$

表 4-6 本研究求解範例的生產成本和生產負荷

	產品編號	週期乘數(k_i)	週期 1	週期 2	週期 3	週期 4
單位時間 生產成本 (千元)	1	1	11.876	11.876	11.876	11.876
	2	2	12.019		12.019	
	3	2	10.891		10.891	
	4	1	21.654	12.040	21.654	12.040
	5	2	10.08		10.08	
生產負荷 (天)	1	1	0.120	0.120	0.120	0.120
	2	2	0.389	0.000	0.3894	0.000
	3	2	0.231	0.000	0.231	0.000
	4	1	1.039	1.792	1.039	1.79
	5	2	0.130	0.000	0.130	0.000
週期負荷(天)			1.912	1.912	1.912	1.912

以產品 2 為例，產品 2 的週期乘數是 2，所以產品 2 在週期 1 生產的產品須滿足週期 1 和 2 的需求，在週期 1 的生產負荷的計算方式如公式(4-1)所示，而產品 2 在週期 1 和 2 的單位時間平均成本是 12.019。而以產品 1 來說，該產品的週期乘數是 1，所以產品 1 必須在每一個週期都生產；所以此產品在各週期的單位時間平均成本是 11.876，生產負荷為 0.11972。此範例的所有產品之最小公倍數是 2，表示所有產品是以 2 個週期來循環生產；因此計算所有產品的單位時間平均成本的總和後，可得時間平均總成本 61.713。因為充份利用各週期的閒置時間，所以各週期的生產負荷等於該週期的長度。

第二節 數據實驗

本章節將以數據實測來檢驗本研究演算法的求解能力和使用固定速率法的對於成本的改善程度。Elmaghraby(1978)和 Moon *et al.*(2002)提出機台的產能使用率

可以反映經濟批量排程問題的求解困難度，產能使用率越高，求解的難度也會跟著提升，產能使用率的計算方式可以以公式(4-2)所表示

$$\sum_{i=1}^n \frac{d_i}{p_i} \quad (4-2)$$

因為本研究會對生產速率的進行降速，Elmaghraby(1978)和 Moon *et al.*(2002)所提出的產能使用率計算方式因無法反應閒置時間的真實程度，所以不能直接套用，必須作出改寫以符合實驗需求，改寫後的計算方式為公式(4-3)，作為產能使用率。本研究將產能利用率由 0.5-0.9 每隔 0.1 劃分成一區間共四個區間。產品數量則分成 3、5、8、10、12、15 成共六類。因此本研究將四個區間的產能利用率和六種產品數量產生 24 種隨機案例組合。每一個案例組合產生 20 個隨機案例，用來評估傳統的基本週期法、郭育璋(2014)和本研究所提出的解法之求解效率和求解品質。

$$f_{UF} = \frac{\sum_{i=1}^n \left(s_i + \frac{d_i B}{p_i} \right)}{B} \quad (4-3)$$

本研究所使用的隨機實驗參數資料由 Carreno(1990)所提出， A_i 為整備成本、 h_i 為存貨持有成本，都是以「千元」作為單位； s_i 為整備時間，以「天」為單位； d_i 為產品需求率、 p_i 為產品生產速率是以「個/天」作為單位。為了避免整備時間設定過高而影響可行解的品質，本研究將整備時時間的平均值調整為 0.05，範圍也為 0.05。產品資料的數值將會介於平均值 \pm (範圍/2)之間，以需求率作為例子，需求率 d_i 的平均值為 2500，範圍為 4800，因此其數值將會介於 $2500 \pm \frac{4800}{2}$ (= [100, 4900]) 之間。詳細產品的參數資料請參考表(4-7)。

表 4-7 數值實驗產品參數資料範圍

	平均值	範圍
存貨持有成本 h_i	0.35	0.70
整備成本 A_i	200	400
整備時間 s_i	0.05	0.05
需求率 d_i	2500	4800
生產率 p_i	14000	5000

