

東海大學資訊管理研究所  
碩士學位論文

以彈性速率法求解經濟批量排程問題的倉儲大  
小與成本

Applying the Flexible Rate Approach to Solve the Cost and  
Warehouse Size of the Economic Lot Scheduling Problem

指導教授：張育仁 博士  
研究生：陳 蘊

中華民國 104 年 06 月

## 致謝

三年前，一場車禍在家休養的時間報考了研究所，卻讓我考上了研究所，展開兩年的研究所生涯，取得碩士學位，真得是因禍得福。這兩年間，時間過得特別快，而我能如期完成論文順利畢業，非常非常感謝張育仁教授，感謝教授在學業上的指導讓我學習到許多學術上的知識，也會教導我們在職場上會遇到問題的時候該如何處理，還給了我實習的機會，讓我在這兩年的碩士生活中獲益良多。另外，也非常感謝口試委員張寧群教授、曾懷恩教授與陳世杰教授，在百忙之中撥空期來指導，且提供寶貴的建議，使本論文能更加完善，在此由衷的感謝口試委員。

在兩年研究所生涯中，要感謝實驗室的同學，進皓、爾廷、侑侑、盈秀、晞臨，那些一起寫論文的的日子，互相砥礪、每天都一起奮鬥的經歷，雖然很辛苦但也很值得。也感謝陪伴我兩年的好朋友們，小比、阿花、阿萍、阿耀、阿雯、杏樺、維尼等所有朋友們，謝謝你們和我一起分享喜、怒、哀、樂。最後，我要感謝我家人給予的支持與鼓勵，總是擔心我太晚回家，要我注意安全，在重要的日子裡也給了我許多的精神喊話，讓我能渡過許多困難關卡，爸爸媽媽我畢業了，謝謝你們。

論文名稱：以彈性速率法求解經濟批量排程問題的倉儲大小與成本

校所名稱：東海大學資訊管理學系研究所

畢業時間：2015 年 6 月

研究生：陳 蘊

指導教授：張育仁 博士

論文摘要：

經濟批量排程問題是指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。而為了貼近現實的生產情況，學者們陸續將經濟排程問題分成各種情境來討論，其中一種考量許多企業生產完產品時，都需要進入儲藏的過程，就會有倉儲租賃成本。因此總成本不可能只去計算存貨持有成本和整備成本，期間的倉儲的租賃成本也該納入考量；本研究假設企業需要租借外部倉儲空間，建立數學模式，求解經濟批量排程問題的倉儲空間與總成本(包括租賃成本)，目標為求得最佳總成本。

傳統上，經濟批量排程問題皆假設機台以最大的生產速率來進行生產，有學者研究發現在機器有閒置時間下，有效的降低生產速率進行生產，可減少存貨持有成本，平均總成本因而下降。因此本研究考慮多個產品的經濟批量排程問題，假設機器在共同週期法下，挑選產品中存貨持有成本最高的產品，使用閒置時間進行彈性速率法來變更生產速率，並求解所需的倉儲空間的大小。

求解此問題的關鍵因素為利用蟻群系統演算法搜尋較佳化的生產順序以求得較佳的可行解。數據結果顯示共同週期法下運用彈性速率法可以比共同週期法下運用固定速率法求解出更低的成本，有明顯的改善程度。本研究提出在共同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、共同週期、彈性速率、倉儲、蟻群系統。

Title of Thesis : Applying the Flexible Rate Approach to Solve the Cost and Warehouse Size of the Economic Lot Scheduling Problem

Name of Institute : Tunghai University, Institute of Information Management

Graduation Time : 2015/06

Student Name : Yun Chen

Advisor Name : Dr. Yu-Jen Chang

Abstract :

The ELSP is concerned with the scheduling of cyclical production of  $n \geq 2$  products on a single facility in equal lots over an infinite planning horizon, assuming stationary and known demands for each product. The objective of the ELSP is to determine the lot size and the schedule of production of each product so as to minimize the total cost incurred per unit time. When a company does not have enough storage space, it must rent an outsourcing warehouse space from other companies to store products. If this condition happening, a manager will need to consider following: (1) how to calculate the exact warehouse size and (2) how does the rent cost affect the average total cost. Therefore, this study discusses the above two issues as a company must rent an outsourcing warehouse space from other companies. This research uses the flexible rate method to solve the warehouse size and average total cost of the economic lot scheduling problem under the common cycle approach.

Our approach applied the extra idle time for changing the production rate of one product in order to reduce the holding cost and warehouse rental cost. This study uses the flexible rate method to pick a particular product with the highest holding cost and change its production rate. In this study, as applying the flexible rate method to adjust the production rate of a product, it does not only affect the holding costs, but also impacts the warehouse rent sizes.

This study will investigate how to apply the ant colony system to find a the optimal production sequence and solution under the common cycle approach. The numerical experiment shows our flexible rate method can obtain better solutions than the fixed rate method. Our study can help decision makers to schedule all products sequence, compute the related costs and rent the exact outsourcing warehouse space.

Keyword: economic lot, common cycle, flexible rate, warehouse, ant colony system.



# 目錄

第一章	緒論.....	1
1.1	研究背景與動機.....	1
1.2	研究目的.....	4
1.3	研究方法與步驟.....	4
1.4	論文架構.....	5
第二章	文獻探討.....	7
2.1	經濟批量排程問題.....	7
2.1.1	經濟批量排程問題常見的求解方法.....	8
2.1.2	經濟批量排程問題延伸研究.....	12
2.2	倉儲相關文獻.....	13
2.3	變更速率生產方法的相關文獻.....	14
2.4	解決順序性問題常見的方法.....	16
2.5	小結.....	21
第三章	ELSP 數學模式建立與求解.....	22
3.1	ELSP 的符號定義與基本假設.....	22
3.1.1	符號定義.....	22
3.1.2	基本假設.....	22
3.1.3	在共同週期法下的倉儲空間推導.....	23
3.2	彈性速率變更下 ELSP 模式.....	26
3.2.1	基本假設與數學符號.....	26
3.2.2	本研究數學符號定義.....	26
3.2.3	在共同週期法下運用彈性速率之公式建立.....	27

3.2.4	成本項分析.....	28
3.2.5	彈性速率下倉儲的租賃成本.....	29
3.2.6	數學模式建構.....	41
3.3	以螞蟻演算法求解生產順序.....	42
3.3.1	螞蟻系統介紹.....	42
3.3.2	蟻群系統介紹.....	45
3.3.3	ACS 求解 ELSP 問題.....	47
3.4	小結.....	51
第四章	數值範例與數據實驗.....	53
4.1	數值範例.....	53
4.2	隨機實驗.....	67
4.3	小結.....	70
第五章	結論與未來研究方向.....	71
5.1	結論.....	71
5.2	未來研究方向.....	71
參考文獻	.....	73

## 表目錄

表 2-1 BP 法下四個產品的生產週期.....	10
表 2-2 在 EBP 法下四個產品的生產週期.....	11
表 2-3 在 EBP 法下四個產品的生產週期.....	11
表 2-4 螞蟻最佳化演算法之應用.....	20
表 4-1 第一組範例的 5 個產品參數資料.....	53
表 4-2 以共同週期法運用固定速率法求解第一組的結果.....	54
表 4-3 以共同週期法運用彈性速率法求解第一組的結果.....	54
表 4-4 彈性速率法與固定速率法求解第一組結果比較.....	54
表 4-5 第二組範例的 5 個產品參數資料.....	55
表 4-6 以共同週期法運用固定速率法求解第二組的結果.....	55
表 4-7 以共同週期法運用彈性速率法求解第二組的結果.....	56
表 4-8 彈性速率法與固定速率法求解第二組結果比較.....	56
表 4-9 第三組範例的 5 個產品參數資料.....	56
表 4-10 以共同週期法運用固定速率法求解第三組的結果.....	57
表 4-11 以共同週期法運用彈性速率法求解第三組的結果.....	57
表 4-12 彈性速率法與固定速率法求解第三組結果比較.....	57
表 4-13 第四組範例的 5 個產品參數資料.....	58
表 4-14 以共同週期法運用固定速率法求解第四組的結果.....	58
表 4-15 以共同週期法運用彈性速率法求解第四組的結果.....	59
表 4-16 彈性速率法與固定速率法求解第四組結果比較.....	59
表 4-17 第五組範例的 5 個產品參數資料.....	59
表 4-18 以共同週期法運用固定速率法求解第五組的結果.....	60
表 4-19 以共同週期法運用彈性速率法求解第五組的結果.....	60

表 4-20 彈性速率法與固定速率法求解第五組結果比較.....	60
表 4-21 第六組範例的 5 個產品參數資料.....	61
表 4-22 以共同週期法運用固定速率法求解第六組的結果.....	61
表 4-23 以共同週期法運用彈性速率法求解第六組的結果.....	62
表 4-24 彈性速率法與固定速率法求解第六組結果比較.....	62
表 4-25 第七組範例的 5 個產品參數資料.....	62
表 4-26 以共同週期法運用固定速率法求解第七組的結果.....	63
表 4-27 以共同週期法運用彈性速率法求解第七組的結果.....	63
表 4-28 彈性速率法與固定速率法求解第七組結果比較.....	63
表 4-29 第八組範例的 5 個產品參數資料.....	64
表 4-30 以共同週期法運用固定速率法求解第八組的結果.....	64
表 4-31 以共同週期法運用彈性速率法求解第八組的結果.....	65
表 4-32 彈性速率法與固定速率法求解第八組結果比較.....	65
表 4-33 第一組範例與第二組範例比較表.....	65
表 4-34 第三組範例與第一組範例比較表.....	66
表 4-35 第五組範例與第一組範例比較表.....	66
表 4-36 產品參數資料範圍.....	68
表 4-37 在共同週期法下彈性速率法求解 ELSP 的表現.....	69

## 圖目錄

圖 1-1 經濟批量排程問題的運作.....	1
圖 1-2 在週期 $T$ 內產品 $i$ 以全速與固定速率生產的時間、存貨與倉儲空間關係 .....	3
圖 1-3 在週期 $T$ 內產品 $i$ 以全速與彈性速率生產的時間、存貨與倉儲空間關係 .....	3
圖 1-4 研究流程圖.....	6
圖 2-1 以週期 $T$ 生產 $n$ 個產品.....	9
圖 2-2 在週期 $T$ 內以固定速率與全速生產之產品 $i$ 的時間與存貨關係.....	15
圖 2-3 在週期 $T$ 內以彈性速率與全速生產之產品 $i$ 的時間與存貨關係.....	15
圖 3-1 多產品倉儲空間可彼此佔用模型.....	24
圖 3-2 在週期 $T$ 內以彈性速率生產產品 $i$ 的時間與存貨關係.....	26
圖 3-3 在週期 $T$ 內以彈性速率生產產品 $i$ 的時間與存貨關係.....	28
圖 3-4 Seq1 順序下兩個產品中第一產品降速圖.....	29
圖 3-5 Seq1 順序下兩個產品中第二產品降速圖.....	30
圖 3-6 Seq2 順序下兩個產品中第一產品降速圖.....	31
圖 3-7 Seq2 順序下兩個產品中第二產品降速圖.....	32
圖 3-8 Seq2 順序下三個產品中第一產品降速圖.....	33
圖 3-9 Seq2 順序下三個產品中第二產品降速圖.....	34
圖 3-10 Seq2 順序下三個產品中第三產品降速圖.....	35
圖 3-11 Seq2 順序下四個產品中第一產品降速圖.....	36
圖 3-12 Seq2 順序下四個產品中第二產品降速圖.....	37
圖 3-13 Seq2 順序下四個產品中第三產品降速圖.....	38
圖 3-14 Seq2 順序下四個產品中第四產品降速圖.....	39
圖 3-15 在週期 $T$ 內以彈性速率生產產品 $i$ 的生產速率.....	41

圖 3-16 螞蟻演算法路徑示意圖.....43

圖 3-17 ACS 流程圖.....47

圖 3-18 螞蟻選擇的路徑.....48

圖 3-19 螞蟻走過的路徑距離.....51



## 符號對照表

1.  $p_i$  : 每單位時間產品  $i$  的生產率。
2.  $d_i$  : 每單位時間產品  $i$  的需求率。
3.  $W$  : 總倉儲空間。
4.  $w$  : 內部倉儲空間。
5.  $\alpha$  : 每單位存貨之倉儲單位時間租金成本。
6.  $S_i$  : 產品  $i$  的整備時間。
7.  $A_i$  : 產品  $i$  的整備成本。
8.  $h_i$  : 產品  $i$  的存貨持有成本。
9.  $T$  : 所有產品在共同週期法下的生產週期長度。
10.  $\rho_i = (d_i / p_i)$  : 產品  $i$  的產能利用率。
11.  $k$  : 選定降速之產品代號。
12.  $p_k$  : 變更生產速率之產品  $k$  的生產率。
13.  $d_k$  : 變更生產速率之產品  $k$  的需求率。
14.  $I$  : 一生產週期內的產能閒置時間(idle time)。
15.  $t_k$  : 一生產週期內變更生產速率之產品  $k$  以  $d_i$  速率的生產時間。
16.  $i$  : 第  $i$  個生產的產品。

# 第一章 緒論

經濟批量排程問題(Economic Lot Scheduling Problem, ELSP)最早在 1958 年由 Rogers 提出，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。

本章節將對本研究做概括性的描述，分為以下四個部分：研究背景與動機、研究目的、研究方法與步驟和論文架構與流程。

## 1.1 研究背景與動機

ELSP 是一個已經被確認具有價值的問題，可以支援決策者做出適當決策依據的數學模式。由於單一機器於同一時間只能用於生產單一產品；當面臨生產多種產品時，需要解決產品之生產排程與每一產品之批量大小的問題，即被稱之為經濟批量排程問題。圖 1-1 說明經濟批量排程問題的運作狀況，目的在解決單一機台生產多種產品；週期  $T$  代表一個生產週期，在週期  $T$  內需生產  $n$  種產品，並以週期  $T$  為單位重複生產。ELSP 假設生產時程為無限長，亦即重複生產沒有終止的時候。過去幾十年來經濟批量排程問題，受到許多學者的矚目，學者們在不同的決策方案下著手研究求解 ELSP。ELSP 能應用在金屬成型、塑膠生產、汽車、油漆、飲料、動物食品、紡織品及地毯產業等(Boctor, 1987; Moon *et al.*, 2002)。經濟批量排程問題的關鍵在於生產排程和存貨系統的效率。

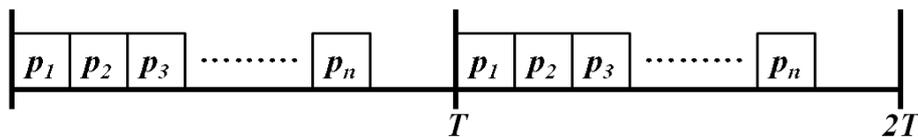


圖 1-1 經濟批量排程問題的運作

傳統 ELSP 模式不會因為產品的生產順序而影響整置時間，週期  $T$  內生產的產品不會因為順序互調而影響總成本，但仍有許多學者進行 ELSP 的延伸研究，

在這些研究中生產順序對總成本有重大影響。Dobson(1992)提出 ELSP 延伸性問題，探討 ELSP 的整備時間與生產順序相依關係(sequence-dependent setup)，透過轉換與鬆弛的技術來求解。而目前有許多 ELSP 延伸性問題像貨架上時效問題；當產品具有保存期限限制，若產品生產完成後到銷售之時間點的這段時間，若超過保存期將會損壞造成損失；還有回流狀況的 ELSP、重製狀況的 ELSP 等延伸性問題，這些問題中發現改變生產順序能得到較佳的總成本。

近年來，在 ELSP 延伸的研究中，多位學者逐漸關注到生產批量與倉儲的問題。在過去求解經濟批量排程問題的過程中，許多研究只對整備成本與存貨成本進行鑽研，大多對於倉儲空間的部分未加以重視，通常以無限倉儲或是有限倉儲來定義，然後計算出經濟批量排程問題的總成本。然而實際情形下，產品生產之後必然會進入儲藏的過程；一企業的倉儲可以分為內部倉儲和外部倉儲。內部倉儲是企業自己的倉儲，不需要支付租賃成本。外部倉儲是指其它企業的倉儲空間；當內部倉儲空間無法再存放產品時，需要租借外部倉儲來儲存，因此就會有倉儲租賃成本。因此總成本不可能只去計算存貨持有成本和整備成本，期間的倉儲的租賃成本也該納入考量，才能符合現實狀況。

傳統上，經濟批量排程問題皆假設機台以最大的生產速率來進行生產，有學者研究發現在機器有閒置時間下，有效的降低生產速率進行生產，可減少存貨持有成本，平均總成本因而下降。降低生產速率的生產方式可以減少存貨的累積，存貨持有成本因此降低，進而得到較低的總成本。降速生產速率的研究文獻中，常見的方法有：固定速率法及彈性速率法。固定速率法是以低於最大速率(全速生產)的方式來生產，當決定一產品較低的生產速率後，此生產速率在製造過程中便不再改變。圖 1-2 為週期  $T$  內固定速率與全速生產產品  $i$  的時間與存貨關係，虛線為低於全速生產的生產速率；灰色區塊則為減少的存貨成本。

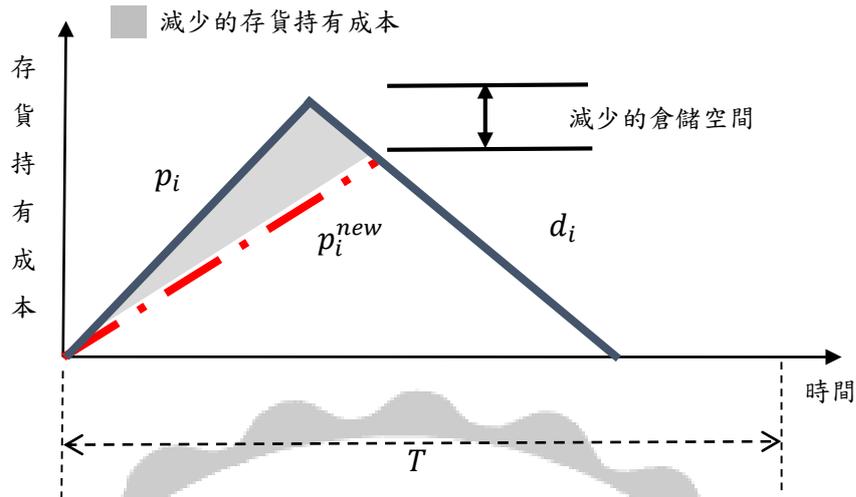


圖 1-2 在週期  $T$  內產品  $i$  以全速與固定速率生產的時間、存貨與倉儲空間關係

彈性速率法為以生產速率等於需求率進行生產一段時間，而後再以最大速率來進行生產。圖 1-3 在週期  $T$  內以彈性速率生產產品  $i$  的時間與存貨關係，虛線為彈性速率；灰色區塊則為所減少的存貨成本。

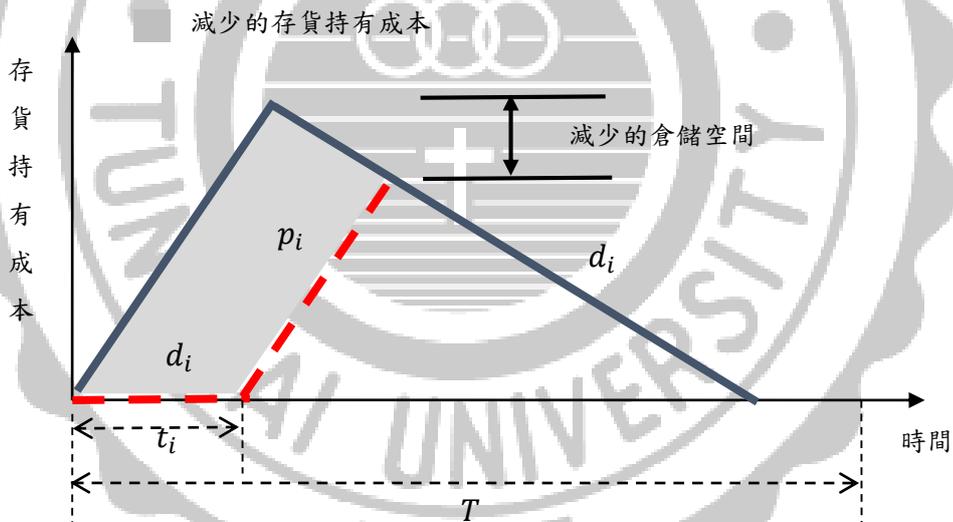


圖 1-3 在週期  $T$  內產品  $i$  以全速與彈性速率生產的時間、存貨與倉儲空間關係

本研究考慮多個產品的經濟批量排程問題，假設機器在共同週期法下，使用閒置時間進行彈性速率法來變更生產速率，並求解所需的倉儲空間的大小。當內部倉儲空間無法容納產品時，需要外部倉儲來存放所生產的產品，因此將有限倉儲空間納入考量。傳統 ELSP 僅考量整置成本和存貨持有成本，本研究進一步將外部倉儲租賃成本納入總成本的考量。在有限倉儲空間下進行生產，並建立數學模式，求解 ELSP 倉儲空間與總成本(包括租賃成本)，目標為求得最佳總成本。

## 1.2 研究目的

本論文應用變更生產速率的 ELSP 以共同週期法(Common Cycle, CC)求解經濟批量排程問題的倉儲空間與總成本，並利用蟻群系統(Ant Colony System, ACS)演算法搜尋較佳化的生產順序以求得較佳的可行解。

本研究主要是提出 ELSP 的成本項目加上外部倉儲的租賃成本的情況下，應用彈性速率法來挑選一產品以變更生產速率，Buzacott and Ozkarahan(1983)建議存貨持有成本最大( $Max\{d_i h_i\}$ )的產品就是應該做降速的產品。一製造機台生產  $n$  種產品時，一生產週期中若有多餘的閒置時間時，適當的分配於某項產品，以變更生產速率來減少存貨持有成本以及外部倉儲的租賃成本。

林禎樺(2014)提出不同的生產順序能找到不同的平均總成本；因此，求得最佳生產順序便是求解本論文研究問題的主要關鍵。本研究發展蟻群系統演算法求解本論文研究問題，求取最佳生產順序的排列組合，達到每單位時間平均總成本最低的目標。所有的產品皆以在週期  $T$  內會依此生產順序進行生產，然後以生產週期  $T$  重複生產。而在決定生產順序時，會一併考量(降速)生產完之後產品所需要的倉儲空間之大小。當內部空間無法滿足產品的需求時，會需要外部的倉儲空間來存放產品，因此會產生倉儲租賃成本，本研究建立了上述問題的數學模式，來求解倉儲空間與總成本(含租賃成本)，目標為找到經濟批量排程問題的最佳解。

## 1.3 研究方法與步驟

傳統的 ELSP 僅考量生產過程的整備成本和持有成本，本研究將外部倉庫的租賃成本加入經濟批量排程問題的目標成本函數上，應用變更速率來求經濟批量排程問題。本研究所設計的數學模式，將一生產週期中若有多餘的閒置時間，將其適度的分配於某項產品，以變更生產速率來減少存貨持有成本以及倉儲的租賃成本。因此本研究是以共同週期法變更生產速率後進行探討。最後，蟻群系統演算法在搜尋

的過程中，會對於螞蟻走過的路徑進行全域更新與區域更新，讓螞蟻可以去探索未知的路徑，得到比其它演算法佳的解，因此本研究使用 ACS 來找尋不同的生產順序，並探索是否擁有較佳的可行解。

本研究主要可分為下列幾個步驟及方法：

1. 搜集、整理及探討經濟批量排程問題的相關文獻及求解的方法。
2. 搜集、整理及探討經濟批量排程問題延伸研究的相關文獻。
3. 搜集、整理及探討倉儲的相關文獻。
4. 搜集變更生產速率生產方法與搜尋生產順序的相關文獻。
5. 搜集各種搜尋演算法，來找出適合用於本研究找尋排列順序的最佳解。
6. 針對使用的求解模式進行修改，使其模式符合本研究需求。
7. 運用變更生產速率中的彈性速率法，減少存貨持有成本及倉庫租賃成本，找出在不同生產順序下，總成本最小者。
8. 利用本研究提供的八組範例，來說明本研究所建構的方法。
9. 本研究提供各產品參數的值域範圍以建立隨機實驗所需的案例，來執行本研究所提出的求解方法。
10. 針對數據實驗的結果，提出結論與建議。

## 1.4 論文架構

本研究論文的內容共分為五章，如圖 1-4 所示，分別描述如下。

第一章緒論：本研究主要是在共同週期法下，探討 ELSP 的成本項目加上倉儲的租賃成本的情況下，並加入變更速率的概念，透過降低生產速率使存貨持有成本與倉庫的租賃成本減少，並利用蟻群系統演算法求 ELSP 問題。

第二章文獻探討：為了讓本研究更為完善，首先介紹經濟批量排程問題與常見的求解方法以及以往經濟批量排程問題的延伸研究，再了解以往經濟批量排程問題加入倉儲的相關研究，接著對變更速率生產方法做介紹，最後對解決順序性方法進行探討。

第三章應用速率變更之 ELSP 數學模式的建立與求解：首先先說明在未降速運用共同週期法來解決經濟批量排程的倉儲問題，再詳述運用共同週期法和彈性速率法下成本項分析，倉儲空間和倉儲租賃成本的數學公式推導與建立，最後說明蟻群系統演算法概念並說明本研究問題模式下的求解運作過程。

第四章數值範例與數據實驗：經由上述所建構出的模式，利用四組數值範例說明實際操作情形，探討變更速率對於倉儲空間與總成本的影響，並以大量隨機實驗測試演算法的成本改善幅度和計算時間效率。

第五章結論與未來發展：總結本研究的實驗數據結果，並藉此提出建議與未來的研究方向。

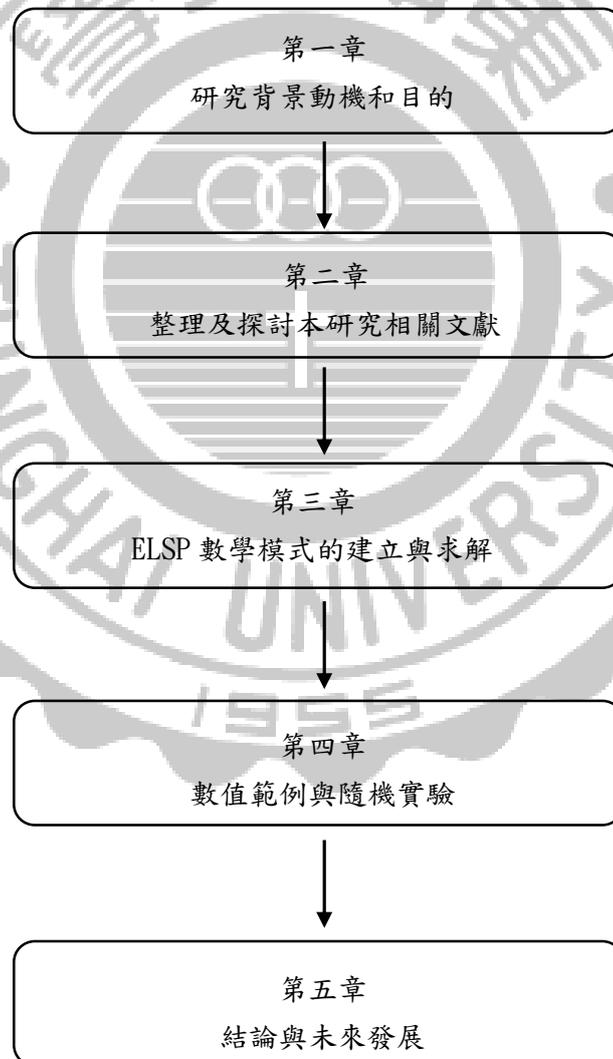


圖 1-4 研究流程圖

## 第二章 文獻探討

本章節分為六個部分進行逐步探討，首先第一節部分探討經濟批量排程問題有哪些常見的求解方法以及延伸性研究的相關文獻，第二部分為經濟批量排程問題結合倉儲相關的文獻，第三部分介紹經濟批量排程問題變更生產速率的相關研究文獻，第四部分介紹關於演算法求解順序性問題的相關文獻，第五部分為第二章做個小結。

### 2.1 經濟批量排程問題

經濟批量排程問題最早是由 Rogers(1958)所提出，探討產品的批量大小，生產次數及週期時間。假設在無限期規劃下，在單一機器多種產品時，調整產品週期性的生產排程，減少生產產品所需的整備空間及成本，在排程為合理可行下，使得平均總成本最小。整備成本(setup costs)及存貨持有成本(inventory holding costs)是 ELSP 模式中常被考慮的成本項目。許多學者發展了各種不同方法來解經濟批量排程問題，本章節將介紹常用的方法來求解經濟批量排程問題。

經濟批量排程問題為 Rogers(1958)所提出，Bomberger(1966), Davis(1990)與 Khouja *et al.*(1998)提出求解的原始 ELSP 問題之以下幾項假設：

1. 此單一機台生產力，可以滿足所有產品的需求量。
2. 整備時間與整備成本只與生產出來的產品有關，會隨著產品的不同而有所不同。
3. 所有產品皆由此機台完成生產。
4. 在任一時間點上，此機台只能生產單一種產品。
5. 在週期生產排程中，每個產品在各週期生產出來的產品批量與循環時間的長度(cycle time)都是相同的。
6. 產品需求是持續不斷的。

7. 同一時間內，所有產品的需求率、生產率、整備時間、整備成本和存貨持有成本都是已知且不隨時間改變。

### 2.1.1 經濟批量排程問題常見的求解方法

Hsu(1983)確認ELSP為非多項時間可求解(NP-hard)之問題，許多學者提出各種不同的方法來求解此問題。Elmaghraby(1978)將求解ELSP問題的解法略分為分析式方法(analytical approach)及啟發式方法(heuristic approach)。

1. 啟發式解法：具備快速求解問題的能力，一般可以得到較好的近似解，不能保證其解為最佳解。
2. 分析式方法：通常可用於原始問題在某些限制或假設下，得到最佳解或近似最佳解，較常見的分析式解法有獨立解法(Independent Solution, IS)、共同週期法(Common Cycle, CC)、基本週期法(Basic Period, BP)及延伸基本週期法(Extend Basic Period, EBP)等四種。

獨立解法：

在獨立解法(Independent Solution, IS)中每個產品只考慮本身的最佳解，再加總各產品的平均成本，其中忽略了單一機台在同一時間只能生產一種產品的假設，並沒有考慮到可用產能是有限的。Jones(1989)所提出的獨立解法不能保證其解的排程是合理可行，但獨立解法的解可以視為 ELSP 模式中可行解的下界。

獨立解法的數學模式如下：

產品  $i$  的平均單位總成本：

$$TC_i = \frac{A_i}{T_i} + \frac{(1-\rho_i)h_i d_i T_i}{2} \quad (2.1)$$

產品  $i$  的最佳獨立循環時間：

$$T_i = \sqrt{\frac{2A_i}{h_i d_i (1-\rho_i)}} \quad (2.2)$$

產品  $i$  的最小平均單位總成本：

$$TC_i^* = \sqrt{2A_i h_i d_i (1-\rho_i)} \quad (2.3)$$

共同週期法：

共同週期法(Common Cycle)是 Hanssmann(1962)所提出，是一個簡單且保證其解為可行解的排程法則。其主要的假設為每個產品在每個週期都要生產，而生產週期內生產順序可以任意排列，意即所有產品的排程都包含在一個共同的循環時間( $T$ )內，每個產品都會在週期  $T$  內生產一次，每個產品都以週期  $T$  時間重複循環，直到產品都生產完畢。

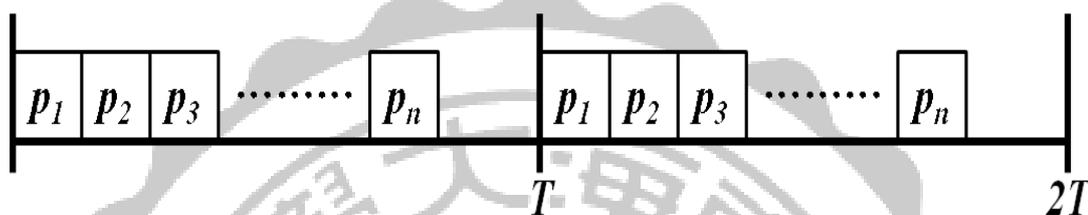


圖 2-1 以週期  $T$  生產  $n$  個產品

共同週期法的數學模式如下：

$$\text{Min } \sum_{i=1}^n \frac{A_i}{T} + \frac{(1-\rho_i)h_i d_i T}{2} \quad (2.4)$$

$$\text{Subject to } \sum_{i=1}^n (s_i + \rho_i T) \leq T \quad (2.5)$$

$$T_{CC} = \text{Max} \left\{ \sqrt{\frac{2 \sum_{i=1}^n A_i}{\sum_{i=1}^n [h_i d_i (1-\rho_i)]}}, \frac{\sum_{i=1}^n s_i}{1 - \sum_{i=1}^n \rho_i} \right\} \quad (2.6)$$

其中方程式(2.4)假設所有產品皆有共同的週期  $T$ ，因此可以得到所有產品的單位時間期望總成本。方程式(2.5)表示所有產品的總生產時間必須要小於等於循環時間。方程式(2.6)為循環時間，其值為取式中之最大值者。

基本週期法：

基本週期法(Basic Period Approach, BP)為 Bomberger(1966)所提出，其主要是使用動態規劃來求解 ELSP。其模式如下：

$$\text{Min } \sum_{i=1}^n \frac{A_i}{T_i} + \frac{h_i d_i (1-\rho_i) T_i}{2} \quad (2.7)$$

$$\text{Subject to } \sum_{i=1}^n (s_i + \rho_i B k_i) \leq B \quad (2.8)$$

$$T_i = k_i B \quad (2.9)$$

$$k_i : \text{integer}, k_i \in \{1, 2, \dots\} \quad (2.10)$$

BP 法將每個產品的生產週期長度設為基本週期  $B$  的整數倍( $k_i$ )，即每  $k_i$  週期生產一次，且必須滿足  $k_i$  個週期需求。BP 法假設所有產品於第一週期皆要生產，往後的週期是否生產，就看每個產品的週期乘數  $k_i$  而定。若每個產品的  $k_i$  皆為 1，其解為共同週期法。BP 法產生的排程除了週期一的機台負荷較重外，其他週期的機台閒置時間較多，因此容易造成設備產能的浪費及不容易找尋到最佳解的機會。

表 2-1 即為 BP 法下四個產品生產週期範例。四個產品的生產週期分別為基本週期時間  $k_i$  的 1、2、3、4 倍。例如產品 2 的  $k_i$  值等於 2，表示每隔 2 個週期產品 2 就要生產一次，因此產品 2 在週期 1、3、5、7 時都需要生產。

表 2-1 BP 法下四個產品的生產週期

產品 \ 週期	1	2	3	4	5	6	7
1( $k_i=1$ )	✓	✓	✓	✓	✓	✓	✓
2( $k_i=2$ )	✓		✓		✓		✓
3( $k_i=3$ )	✓			✓			✓
4( $k_i=4$ )	✓				✓		

註：✓表示在該週期要生產

延伸基本週期法：

延伸基本週期法(Extended Basic Period Approach, EBP)的架構是 Hasessler and Hogue(1976)所提出，利用一整數規劃模式進行求解。而後 Elamghraby(1978)及 Davis(1990)相繼提出較佳的整數規劃模式。

EBP 法改善了 BP 法中產品在第一個週期皆要生產的限制，讓每一個產品依其週期乘數  $k_i$  自由調整其排程，避免機器在除了第一週期外的其他週期可能有更多閒置時間的缺點。表 2-2 在 EBP 法下，四個產品生產週期範例，總共有四個產品生產週期分別為基本週期時間  $k_i$  的 1、2、3、4 倍。EBP 會遇到的困難為如何尋找產品的  $k_i$  值，與當找到產品的  $k_i$  值時，該如何決定產品的起始生產週期。

例如，產品 2 的  $k_i$  值等於 2，起始生產週期可以落在第 1 個週期(如表 2-2)，之後一樣每隔兩個週期生產，因此產品 2 在週期 1、3、5 和 7 時都需要生產；或是起始生產週期可以落在第 2 個週期(如表 2-3)，之後一樣每隔兩個週期生產，因此產品 2 在週期 2、4、6 和 8 時都需要生產。產品 3 的  $k_i$  值等於 3，起始生產週期可以落在第 1 週期，因此產品 3 在週期 1、4、7 都需要生產，如圖 2-3 所示。

表 2-2 在 EBP 法下四個產品的生產週期

產品 \ 週期	1	2	3	4	5	6	7
1( $k_i=1$ )	✓	✓	✓	✓	✓	✓	✓
2( $k_i=2$ )	✓		✓		✓		✓
3( $k_i=3$ )		✓			✓		
4( $k_i=4$ )				✓			

註：✓表示在該週期要生產

表 2-3 在 EBP 法下四個產品的生產週期

產品 \ 週期	1	2	3	4	5	6	7
1( $k_i=1$ )	✓	✓	✓	✓	✓	✓	✓
2( $k_i=2$ )		✓		✓		✓	
3( $k_i=3$ )	✓			✓			✓
4( $k_i=4$ )			✓				✓

註：✓表示在該週期要生產

EBP 因產品的起始週期並無限制，因此一組  $k_i$  可以產生許多可能的排程，EBP 需要進行確認排程為可行或不可行。若一組  $k_i$  產生的排程為不可行，則能調整產品的起始週期再進行一次確認是否可行，由於需再重新確認排程的解是否可行，因此與 BP 相較之下 EBP 顯得困難許多。

啟發式解法：

啟發式方法多運用反覆式(iterative)搜尋程序，以求問題的最佳解或近似最佳解。其解法一般是從特定的生產週期長度開始，先嘗試求得此週期下每種產品最佳  $k_i$ 。再以此最佳  $k_i$ ，搜尋下一個生產週期長度，如此反覆進行最佳化與搜尋，直到無法找到更好的解。Park and Yun(1987)、Boctor(1987)及 Geng and Vickson(1988)等人提出各自的啟發式解法皆屬於貪婪式搜尋法，使用了延伸基本週期法，說明如何來安排每個產品的生產週期，進而判斷該組解是否為可行解。但貪婪式搜尋法往往只能得到局部最佳解，而無法保證其解的品質。

### 2.1.2 經濟批量排程問題延伸研究

傳統ELSP模式是假設各產品間的整備時間與生產順序的關係是互為獨立的，但仍有某些學者進行ELSP的延伸研究，在這些研究中整備時間與生產順序是相依的。Dobson(1992)提出ELSP的延伸模式，探討ELSP的整備時間與生產順序相依關係(sequence-dependent setups)，透過轉換與鬆弛的技術來求解。

ELSP問題中，多數文獻皆假設有足夠的產能來滿足需求，並不考慮缺貨的狀況，Gallego *et al.*(1996)將此因素納入考量，在有限的規劃之下，考量了產能不足的情況。Khouja(1997)則探討在一個高效率的製造系統中，如何增加生產率，來降低平均單位成本，避免大量的訂單數量及過多的持有成本。

Wagner and Davis(2002)提出一啟發式搜尋演算法來解決相同的問題，不同於過去只有單一最佳解方式，他們所提出的演算法可以有一組解供決策者選擇，以應付多變且動態的現實生產環境，此方法實驗結果優於Dobson(1992)。

但有學者發現生產順序和總成本則是有相互影響的關係，也開始有許多ELSP延伸研究。根據Haessler(1979)所提出的基本週期法求解相同問題的程序建立模式，Soman *et al.*(2004)提出產品於貨架上時效問題(shelf life)的ELSP模式，發展類似於分支界線法(branch and bound)的啟發式演算法求解，其數值實驗結果

顯示其結果優於共同週期法。

Tang and Teunter(2006)提出考慮產品有回收狀況的 ELSP，模式中假設所有作業都在單一生產線上，藉由外部回收的瑕疵品修復後加以滿足生產需求。Teunter *et al.*(2008)根據 Tang and Teunter(2006)的模式發展出多產品且有退貨狀況的 ELSP，在模式中假設生產線有二條，分別負責產品製造與外部回收瑕疵品重新製作。

Öner and Bilgiç(2008)提出考慮有副產品產生的 ELSP，假設各個產品有各自市場且需求無可取代，副產品的產出有一定的比例會成為次級品。他們提出允許計畫性的缺貨及不允許缺貨發生兩種狀況來求最佳解。

Chang and Yao(2011)提出有重製(reworks)狀況的 ELSP(Economic Lot Scheduling Problem with Reworks, ELSPR)，兩位學者指出有重製與有回流(Return)狀況的 ELSP 是相似的問題，目標皆最小化理想成本和額外成本的總和，並提出一簡單排程法及二分搜尋法以求得最佳週期時間及最佳產品批量生產順序。

## 2.2 倉儲相關文獻

過去求解 ELSP 問題都只對整置與存貨成本進行深入研究，並未重視倉儲空間，但實際情形下，總成本不只包含整備時間和存貨持有成本，產品生產後會進行儲藏，儲藏期間的倉儲租賃成本也該納入考量，符合現實狀況。

Rogers(1958)在討論 ELSP 問題時，發現一群群組的產品必須同時被取得，它的成本項包含了整備成本、存貨持有成本以及倉儲租賃成本，但沒有考慮倉儲空間大小的問題。

Jarupongsa *et al.*(2004)假設一個簡單的供應鏈來考量倉庫大小與其產品的交貨時間窗的動態批量大小，對於單一產品的交貨時間窗以及倉庫容量限制兩個動態批量，過早交貨則會帶有處罰。

張孝裕(2006)探討在倉儲有限的情況下可變動生產速率來求解經濟批量排

程問題，ELSP 可藉由降低生產速率來減少存貨成本以及倉庫的租賃成本，得到較低的總成本。

鄭舜維(2009)在倉儲受限下，產品的損毀率對於其倉儲儲存量的決策問題進行了研究，並且把外租倉庫的存貨損毀機率以及其租金成本納入考量，來進行決策推導以求其最佳解。

Transchel and Minner(2009)探討在倉儲有限的情況下，多產品共用一個倉庫時，公司允許不斷調整銷售價格(動態定價)與倉儲的補給調度問題，分析影響銷售價格與補貨比例的協調決策，目的是透過最佳定價策略與最優批量大小及排放順序來達成利潤最大化。

Minner(2009) 則對倉儲有限的多產品動態需求進行了三種簡單啟發式分析，分別是(1)根據成本為基礎來延長補給批量的優先權規則、(2)假設每個產品生產第一步為獨立生產，然後藉由平滑機制處理超過負荷的機台，(3)單一產品批量無容量限制，對多產品機台負荷受限的問題，根據節約成本為優先權原則藉由增資來改善先後的排程時間。

Liang *et al.*(2011)則是建立兩個有限倉庫來探討會在有條件允許延遲付款的情況下，不斷的需求會隨時間損毀的商品，目的在尋找總庫存成本最小化的最佳補貨政策，將已經得到的最佳解決方案成為有用的理論。

## 2.3 變更速率生產方法的相關文獻

固定速率法(fixed rate approach)及彈性速率法(flexible rate approach)是常被用來變更生產速率的方法。固定速率法是以低於最大速率的方式進行生產。當一產品降低生產速率進行生產後，此產品的生產速率在製造過程中不再改變。圖 2-2 顯示，在週期  $T$  內使用固定速率與全速生產的產品  $i$  時間與存貨關係，虛線為降低產品的生產速率( $p_i^{new}$ )，灰色區塊為減少的存貨持有成本。