表 4-7 為本研究的隨機實驗結果，左邊數據是郭育瑋(2014)的解法所得出的結果。郭育瑋(2014)的解法是應用固定速率法於基本週期法，並以第一個生產週期的閒置時間作為各週期的閒置時間來進行降速。右邊數據是本研究所提出的解法之求解結果，除了同樣應用固定速率法於基本週期法之外，本研究在每個生產週期都計算其閒置時間，以充分的閒置時間來進行降速。本研究以傳統基本週期法作為績效比較基準，來突顯郭育瑋(2014)的解法的求解品質和效率。

從表 4-8 中，郭育瑋(2014)的解法在 3 個產品，產能利用率 0.5-0.6 的情況下，平均改善率為 11.33%，證明加入利用各週期閒置時間的考量確實能提升固定速率法的效能；而本研究在同樣的案例組合下，平均改善率為 14.31%，優於郭育瑋(2014)之解法的求解表現。而 2 個解法在這案例組合的求解時間就差 0.4 秒。在 10 個產品，產能利用率 0.7-0.8 的案例組合下，相較於傳統基本週期法，郭育瑋(2014)的解法平均改善率為 1.34%，而本研究在同樣的情況下，平均改善率為 3.48%，兩者求解時間差 0.9 秒，依然優於郭育瑋(2014)的解法。最後在產品數 15，產能利用率 0.8-0.9 的案例組合下，相較於傳統基本週期法，本研究的平均改善率為 1.05%，郭育瑋(2014)的解法的平均改善率為 0.21%，求解時間相差 1.4 秒。

綜合上述，當產能利用率愈高和產品數量愈多的時候，郭育瑋(2014)和本研究的平均總成本改善幅度會明顯的下降。當產品數和產能利用率到達最高時，郭育瑋(2014)的改善率只剩 0.21%，而本研究的改善率也只有 1.05%。這是因為高產能利用率和多項產品會使得設備閒置時間減少，令到降低生產速率的閒置時間愈

來愈少，最後影響到平均總成本的改善幅度。但是在任何一個案例組合下本研究提出的解法都可以得到比郭育瑋(2014)和傳統基本週期法更低的生產成本。而從表 4-7 的實驗數據可以發現，雖然本研究的解法優於郭育瑋(2014)的表現;但是可以發現郭育瑋(2014)相較於傳統基本週期法的成本改善幅度遠高於本研究相較於郭育瑋(2014)的成本改善幅度。這有一個合理的解釋，郭育瑋(2014)的解法已利用大部份的閒置時間，所以可以取得較大幅度的改善績效;雖然本研究可以充分利用各週期的閒置時間，但相較於郭育瑋(2014)的解法可以取得的閒置時間已經不多了;所以本研究的解法可以取得比郭育瑋(2014)更好的表現但其改善幅度無法達到如同郭育瑋(2014)的改善效果。



表 4-8 隨機實驗的結果

產品數目	產能利 用率	郭育璋(2014)解法			執行時間 (秒)	本研究解法(%)			執行時間 (秒)
		Min	Max	Avg		Min	Max	Avg	
3	[0.5-0.6)	8.48	13.8	11.33	1.8	9.94	24.95	14.31	2.2
	[0.6-0.7)	6.44	11.54	10.16	1.0	10.26	14.196	12.21	1.8
	[0.7-0.8)	4.53	12.75	8.13	3.6	6.59	11.305	8.54	1.6
	[0.8-0.9)	1.87	8.10	4.77	1.2	3.89	8.10	5.51	1.2
5	[0.5-0.6)	1.72	7.36	4.75	3	6.65	11.02	8.77	6.8
	[0.6-0.7)	0.28	7.73	4.34	4.2	4.93	8.82	7.38	5.7
	[0.7-0.8)	1.12	5.41	3.32	1.6	3.22	7.76	5.96	5.2
	[0.8-0.9)	1.71	3.62	2.39	1.0	1.84	4.48	3.62	1.2
8	[0.5-0.6)	1.44	8.30	5.05	8.2	2.56	13.23	6.51	8.3
	[0.6-0.7)	0.35	1.63	1.10	5.8	2.36	5.43	3.55	8.4
	[0.7-0.8)	0.44	2.90	1.34	6.2	2.34	4.15	3.48	7.1
	[0.8-0.9)	0.37	1.40	0.93	2.6	1.37	3.16	2.06	3.0
10	[0.5-0.6)	0.92	5.16	3.43	7.4	3.5	6.45	4.48	9.2
	[0.6-0.7)	0.83	4.51	2.15	7.8	2.2	5.45	4.16	10.2
	[0.7-0.8)	0.45	3.61	1.43	8.2	2.1	4.04	2.79	8.9
	[0.8-0.9)	0.07	1.32	0.84	7.8	0.79	2.54	1.49	7.2
12	[0.5-0.6)	0.18	3.26	2.11	7.5	2.93	5.98	4.26	7.2
	[0.6-0.7)	0.04	4.53	2.35	6.6	2.75	3.57	3.29	7.8
	[0.7-0.8)	0.04	0.60	0.32	6.5	1.64	2.58	1.91	6.4
	[0.8-0.9)	0.62	1.16	0.47	6.2	0.65	1.67	1.20	3.2
15	[0.5-0.6)	0.03	4.18	2.26	8.4	2.91	3.89	3.52	7.1
	[0.6-0.7)	0.48	4.36	1.69	8.6	1.85	3.43	2.65	6.6
	[0.7-0.8)	0.56	3.84	1.73	7.1	1.57	2.19	2.62	7.0
	[0.8-0.9)	0.01	0.51	0.21	8.2	0.75	1.59	1.05	9.6