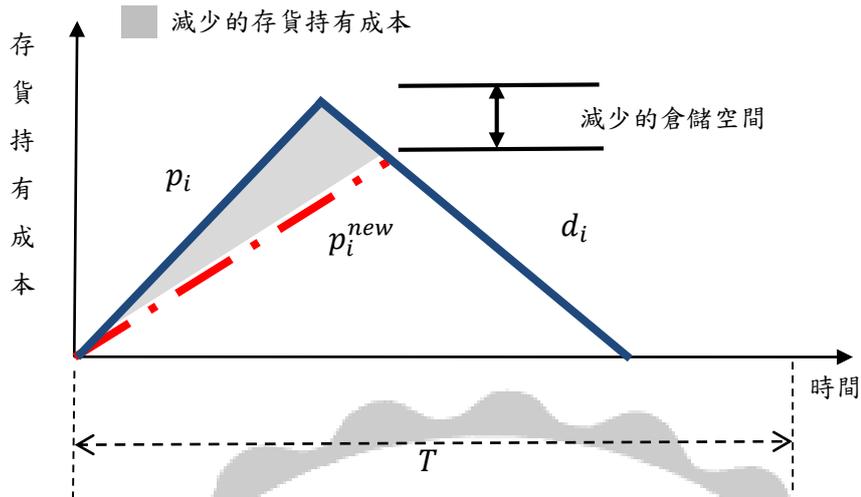


圖 2-2 在週期  $T$  內以固定速率與全速生產之產品  $i$  的時間與存貨關係

彈性速率法為以生產速率( $p_i$ )等於需求率( $d_i$ )進行生產一段時間，再以最大速率來進行生產。如圖 2-3 顯示週期  $T$  內以彈性速率與全速生產的產品  $i$  時間與存貨關係，虛線為彈性速率，灰色區塊則為所減少的存貨成本。

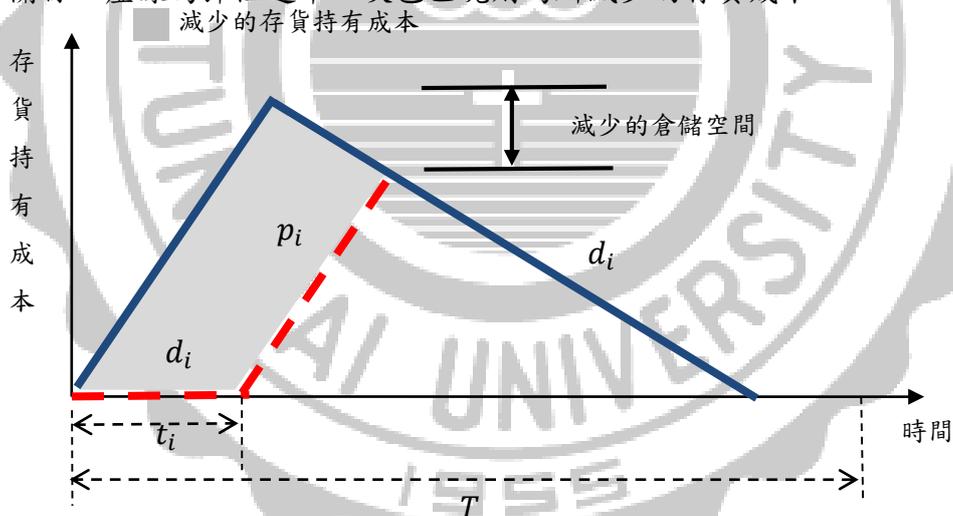


圖 2-3 在週期  $T$  內以彈性速率與全速生產之產品  $i$  的時間與存貨關係

Buzacott and Ozkarahan (1983)提出應用固定低速率法求解ELSP,其中說明只需要降低一種產品的生產速率，其餘產品仍以最大生產速率進行生產。Silver (1990)在共同週期法下求解出最佳生產速率之封閉型式解與最佳共同生產週期，研究中顯示出只須降低一種產品的生產速率。Khouja (1999)根據Silver (1990)的

研究，利用固定低速率法分析不可靠生產系統(unreliable production systems)的經濟批量模式。其研究發現在加入不可靠生產系統的假設之後，最佳期望共同生產週期及生產批量將會優於Silver(1990)所求出之解。

Sheldon (1986)提出彈性速率法，利用機器的閒置時間在生產的過程中改變機器的生產速率以進行生產。其假設各種產品在生產開始時以生產速率等於需求率進行生產減少此段時間的存貨成本，而後再以最大速率來進行生產。Moon *et al.*(1991)提出使用彈性速率法在共同週期法下求解ELSP，發現在ELSP中使用彈性速率法求解可比傳統法(全速生產)及固定速率法(降速生產)進行生產所得到的最佳期望平均總成本解更低。Gallego(1993)及Eynan(2003)在共同週期法下比較了傳統法、固定速率法以及彈性速率法之經濟批量排程問題。

Moon and Christy (1998)指出在固定速率法下，可降低的生產速率有其下限，即為產品的需求率，而上限則為原有的最大生產速率，並指出生產速率增加時，平均總成本也會隨之增加。

蔣其軒(2007)提出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，假設機器在生產開始前必定處於操控狀態下，分別考慮在不進行檢驗下、連續檢驗政策下與定期檢驗下的不完美生產系統之單位時間期望總成本模式。

陳世杰等(2012)提出變更生產速率的經濟批量檢驗與排程問題模式(ELISP-FPR)，探討利用固定速率法將會對生產週期時間、生產排程與檢驗排程的影響。

張育仁等(2014)提出以簡單固定速率法求解在基本週期法下經濟批量排程問題，探討如何計算一產品在不同生產速率下計算單位時間平均成本的方法，並發展一個有效率的搜尋方法搭配固定速率法，以找出最佳的 $\{k_i\}$ 。

## 2.4 解決順序性問題常見的方法

本研究探討產品的最佳生產順序問題，而目前有許多解決順序性問題的方法，

以下挑選出幾個常用來解決順序性問題的啟發式演算法，加以介紹。

### 基因演算法：

在基本週期法下，產品週期乘數的組合對解有重要的影響，為了能搜尋出最佳的產品週期乘數組合，有些研究嘗試以搜尋演算法來求解經濟批量排程問題，如基因演算法(Genetic Algorithm, GA)，能找到最佳的週期乘數來降低成本。基因演算法是由Holland(1975)所提出，其模仿自然界物競天擇、適者生存不適者淘汰的方法，訂出染色體、適應能力、基因交換及基因突變等規則性之演算過程來求最佳解，其解為全區域的最佳解。

Khouja *et al.*(1998)使用基因演算法和基本週期法來求解ELSP，並使用 Bomberger (1966)的產品資料作數據實驗，找出產品的週期乘數，得到的成本比獨立解法(IS)高出許多。

Moon *et al.*(2002)使用時間變動批量大小(Time Varying Lot Sizes, TVLS)求解ELSP，並假設每個產品每次生產的批量可以不同，每批量之間的週期可以不同，Moon *et al.*(2002)應用混合式的基因演算法來求解上述問題的順序、批量和週期。

Sarker and Newton(2002)使用基因演算法來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)提出一類似分界法之深度優先的搜尋演算法，來求基本週期法具有保存期限限制的ELSP問題，數值實驗顯示此方法優於共同週期法得到更好的解。

Chang and Yao (2005)以基因演算法求解一般整數策略下之多機ELSP問題，探討單一生產系統中具有多不相同類型機器，生產多種產品的ELSP，實驗數據顯示其研究解法兼具求解的品質和時間的要求。

Moon *et al.* (2006)提出以啟發式演算法與應用混和基因演算法(hybrid genetic algorithm)解決群體技術的經濟批量排程問題，以共同週期法與時間變動

批量大小為基礎建立模式。

林禎樺(2014)提出在共同週期法下應用固定速率法求解ELSP倉儲大小與成本，運用基因演算法找尋出不同的生產順序，是否在某種生產順序下擁有較佳的可行解。

粒子群演算法：

粒子群演算法(Particle Swarm Optimization, PSO)起源於觀察自然界中鳥群的群體覓食最佳化行為，模擬鳥類的群體覓食行為所建構的群體智能模式，透過鳥群覓食時，個體之間的協調和訊息共享來尋找最佳解。粒子群演算法是由Kenned and Eberhart(1995)提出，具有快速收斂、較少的參數設定和適用於動態環境等優點，有許多學者使用粒子群演算法來做研究。

Eberhart and Shi(1998)加入了慣性權重(Inertia Weight)等因素，為了避免收斂過快而陷入區域最佳解，而有更好的全域搜尋能力，並成為了粒子群演算法的標準版本，衍伸出許多應用。Merwe and Engelbrecht(2003)利用粒子群演算法來處理資料分群問題，快速找出最佳化的分群。Lian *et al.* (2006)和賴阮明(2009)整合粒子群演算法應用於求解彈性零工工廠排程問題(Job-Shop Scheduling Problem, JSP)。Ai and Kachitvichyanukul (2009)發表了利用 PSO 求解車輛途程問題的研究。

Lima *et al.*(2010)提出透過 PSO 與 K-means 實作網路異常偵測，結合此兩種演算法建立網路異常偵測模型，並以真實網路流量做實證，顯示該模型能快速的辨識出異常行為，在偵測率和誤判率都有良好的表現。

林信宏(2010)應用 PSO 於多目標排程問題研究，並比較利用 PSO 求解與基因演算法求解的結果，顯示 PSO 於該問題能求得更好的解答。Li *et al.*(2011)也利用 PSO 和 K-means 偵測網路異常行為，並以 KDD CUP 99 資料庫做驗證，結果顯示其偵測機制具有高偵測率及低誤判率。

郭育璋(2014)提出在基本週期下對具有保存期限的 ELSP 問題進行探討，求

解問題時關鍵因素為找出每個產品的週期乘數；郭育璋(2014)應用 PSO 搜尋最佳的產品週期乘數之組合，以求得最低的平均總成本。

### 螞蟻演算法：

螞蟻系統演算法(Ant System, AS)為最近藉由模擬螞蟻行為所發展出求解最佳化問題的搜尋演算法之統稱。螞蟻演算法屬於仿生物演算法的一種，核心在於模仿螞蟻尋找食物的程序，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，並尋找出蟻巢與食物間的最短路徑。

Bullnheimer *et al.* (1999)研究報告指出若用傳統的螞蟻系統(Ant System, AS)來求解較複雜的旅行推銷員問題(Traveling Salesman Problem, TSP)問題，不但耗時，解題的難度也提高。Bullnheimer 等人提出藉由螞蟻演算法適合做平行化的天性，分別提出同步化(synchronous)平行法及非全然同步法(partially asynchronous)平行法的概念，應用在各種的 TSP 問題上，並在平行化後進行速度增益比(speedup)、效率(efficiency)及功效(efficacy)等三項的效能評估。其結果發現，使用非全然同步化平行法的效能勝過同步化平行法。

Maniezzo(1999)採用不同之機率選擇規則，將螞蟻系統演算法應用於二次分配問題(Quadratic assignment problem, QAP)，結果顯示螞蟻系統演算法在求解二次分配問題上具有優良之表現。螞蟻系統演算法並非全收斂至同一個解，而是收斂至一群好的解上，進而找到最佳解。

McMullen(2001)提出利用二維與三維資料並且針對條件的不同來設計不同型式的螞蟻最佳化演算法(Ant Colony Optimization, ACO)演算法，來處理即時生產系統問題，研究結果顯示 ACO 演算法優於其他方法。

Maier *et al.* (2003)首度利用蟻群系統演算法來求解配水管網的問題，並依據費洛蒙的更新方式分成群體更新及僅單一最佳解更新兩種不同的模式，來求解十四根管徑問題和紐約案例。在這兩個案例中，都顯示蟻群系統演算法的效率比基因演算法來的好。

Emanuel *et al.* (2007)提出以螞蟻演算法解決車間作業調度問題。使用一組車間作業調度問題與最先進的技術進行比較，發現螞蟻演算法可以降低評估的數量。

Moghadom *et al.*(2010)應用穩健規劃法(robust optimization)並提出了先以分之切割法求得小規模問題的精確解，再擴大為螞蟻最佳化演算法的求解方式。最後做出固定型與穩定型的精確解與啟發解的比較，得到在大規模問題中，穩健規劃的啟發解可以在可接受的運算時間內得到接近精確的解。

Chang(2014)指出有些作業需要在非平行生產的情況下先生產，然後分發到相應受容量限制的車輛的客戶，沒有中間的庫存與生產和分配等問題。Chang (2014)用螞蟻演算法設計數學模型，目標是找到一個合作生產和分配計畫，使總作業時間與經銷成本能得到最小化。

表 2-4 螞蟻最佳化演算法之應用

問題名稱	作者	年份
Traveling salesman problem	Dorigo, Maniezzo, and Coloni	1991
	Dorigo and Gambardella	1997
	Stützle and Hoos	1999
	Karaboga, D. ,Gorkemli, B.	2011
	Shyi-Ming Chen	2011
Quadratic assignment problem	Gambardella, Coloni, and Dorigo	1994
	Stützle and Hoos	2000
	Maniezzo	1999
	Maniezzo and Coloni	1999
	Cheng-Fa Tsai, Chun-Wei Tsai, Ching-Chang Tseng	2003
	Hamed Qahri Saremi	2007
Scheduling problems	Dorigo, Maniezzo, and Coloni	1994
	Stützle	1998
	Den Besten, Stützle and Dorigo	2000
	Stefka Fidanova	2006
Machine learning	Paepinelli, Lopes, and Freitas	2002
	Casillas, Cordon, and Herrera	2000
	Mohammad Ali Jan Ghasab	2014
Network routing	Di Caro and Dorigo	1998
	White, Pagurek and Oppacher	1998
	Schoonderwoerd et al.	1996
	Jun-Zhong Ji	2009
Vehicle routing problem	Bullnheimer, Hartl, and Strauss	1999
	Gambardella, Taillard, and Agazzi	1999
	Bin Yu	2008
Sequential ordering	Gambardella and Dorigo	2000
Optical networks routing	Leguizamon and Michalewicz	1999

(資料來源：修改自 Dorigo and Stützle, 2002)

在蟻群系統演算法被提出來之後，有許多學者開始應用來求解各個領域的問題，如上表2-4所分類。

以上述螞蟻演算法的相關研究可以發現目前並未有與ELSP的相關研究，而Dorigo and Gambardella(1997)利用旅行銷售員問題資料庫(TSPLIB)，實作螞蟻理論，並與其他較常被應用的演算法如基因演算法、模擬退火法、粒子群演算法等做比較，結果證實螞蟻理論無論是在效能或效率上都有傑出的表現，因此螞蟻演算法被利用來解決許多複雜的最佳化問題，因此本研究使用螞蟻演算法來解決產品的生產順序問題。

## 2.5 小結

從上述關於ELSP的文獻中，以往的學者很少同時探討ELSP和倉儲的問題。學者以往應用共同週期法求解ELSP模式時，都局限於利用固定速率法變更生產速率；以往討論ELSP和倉儲問題時，沒有學者探討在共同週期法下運用彈性速率法變更生產速率。然而，彈性速率法能減少更多的倉儲空間與存或持有成本，能求出比固定速率法更好的解，因此本研究便在共同週期法下運用彈性速率法，對以往的模式做創新的突破。此外不同的生產順序能找到不同的平均總成本；因此求得最佳生產順序便是求解本論文研究問題的主要關鍵。本研究利用蟻群演算法，利用開發與探索的特性，期望能達到優良的求解效果。

## 第三章 ELSP 數學模式建立與求解

本研究在變更生產速率的 ELSP 以共同週期法求解經濟批量排程問題的倉儲空間與總成本。本章節先對於未降速在共同週期法下的倉儲空間推導，接著說明基本假設及數學符號定義，之後對彈性速率在共同週期法下的倉儲空間推導建構出問題的數學模式；最後詳細說明以蟻群系統演算法的運作方式，並以蟻群系統演算法對模式進行求解。

### 3.1 ELSP 的符號定義與基本假設

#### 3.1.1 符號定義

考量倉儲空間與成本的 ELSP 數學符號：

1.  $p_i$ ：每單位時間產品  $i$  的生產率。
2.  $d_i$ ：每單位時間產品  $i$  的需求率。
3.  $W$ ：總倉儲空間。
4.  $w$ ：內部倉儲空間。
5.  $\alpha$ ：每單位存貨之倉儲單位時間租金成本。
6.  $S_i$ ：產品  $i$  的整備時間。
7.  $A_i$ ：產品  $i$  的整備成本。
8.  $h_i$ ：產品  $i$  的存貨持有成本。
9.  $T$ ：所有產品在共同週期法下的生產週期長度。
10.  $\rho_i = (d_i / p_i)$ ：產品  $i$  的產能利用率。

#### 3.1.2 基本假設

考量倉儲空間與成本的基本假設如下所示：

1. 單一機器生產多種產品，任一時間點上，生產機台只生產一種產品。

2. 該製程中所有產品皆由一個生產設備完成生產。
3. 缺貨狀況不被允許。
4. 在任何時間內，所有產品的需求率、生產率、生產整備時間、生產整備成本、及存貨持有成本等，為確定已知常數不隨著時間而改變。
5. 假設每個產品占用的儲位空間是可互換的。
6. 在整個生產週期，倉儲大小是固定不變的。
7. 生產過程中生產速率可以變動。

### 3.1.3 在共同週期法下的倉儲空間推導

ELSP 的基本求解模式為單位時間整備成本加上單位時間持有成本，求得最小化的單位時間總成本。沒有變更生產速率的 ELSP 一般都視為傳統的 ELSP。本研究參考許柏彥(2012)論文中，在未降速的情況下，探討 ELSP 的成本函數加上倉儲的租金大小，預先求得最小的倉儲空間大小，再取得倉儲租金大小，以求得 ELSP 的總成本。許柏彥(2012)探討 ELSP 的倉儲問題時，所設定 ELSP 目標成本函數可以表達成如公式(3.1)所示。而所有產品的生產時間加上整備時間須小於或等於整個週期的長度，可以表達如公式(3.2)所示。最佳的週期時間之長度則如公式(3.3)所示。

$$\text{Min } TC = \sum_{i=1}^n \left[ \frac{A_i}{T} + \frac{Th_i d_i (1 - \rho_i)}{2} \right] + (W - w)\alpha \quad (3.1)$$

$$\text{Subject to } \sum_{i=1}^n (S_i + \rho_i T) \leq T$$

(3.2)

$$T = \sqrt{\frac{2 \sum_{i=1}^n A_i}{\sum_{i=1}^n [h_i d_i (1 - \rho_i)]}}$$

(3.3)

本研究假設每個產品占用的儲位空間是可互換的，因此針對倉儲空間可互換的推導做說明，由於產品生產完之後會存放在倉庫中繼續販售，因此我們能得知當產品生產完畢的時間點為倉儲使用空間的最大值，然而倉儲空間便會隨著產品

販售完畢而降到 0。本研究的假設參考許伯彥(2012)的論文，假設前一個產品生產完之後，會因產品販售而空閒出來的倉儲空間能被下一個生產產品佔用。假設有  $n$  個產品，可能會產生  $n!$  種生產順序，這會使數學模式變得複雜，需花費更多的時間來求解。為了簡化數學模式的推導過程，本研究僅考慮下列兩種排列順序：

$$\text{Seq1} : d_i > p_{(i+1)} - d_{(i+1)}$$

$$\text{Seq2} : d_i < p_{(i+1)} - d_{(i+1)}$$

依據  $d_i > p_{(i+1)} - d_{(i+1)}$  排出生產順序的情況下，前一個產品所佔用的倉儲空間會在下一個產品生產時釋出足以容納新產品庫存的倉儲空間。而  $d_i > p_{(i+1)} - d_{(i+1)}$  此種情況在一企業的倉儲發生的機率較低，若發生此狀況，其最佳倉儲空間大小必然為第一個產品所佔用的倉儲空間大小，公式如(3.4)所示：

$$W = (p_1 - d_1) \left( \frac{d_1 T}{p_1} \right) \quad (3.4)$$

在  $d_i < p_{(i+1)} - d_{(i+1)}$  的情況下，由於前一個產品所佔用的倉儲空間無法在下一個產品生產時釋出足以容納新產品庫存的倉儲空間；所以第二個產品需要額外的倉儲空間，而  $d_i < p_{(i+1)} - d_{(i+1)}$  此種情況在一企業倉儲發生的機率較高。

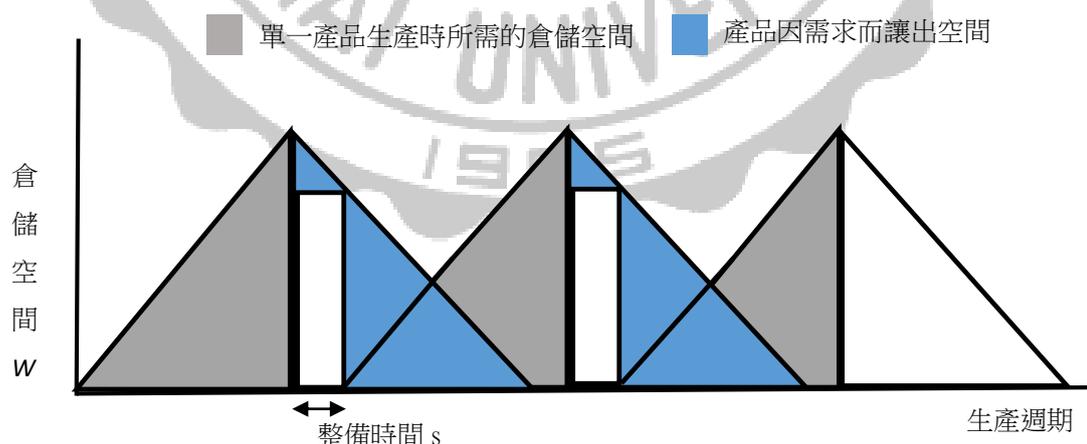


圖 3-1 多產品倉儲空間可彼此佔用模型

本研究以 2 個產品為例，說明倉儲空間大小公式的推導。產品的生產順序若

是依據 $d_i < p_{(i+1)} - d_{(i+1)}$ 之順序排列，當產品 1 產品 2 生產完所達到最大的倉儲空間，再減去在產品 2 生產以及整備時產品 1 因為商品需求的關係而讓出的倉儲空間，即可用公式(3.5)求得倉儲空間。

$$W = (p_1 - d_1) \left( \frac{d_1 T}{p_1} \right) + (p_2 - d_2) \left( \frac{d_2 T}{p_2} \right) - d_1 \left( \frac{d_2 T}{p_2} + S_2 \right) \quad (3.5)$$