小結

整體來說，本研究透過取得各週期的閒置時間來對產品進行充分的降速，求解品質優於比郭育璋(2014)和傳統的基本週期法，但由於郭育璋(2014)已經利用了大部份的閒置時間，所以本研究的改善幅度比不上郭育璋(2014)相對於傳統的基本週期法的改善幅度。但因為已完全利用各週期的閒置時間，所以本研究所提出的解法成效比傳統的基本週期法和郭育璋(2014)的解法要好。而當產能利用率提高時，本研究所提出的解法也會因為閒置時間變少而使降速改善成本的效果變得不明顯。



第五章 結論與未來研究

第一節 結論

本研究建立出一個在基本週期法下結合蜂群演算法和固定速率法經濟批量排程問題的模型。由於在以往的文獻中並沒有運用蜂群演算法來求解基本週期法下變更生產速率的研究，因此這研究是一個全新的嘗試，以往的 ELSP 降速是以第一生產週期的閒置時間作為以後週期的閒置時間，為更貼近現實的生產情況，本研究將加入每個生產週期的閒置時間都重新計算的條件。

本研究探討在基本週期法下變更生產速率所產生的問題，包括：

1. 機器在每個週期的閒置時間長度都不一樣。
2. 產品的平均單位時間成本的計算方式。
3. 如何產生合理可行的解決方式。
4. 每個週期降速的產品並不相同。

而本研究是首次利用蜂群演算法來求解變更生產速率的經濟批量排程問題的研究，透過其跳脫區域最佳解和群體資訊交換的特性，來搜尋最佳的產品週期乘數組合 k_i ，進而得到最低總成本。經過第四章的數值實驗後，證明本研究提出的基本週期法下應用固定速率法結合蜂群演算法所產生出來的模式，能得出比以往研究更低的生產總成本。

第二節 未來研究

本研究尚有不足的地方，以下將針對其中明顯的部分作為未來研究方向：

1. 加入更多的限制條件：

本研究除了計算出各週期的閒置時間之外，其他條件均為基本的在基本週期法下應用固定速率法的 ELSP 模式；如果能加入更多的限制條件，如加入產品保存期限、機器耗損的情況、產品退回、有機會生產出瑕疵品、銷售損失等限制式，將能更進一步的符合現實生產的情況。

2. 演算法的改進：

本研究所利用的蜂群演算法比起基本版本，雖然在移動方式有作出改良使得效率有所提升，但改良移動方式並非唯一的改善方法，像是觀察蜂挑選工蜂的方式、偵察蜂產生新食物源的方式等，還有不少方法可以作出改良，日後如果能結合各種改良方法並針對 ELSP 問題作出修改，相信求解品質還可以再進一步的提升。

3. 探討應用在彈性速率法和延伸基本週期法上：

根據以往文獻，彈性速率法和延伸基本週期法由於彈性較高，所以大多能得出更好的可行或最佳解。但因為結構相對複雜不少，在初次研究就在應用在蜂群演算法上有一定難度。日後若能將蜂群演算法順利的應用於在彈性速率法跟延伸基本週期法下的 ELSP 問題上，相信將能得更好的最佳解。

參考文獻

1. 林子元 (2009), 《應用粒子群演算法於半導體晶圓廠設施佈置問題之研究》, 國立台北科技大學工業工程與管理研究所碩士論文, 未出版。
2. 林信宏 (2010), 《粒子群演算法於多目標排程問題之研究》, 國立交通大學工業工程與管理系所博士論文, 未出版。
3. 李維平、黃郁授、戴彰廷 (2008), 《自適應慣性權重改良粒子群演算法之研究》, 國立屏東教育大學, *資訊科學應用期刊*, 第四卷, 第一期。
4. 李維平、李元傑、謝明勳(2014), 以群中心策略改良人工蜂群演算法, *資訊管理學報*, 第二十一卷, 第一期, 第 25-44 頁。
5. 李世炳、鄒忠毅(2002), 簡介導引模擬退火法及其應用, *物理雙月刊*, 第二十四卷, 第二期, 第 308-319 頁。
6. 陳世杰、張育仁、詹智欽、姚銘忠 (2011), 以遺傳演算法求解多機台經濟批量檢驗與排程問題, 2011 工業工程與管理年會暨學術研討會。
7. 陳世杰、張育仁、黃煜舜、黃晞臨 (2012), 應用固定生產速率求解生產系統下經濟批量檢驗與排程問題, 2012 工業工程學會年會暨學術研討會。
8. 陳貴敏、賈建 (2006), 粒子群優化算法的慣性權重遞減策略研究, *西安交通大學學報*, 第四十卷, 第一期。
9. 張育仁、姚銘忠 (2005), 以遺傳演算法求解一般整數策略下之多機經濟批量排程問題, *計量管理期刊*, 第二卷, 第 1~14 頁。
10. 張孝裕(2006), 《倉儲受限下可變動生產速率之兩產品經濟批量排程問題》, 國立成功大學工業與資訊管理學系碩士論文, 未出版。
11. 張育仁 (2011), 以基本週期法求解有回流狀況的經濟批量排程問題, 第六屆流通與全球運籌論文研討會。
12. 郭育瑋(2014), 《以簡單固定速率法求解在基本週期下有保存期限限制的經濟批量排程問題》, 私立東海大學資訊管理學系碩士論文, 未出版。
13. 潭立靜、樂麗君、牛奔、虛並寧 (2006), 《一種求解函數優化的新型混合優化算法》, 遼寧工程技術大學機械工程學院。

14. 蔣其軒 (2007), 《利用共同週期法求解不完美生產系統下可變動生產速率之經濟批量排程問題》, 國立成功大學工業與資訊管理學系碩士論文, 未出版。
15. 蔡季軒 (2012), 《生產一般產品與再生產品製造商利潤最大化經濟批量排程問題之研究》, 國立交通大學運輸科技與管理學系碩士論文, 未出版。
16. 賴阮明 (2009), 《整合粒子群最佳化與蜂群演算法求解彈性零工式生產排程問題之研究》, 國立台北科技大學工業工程與管理研究所碩士論文, 未出版。
17. 賴閔揚 (2014), 《多目標基因演算法應用於混合流程型生產排程問題》, 國立清華大學工業工程與工程管理學系碩士論文, 未出版。
18. 羅榮武 (2009), 考慮貨架存放期的經濟批量排程問題的改良演算法, *運籌與管理*, 第 18 卷, 第 2 期, 第 143-148 頁。
19. Adhisatya Filemon Yoga, Jin Ai The and Gong Dah-Chuan (2015). Economic lot scheduling problem with two imperfect key modules, *Industrial Engineering and Service Science 2015, Procedia Manufacturing*, 4, 30–37
20. Ai and Kachitvichyanukul (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery, *Computers and Operations Research*, 36, 1693-1702
21. Akbaria Reza, Hedayatzadeha Ramin, Ziaratib Koorush and Hassanizadeha Bahareh (2012). A multi-objective artificial bee colony algorithm, *Swarm and Evolutionary Computation*, 2, 39–52.
22. Ariyasingha, I.D.I.D. and Fernando, T.G.I. (2015). Performance analysis of the multi-objective ant colony optimization algorithms for the traveling salesman problem, *Swarm and Evolutionary Computation*, 23, 11–26
23. Boctor, F.F. (1987). The g-group heuristic for single machine lot scheduling, *International Journal of Production Research*, 25, 363-379.
24. Bomberger, E. (1966). A dynamic programming approach to a lot size scheduling problem, *Management Science*, 12, 778-784.
25. Bukata Libor, Šůcha Přemysl and Hanzálek Zdeněk (2015). Solving the resource constrained project scheduling problem using the parallel tabu search designed for the CUDA platform, *Journal of Parallel and Distributed Computing*, 77, 58–68.