以三個產品為例：

三產品在 $d_i < p_{(i+1)} - d_{(i+1)}$ 的生產順序排列，當產品 1、產品 2 與產品 3 生產完所達到最大的倉儲空間，再減去在產品 2 產品 3 生產以及整備時產品 1 因為商品需求的關係而讓出的倉儲空間，再減去在產品 3 生產以整備時產品 2 因為商品需求的關係而讓出的倉儲空間，即可用公式(3.6)求得倉儲空間。

$$W = (p_1 - d_1) \left( \frac{d_1 T}{p_1} \right) + (p_2 - d_2) \left( \frac{d_2 T}{p_2} \right) + (p_3 - d_3) \left( \frac{d_3 T}{p_3} \right) - d_1 \left( \frac{d_2 T}{p_2} + \frac{d_3 T}{p_3} + S_2 + S_3 \right) - d_2 \left( \frac{d_3 T}{p_3} + S_3 \right) \quad (3.6)$$

由於數學式(3.7)(3.8)的拆解式有一定的規律，因此可以推導出再 $d_i < p_{(i+1)} - d_{(i+1)}$ 的生產順序情況下，不管是多少個產品，我們都能運用公式(3.9)算出多產品的倉儲空間大小。

數學式(3.7)：

$$W = (p_1 - d_1) \left( \frac{d_1 T}{p_1} \right) - d_1 \left( \frac{d_2 T}{p_2} + S_2 \right) + (p_2 - d_2) \left( \frac{d_2 T}{p_2} \right) \quad (3.7)$$

數學式(3.8)：

$$W = (p_1 - d_1) \left( \frac{d_1 T}{p_1} \right) - d_1 \left( \frac{d_2 T}{p_2} + \frac{d_3 T}{p_3} + S_2 + S_3 \right) + (p_2 - d_2) \left( \frac{d_2 T}{p_2} \right) - d_2 \left( \frac{d_3 T}{p_3} + S_3 \right) + (p_3 - d_3) \left( \frac{d_3 T}{p_3} \right) \quad (3.8)$$

多產品的倉儲空間大小之總結式為公式(3.9)：

$$W = \sum_{i=1}^n \left[ (p_i - d_i) \frac{d_i T}{p_i} \right] - \sum_{i=1}^{n-1} (d_i) \left[ \sum_{j=i+1}^n \left( \frac{d_j T}{p_j} + S_j \right) \right] \quad (3.9)$$

## 3.2 彈性速率變更下 ELSP 模式

### 3.2.1 基本假設與數學符號

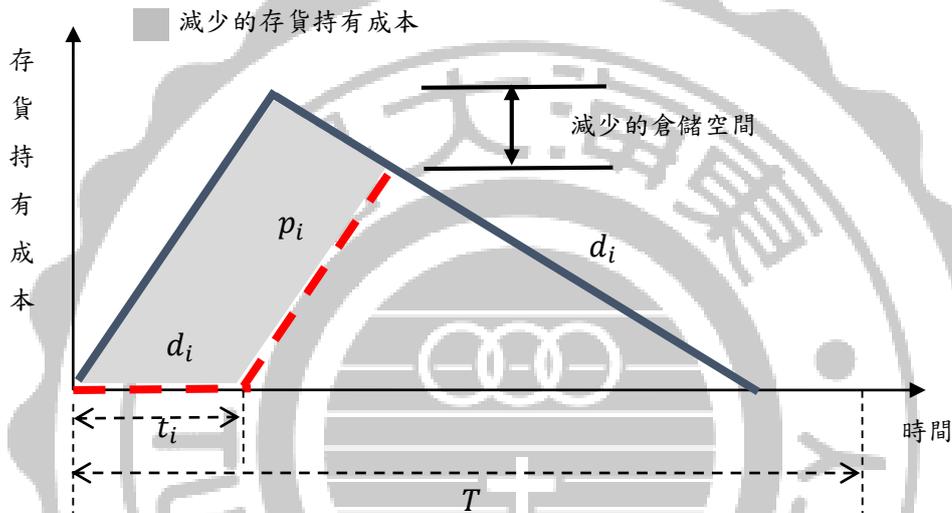


圖 3-2 在週期  $T$  內以彈性速率生產產品  $i$  的時間與存貨關係

彈性速率法設定的生產速率等同於需求率進行生產一段時間後，再以最大速率來進行生產。圖 3-1 在週期  $T$  內以彈性速率生產產品  $i$  的時間與存貨關係，虛線為彈性速率，灰色區塊為減少的存貨成本，如圖 3-2 所示。

### 3.2.2 本研究數學符號定義

以下為彈性速率模式下考量倉儲空間與成本的 ELSP 數學符號：

1.  $k$ ：選定降速之產品代號。
2.  $p_k$ ：變更生產速率之產品  $k$  的生產率。
3.  $d_k$ ：變更生產速率之產品  $k$  的需求率。
4.  $I$ ：一生產週期內的產能閒置時間(idle time)。
5.  $t_k$ ：一生產週期內變更生產速率之產品  $k$  以  $d_i$  速率的生產時間。

6.  $i$  : 第  $i$  個生產的產品。

其它數學符號與 3.1.1 的符號定義一樣。本研究假設生產時間、整備時間及閒置時間的總和不超过  $T$ ，其它的基本假設條件與 3.1.2 的基本假設一樣。

### 3.2.3 在共同週期法下運用彈性速率之公式建立

以下為彈性速率模式下的公式推導：

1. 在全速生產狀況下，生產週期為生產時間加上產能閒置時間，式子如下：

$$\sum_{i=1}^n \frac{d_i T}{p_i} + I = T \quad (3.10)$$

2. 在彈性速率生產狀況下，生產週期為降速生產時間加上全速生產時間，式子如下：

$$\sum_{i \in J} \frac{d_i T}{p_i} + t_k + \frac{d_k(T - t_k)}{p_k} = T, \quad J = \{1, 2, 3, \dots, n\} \setminus \{k\}, \quad k \in \{1, 2, 3, \dots, n\} \quad (3.11)$$

3. 將公式(3.10)及公式(3.11)兩式合併，可得出降速生產時間如公式(3.17)：

$$\sum_{i=1}^n \frac{d_i T}{p_i} + I = \sum_{i \in J} \frac{d_i T}{p_i} + t_k + \frac{d_k(T - t_k)}{p_k} \quad (3.12)$$

$$\frac{d_k T}{p_k} + I = t_k + \frac{d_k(T - t_k)}{p_k} \quad (3.13)$$

$$t_k = \frac{d_k T}{p_k} + I - \frac{d_k(T - t_k)}{p_k} \quad (3.14)$$

$$t_k = I + \rho_k t_k \quad (3.15)$$

$$(1 - \rho_k)t_k = I \quad (3.16)$$

$$t_k = \frac{I}{(1 - \rho_k)} \quad (3.17)$$

### 3.2.4 成本項分析

#### 1. 存貨持有成本(holding cost)

產品  $i$  在一生產週期  $T$  內的時間與存貨關係如圖 3-3：

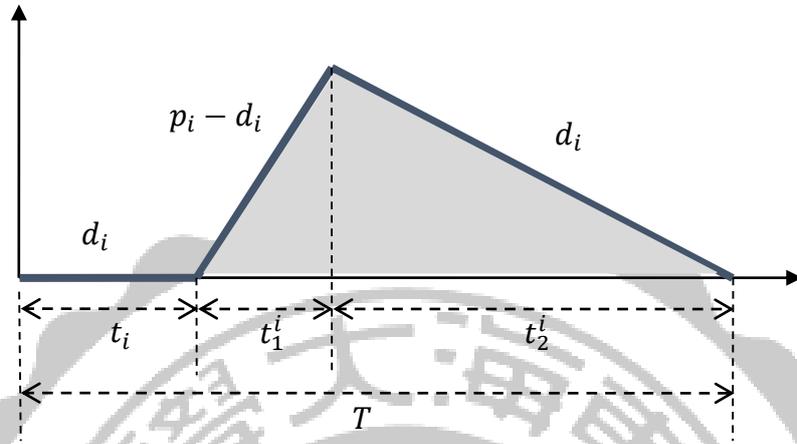


圖 3-3 在週期  $T$  內以彈性速率生產產品  $i$  的時間與存貨關係

$$(p_i - d_i)t_1^i = d_i t_2^i \text{ 且 } t_1^i + t_2^i = T - t_i$$

本研究針對生產週期  $T$  內產品中或持有成本最高 ( $\text{Max}\{d_i h_i\}$ ) 之產品  $i$  以速率  $d_i$  生產  $t_i$  時間後，再以速率  $p_i$  生產。由於本研究考量存貨持有成本，倉儲的租賃成本也考慮在內，因此在生產週期  $T$  內本研究有可能不針對產品的持有成本最高的產品進行降速。根據上節的公式推導，若產品  $i$  為變更生產速率的產品，生產週期  $T$  內將以速率  $d_i$  生產  $t_i$  的時間，否則  $t_i = 0$  (生產週期內以速率  $p_i$  生產)。

簡化後得：

$$t_i = \frac{I}{(1 - \rho_i)}$$

$$t_1^i = \rho_i (T - t_i), \quad t_2^i = (1 - \rho_i)(T - t_i)$$

因此，在生產週期  $T$  內，產品  $i$  的平均存貨持有水準為上圖灰色面積，等於：

$$\frac{1}{2} \times (T - t_i) \times d_i (1 - \rho_i)(T - t_i) = \frac{d_i (1 - \rho_i)(T - t_i)^2}{2}$$

所以，在生產週期  $T$  內，產品  $i$  的平均存貨持有成本為：

$$\text{每單位存貨持有成本} \times \text{平均存貨持有水準} = \frac{h_i d_i (1 - \rho_i)(T - t_i)^2}{2} \quad (3.18)$$

### 3.2.5 彈性速率下倉儲的租賃成本

本研究探討多個產品的倉儲空間是可互換的，因為對倉儲空間可互換進行公是推導，由於產品生產完之後會進入倉儲儲藏，產品會存放在倉儲中繼續販售，因此可以得知產品生產完畢的時間點是倉儲使用空間的最大值，當產品販售完畢時倉儲空為零。假設前一個產品生產完之後，倉儲空間空閒出來能被下一個生產的產品占用。以 15 個產品為例，可能的生產順序會有  $15!$  種，這會使數學模式變得難以推導。為了簡化數學模式的推導過程，本研究僅考慮以下兩種排列順序：

$$\text{Seq1} : d_i > p_{(i+1)} - d_{(i+1)}$$

$$\text{Seq2} : d_i < p_{(i+1)} - d_{(i+1)}$$

在分析存貨時有成本時，提到在生產週期  $T$  內本研究有可能不針對產品的持有成本最高的產品進行降速，但由於本研究為第一篇研究以彈性速率來解決 ELSP 倉儲問題，因此我們以持有成本最高的產品進行降速。因此本研究接下來進行 Seq1 與 Seq2 每個產品數不同產品降速進行數學模式推導。

以 Seq1 :  $d_i > p_{(i+1)} - d_{(i+1)}$  兩個產品進行數學模式說明：

依據 Seq1 :  $d_i > p_{(i+1)} - d_{(i+1)}$  生產排列情況下，當降速為產品 1 時，產品所需要的倉儲空間如公式(3.19)，由於產品 1 的需求率  $d_i$  大於生產中產品庫存速率  $p_{(i+1)} - d_{(i+1)}$ ，因此降速後的產品 1 會釋出足以容納產品 2 庫存的倉儲空間，使得產品 2 所需的倉儲空間為 0。

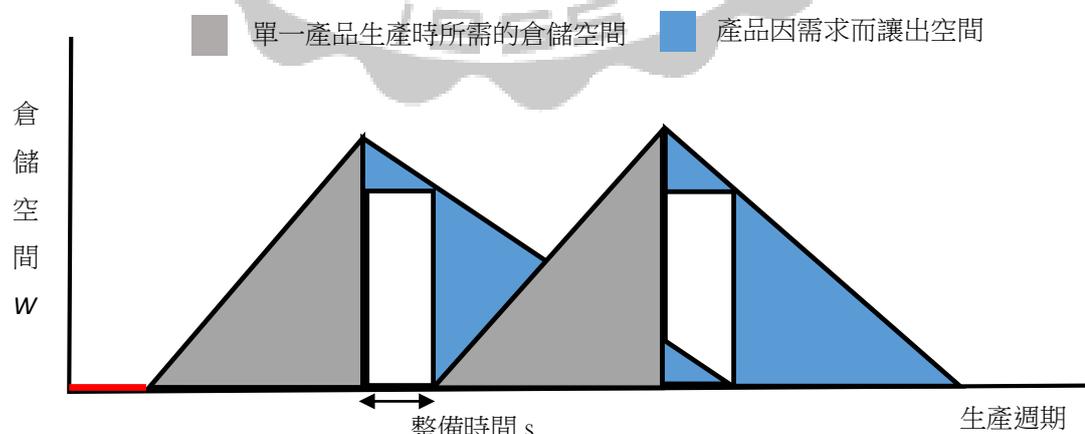


圖 3-4 Seq1 順序下兩個產品中第一產品降速圖

公式推導：

$$W = (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) \quad (3.19)$$

依據 Seq1： $d_i > p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 所需要的倉儲空間如公式(3.20)，由於產品 1 的需求率 $d_i$ 大於生產中產品庫存速率 $p_{(i+1)} - d_{(i+1)}$ ，因此產品一會釋出足以容納產品 2 降速後庫存的倉儲空間，產品 2 生產變更速率並不影響倉儲空間，使得產品 2 所需的倉儲空間為 0。

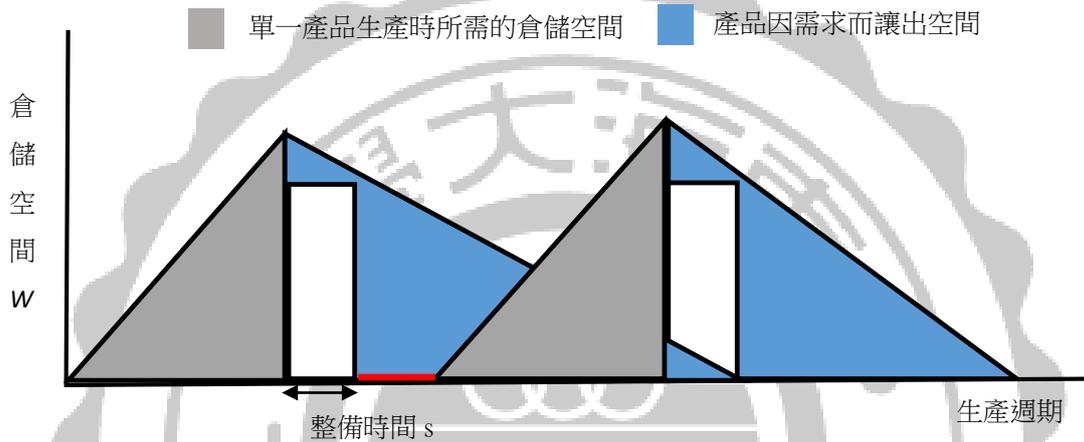


圖 3-5 Seq1 順序下兩個產品中第二產品降速圖

公式推導：

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) \quad (3.20)$$

由於 Seq1： $d_i > p_{(i+1)} - d_{(i+1)}$  在現實世界的出現機率較低，若出現，其最佳倉儲空間大小必為第 1 個產品所佔用的倉儲空間大小，合併公式(3.19)與公式(3.20)，公式如(3.21)所示：

$$W = (p_i - d_i) \left( \frac{(d_i(T - t_i))}{p_i} \right) \quad (3.21)$$

以 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  兩個產品進行數學模式說明：

依據 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，當產品 1 降速後的需求率 $d_i$ 較生產中的產品庫存 $d_i < p_{(i+1)} - d_{(i+1)}$ 速率小，因此產品 1 所佔用的倉儲空間無法在下一個產品生產時釋出足以容納新產品庫存的倉儲空間，以下為倉儲推導。

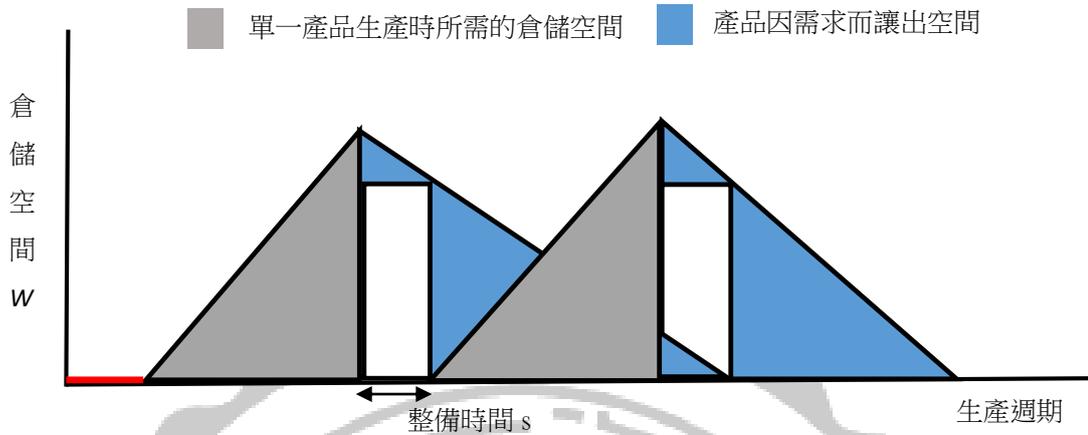


圖 3-6 Seq2 順序下兩個產品中第一產品降速圖

$$W = (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) \quad (3.22)$$

$$W = (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) \quad (3.23)$$

$$W = d_1 \left( \frac{d_2T}{p_2} + S_2 \right) \quad (3.24)$$

當產品 1 為降速產品生產完成的倉儲空間如公式(3.22)，加上產品 2 每單位所生產的時間如公式(3.23)，再減去產品 2 生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.24)。

結合(3.22)(3.23)(3.24)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.25)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) - d_1 \left( \frac{d_2T}{p_2} + S_2 \right) \quad (3.25)$$

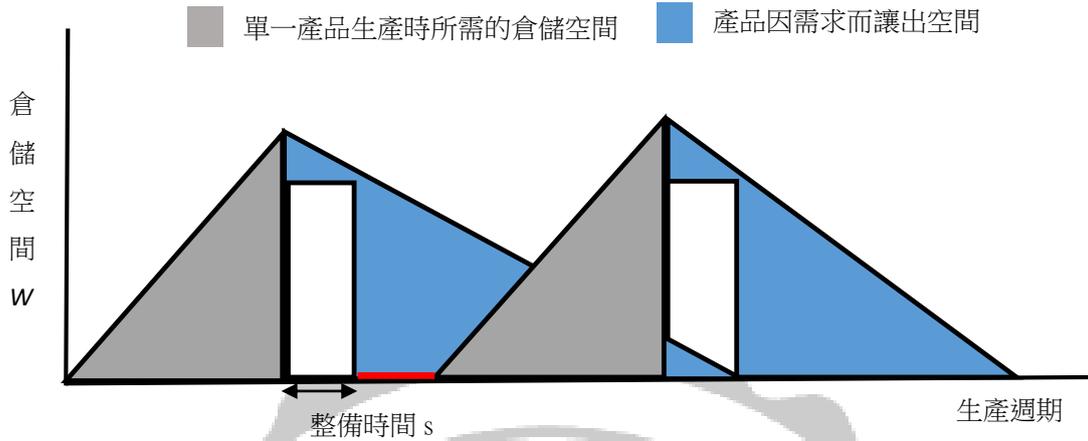


圖 3-7 Seq2 順序下兩個產品中第二產品降速圖

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) \quad (3.26)$$

$$W = (p_2 - d_2) \left( \frac{(d_2 (T - t_2))}{p_2} \right) \quad (3.27)$$

$$W = d_1 \left( S_2 + t_2 + \frac{d_2 (T - t_2)}{p_2} \right) \quad (3.28)$$

當產品 2 降速時，依據 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間如公式(3.26)，加上降速後產品 2 每單位所生產的時間如公式(3.27)，再減去產品 2 生產和整備時間和變更生產時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.28)。

結合(3.26)(3.27)(3.28)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.29)。

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 (T - t_2))}{p_2} \right) - d_1 \left( S_2 + t_2 + \frac{d_2 (T - t_2)}{p_2} \right) \quad (3.29)$$

以依據 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  三個產品進行數學模式說明：

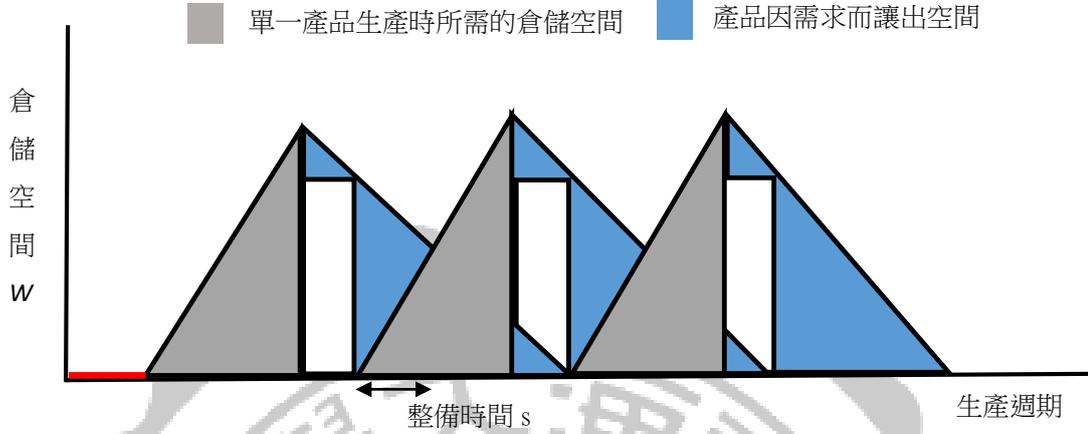


圖 3-8 Seq2 順序下三個產品中第一產品降速圖

依據 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  當產品 1 降速時，依據生產排列情況下，當產品 1 為降速產品生產完成的倉儲空間如公式(3.30)，加上產品 2 產品 3 每單位所生產的時間如公式(3.31)，再減去產品 2 生產和整備時間與產品 3 生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.32)，再減去產品 3 生產和整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.33)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) \quad (3.30)$$

$$W = (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3T)}{p_3} \right) \quad (3.31)$$

$$W = d_1 \left( S_2 + S_3 + \frac{(d_2T)}{p_2} + \frac{(d_3T)}{p_3} \right) \quad (3.32)$$

$$W = d_2 \left( S_3 + \frac{(d_3T)}{p_3} \right) \quad (3.33)$$

結合(3.30)(3.31)(3.32)(3.33)公式，可以推導在 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.34)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3T)}{p_3} \right) - d_1 \left( S_2 + S_3 + \frac{(d_2T)}{p_2} + \frac{(d_3T)}{p_3} \right) - d_2 \left( S_3 + \frac{(d_3T)}{p_3} \right) \quad (3.34)$$