26. Bulut Onder and Tasgetiren M. Fatih (2011). A discrete artificial bee colony algorithm for the economic lot scheduling problem, *2011 IEEE Congress of Evolutionary Computation (CEC)*, 347–353
27. Bulut Onder and Tasgetiren M. Fatih (2014). A discrete artificial bee colony algorithm for the Economic Lot Scheduling problem with returns, *2014 IEEE Congress on Evolutionary Computation (CEC)*, 551-557
28. Buzacott, J.A. and Ozkarahan, I.A. (1983). One- and two-stage scheduling of two products with distributed inserted idle time: the benefits of a controllable productions rate, *Naval Research Logistics Quarterly*, 30, 675-696.
29. Capa Canan, Akgunduz Ali ,Demirli Kudret (2014). A bi-objective genetic algorithm for economic lot scheduling problem under uncertainty, *2014 Industrial and Systems Engineering Research*, 2692-2701
30. Carreno, J.J. (1990). Economic lot scheduling for multiple products on parallel identical processors, *Management Science*, 36, 348-358.
31. Cesaret Bahriye, Oğuz Ceyda, Sibel Salman, F. (2012). A tabu search algorithm for order acceptance and scheduling, *Computers & Operations Research, Special Issue on Scheduling in Manufacturing Systems*, 39(6), 1197–1205
32. Chang, Y.J. and M.J. Yao (2011). New heuristics for solving the economic lot scheduling problem with reworks, *Journal of Industrial Management Optimization*, 7(1), 229-251.
33. Chung Sai Ho and Chan Hing Kai (2011). A two-level genetic algorithm to determine production frequencies for economic lot scheduling problem, *IEEE Industrial Electronics Society*, 611-619.
34. Clausen, J. and Ju, S. (2006). A hybrid algorithm for solving the economic lot and delivery scheduling problem in the common cycle case, *European Journal of Operational Research*, 175, 1141-1150.
35. Davis, S.G. (1990). Scheduling economic lot size production runs, *Management Science*, 36, 985-998.
36. Dobson, G. (1992). The cyclic lot scheduling problem with sequence-dependent setups, *Operations Research*, 40(4), 736-749.
37. Dobson, W.L. (1964). Scheduling of economic lot sizes. *Naval Research Logistics*
38. Eberhart, R.C. and Kennedy, J. (1995). New optimizer using particle swarm theory, *Proc. Sixth International Symposium on Nagoya, Japan*, 39-43.
39. Eberhart, R.C. and Shi, Y. (1998). Comparison between genetic algorithms and

- particle swarm optimization, *Annual Conference on Evolutionary Programming*, San Diego.
40. Eberhart, R.C. and Shi, Y. (2001). Particle swarm optimization: developments, applications and resources, *Proc. IEEE Int. Conf. On Evolutionary Computation*, 81-86.
 41. Elmaghraby, S.E. (1978). The economic lot scheduling problem (ELSP): review and extension, *Management Science*, 24, 587-597.
 42. Geng, P.C. and Vickson, R.G. (1988). Two heuristics for the economic lot scheduling problem: an experimental study, *Naval Research Logistics*, 35, 605-617.
 43. Giri, B. C., Moon, I. and Yun, W. Y. (2003). Scheduling Economic Lot Sizes in Deteriorating Production Systems, *Naval Research Logistics*, 50, Pages 650-661.
 44. Hanssmann, F. (1962). Operation research in production and inventory control, New York, Wiley.
 45. Hing Kai Chan, Sai Ho Chung and Tak Ming Chan (2012). Combining genetic approach and integer programming to solve multi-facility economic lot-scheduling problem, *Journal of Intelligent Manufacturing*, 23(6), 2397-2405
 46. Hsu, W. L. (1983). On the general feasibility test of scheduling lot sizes for several products on one machine, *Management Science*, 29, 93-105.
 47. Jarosław Pempera, Czesław Smutnicki and Dominik Żelazny (2013). Optimizing bi-criteria flow shop scheduling problem by simulated annealing algorithm, *Procedia Computer Science*, 2013 International Conference on Computational Science, 18, 936-945.
 48. Jenabia, M., Fatemi Ghomia, S.M.T., Torabib, S.A. and Karimia, B. (2007). Two hybrid meta-heuristics for the finite horizon ELSP in flexible flow lines with unrelated parallel machines, *Applied Mathematics and Computation*, 186(1), 230-245.
 49. Karaboga Dervis (2005). An idea based on honey bee swarm for numerical optimization, *Technical Report-TR06*.
 50. Karaboga Dervis and Basturk Bahriye (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, *Journal of Global Optimization*, 39(3), 459-471
 51. Kayvanfar V. and Zandieh M. (2012). The economic lot scheduling problem with deteriorating items and shortage: an imperialist competitive algorithm, *The*

- International Journal of Advanced Manufacturing Technology*, 62(5), 759-773.
52. Khouja, M. (1999). A note on 'Deliberately slowing down output in a family production context', *International Journal of Production Research*, 37, 4067-4077.
53. Khouja, M., Michalewicz, Z. and Wilmot, M. (1998). The use of genetic algorithms to solve the economic lot size scheduling problem, *European Journal of Operation Research*, 110, 509-524.
54. Khoury, B. N., Abboud, N. E. and Tannous, M. M. (2001). The common cycle approach to the ELSP problem with insufficient capacity, *International Journal of Production Economics*, 73, 189-199.
55. Leachman, R. C. and Gascon, A. (1988). A heuristic scheduling policy for multi-item, single-machine production systems with time-varying, stochastic demands, *Management Science*, 34, 377-390.
56. Li, Z., Li, Y. and Xu, L. (2011). Anomaly intrusion detection method based on k-means clustering algorithm with particle swarm optimization, information technology, *Computer Engineering and Management Sciences*, 2, 157-161.
57. Lia Heng and Zhang Hong (2013). Ant colony optimization-based multi-mode scheduling under renewable and nonrenewable resource constraints, *Automation in Construction*, 35, 431-438
58. Lian Zhigang, Jiao Bin and Gu Xingsheng (2011). A similar particle swarm optimization algorithm for job-shop scheduling to minimize makespan, *Applied Mathematics and Computation*, 183, 1008-1017.
59. Lima, M.F., Sampaio, L.D.H., Zarpelao, B.B., Rodrigues, J.J.P.C., Abrao, T. and Proenca, M.L. (2010). Networking anomaly detection using dns and particle swarm optimization with re-clustering, *Global Telecommunications Conference*, 1-6.
60. Lin, C.S., Chen, C.H. and Kroll, D.E. (2003). Integrated production-inventory models for imperfect production processes under inspection schedules, *Computers and industrial Engineering*, 44, 633-650.
61. Löhndorfa Nils, Riel Manuel and Minnerb Stefan (2014). Simulation optimization for the stochastic economic lot scheduling problem with sequence-dependent setup times, *International Journal of Production Economics*, 157, 170-176
62. Luo Jianping, Liu Qiqi, Yang Yun, Li Xia, Chen Min-rong and Cao Wenming. (2017). An artificial bee colony algorithm for multi-objective optimization, *Applied Soft Computing*, 50, 235-251