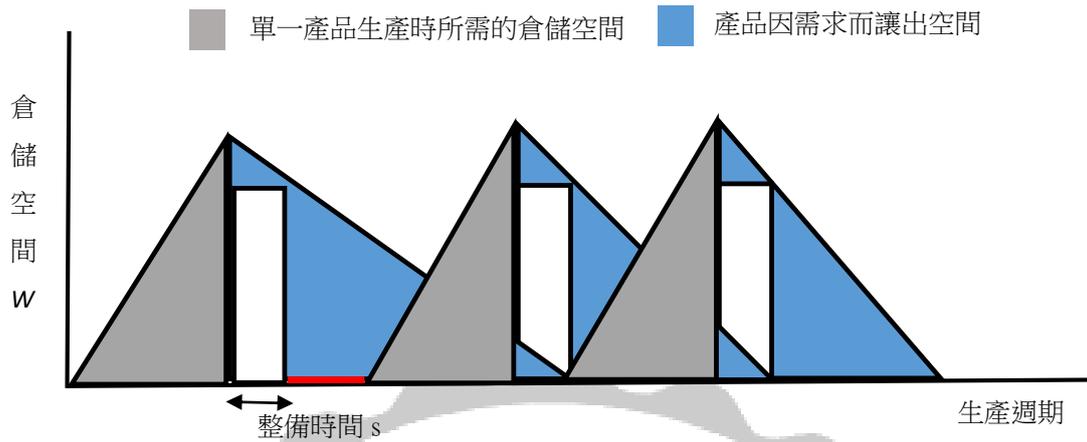


圖 3-9 Seq2 順序下三個產品中第二產品降速圖

當產品 2 降速時，依據 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間如公式(3.35)，加上降速後產品 2 每單位所生產的時間如公式(3.36)，產品 3 每單位所生產的時間如公式(3.37)，再減去產品 2 生產和整備時間和變更生產時間與產品 3 生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.38)，再減去產品 3 生產和整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.39)。

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) \quad (3.35)$$

$$W = (p_2 - d_2) \left( \frac{(d_2 (T - t_2))}{p_2} \right) \quad (3.36)$$

$$W = (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) \quad (3.37)$$

$$W = d_1 \left( S_2 + S_3 + t_2 + \frac{(d_2 (T - t_2))}{p_2} + \frac{(d_3 T)}{p_3} \right) \quad (3.38)$$

$$W = d_2 \left( S_3 + \frac{(d_3 T)}{p_3} \right) \quad (3.39)$$

結合(3.35)(3.36)(3.37) (3.38) (3.39)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.40)。

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 (T - t_2))}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) - d_1 \left( S_2 + S_3 + t_2 + \frac{(d_2 (T - t_2))}{p_2} + \frac{(d_3 T)}{p_3} \right)$$

$$-d_2 \left( S_3 + \frac{(d_3 T)}{p_3} \right) \quad (3.40)$$

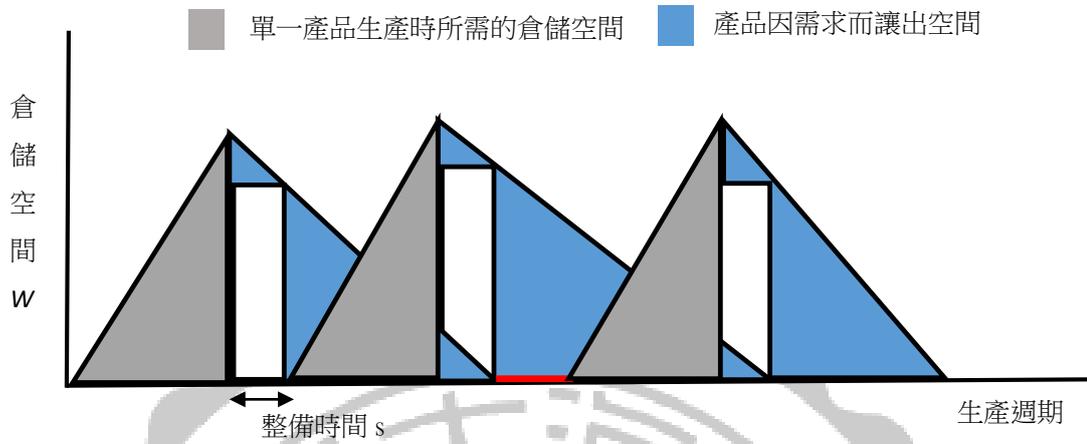


圖 3-10 Seq2 順序下三個產品中第三產品降速圖

當產品 3 降速時，依據 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間，加上產品 2 每單位所生產的時間如公式(3.41)，降速後產品 3 每單位所生產的時間如公式(3.42)，再減去產品 2 生產和整備時間與產品 3 生產和整備時間和變更生產時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.43)，再減去產品 3 生產和整備時間和變更生產時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.44)。

$$W = (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) \quad (3.41)$$

$$W = (p_3 - d_3) \left( \frac{(d_3(T - t_3))}{p_3} \right) \quad (3.42)$$

$$W = d_1 \left( S_2 + S_3 + t_3 + \frac{(d_2 T)}{p_2} + \frac{(d_3(T - t_3))}{p_3} \right) \quad (3.43)$$

$$W = d_2 \left( S_3 + t_3 + \frac{(d_3(T - t_3))}{p_3} \right) \quad (3.44)$$

結合(3.41)(3.42)(3.43) (3.44)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.45)。

$$\begin{aligned}
W &= (p_1 - d_1) \left( \frac{(d_1 T)}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3 (T - t_3))}{p_3} \right) \\
&\quad - d_1 \left( S_2 + S_3 + t_3 + \frac{(d_2 T)}{p_2} + \frac{(d_3 (T - t_3))}{p_3} \right) \\
&\quad - d_2 \left( S_3 + t_3 + \frac{(d_3 (T - t_3))}{p_3} \right)
\end{aligned} \tag{3.45}$$

以 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  四個產品進行數學模式說明：

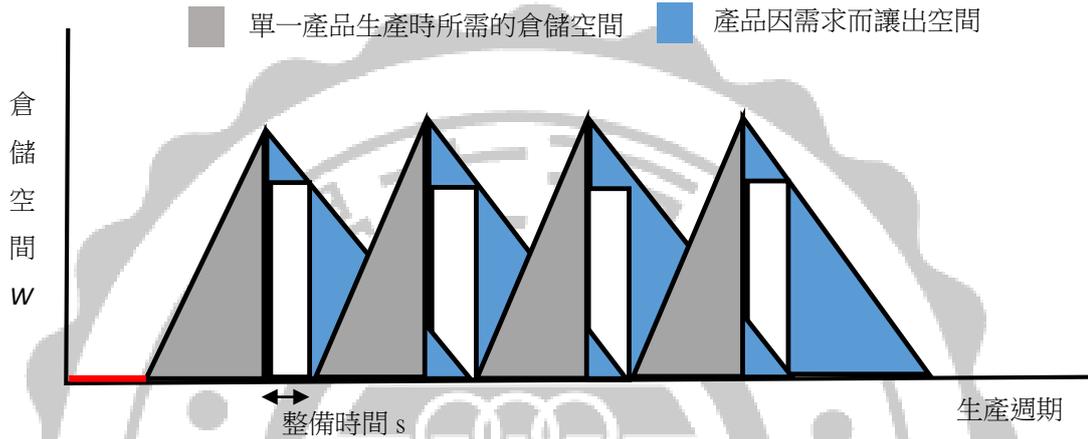


圖 3-11 Seq2 順序下四個產品中第一產品降速圖

當產品 1 降速時，依據 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，當產品 1 為降速產品生產完成的倉儲空間，加上產品 2 產品 3 產品 4 每單位所生產的時間如公式(3.46)，再減去產品 2、產品 3、產品 4 生產整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.47)，再減去產品 3、產品 4 生產整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.48)，再減去產品 4 生產整備時間，產品 3 因為商品需求的關係而讓出的倉儲空間如公式(3.49)。

$$\begin{aligned}
W &= (p_1 - d_1) \left( \frac{(d_1 (T - t_1))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) \\
&\quad + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right)
\end{aligned} \tag{3.46}$$

$$W = d_1 \left( S_2 + S_3 + S_4 + \frac{(d_2 T)}{p_2} + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) \tag{3.47}$$

$$W = d_2 \left( S_3 + S_4 + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) \tag{3.48}$$

$$W = d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \tag{3.49}$$

結合(3.46)(3.47)(3.48)(3.49)公式，可以推導在 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.50)。

$$\begin{aligned}
 W = & (p_1 - d_1) \left( \frac{(d_1(T - t_1))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) \\
 & + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right) - d_1 \left( S_2 + S_3 + S_4 + \frac{(d_2 T)}{p_2} + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) \\
 & - d_2 \left( S_3 + S_4 + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) - d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \quad (3.50)
 \end{aligned}$$

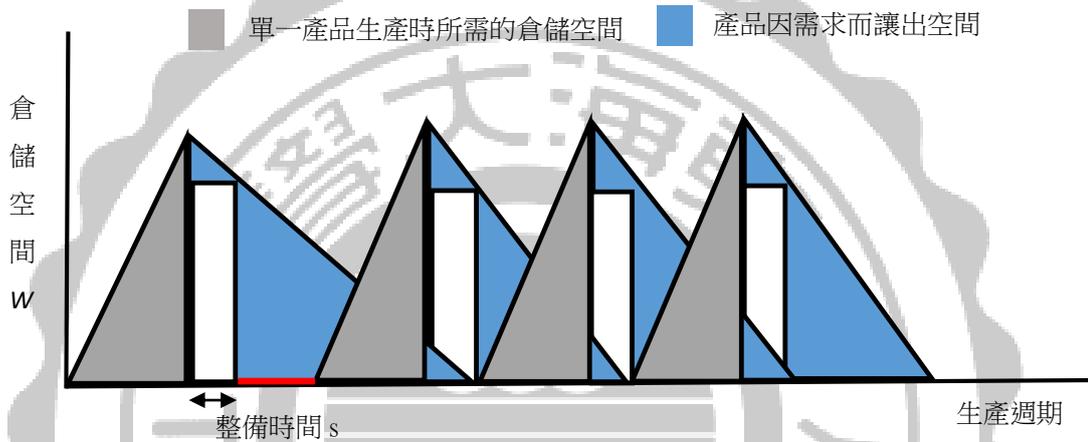


圖 3-12 Seq2 順序下四個產品中第二產品降速圖

當產品 2 降速時，依據 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間，加上降速後產品 2 每單位所生產的時間如公式(3.51)，產品 3 產品 4 每單位所生產的時間如公式(3.52)，再減去產品 2 生產和整備時間和變更生產時間與產品 3 產品 4 生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.52)，再減去產品 3 產品 4 生產和整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間，產品 4 生產和整備時間，產品 3 因為商品需求的關係而讓出的倉儲空間如公式(3.54)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2(T - t_2))}{p_2} \right) \quad (3.51)$$

$$W = (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right) \quad (3.52)$$

$$W = d_1 \left( S_2 + S_3 + S_4 + t_2 + \frac{(d_2(T - t_2))}{p_2} + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) \quad (3.53)$$

$$W = d_2 \left( S_3 + S_4 + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) - d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \quad (3.54)$$

結合(3.51)(3.52)(3.53)(3.54)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.55)。

$$\begin{aligned} W = & (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2(T - t_2))}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3 T)}{p_3} \right) \\ & + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right) - d_1 \left( S_2 + S_3 + S_4 + t_2 + \frac{(d_2(T - t_2))}{p_2} + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) \\ & - d_2 \left( S_3 + S_4 + \frac{(d_3 T)}{p_3} + \frac{(d_4 T)}{p_4} \right) - d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \end{aligned} \quad (3.55)$$

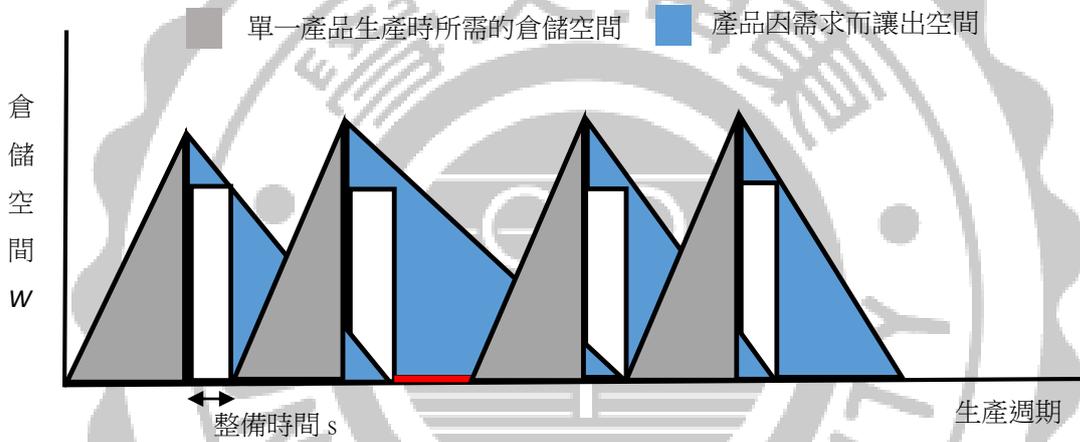


圖 3-13 Seq2 順序下四個產品中第三產品降速圖

當產品 3 降速時，依據 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間，加上產品 2、降速後產品 3、產品 4 每單位所生產的時間如公式(3.56)，再減去產品 2 生產和整備時間和產品 3 變更生產時間與產品 4 生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.57)，再減去產品 3 變更生產時間與產品 4 生產和整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.58)，產品 4 生產和整備時間，產品 3 因為商品需求的關係而讓出的倉儲空間如公式(3.59)。

$$\begin{aligned} W = & (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3(T - t_3))}{p_3} \right) \\ & + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right) \end{aligned} \quad (3.56)$$

$$W = d_1 \left( S_2 + S_3 + S_4 + t_3 + \frac{(d_2 T)}{p_2} + \frac{(d_3(T - t_3))}{p_3} + \frac{(d_4 T)}{p_4} \right) \quad (3.57)$$

$$W = d_2 \left( S_3 + S_4 + t_3 + \frac{(d_3(T - t_3))}{p_3} + \frac{(d_4 T)}{p_4} \right) \quad (3.58)$$

$$W = d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \quad (3.59)$$

結合(3.56)(3.57)(3.58)(3.59)公式，可以推導在 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.60)。

$$\begin{aligned} W = & (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2 T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3(T - t_3))}{p_3} \right) \\ & + (p_4 - d_4) \left( \frac{(d_4 T)}{p_4} \right) - d_1 \left( S_2 + S_3 + S_4 + t_3 + \frac{(d_2 T)}{p_2} + \frac{(d_3(T - t_3))}{p_3} + \frac{(d_4 T)}{p_4} \right) \\ & - d_2 \left( S_3 + S_4 + t_3 + \frac{(d_3(T - t_3))}{p_3} + \frac{(d_4 T)}{p_4} \right) - d_3 \left( S_4 + \frac{(d_4 T)}{p_4} \right) \end{aligned} \quad (3.60)$$

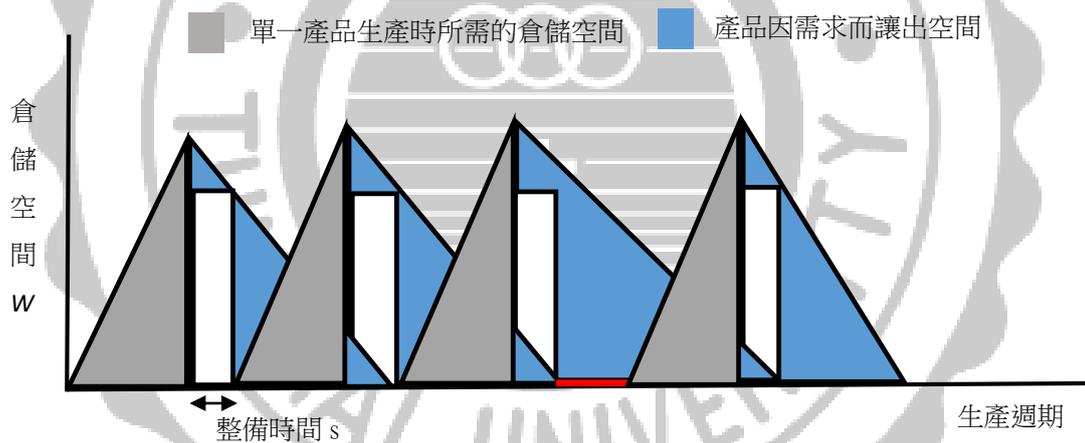


圖 3-14 Seq2 順序下四個產品中第四產品降速圖

當產品 4 降速時，依據 Seq2 :  $d_i < p_{(i+1)} - d_{(i+1)}$  生產排列情況下，產品 1 生產完成的倉儲空間，加上產品 2、產品 3、降速後產品 4 每單位所生產的時間如公式(3.61)，再減去產品 2、產品 3 生產和整備時間和產品 4 變更生產時間和生產和整備時間，產品 1 因為商品需求的關係而讓出的倉儲空間如公式(3.62)，再減去產品 3 生產和整備時間與產品 4 變更生產時間和生產和整備時間，產品 2 因為商品需求的關係而讓出的倉儲空間如公式(3.63)，產品 4 生產整備時間和變更生產時間，產品 3 因為商品需求的關係而讓出的倉儲空間如公式(3.64)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3T)}{p_3} \right) \quad (3.61)$$

$$+ (p_4 - d_4) \left( \frac{(d_4(T-t_4))}{p_4} \right)$$

$$W = d_1 \left( S_2 + S_3 + S_4 + t_4 + \frac{(d_2T)}{p_2} + \frac{(d_3T)}{p_3} + \frac{(d_4(T-t_4))}{p_4} \right) \quad (3.62)$$

$$W = d_2 \left( S_3 + S_4 + t_4 + \frac{(d_3T)}{p_3} + \frac{(d_4(T-t_4))}{p_4} \right) \quad (3.63)$$

$$W = d_3 \left( S_4 + t_4 + \frac{(d_4(T-t_4))}{p_4} \right) \quad (3.64)$$

結合(3.61)(3.62)(3.63)(3.64)公式，可以推導在 Seq2： $d_i < p_{(i+1)} - d_{(i+1)}$  的生產順序情況下，產品倉儲空間大小總結式如公式(3.65)。

$$W = (p_1 - d_1) \left( \frac{(d_1(T))}{p_1} \right) + (p_2 - d_2) \left( \frac{(d_2T)}{p_2} \right) + (p_3 - d_3) \left( \frac{(d_3T)}{p_3} \right) \quad (3.65)$$

$$+ (p_4 - d_4) \left( \frac{(d_4(T-t_4))}{p_4} \right) - d_1 \left( S_2 + S_3 + S_4 + t_4 + \frac{(d_2T)}{p_2} + \frac{(d_3T)}{p_3} + \frac{(d_4(T-t_4))}{p_4} \right)$$

$$- d_2 \left( S_3 + S_4 + t_4 + \frac{(d_3T)}{p_3} + \frac{(d_4(T-t_4))}{p_4} \right) - d_3 \left( S_4 + t_4 + \frac{(d_4(T-t_4))}{p_4} \right)$$

以二個、三個及四個產品的任一項產品降速的推導過程中，能發現數學式有一定的規律，因此我們可以推導在 $d_i < p_{(i+1)} - d_{(i+1)}$ 的生產情況下，多產品的倉儲空間大小之總結式如公式(3.66)：

$$W = \sum_{i=1}^n \left[ (p_i - d_i) \frac{d_i(T - t_i)}{p_i} \right] - \sum_{i=1}^{n-1} (d_i) \left[ \sum_{j=i+1}^n \left( \frac{d_j(T - t_j)}{p_j} + S_j + t_j \right) \right] \quad (3.66)$$

本研究考慮到產品最佳生產順序，因此產品倉儲可以互換，依照生產順序來進行生產，本研究以螞蟻最佳化演算法來尋出最佳的生產順序，本研究目標是找到最低的倉儲空間與最低的總成本，3.3 節我們會詳細說明螞蟻最佳化演算法找尋最佳生產順序。