63. Mansouri, S. Afshin (2006). A simulated annealing approach to a bi-criteria sequencing problem in a two-stage supply chain, *Computers & Industrial Engineering*, 50(1-2), 105-119.
64. McClain, J. O. (1992). *Operations Management: Production of Goods and Services*, 3rd ed. Prentice Hall.
65. Merwe, D.W. and Engelbrecht, A.P. (2003). Data clustering using particle swarm optimization, *Proceedings of IEEE Congress on Evolutionary Computation 2003*, 215-220.
66. Mohammadi Maryam, Nurmaya Musa, S. and Bahreininejad Ardeshir (2014). Optimization of mixed integer nonlinear economic lot scheduling problem with multiple setups and shelf life using metaheuristic algorithms, *Advances in Engineering Software*, 78, 41–51.
67. Moon, D. H. and Christy P. D. (1998). Determination of optimal production rates on a single facility with dependent mold lifespan, *International Journal of Production Economics*, 54(1), 29-40.
68. Moon, I., Silver, E.A. and Choi, S. (2002). Hybrid genetic algorithm for the economic lot-scheduling problem, *International Journal of Production Research*, 40, 809-824.
69. Narasimhan Hari Krishna (2009). Parallel artificial bee colony (PABC) algorithm, *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, 306-311
70. Pan Quan-Ke , Tasgetiren, M. Fatih , Suganthan, P.N. and Chua, T.J. (2011). A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem, *Information Sciences* 181, 2455–2468.
71. Park, K.S. and Yun, D.K. (1987). Feasibility test for multi-product lot size scheduling on one machine, *Policy and Information*, 11, 01-108.
72. Pavani Gustavo Sousa, and Tinini Rodrigo Izidoro (2016). Distributed meta-scheduling in lambda grids by means of ant colony optimization, *Modeling and Management for Big Data Analytics and Visualization, Future Generation Computer Systems*, 63, 15–24
73. Petrović Milica , Vuković Najdan , Mitić Marko and Miljković Zoran (2016) Integration of process planning and scheduling using chaotic particle swarm optimization algorithm, *Expert Systems with Applications*, 64, 569–588. *Quarterly*, 11(2-3):89- .

74. Robinson, Jacob, Sinton, Seelig and Rahmat-Samii, Yahya (2002). Particle swarm genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna, *IEEE Antennas and Propagation Society, AP-S International Symposium*, 314-317.
75. Rogers, J. (1958). A computational approach to the economic lot scheduling problem, *Management Science*, 4, 264-291.
76. Sarker, R. and Newton, C. (2002) A genetic algorithm for solving economic lot size scheduling problem, *Computers & Industrial Engineering*, 42, 189-198.
77. Sifaleras Angelo , Konstantaras Ioannis and Mladenović Nenad (2015). Variable neighborhood search for the economic lot sizing problem with product returns and recovery, *International Journal of Production Economics*, 160, 133–143
78. Silver, E. A. (1989). Shelf life considerations in a family production context, *International Journal of Production Research*, 27(12), 2021–2026.
79. Silver, E. A. (1995). Dealing with a shelf life constraint in cyclic scheduling by adjusting both cycle time and production rate, *International Journal of Production Research*, 33(3), 623–629.
80. Silver, E.A. (1990). Deliberately slowing down output in a family production context, *International Journal of Production Research*, 28, 17-27.
81. Soman, C. A., Van Donk, D. P. and Gaalman, G. (2003). Combined make-to-order and make-to-stock in a food production system, *International Journal of Production Economics*, 90(2), 223-235.
82. Soman, C.A., van Donk, D.P. and Gaalman, G.J.C. (2004). A basic period approach to the economic lot scheduling problem with shelf life considerations, *International Journal of Production Research*, 42(8), 1677-1689.
83. Tang, O. and Teunter, R.H. (2006). Economic lot scheduling problem with returns, *Production and Operations Management*, 15(4), 488-497.
84. Teunter, R.H., Kaparis, K. and Tang, O. (2008). Multi-product economic lot scheduling problem with separate production lines for manufacturing and remanufacturing, *European Journal of Operational Research*, 191, 1241-1253.
85. Transchel, S. and Minner, S. (2009). The impact of dynamic pricing on the economic ordering decisions, *European Journal of Operational Research*, 198(3), 773-789.
86. Tsai Pei-Wei, Pan Jeng-Shyang, Liao Bin-Yih, and Chu Shu-Chuan (2009). Enhanced artificial bee colony optimization, *International Journal of*

- Innovative, Computing, Information and Control*, 5(12), 01-108.
87. Van Donk, D. P. (2001). Make to stock or make to order: the decoupling point in the food processing industries, *International Journal of Production Economics*, 69(3), 297-306.
 88. Viswanathan, S. (1995) A note on effect of production cost on shelf life, *International Journal of Production Research*, 33(12), 3485-3486.
 89. Viswanathan, S. and Goyal, S. K. (1997). Optimal cycle time and production rate in a family production context with shelf life considerations, *International Journal of Production Research*, 35(6), 1703-1711.
 90. Viswanathan, S. and Goyal, S. K. (2000). Incorporating planned backorders in a family production context with shelf-life considerations, *International Journal of Production Research*, 38(4), 829-836.
 91. Yang, J., Yan, H. and Taksar, M. I. (2000). Optimal production and setup scheduling: a one-machine, two-product system, *Annals of Operations Research*, 98, 291-311.
 92. Yao, M.J., Chen, S.C., Chang, Y.J. and Tseng, T.Y. (2012). Solving the economic lot and inspection scheduling problem using the extended basic period approach under power-of-two policy, *Journal of the Chinese Institute of Industrial Engineers*, 29(1), 43-60.
 93. Yu Shun-Chi (2014). Elucidating multiprocessors flow shop scheduling with dependent setup times using a twin particle swarm optimization, *Applied Soft Computing*, 21, 578–589.
 94. Zandoni Simone, Segerstedt Anders, Tang Ou and Mazzoldi Laura (2012). Multi-product economic lot scheduling problem with manufacturing and remanufacturing using a basic period policy, *Computers & Industrial Engineering*, 62(4), 1025–1033.
 95. Zhao Fuqing , Tang Jianxin , Wang Junbiao and Jonrinaldi (2014) An improved particle swarm optimization with decline disturbance index (DDPSO) for multi-objective job-shop scheduling problem, *Computers & Operations Research*, 45, 38–50
 96. Zheng, Y., Ma ,L., Zhang, L., and Qian, J. (2003). Empirical study of particle swarm optimizer with an increasing inertia weight, *Proceeding of IEEE Congress on Evolutionary Computation*, Canbella, Australia, 221-226.
 97. Zhua Guopu and Kwong Sam (2010). Gbest-guided artificial bee colony

algorithm for numerical function optimization, *Applied Mathematics and Computation*, 217(7), 3166–3173



附錄一

Soman *et al.*(2004)提出的方式來求出週期乘數 k_i 的上限，其步驟如下：

Step 1. 以獨立解法(公式 1)求出各產品 i 所需的週期時間 T_i ；

$$T_i = \sqrt{(2a_i/[h_i d_i (1 - \frac{d_i}{p_i})])} \quad (1)$$

Step 2. 選擇最小的 T_i 作為基本週期法的週期時間 B (公式 2)。

$$B = \min(T_i) \quad (2)$$

Step 3. 求出週期乘數的下界 k_i^- 和上界 k_i^+ ，其中 $k_i^- = \{1, 2, 4, 8, 16, \dots\}$ 做為週期乘數的下界， $k_i^+ = \{1, 2, 4, 8, 16, \dots\}$ 做為週期乘數的上界(公式 3)。

$$k_i^- \leq \frac{T_i}{B} \leq k_i^+ \quad (3)$$

Step 4. 求出週期乘數的下界 k_i^- 和上界 k_i^+ ，其中 $k_i^- = \{1, 2, 4, 8, 16, \dots\}$ 做為週期乘數的下界， $k_i^+ = \{1, 2, 4, 8, 16, \dots\}$ 做為週期乘數的上界。

Step 5. 將 k_i^- 和 k_i^+ 兩者帶入公式(1)計算成本，並取兩者中成本較低的值做為 k_i 。

Step 6. 將 Step4 取得的 k_i 帶入公式(4)，重新計算週期時間 B 。

$$B^{new} = \sqrt{\frac{2[\sum \frac{a_i}{k_i}]}{\sum h_i d_i k_i (1 - \frac{d_i}{p_i})}} \quad (4)$$

Step 7. 藉由 Step5 找出的新週期時間 B^{new} 帶回 Step3 求取各產品的新 k_i^- 和 k_i^+ ，以
求出各產品的 k_i 。當所有產品的 k_i 和上一次求出的 k_i 完全相同時，此 k_i 即
為該產品的 k_i 上界，則結束此程序。

透過以上的步驟即可得到各產品的 k_i 上界。