### 3.2.6 數學模式建構

以下為本研究設計在降速 ELSP 問題的數學模式，表式如下：

$$MinTC = \sum_{i=1}^n \left[ \frac{A_i}{T} + \frac{h_i d_i (1 - \rho_i) (T - t_i)^2}{2T} \right] + (W - w)\alpha \quad (3.67)$$

$$Subject\ to \sum_{i=1}^n [S_i + \rho_i T] \leq T \quad (3.68)$$

$$T = \sqrt{\frac{2 \sum_{i=1}^n A_i}{\sum_{i=1}^n [h_i d_i (1 - \rho_i)]}} \quad (3.69)$$

$$t_i = \begin{cases} \frac{I}{(1 - \rho_i)} & \text{產品 } i \text{ 被選中進行降速，} t_i \text{ 時間內以 } d_i \text{ 速率進行生產} \\ 0 & \text{未降速產品以 } d_i \text{ 速率生產的時間}(t_i) \text{ 為 } 0 \end{cases} \quad (3.70)$$

$$I = T - \sum_{i=1}^n (S_i + \rho_i T) \quad (3.71)$$

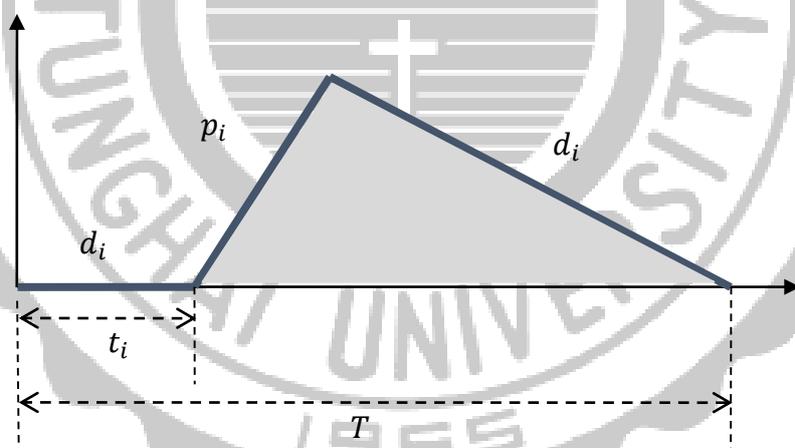


圖 3-15 在週期  $T$  內以彈性速率生產產品  $i$  的生產速率

公式(3.67)為目標成本函數，成本包含整備成本、存貨持有成本及倉儲成本；為了方便計算倉儲空間，由於本研究基本假設  $W$  為總倉儲， $w$  為內部倉儲空間，為了方便計算本研究假設內部倉儲空間( $w$ )為 0，因此所算出的倉儲空間為外部倉儲空間，運用彈性速率求解 ELSP 倉儲空間大小與單位時間最低的總成本。公式(3.68)為生產產能限制式，所有產品的整備時間加上生產時間必須小於等於整個

生產週期  $T$ 。公式(3.69)計算出最佳的生產週期時間長度。公式(3.70)為找出變更生產速率的產品做降速，當產品降速先以  $t_i = \left(\frac{I}{1-\rho_i}\right)$  時間內以  $d_i$  速率進行生產，再以  $p_i$  的速率全速生產，當不為降速的產品皆以  $p_i$  的速率進行生產。公式(3.71)為生產週期減去整備時間加上生產時間的時間後，得到在週期內的產能閒置時間。

### 3.3 以螞蟻演算法求解生產順序

螞蟻系統(Ant System, AS)觀念最早是由 Dorigo(1992)在博士論文中提出利用螞蟻群體合作尋找食物的行為，所設計出一套用於處理最佳化問題的螞蟻系統；Dorigo *et al.*(1996)等學者於 IEEE 國際期刊上提出第一個最佳化模型，描述螞蟻理論的基本定義與演算步驟，並在研究中提出螞蟻理論求解問題最佳化方法。而螞蟻演算法已被廣泛地應用於求解許多組合最佳化問題，如：旅行推銷員問題(Traveling Salesman Problem, TSP)、二次分配問題(Quadratic Assignment Problem, QAP)、排程問題(scheduling problem)、車輛途程問題(Vehicle Routing Problem, VRP)等問題。

Dorigo 和 Gambardella(1997)針對 AS 做了三處改良，加入了轉換規則(transition rule)、改變費洛蒙路徑更新方法及增加局部更新法(local updating)，成功的應用螞蟻族群系統(Ant Colony System, ACS)於 TSP 上，結果與其他啟發式演算法相比較具有不錯的效率，於是許多學者 Maniezzo(1999)、McMullen(2001)、Chang (2014)紛紛開始將其應用在各領域研究問題上，本研究也以 ACS 演算法來找尋最佳產品生產順序。

#### 3.3.1 螞蟻系統介紹

螞蟻系統起源自於螞蟻尋找食物的過程，螞蟻在尋找食物的過程中，會在走過的地方殘留一種分泌，被稱費洛蒙(pheromone)的化學物質，之後的螞蟻會根

據前面螞蟻留下的費洛蒙濃度，作為選擇路徑的依據。螞蟻演算法模仿螞蟻尋找食物的過程，當遇到岔路時，會有許多單一螞蟻任意去嘗試各種方向的路徑，在走過的路徑上也都會殘留一定的費洛蒙量，而螞蟻會依循費洛蒙強度較高的路徑去走。隨著時間增長，殘留的費洛蒙也會蒸發。較短的路徑由於螞蟻走過的次數增加，使得費洛蒙濃度會逐漸累積讓濃度增強，相反的較遠的路徑，螞蟻走過的次數會漸漸減少，因此費洛蒙的濃度反而會因為時間長了而蒸發，使得費洛蒙濃度降低。上述描述螞蟻找到最短路徑的方法之模擬，如下圖 3-16 所示：

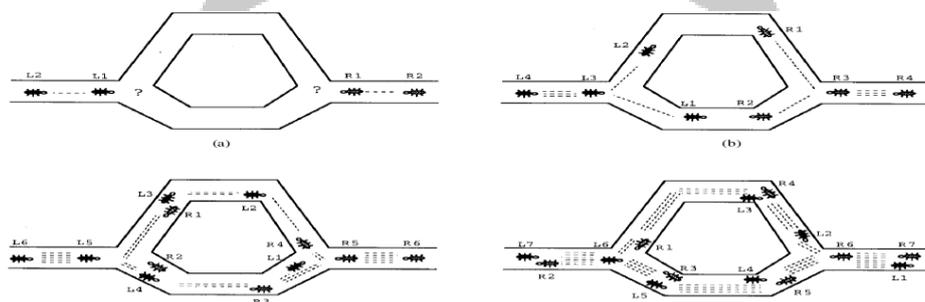


圖 3-16 螞蟻演算法路徑示意圖

(資料來源：Dorigo and Gambardella, 1997)

螞蟻演算法是將上面所敘述的螞蟻尋找食物的行為，進一步的轉換成演算法求解的方式，而螞蟻系統(Ant System, AS)為蟻群優化 (Ant Colony Optimization, ACO) 演算法中最早被提出來的方法(Stützle and Dorigo, 1999)，後續許多學者相繼提出各式不同求解機制的螞蟻演算法，目前改善效果較為顯著的有蟻群系統 (Ant Colony System, ACS)、極大極小螞蟻系統(MAX-MIN Ant System, MMAS)和基於排序的螞蟻系統(Rank-Based Version of Ant System, ASrank)。

螞蟻系統演算法求解的模式主要有以下幾個步驟：

(一) 初始化：

首先第一步須先設定相關參數，費洛蒙濃度初始值 $\tau_0$ 、決定費洛蒙濃度權重參數 $\alpha$ 、絕對距離參數 $\beta$ 、費洛蒙揮發參數 $\rho$ 、人工螞蟻數目  $m$ ，設定起始迭代次數  $t=0$ ，並平均在每條路徑上置放固定常數的初始費洛蒙量，以提供螞蟻進行搜

尋的依據。每一次循環開始以隨機的方式產生每隻螞蟻的起始點，同一個起點能放多隻螞蟻進行探索。

(二)轉換機率：

螞蟻系統為最早將螞蟻理論實現的方法，Colormi *et al.* (1996)明確定義人工螞蟻進行選擇要走的節點與經過的路徑時，依據公式(3.72)轉換機率來選擇下一個要走的節點，而考慮的因素為費洛蒙濃度與路徑長度轉換成機率值來決定，當 $\alpha$ 值比 $\beta$ 大很多時則表示指引螞蟻方向的僅有費洛蒙，並沒有距離參數，因此容易在求解過程產生停滯，陷入區域解。

$$p_{ij}^k(t) = \begin{cases} \frac{(\tau_{ij}(t))^\alpha (\eta_{ij})^\beta}{\sum_{u \in J_k(i)} (\tau_{iu}(t))^\alpha (\eta_{iu})^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases} \quad (3.72)$$

其中，

$\tau_{ij}(t)$ ：在時間點  $t$  時路徑  $(i, j)$  上所殘留的費洛蒙數量。

$J_k(i)$ ：位於節點  $i$  之螞蟻  $k$  尚未拜訪過的鄰近節點集合。

$\eta_{ij}$ ：代表期望值，等於路徑  $(i, j)$  長度的倒數。

$\alpha, \beta$ ：控制費洛蒙與能見度間相對重要性的參數。

$p_{ij}^k(t)$ ：第  $k$  隻螞蟻從節點  $i$  要到節點  $j$  的轉移機率。

(三)更新費洛蒙濃度：

假如螞蟻走過此路徑，就會改變該路徑上的費洛蒙濃含量，否則該路徑的費洛蒙含量就會慢慢揮發；費洛蒙含量揮發計算方式如公式(3.73)(3.74)所示。

$$\tau_{ij}(t+1) = (1 - \rho) \times \tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (3.73)$$

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & , \text{if } k_{th} \text{ ant uses edge } ij \text{ in its tour} \\ 0 & , \text{otherwise} \end{cases} \quad (3.74)$$

其中

$\tau_{ij}(t+1)$ ：於  $t+1$  時，由節點  $i$  至  $j$  的費洛蒙值。

$\tau_{ij}(t)$ ：在時間點  $t$  時路徑  $(i, j)$  所殘留的費洛蒙數量。

$\rho$ ：費洛蒙揮發參數， $0 < \rho \leq 1$ 。

$\Delta\tau_{ij}^k$ ：第  $k$  隻螞蟻殘留在路徑  $(i, j)$  上的每單位距離費洛蒙數量。

$Q$ ：為一個參數值，表示費洛蒙的強度，此參數一定程度上會影響到收斂速度。

$L_k$ ：為螞蟻  $k$  所完成路徑之總長度。

(四)終止條件：

為了停止螞蟻搜尋的進行，必須先定義執行的終止條件，常見的主要分為下列三種類型：

1. 限制固定的搜尋回合數，如：執行  $N$  次的迭代(Iteration)就結束，有學者建議迭代數為 100 或 1000，會得到較好的解。
2. 限制執行的時間，如：程式執行 100 秒後即停止。
3. 連續取得相同解的次數、如：連續 50 次迭代未取得更佳解相同解的次數，則視為已收斂至最佳解，並結束搜尋的任務。

### 3.3.2 蟻群系統介紹

Dorigo *et al.* (1997)提出了 ACS，ACS 承襲了先前的螞蟻系統，在方法上則改良了路徑搜尋(path search)、區域費洛蒙更新(local update pheromone)和全域費洛蒙更新(global update pheromone)三個主要的步驟，以下會針對這三點改良加以說明。

ACS 利用轉換規則，取代 AS 中的轉換機率。Dorigo and Gambardella (1997)定義在螞蟻最佳化演算法中，螞蟻在節點  $i$  選擇下一個節點  $j$  的機率選擇規則如公式(3.75)：

$$J = \begin{cases} \arg \max_{u \in J_k(i)} \{[\tau_{iu}(t)]^\alpha \times [\eta_{iu}]^\beta\}, & \text{if } q \leq q_0 \\ J & , \text{otherwise} \end{cases} \quad (3.75)$$

其中

$q$ ：為 0,1 間呈均勻分配之隨機亂數。

$q_0$ ：為一設定參數， $0 \leq q_0 \leq 1$ 。

$J$ ：為計算機率後再隨機選取節點。

$\max_{u \in J_k(i)}$ ：是指找出高費洛蒙濃度( $\tau$ )及短距離( $\eta$ )的節點  $u$ 。

當  $q \leq q_0$  時，雖然費洛蒙濃度高及距離短的節點被選擇的機率較高，但因為螞蟻仍是以隨機的方式選擇節點，因此機率最高的節點未必會被選到，螞蟻仍有選擇其他節點的可能性，此行為較偏向「探索」(exploration)，意即此行為較有可以發現新的解答。

當  $q \geq q_0$  時，螞蟻被設定必須選擇費洛蒙濃度高及距離短的節點，此行為較偏向「開發」(exploitation)，意即此行為較容易落入區域最佳解。

為了能夠得到最佳的解，螞蟻所殘留的費洛蒙濃度必須更新，反映螞蟻的表現及他們求解的品質。費洛蒙更新是 ACS 中最關鍵的一項因素，因此 ACS 費洛蒙更新有兩個部分：區域費洛蒙更新與全域費洛蒙更新。ACS 全域費洛蒙更新，當所有螞蟻都建構完可行的路徑後，便執行全域的費洛蒙更新如公式(3.76)，只會對表現最好的那一隻螞蟻才有遺留費洛蒙的權利，其餘路線沒有螞蟻行經，費洛蒙濃會蒸發，全域性更新費洛蒙的方法採用公式(3.76)：

$$\tau_{ij}(t+1) = (1 - \rho) \times \tau_{ij}(t) + \rho \Delta \tau_{ij} \quad (3.76)$$

其中：

$\rho$ ：全域費洛蒙揮發參數， $0 \leq \rho \leq 1$ 。

$\Delta \tau_{ij}$ ：所有螞蟻完成之旅程中最佳解，即最短之路徑長度。

若  $(i, j)$  屬於最佳路徑，則  $\Delta \tau_{ij} = Q/L^+$ ；否則  $\Delta \tau_{ij} = 0$ 。

ACS 與 AS 兩者最大的差異，在於 ACS 加入區域費洛蒙更新的觀念。當每一隻螞蟻選擇完下一節點時，即針對該路段作一次費洛蒙更新。這種區域更新費洛蒙的方式，是降低前一隻螞蟻所走路徑的費洛蒙含量，避免之後做路徑搜尋的螞蟻選擇相同的路徑走，陷入局部解，區域更新費洛蒙如公式(3.77)：

$$\tau_{ij}(t+1) = (1 - \rho) \times \tau_{ij}(t) + \rho \tau_0 \quad (3.77)$$

$\rho$ ：全域費洛蒙揮發參數， $0 \leq \rho \leq 1$ 。

$\tau_0$ ：為起始費洛蒙濃度，通常設定 $\tau_0 = (N \times L_{nn})^{-1}$ ， $N$ 為總節點數， $L_{nn}$ 是以鄰近搜尋法所求解之總路徑長。

### 3.3.3 ACS 求解 ELSP 問題

本節將介紹 ACS 應用於 ELSP 倉儲空間與總成本(包括租賃成本)的求解步驟。在 ELSP 倉儲空間與總成本中，螞蟻走過的節點代表生產的產品，而走過的路徑(即節點順序)可以視為所有產品的生產順序。找尋出哪隻螞蟻走過的路徑之總成本最低，即可視為本研究的最佳解。

應用 ACS 求解 ELSP 的倉儲空間與總成本之流程說明如下(如圖 3-17 所示)：

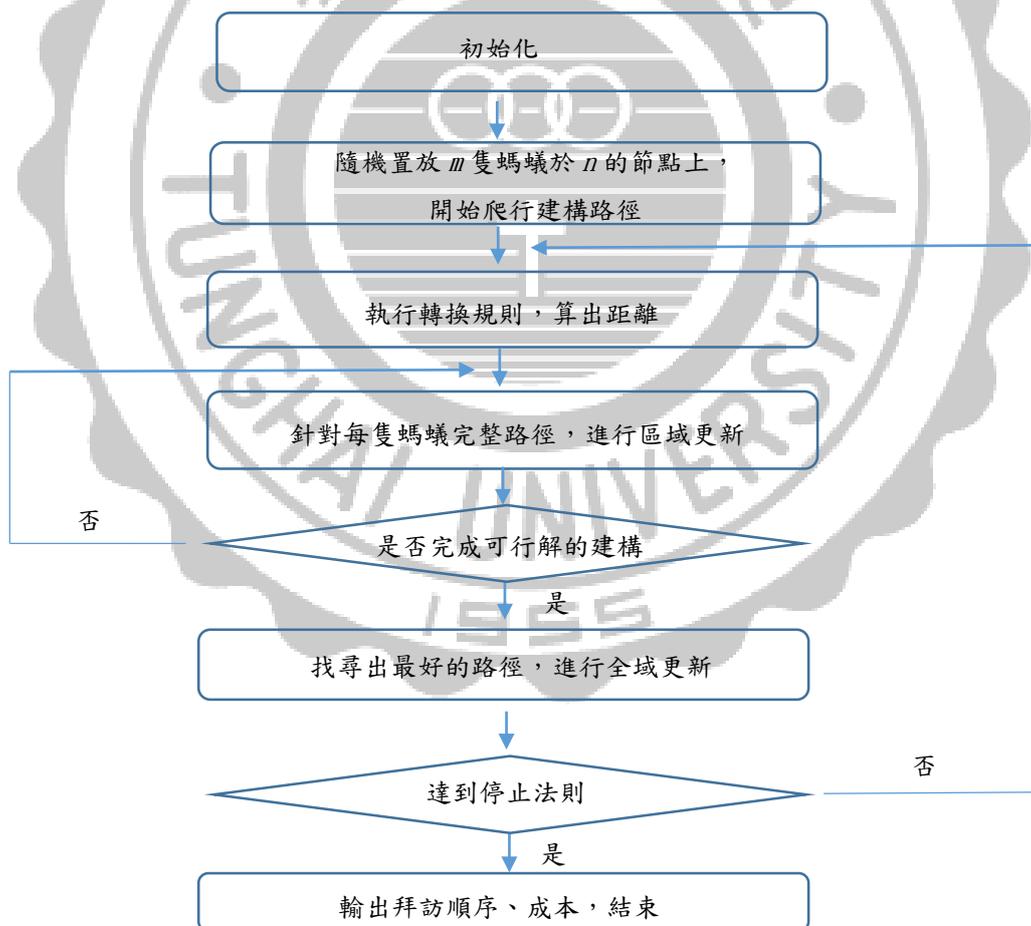


圖 3-17 ACS 流程圖

1. 設定初始化參數：

在蟻群系統的初始化中，必須設置初始化參數值，費洛蒙濃度出始值 $\tau_0$ 、決定費洛蒙濃度權重參數 $\alpha$ 、絕對距離參數 $\beta$ 、費洛蒙揮發參數 $\rho$ 、人工螞蟻數目  $m$ ，設定起始疊代次數  $t=0$ 。這些參數都是經驗參數，可以經由調整這些參數，使演算法能夠得到較好的解。本研究以總成本(即公式(3.67))當成螞蟻走過的距離。

2. 建構路徑：

一開始隨機置放  $m$  隻螞蟻於  $n$  個節點上，螞蟻開始爬行建構路徑。

3. 執行轉換規則：

將  $m$  隻螞蟻隨機放置在節點上，每隻螞蟻從一開始放置位置出發，計算轉換規則，以便選擇下一個節點、逐步完成一趟完整的路徑，下面會介紹螞蟻如何選擇下一個節點，如圖(3-18)所示。

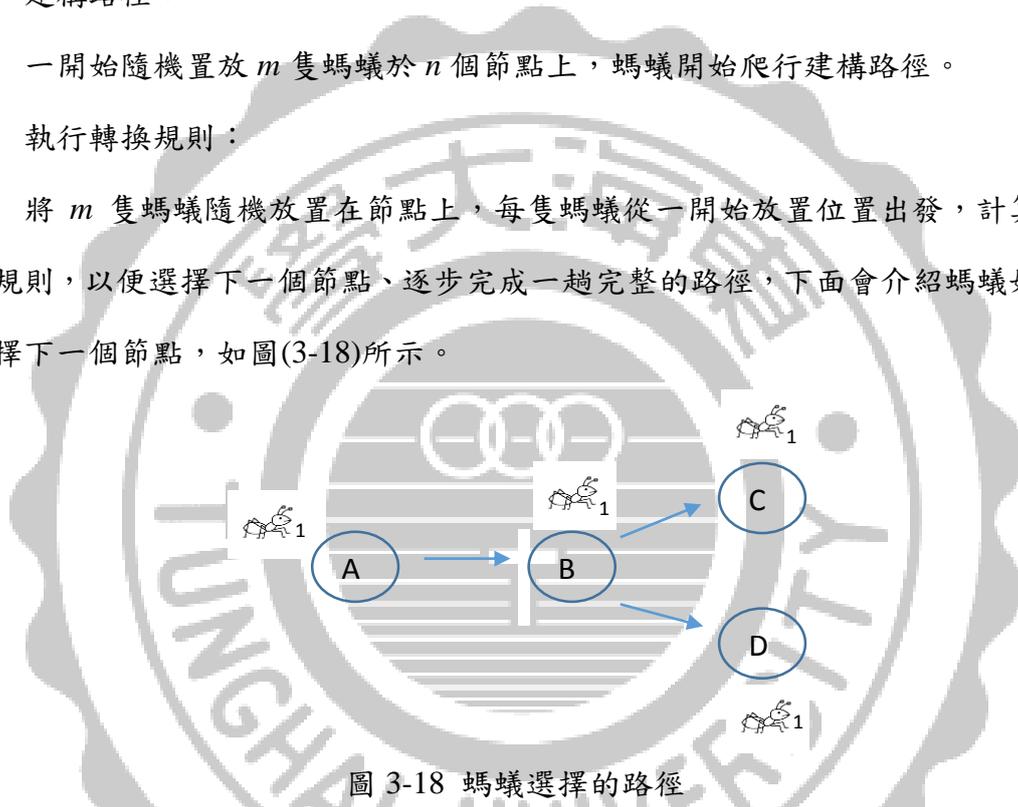


圖 3-18 螞蟻選擇的路徑

當一隻螞蟻放在 A 節點時，走的路徑可能為(A,B,C)或(A,B,D)兩種選擇，在選擇下一個節點前要先求出節點間的距離，再依照轉換規則選取下一個節點。一螞蟻走過路徑為(A,B,C)時，即可視為此螞蟻找到的產品生產順序為(A,B,C)。當螞蟻走的路徑為(A,B,C)時(B, C)之距離的計算方式為走過(A,B,C)路徑之總成本減掉走過 (A, B) 路徑之總成本得到的解為(B, C)兩節點之成本(即可視為節點 B 和 C 之間的距離)，計算方式如公式(3.78)所示。螞蟻走的路徑為(A,B,D)時，(B, D)之距離的計算方式為走過(A,B,D)路徑之總成本減掉走過 (A,B) 路徑之總成本得到的解為(B, D)兩節點之成本(即可視為節點 B 和 D 之間的距離)。

$$\begin{aligned}
MinTC_{(B,D)} = & \left\{ \sum_{i \in \{A,B,D\}} \left[ \frac{A_i}{T} + \frac{h_i d_i (1 - \rho_i) (T - t_i)^2}{2T} \right] + (W - w)\alpha \right\} \\
& - \left\{ \sum_{i \in \{A,B\}} \left[ \frac{A_i}{T} + \frac{h_i d_i (1 - \rho_i) (T - t_i)^2}{2T} \right] + (W - w)\alpha \right\} \quad (3.78)
\end{aligned}$$

由於螞蟻走過節點 B 和 C 的距離可以視為先生產產品 B 再生產產品 C 的成本，因此這裡將說明如何計算生產所有產品的總成本：當已知一組生產順序的情況下，依序挑選出一產品做降速，並求出平均總成本。而以共同週期法運用彈性速率求解 ELSP 的倉儲空間與平均總成本的方法如下：

Step1 由公式(3.69)求解  $T$ 。藉由公式(3.68)檢查產品的生產產能限制式，測試出  $T$  的結果是否為可行，若為可行，則執行 Step2；若測出的  $T$  為不可行，則無可行解的存在。

Step2 由公式(3.71)求解閒置時間  $I$ 。

Step3 本研究以  $d_i h_i$  最大的產品進行降速，並令  $k$  等於此產品的代號( $=i$ )。

Step4 用公式(3.70)求解產品  $k$  的  $t_i$  值，產品  $k$  會以生產速率  $d_i$  生產一段時間(即  $t_i$ )後，再以  $p_i$  的生產速率繼續生產；其他不變更生產速率的產品，則以原生產速率  $p_i$  生產。

Step5 根據產品順序 Seq1 與 Seq2 的限制，找出相對生產的順序和總成本。

Step5.1 此生產順序若符合 Seq1，則將公式(3.69)所算出的  $T$  代入公式(3.21)，計算倉儲大小，令  $W = W_1$ ，並計算出相對的總成本，計算出倉儲大小  $W_1$  代入公式(3.67)計算出總成本，若未符合此限制的生產順序，則總成本會無限大。

Step5.2 此生產順序若符合 Seq2，則將公式(3.69)所算出的  $T$  代入公式(3.66)，計算倉儲大小，令  $W = W_2$ ，並計算出相對的總成本，計算出倉儲大小  $W_2$  代入公式(3.67)計算出總成本，若未符合此限制的生產順序，則總成本會無限大。

Step6 比照 Seq1 和 Seq2 之總成本,取小者為該順序的最佳成本,執行完畢後,輸出平均總成本(即路徑距離)。

當求出路徑距離後,依照轉換規則公式(3.75)進行選擇下一個節點,當每隻螞蟻走完節點時,建立成一個路徑時,將對每隻螞蟻走過的路徑進行區域更新。

#### 4. 區域更新：

區域更新的目的是為了讓後續搜尋的螞蟻能夠往不同的方向搜尋,當每隻螞蟻建立一路徑時,在螞蟻走過路徑上的費洛蒙依照區域更新法則更新,目的為了找出更多不同的解,區域更新根據公式(3.77)所示。

#### 5. 是否完成可行解的建構

查看每隻螞蟻是否都完成路徑的建構,若螞蟻未完成路徑建構,則回到步驟 3 繼續爬行,直到所有螞蟻皆完成路徑。

#### 6. 進行全域更新：

當  $m$  之螞蟻反覆搜尋完一次稱為一迭代,一迭代之後進行全域更新,在全域更新中,選擇最好的路徑加強費洛蒙的強度,全域更新公式如(3.76)。

#### 7. 停止條件：

若演算法的執行符合以下某個停止條件就停止執行。停止條件有以下三個：一般來說,分為執行 100 次的迭代(Iteration)就結束、限制執行 100 秒或連續 50 代未取得更佳解。

#### 8. 輸出結果：

而在本系統中,每隻螞蟻完成後挑選路徑長度最短的路徑(最低總成本)則視為本研究最佳解。本研究目標為找到最小總成本,因此經由  $N$  個迭代後可找出最好的那隻螞蟻走的路徑距離(總成本)及走過路徑的順序,最後輸出最佳順序與總成本。

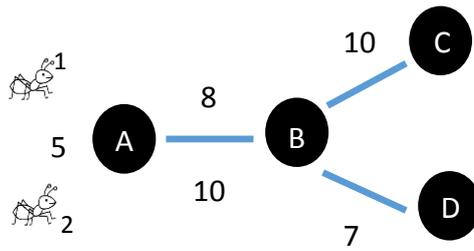


圖 3-19 螞蟻走過的路徑距離

ACS 在求解順序性問題時，如求解 TSP 問題，假設兩節點間的路徑距離是固定的。但由於本研究在螞蟻走過的路徑上會挑選  $d_{ij}$  最大的一產品來進行降速再計算螞蟻走過路徑的距離(成本)，如圖 3-19 所示，當螞蟻 1 選擇走的路徑為 (A,B,C) 時，三個點中 B 節點的  $d_{ij}$  最大，因此選擇 B 節點進行降速，此時 (A,B,C) 中的 (A,B) 距離為 8；當螞蟻 2 選擇走的路徑為 (A,B,D) 時，三個點中 D 節點的  $d_{ij}$  最大，因此選擇 D 節點進行降速，此時 (A,B,D) 中的 (A,B) 距離為 10。這時會發現到兩隻螞蟻同樣走 (A,B) 兩節點，但兩隻螞蟻走過的距離卻不一樣。因此本研究找尋許多相關文獻研究後發現，在 ACS 的演算過程中，螞蟻走過所有節點的總距離主要是用來更新各路徑上費洛蒙含量(如公式(3.76)所示)。而所有 ACS 都是用總距離，而非節點  $i$  和  $j$  之間的距離，來更新各路徑上的費洛蒙含量。由此來看，對兩隻走過不同路徑((A,B,C)和(A,B,D))的螞蟻來說，節點  $i$  和  $j$  之間的距離分別是 X 和 Y，並不影響本研究運用螞蟻走過所有節點的總距離來更新各路徑上費洛蒙含量之計算過程。

而當節點(A, B)為最佳解路徑的一部分時，不管每隻螞蟻走過(A, B)的路徑長度為多少，還是會有許多螞蟻會挑選(A, B)這條路徑行走，使得此路徑上的費洛蒙濃度會持續增加；因此不會因為每隻螞蟻走過(A, B)路徑的距離不同，而影響到最後的最佳解。

### 3.4 小結

本研究在產品倉儲可互換的情況下，以共同週期法運用彈性速率建立一個新

模型，並用這模式進行 ELSP 下的倉儲空間的計算與平均總成本的求解問題。本研究  
研究的成本項中除了存貨持有成本還加上了倉儲的租賃成本，運用 ACS 演算法  
找尋出最佳生產順序，在找尋最佳生產順序中，並以  $Max\{d_i h_i\}$  最大的產品作為  
降速的產品，找出產品最佳的生產順序，並算出最佳的倉儲空間與總成本。



## 第四章 數值範例與數據實驗

本章節分成兩個部分，為了瞭解倉庫的租金成本、持有成本和產品需求率對目標成本的影響，本研究參考林禎樺(2014)提出的數值範例進行修改。第一節先以八組數值範例(numerical example)介紹實際操作情形，本研究將每組數據皆列出共同週期法運用固定速率法和共同週期法運用彈性速率法兩種方法求解的改善結果，再以表格方式比較兩者的結果加以說明。第二節以隨機實驗(random experiment)說明 ACS 演算法求解效率。本研究的所有案例之成本單位皆以「千元」為單位；倉儲空間大小的單位皆以「儲位」為單位；而時間皆以「天」為單位。本研究的程式使用 Eclipse 來撰寫 Java 程式的求解方法，並於 Lenovo E420s，CPU：Intel Core i5-2410M 2.30GHz，RAM：8.00GB DDR III 的環境下執行。

### 4.1 數值範例

本節提出八組數值範例說明在共同週期法下應用彈性速率法求解倉儲空間和倉儲租金成本的改善程度。第一組範例為倉儲租金成本為 0.1( $\alpha = 0.1$ )的情況下，考慮 5 個產品的問題，目的是用來示範 ACS 演算法的求解過程與品質。以下列舉第一範例的五個產品的參數設定，相關參數如表 4-1 所示。

表 4-1 第一組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350	15	0.05	0.1	0.1
2	4000	400	30	0.01		0.1
3	6350	635	50	0.45		0.1
4	7000	700	20	0.03		0.1
5	15000	1500	150	0.01		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-2 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-3 所示；此範例的求解結果比較表，如表 4-4 所示。

表 4-2 以共同週期法運用固定速率法求解第一組的結果

最佳排列順序	5	4	2	3	1
最佳 W(儲位)	1523				
平均總成本(千元)	534.76				
執行時間(秒)	5				

表 4-2 的倉儲空間大小求解結果為 1523 儲位，而平均總成本的求解結果為 534.76 千元，執行時間為 5 秒。

表 4-3 以共同週期法運用彈性速率法求解第一組的結果

最佳排列順序	5	4	2	3	1
最佳 W(儲位)	1523				
平均總成本(千元)	515.54				
執行時間(秒)	5				

表 4-3 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,3,1)，產品三的 $d_i h_i$ 最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 1523 儲位，而平均總成本的求解結果為 515.54 千元，執行時間為 5 秒。

表 4-4 彈性速率法與固定速率法求解第一組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	1523	1523	0.00%
平均總成本(千元)	534.76	515.54	3.59%
執行時間(秒)	5	5	—

彈性速率法在倉儲大小與成本的改善程度之算法分別如公式(4.1)以及公式(4.2)所示。倉儲的空間的改善程度，為共同週期法下運用固定速率法求得的最佳倉儲空間大小( $W_{RC}$ )做為與共同週期法下運用彈性速率法求解倉儲空間大小( $W_{BB}$ )的比較基準。總成本改善程度為固定速率法求得的最佳平均單位時間總成本( $TC_{RC}$ )做為與彈性速率法下求解成本( $TC_{BB}$ )的比較基準。

$$\text{倉儲空間改善程度} = \frac{W_{RC} - W_{BB}}{W_{RC}} \times 100\% \quad (4.1)$$

$$\text{總成本改善程度} = \frac{TC_{RC} - TC_{BB}}{TC_{RC}} \times 100\% \quad (4.2)$$

從表 4-4 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 3.59%，而倉儲空間大小並沒有改變。

以下列舉第二組範例的 5 個產品的參數設定，第二組範例與第一組範例相似，為了瞭解在倉庫的租金成本較高( $\alpha=0.8$ )的情況下，本研究演算法的求解效果，因此倉庫的租金成本調高，相關參數如表 4-5 所示。

表 4-5 第二組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$s_i$
1	3500	350	15	0.05	0.8	0.1
2	4000	400	30	0.01		0.1
3	6350	635	50	0.45		0.1
4	7000	700	20	0.03		0.1
5	15000	1500	150	0.01		0.1

以下列出第二組範例在共同週期法下運用固定速率法求解的結果，如表 4-6 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-7 所示；此範例的求解結果比較表，如表 4-8 所示。

表 4-6 以共同週期法運用固定速率法求解第二組的結果

最佳排列順序	5	4	2	3	1
最佳 W(儲位)	1523				
平均總成本(千元)	1600.57				
執行時間(秒)	4				

表 4-6 的倉儲空間大小求解結果為 1523 儲位，而平均總成本的求解結果為 1600.57 千元，執行時間為 4 秒。

表 4-7 以共同週期法運用彈性速率法求解第二組的結果

最佳排列順序	5	4	2	3	1
最佳 W(儲位)	1523				
平均總成本(千元)	1581.35				
執行時間(秒)	3				

表 4-7 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,3,1)，產品三的 $d_i h_i$ 最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 1523 儲位，而平均總成本的求解結果為 1581.35 千元，執行時間為 3 秒。

表 4-8 彈性速率法與固定速率法求解第二組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	1523	1523	0.00%
平均總成本(千元)	1600.57	1581.35	1.20%
執行時間(秒)	4	3	—

從表 4-8 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 1.20%，而倉儲空間大小並沒有改善。

以下舉例第三組範例的 5 個產品的參數設定，第三範例與第一組範例相似，但為了比較產品的存貨持有成本( $h_i/240$ )改變是否會成為影響的因素，倉庫的租金成本維持不變( $\alpha=0.1$ )、只調整了持有成本 $h_i$ ，相關參數如表 4-9 所示。

表 4-9 第三組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$s_i$
1	3500	350	15	0.05/240	0.1	0.1
2	4000	400	30	0.01/240		0.1
3	6350	635	50	0.45/240		0.1
4	7000	700	20	0.03/240		0.1
5	15000	1500	150	0.01/240		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-10 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-11 所示；此範例的求解結果比較表，如表 4-12 所示。

表 4-10 以共同週期法運用固定速率法求解第三組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	11196				
平均總成本(千元)	1139.98				
執行時間(秒)	5				

表 4-10 的倉儲空間大小求解結果為 11196 儲位，而平均總成本的求解結果為 1139.98 千元，執行時間為 5 秒。

表 4-11 以共同週期法運用彈性速率法求解第三組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	11196				
平均總成本(千元)	1137.26				
執行時間(秒)	3				

表 4-11 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的  $d_i h_i$  最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 11196 儲位，而平均總成本的求解結果為 1137.26 千元，執行時間為 5 秒。

表 4-12 彈性速率法與固定速率法求解第三組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	11196	11196	0.00%
平均總成本(千元)	1139.98	1137.26	0.24%
執行時間(秒)	5	3	—

從表 4-12 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並

透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 0.24%，而倉儲空間大小並沒有改善。

以下舉例第四組範例的五個產品的參數設定，第四組範例的目的與第三組範例相似，為了比較持有成本( $h_i/240$ )變小的情況下倉儲的租金成本提高( $\alpha=0.8$ )時，是否會成為影響的因素，相關參數如表 4-13 所示。

表 4-13 第四組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350	15	0.05/240	0.8	0.1
2	4000	400	30	0.01/240		0.1
3	6350	635	50	0.45/240		0.1
4	7000	700	20	0.03/240		0.1
5	15000	1500	150	0.01/240		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-14 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-15 所示；此範例的求解結果比較表，如表 4-16 所示。

表 4-14 以共同週期法運用固定速率法求解第四組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	11196				
平均總成本(千元)	8977.13				
執行時間(秒)	5				

表 4-14 的倉儲空間大小求解結果為 11196 儲位，而平均總成本的求解結果 8977.13 千元，執行時間為 5 秒。

表 4-15 以共同週期法運用彈性速率法求解第四組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	11196				
平均總成本(千元)	8974.42				
執行時間(秒)	3				

表 4-13 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的 $d_i h_i$ 最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 11196 儲位，而平均總成本的求解結果為 8974.42 千元，執行時間為 5 秒。

表 4-16 彈性速率法與固定速率法求解第四組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	11196	11196	0.00%
平均總成本(千元)	8977.13	8974.42	0.03%
執行時間(秒)	5	3	—

從表 4-16 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 0.03%，而倉儲空間大小並沒有改善。

以下舉例第五組範例的 5 個產品的參數設定，第五組範例的目的與第一組範例相似，為了比較產品需求率減半( $d_i/2$ )的情況下是否會成為影響的因素，相關參數如表 4-17 所示。

表 4-17 第五組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350/2	15	0.05	0.1	0.1
2	4000	400/2	30	0.01		0.1
3	6350	635/2	50	0.45		0.1
4	7000	700/2	20	0.03		0.1
5	15000	1500/2	150	0.01		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-18 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-19 所示；此範例的求解結果比較表，如表 4-20 所示。

表 4-18 以共同週期法運用固定速率法求解第五組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	663				
平均總成本(千元)	299.21				
執行時間(秒)	5				

表 4-18 的倉儲空間大小求解結果為 663 儲位，而平均總成本的求解結果為 299.21 千元，執行時間為 5 秒。

表 4-19 以共同週期法運用彈性速率法求解第五組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	663				
平均總成本(千元)	268.65				
執行時間(秒)	4				

表 4-19 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的  $d_i/h_i$  最大，本研究以產品三進行降速，倉儲空間大小求解結果為 663 儲位，而平均總成本的求解結果為 268.65 千元，執行時間為 5 秒。

表 4-20 彈性速率法與固定速率法求解第五組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	663	663	0.00%
平均總成本(千元)	299.21	268.65	10.21%
執行時間(秒)	5	4	—

從表 4-20 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並

透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 10.21%，而倉儲空間大小並沒有改善。

以下舉例第六組範例的 5 個產品的參數設定，第六組範例的目的與第五組範例相似，為了比較產品需求率減半( $d_i/2$ )和倉儲的租金成本又提高時( $\alpha = 0.8$ )是否會成為影響的因素，相關參數如表 4-21 所示。

表 4-21 第六組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350/2	15	0.05	0.8	0.1
2	4000	400/2	30	0.01		0.1
3	6350	635/2	50	0.45		0.1
4	7000	700/2	20	0.03		0.1
5	15000	1500/2	150	0.01		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-22 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-23 所示；此範例的求解結果比較表，如表 4-24 所示。

表 4-22 以共同週期法運用固定速率法求解第六組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	663				
平均總成本(千元)	763.23				
執行時間(秒)	4				

表 4-22 的倉儲空間大小求解結果為 663 儲位，而平均總成本的求解結果為 763.23 千元，執行時間為 4 秒。

表 4-23 以共同週期法運用彈性速率法求解第六組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	663				
平均總成本(千元)	732.67				
執行時間(秒)	3				

表 4-23 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的  $d_i h_i$  最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 663 儲位，而平均總成本的求解結果為 732.67 千元，執行時間為 5 秒。

表 4-24 彈性速率法與固定速率法求解第六組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	663	663	0.00%
平均總成本(千元)	763.23	732.67	4.00%
執行時間(秒)	4	3	—

從表 4-24 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 4.00%，而倉儲空間大小並沒有改善。

以下舉例第七組範例的 5 個產品的參數設定，第七組範例的目的與第三組範例相似，為了比較產品需求率減半( $d_i/2$ )的情況下是否會成為影響的因素，相關參數如表 4-25 所示。

表 4-25 第七組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350/2	15	0.05/240	0.1	0.1
2	4000	400/2	30	0.01/240		0.1
3	6350	635/2	50	0.45/240		0.1
4	7000	700/2	20	0.03/240		0.1
5	15000	1500/2	150	0.01/240		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-26 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-27 所示；此範例的求解結果比較表，如表 4-28 所示。

表 4-26 以共同週期法運用固定速率法求解第七組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	3988				
平均總成本(千元)	411.60				
執行時間(秒)	5				

表 4-26 的倉儲空間大小求解結果為 3988 儲位，而平均總成本的求解結果為 411.60 千元，執行時間為 5 秒。

表 4-27 以共同週期法運用彈性速率法求解第七組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	3988				
平均總成本(千元)	410.20				
執行時間(秒)	3				

表 4-27 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的  $d_i h_i$  最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 3988 儲位，而平均總成本的求解結果為 410.20 千元，執行時間為 5 秒。

表 4-28 彈性速率法與固定速率法求解第七組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	3988	3988	0.00%
平均總成本(千元)	411.60	410.20	0.33%
執行時間(秒)	5	3	—

從表 4-28 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並

透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 0.33%，而倉儲空間大小並沒有改善。

以下舉例第八組範例的 5 個產品的參數設定，第八組範例的目的與第七組範例相似，為了比較存貨持有成本( $h_i/240$ )與產品需求率( $d_i/2$ )不變的情況下，提高倉儲的租金成成本是否會成為影響的因素，相關參數如表 4-13 所示。

表 4-29 第八組範例的 5 個產品參數資料

產品	$p_i$	$d_i$	$A_i$	$h_i$	$\alpha$	$S_i$
1	3500	350/2	15	0.05/240	0.8	0.1
2	4000	400/2	30	0.01/240		0.1
3	6350	635/2	50	0.45/240		0.1
4	7000	700/2	20	0.03/240		0.1
5	15000	1500/2	150	0.01/240		0.1

以下列出第一組範例在共同週期法下運用固定速率法求解的結果，如表 4-30 所示；在共同週期法下運用彈性速率法求解的結果，如表 4-31 所示；此範例的求解結果比較表，如表 4-32 所示。

表 4-30 以共同週期法運用固定速率法求解第八組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	3988				
平均總成本(千元)	3202.62				
執行時間(秒)	5				

表 4-30 的倉儲空間大小求解結果為 3988 儲位，而平均總成本的求解結果 3202.62 千元，執行時間為 5 秒。

表 4-31 以共同週期法運用彈性速率法求解第八組的結果

最佳排列順序	5	4	2	1	3
最佳 W(儲位)	3988				
平均總成本(千元)	3201.22				
執行時間(秒)	4				

表 4-31 顯示 ACS 演算法搜尋出最佳排列順序(5,4,2,1,3)，產品三的  $d_i h_i$  最大，本研究以產品 3 進行降速，倉儲空間大小求解結果為 3988 儲位，而平均總成本的求解結果為 3201.22 千元，執行時間為 5 秒。

表 4-32 彈性速率法與固定速率法求解第八組結果比較

方法	共同週期法下固定速率法	共同週期法下彈性速率法	改善程度(%)
最佳 W(儲位)	3988	3988	0.00%
平均總成本(千元)	3202.62	3201.22	0.04%
執行時間(秒)	5	4	—

從表 4-32 顯示在共同週期法下運用彈性速率法進行求解，透過生產週期中若有多餘的閒置時間，適度分配於降速產品，再經由本研究的 ACS 演算法，並透過降速產品的挑選和可行解的判斷後，與共同週期法下運用固定速率法做比較，平均單位時間總成本的改善程度為 0.04%，而倉儲空間大小並沒有改善。

接下來說明產品數據的改變如產品的需求率( $d_i$ )、存貨持有成本( $h_i$ )或倉儲的租金成本( $\alpha$ )改變時，是否會成為影響的因素，因此本研究以範例數據來做比較加以說明。

表 4-33 第一組範例與第二組範例比較表

方法	共同週期法下固定速率法總成本(千元)	共同週期法下彈性速率法總成本(千元)	改善程度(%)
第一組範例 ( $\alpha = 0.1$ )	534.76	515.54	3.59%
第二組範例 ( $\alpha = 0.8$ )	1600.57	1581.35	1.20%

第二組範例與第一組範例相似，當倉儲空間的成本從 $\alpha = 0.1$ 調高至 $\alpha = 0.8$ 時，可以發現共同週期法下彈性速率法不管在第一組範例或第二組範例中，總成本都優於共同週期法下固定速率法，由於倉儲的租金成本提高，總成本也會變高，因此第二組範例的改善程度沒有第一組範例的改善程度明顯。

表 4-34 第三組範例與第一組範例比較表

方法	共同週期法下固定速率法總成本(千元)	共同週期法下彈性速率法總成本(千元)	改善程度(%)
第一組範例 ( $h_i$ )	534.76	515.54	3.59%
第三組範例 ( $h_i/240$ )	1139.98	1137.26	0.24%

第三組範例與第一組範例相似，當倉儲空間的成本皆為 $\alpha = 0.1$ 時，調整了存貨持有成本( $h_i$ )將存貨持有成本減少 240 倍，可以發現共同週期法下彈性速率法不管在第一組範例或第三組範例中，總成本都優於共同週期法下固定速率法，當存貨持有成本減少 240 倍，存貨持有成本過於小看不太出改善的效果，因此改善的程度極為小。

表 4-35 第五組範例與第一組範例比較表

方法	共同週期法下固定速率法總成本(千元)	共同週期法下彈性速率法總成本(千元)	改善程度(%)
第一組範例 ( $d_i$ )	534.76	515.54	3.59%
第五組範例 ( $d_i/2$ )	299.21	268.65	10.21%

第五組範例與第一組範例相似，當產品需求率減半時( $d_i/2$ )，倉儲空間的成本為( $\alpha = 0.1$ )與存貨持有成本 ( $h_i$ ) 不變時，可以發現共同週期法下彈性速率法不管在第一組範例或第五組範例中，總成本都優於共同週期法下固定速率法，由於需求率減半，有更多的閒置時間去生產降速的產品，總成本會跟著降低，因此第五組範例的改善程度明顯高於第一組範例。

## 4.2 隨機實驗

此章節將以大量的數據來測試演算法的效能和使用彈性速率的改善程度。Elmaghraby (1978) 和 Moon *et al.* (2002) 指出機台產能使用率(公式 4.3)的高低可以反映 ELSP 的困難度，產能使用率越高時，求解的困難度越高。而由於本研究考慮降低生產速率的模式，本研究考量的兩個實驗變數，一個是產能利用率，另一個是產品數量。Elmaghraby (1978) 和 Moon *et al.* (2002) 提出的產能使用率計算方式，因為沒有考量整備時間的因素，除了無法具體反應出真實使用率，也無法反應出本研究求解 ELSP 案例的閒置時間的多寡；因此本研究將產能利用率將改寫為公式(4.4)，可以真實反應閒置時間的多寡，並將產能利用率的區間設為 0.5 至 0.9，每隔 0.1 劃分為一個區間；產品的數目有 6 類，分別為 5、10、15、20、25 和 30。每種產品數在不同的產能利用率水準下會隨機產生 20 個案例。記錄每一個問題找到的最佳解及所花費的時間，最後將每種產品數在不同產能利用率水準之 20 個問題的求解表現資料求平均值，來比較所找到解的品質及所花費的時間。

本研究參考 Carreno (1990) 提出的產品參數值域來產生隨機案例。由於原資料中整備時間設定較高，當產品數量過多時，容易因為所有整備時間占用產能的比例過高，使解成為不可行解，或是讓降速的效果變差。因此本研究將其平均值調整為 0.05，範圍也調整為 0.05。本研究的產品資料的範圍如表 4-36 所示。產品資料的數值將會介於平均值±(範圍/2)之間。

$$\sum_{i=1}^n \frac{d_i}{p_i} \quad (4.3)$$

$$f_{UF} = \frac{\sum_{i=1}^n (S_i + \frac{d_i(T)}{p_i})}{T} \quad (4.4)$$

表 4-36 產品參數資料範圍

	平均值	範圍
生產整備成本	200.0	400.0
持有成本	0.35	0.70
需求率	2500.0	4800.0
生產率	14000.0	5000.0
生產整備時間	0.05	0.05
倉庫租金成本	0.15	0.05

(資料來源：修改自 Carreno (1990))

本研究將共同週期法下運用固定速率法求得的解做為成本的比較基準 ( $TC_{RC}$ )，與彈性速率法所求得成本 ( $TC_{BB}$ ) 做比較。Max(%) 表示該求解方法在 20 個問題中最小的改善程度；Min(%) 表示該求解方法在 20 個問題中最大的改善程度；Avg(%) 表示該求解方法在 20 個問題中平均的改善程度，以彈性速率法為例，其平均改善程度如公式(4.5)所示。

$$Avg(\%) = \frac{\sum_{i=1}^{20} \left( \frac{TC_{BB} - TC_{RC}}{TC_{BB}} \right) \times 100\%}{20}$$

(4.5)

表 4-37 在共同週期法下彈性速率法求解 ELSP 的表現

產品數目	產能利用率	彈性速率			
		Min(%)	Max(%)	Avg(%)	Time
5	[0.5-0.6)	-39.70	-21.09	-31.21	3.05
	[0.6-0.7)	-41.84	-23.85	-33.48	3.24
	[0.7-0.8)	-40.41	-20.98	-30.79	3.30
	[0.8-0.9)	-40.45	-20.18	-31.38	3.35
10	[0.5-0.6)	-40.00	-20.75	-28.92	7.95
	[0.6-0.7)	-38.38	-7.89	-27.40	7.55
	[0.7-0.8)	-39.55	-16.46	-30.183	9.45
	[0.8-0.9)	-40.471	-18.36	-31.14	9.05
15	[0.5-0.6)	-36.23	-0.815	-20.148	12.00
	[0.6-0.7)	-41.67	-4.96	-29.09	14.00
	[0.7-0.8)	-40.29	-12.06	-27.20	11.55
	[0.8-0.9)	-40.93	-9.15	-29.66	14.12
20	[0.5-0.6)	-32.46	-0.78	-21.34	18.90
	[0.6-0.7)	-38.88	0.07	-21.57	16.95
	[0.7-0.8)	-38.62	-13.24	-25.03	16.70
	[0.8-0.9)	-38.52	-14.43	-27.18	17.35
25	[0.5-0.6)	-37.33	1.599	-21.54	21.45
	[0.6-0.7)	-41.34	-21.15	-21.15	22.5
	[0.7-0.8)	-39.53	-6.735	-22.81	25.65
	[0.8-0.9)	-38.56	-18.05	-27.72	23.34
30	[0.5-0.6)	-33.66	5.73	-16.85	23.50
	[0.6-0.7)	-40.68	31.085	-20.37	25.15
	[0.7-0.8)	-38.92	-2.75	-26.54	25.27
	[0.8-0.9)	-38.65	-2.99	-27.58	25.95

表 4-37 為實驗數據的結果，在表中可以發現，當產品數量越多時，可以發現平均的執行時間較長，而固定速率法平均執行時間，比彈性速率法來的短，這表示固定速率法使用基因演算法時有快速收斂的特性，因此較容易掉入區域最佳解；當產品數目越多，平均改善程度越低，但 ACS 演算法在 30 個產品的時候，可以發現到產品利用率 0.5 到 0.7 跑出來的解比固定速率差，因此 ACS 在產品數

量多時，可能搜尋出一兩個案例比固定速率法來的差，但還是有明顯的改善程度。因此 ACS 在產品數量多時，可能會無法達到預期的效果。以往在產能利用率越高及產品數量越多時，機器閒置時間會相對較少，因此，將閒置時間分配於一產品進行降速生產的速率降低程度有限，而導致降速後的存貨持有成本減少程度不高，平均總成本的改善程度就較低。但本研究發現，當相同的產品數量時，產能利用率越高時，改善的程度較高，因此未來我們可以針對這現象進行探討。整體看來，在共同週期法下應用彈性速率法，能節省一定程度的成本，在產能利用率不高的情況下能有 31.21% 的改善程度，而在產品多達 30 個時，也能保持 27.58% 的改善程度。

### 4.3 小結

從本研究的八組數據範例發現，當租金成本、產品需求率和存貨持有成本有所變動時，共同週期法下運用彈性速率法的總成本都比共同週期法下運用固定速率法的總成本來得佳；在隨機實驗裡我們可以發現，產品數量越多時，平均改善程度越低。

## 第五章 結論與未來研究方向

### 5.1 結論

本研究建立在共同週期法下運用彈性速率法求解經濟批量排程問題，並利用 ACS 演算法進行求解。該題目為一個全新的問題，從文獻中可以發現已討論 ELSP 和倉儲問題的研究中，並沒有學者探討在共同週期法下運用彈性速率法變更生產速率而本研究更進一步探討產品的生產順序，本研究認為，一生產週期中若有多餘閒置的時間，適當的分配到某項產品，以變更生產速率可以減少存貨持有成本以及倉儲的租賃成本。

本研究將倉庫的租賃成本加入 ELSP 目標成本函式，並在變更某項產品的生產速率後，依據  $d_i < p_{(i+1)} - d_{(i+1)}$  這個數學模式排出生產順序。本研究經由數值範例的驗證，顯示在生產過程中變更生產速率（以低於最大速率的方式來生產）後，不同的生產順序下所得到的單位時間平均總成本也不一樣。

而不同於以往的搜尋演算法，本研究也首次嘗試以蟻群系統演算法來求解經濟批量排程問題，利用開發與探索的特性，找到最佳的產品生產順序，進而得到最低總成本。而在進行大量隨機實驗後，從結果顯示本研究提出的共同週期法下應用彈性速率法的模式，能得到較以往更低的總成本，而降低存貨持有成本以及倉儲的租賃成本，使排程轉為可行解的方式也確實可行。

### 5.2 未來研究方向

針對本研究有所不足的地方，提出以下幾點未來研究方向：

1. 本研究在共同週期法下運用變更速率法發現不同生產順序能得到較佳的解，因此未來可以加入在基本週期法或延伸基本週期法上運用生產速率變更後的生產順序之變化，並比較其優劣。

2. 經由數據實驗發現，倉儲空間的改善程度並沒有達到預期效果；倉儲空間若能大幅降低，對於總成本的改善將有重大的影響，希望未來能對此點進行探討。
3. 本研究將每個產品的倉儲租金成本設為一致，未來可以每個產品的倉儲租賃成本可以不同，因此可以探討降速的產品可以不以 $Max\{d_{ih}\}$ (如租賃成本高昂)的產品進行降速做考量。
4. 經濟批量排程問題中，尚有加入檢驗的經濟批量檢驗與排程問題(Economic Lot Inspection and Scheduling Problem, ELISP)與有重製狀況的經濟批量排程問題(Economic Lot Scheduling Problem with Reworks, ELSPRw)等相關延伸題目，可應用本研究對這些題目來進行倉儲空間與成本探討。
5. 在正常的情況下，大多數的存貨持有過程中，會隨著時間導致存貨發生變質現象，存貨的價值和數量就因而減少。在未來研究中可加入保存期限的議題，來應用倉儲空間與成本的探討。
6. 由於目前在 ELSP 倉儲延伸研究中，還沒有公用的數據來做標準，Bomberger(1966)在 ELSP 領域中提出六個公開範例數據，未來我們將修改自 Bomberger 的六組範例數據並使用分枝界限法找到全部的可能解，此數據未來能在 ELSP 的倉儲延伸研究中當做參考範例，做為 ELSP 之倉儲研究的標準。

## 參考文獻

1. 林禎樺 (2014), 《以固定速率法求解經濟批量排程問題的倉儲大小與成本》, 東海大學資訊管理學系碩士論文, 未出版。
2. 郭育璋 (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》, 東海大學資訊管理學系碩士論文, 未出版。
3. 陳世杰、張育仁、黃裕舜、黃晞臨 (2012), 〈應用固定生產速率求解生產系統下經濟批量排程與檢驗問題〉, 發表於2012工業工程學會年會暨學術研討會, 大葉大學。
4. 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。
5. 張育仁、陳世杰、黃晞臨、郭再添 (2014), 〈以簡單固定速率法求解在基本週期法下經濟批量排程問題〉, 發表於第十二屆 (2014) 產業管理創新研討會-精實創新與創意思考, 修平科技大學。
6. 許柏彥 (2012), 《經濟批量排程問題之倉儲空間與成本的探討》, 東海大學資訊管理學系碩士論文, 未出版。
7. 鄭舜維 (2009), 《庫存空間受限下最佳損毀率與儲存量決策問題之研究》, 東吳大學會計學系碩士論文, 未出版。
8. 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, 出版。
9. 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。
10. 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.

11. Bomberger, E. (1966). A dynamic programming approach to a lot size scheduling problem, *Management Science*, 12, 778-784.
12. Boctor, F.F. (1987). The g-group heuristic for dingle machine lot scheduling, *International Journal of Production Research*, 25, 363-379.
13. Buzacott, J.A. and I.A. Ozkarahan (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.
14. Bullnheimer, R.F. Hartl and C, Strauss (1997). An improved ant system algorithm for the vehicle routing problem, *Meta-Heuristics : Advances and Trends in Local Search Paradigms for Optimization*, 109.
15. Bullnheimer B., R. F. Hartl., and C. Strauss (1999). A new rank based version of the ant system- A computational study, *European Journal for Operations Research and Economics*, 7, 25-38
16. Carreno, J.J. (1990). Economic lot scheduling for multiple products on parallel identical processors, *Management Science*, 36, 348-358.
17. 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, 229-251.
18. Chang Y.C., C.Li. Vincent and Chia-Ju Chiang (2014). An ant colony optimization heuristic for an integrated production and distribution scheduling problem, *Engineering Optimization*, 46, 503-520.
19. Davis, S.G. (1990). Scheduling economic lot size production runs, *Management Science*, 36, 985-998.
20. Dobson, G. (1992). The cyclic lot scheduling problem with sequence-dependent setups, *Operations Research*, 40, 736-749.
21. Dorigo, M., and T. Stützle (2002). The ant colony optimization metaheuristic : Algorithms, application and advances, In Glover, F., & G Kochenberger.(Eds.),

*Handbook of Metaheuristics*, Kluwer Academic Publishers.

22. Dorigo, M., and L.M Gambardella (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem, *IEEE Transactions on Evolutionary Computation*, 1, 53-66.
23. Dorigom, M. (1992). *Optimization, learning and natural algorithms*, PH.D. Thesis, Dip. Elettronicae Informaztion, Politecnico di Milano, Italy.
24. Dorigom, M.,and L.M. Gambardella (1997). Ant colonies for the traveling salesman problem.*BioSystems*,43, 73-81.
25. Elmaghraby, S.E. (1978). The economic lot scheduling problem (ELSP): review and extension, *Management Science*, 24, 587-597.
26. Eynan, A. (2003). The benefits of flexible production rates in the economic lot scheduling problem, *IIE Transactions*, 35, 1057-1064.
27. Eberhart, R.C. and Y Shi (1998). Comparison between genetic algorithms and particle swarm optimization, *Annual Conference on Evolutionary Programming*, San Diego.
28. Eberhart, R.C. and J Kennedy (1995). New optimizer using particle swarm theory, *Proc. Sixth International Symposium on Nagoya, Japan*, 39-43.
29. Emanuel T., M Efr'én, and A Carlos (2007). An ant system with steps counter for the job shop scheduling problem, *IEEE Congress on Evolutionary Computation*, September, 19, 477-484.
30. Geng, P. C. and R. G. Vickson (1988). Two heuristics for the economic lot scheduling problem: an experimental study, *Naval Research Logistics*, 35, 605-617.
31. Gallego, G. (1993). Reduced production rates in the economic lot scheduling problem, *International Journal of Production Research*, 31, 1035-1046.

32. Gallego, G. and I. Moon (1996). How to avoid stocks when producing several items in a single facility? What to do if you can't. *Computers & Operations Research*, 23, 1-12.
33. Hsu, W. L. (1983). On the general feasibility test of scheduling lot sizes for several products on one machine, *Management Science*, 29, 93-105.
34. Hanssmann, F. (1962). *Operation research in production and inventory control*, Wiley, New York.
35. Haessler, R.W. and S.L. Hogue (1976). A note on the single machine multi-product lot scheduling problem, *Management Science*, 22, 909-912.
36. Haessler, R.W. (1979). An improved extended basic period procedure for solving the economic lot scheduling problem, *AIIE Transactions*, 11(4), 336-340.
37. Holland, J.H. (1975). *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor, MI.
38. Jones, P. C. and R. R. Inman (1989). When Is The Economic Lot Scheduling Problem Easy?, *IIE Transaction*, 21, 11-20.
39. Jaruphongsa, W., S. Cetinkaya and C. Lee (2004). Warehouse space capacity and delivery time window considerations in dynamic lot-sizing for a simple supply chain, *International Journal of Production Economics*, 92, 169–180.
40. Khouja, M., Z. Michalewicz and M. Wilmot (1998). The use of genetic algorithms to solve the economic lot size scheduling problem, *European Journal of Operation Research*, 110, 509-524.
41. Khouja, M. (1999). A note on 'Deliberately slowing down output in a family production context', *International Journal of Production Research*, 37, 4067-4077.
42. Khouja, M. (1997). The economic lot scheduling problem under volume flexibility, forthcoming, *International Journal of Production Economics*, 48 73-86.

43. Liang, Yanlai and Fangming Zhou (2011). A two-warehouse inventory model for deteriorating items under conditionally permissible delay in payment, *Applied Mathematical Modelling*, 35, 2221-2231.
44. Lima, M.F., L.D.H. Sampaio, B.B. Zarpelao, J.J.P.C. Rodrigues, T Abrao. and M.L Proenca (2010). Networking anomaly detection using dns and particle swarm optimization with re-clustering, *Global Telecommunications Conference*, 1-6.
45. 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.
46. Minner, S. (2009). A comparison of simple heuristics for multi-product dynamic demand lot-sizing with limited warehouse capacity, *International Journal of Production Economics*, 118, 305-310.
47. Moghadam, B.F., S.J. Sadjadi and S.M Seyedhosseini (2010). Comparing mathematical and heuristic methods for robust vehicle routing problem. *International Journal of Research and Reviews in Applied Science*, 2(2),108-116.
48. Moon, I., E.A. Silver and S. Choi (2002). Hybrid genetic algorithm for the economic lot-scheduling problem, *International Journal of Production Research*, 40, 809-824.
49. Merwe, D.W. and A.P Engelbrecht (2003). Data clustering using particle swarm optimization, *Proceedings of IEEE Congress on Evolutionary Computation*, 215-220.
50. Moon, I.K., B.C. Cha and H.C. Bae (2006). Hybrid genetic algorithm for group technology economic lot scheduling problem, *International Journal of Production Research*, 44, 4551-4568.
51. Moon, D. H. and P. D. Christy (1998). Determination of optimal production rates

- on a single facility with dependent mold lifespan, *International Journal of Production Economics*, 54(1), 29-40.
52. Merwe, D.W. and A.P. Engelbrecht (2003). Data clustering using particle swarm optimization, *Proceedings of IEEE Congress on Evolutionary Computation 2003*, 215-220.
53. Maniezzo, V. (1999). Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem. *INFORMS Journal on Computing*, 11, 358-369.
54. McMullen, P. R. (2001). An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives. *Artificial Intelligence in Engineering*, 15, 309-317.
55. Maier, H.R., A. R. Simposon, A. C. Zecchin, W. K. Foong, K. Y. Phang, H. Y. Seah, and C. L. Tan (2003). Ant Colony optimization for water distribution system. *J. Water Resour. Plng. And Mgmt., ASCE*, 129(3), 200-209.
56. Öner, S. and T. Bilgiç (2008). Economic lot scheduling with uncontrolled co-production, *European Journal of Operational Research*, 188, 793-810.
57. Park, K.S. and D.K. Yun (1987). Feasibility test for multi-product lot size scheduling on one machine, *Policy and Information*, 11, 101-108.
58. Rogers, Jack (1958). A computational approach to the economic lot scheduling problem, *Management Science*, 4, 264-291.
59. Soman, C.A., D.P. van Donk and G.J.C. Gaalman (2004). A basic period approach to the economic lot scheduling problem with shelf life considerations, *International Journal of Production Research*, 42(8), 1677-1689.
60. Sheldon, R. (1986). The lot scheduling problem in the hierarchy of decision models. Ph.D. dissertation. School of OR & IE, Cornell University, Ithaca, NY 14850, USA.

61. Silver, E.A. (1990). Deliberately slowing down output in a family production context, *International Journal of Production Research*, 28, 17-27.
62. Stützle, T., and M Dorigo (1999). ACO algorithms for the traveling salesman problem, *Evolutionary Algorithms in Engineering and Computer Science*, 3, 163-183.
63. Tang, O. and R.H. Teunter (2006). Economic lot scheduling problem with returns, *Production and Operations Management*, 15(4), 488-497.
64. Tang, O. and R.H. Teunter (2006). Economic lot scheduling problem with returns, *Production and Operations Management*, 15(4), 488-497.
65. Teunter, R.H., K. Kaparis and O. Tang (2008). Multi-product economic lot scheduling problem with separate production lines for manufacturing and remanufacturing, *European Journal of Operational Research*, 191, 1241-1253.
66. Transchel, S. and S. Minner (2009). Dynamic pricing and replenishment in the warehouse scheduling problem – A common cycle approach, *International Journal of Production Economics*, 118, 331-338.
67. Wangner, B.J. and D.J. Davis (2002). A search heuristic for the sequence-dependent economic lot scheduling problem, *European Journal of Operational Research*, 141, 133-146.