

東 海 大 學
工業工程與經營資訊學系

博士論文

會產出瑕疵品的生產系統下經濟批量、檢驗
與排程問題之研究

研 究 生：陳世杰

指導教授：姚銘忠 教授

曾宗瑤 副教授

中 華 民 國 一 〇 一 年 五 月

The Study of Economic Lot, Inspection and Scheduling Problem for the Imperfect Production Systems

By
Shih-Chieh Chen

Advisor : Prof. Ming-Jong Yao
Associate Prof. Tsueng-Yao Tseng

A Dissertation
Submitted to the Institute of Industrial Engineering and Enterprise
Information at Tunghai University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in
Industrial Engineering and Enterprise Information

May 2012
Taichung , Taiwan , Republic of China

會產出瑕疵品的生產系統下經濟批量、檢驗排程問題之研究

學生：陳世杰

指導教授：姚銘忠 教授

曾宗瑤 副教授

東海大學工業工程與經營資訊學系

摘 要

本研究論文主要提出在會產出瑕疵品的生產系統下，經濟批量、檢驗與排程問題 (Economic Lot, Inspection and Scheduling Problem, ELISP) 的模式與其求解的研究方法。實務上，在會產出瑕疵品之生產系統的製造過程中，產品檢驗工作扮演非常重要的角色。雖然，檢驗工作會帶來機台設置與相關人事成本的增加，但若能在出貨前，以檢驗程序及時發現製程進入「非正常狀態」，可以儘早調整/修護整理，使其避免持續製造不良品，增加修護成本與產能使用的機會成本。因此，本論文設計的數學模式將求取各產品最佳生產週期與在生產週期中之最佳檢驗次數，達成單位時間總成本之最小化。同時，為考量到生產的合理性，所設計模式將會以生產產能與檢驗產能列入限制條件。本論文主要針對共同週期法 (Common Cycle approach, CC) 與延伸基本週期法 (Extended Basic Period approach, EBP) 的週期型態進行探討，並實行全數檢驗 (Consensus Inspection) 的檢驗策略。本論文首先發展一搜尋演算法 (Search Algorithm) 來求解共同週期型態之模式的最佳解，後續設計一混合基因演算法 (Hybrid Genetic Algorithm, HGA) 結合搜尋演算法，在二冪策略 (Power of Two, PoT) 下來求解延伸基本週期型態之模式的最佳解。本論文針對兩種週期型態提出隨機實驗，分別驗證搜尋演算法與混合基因演算法是求解 ELISP 可行且有效的解法。同時，應用延伸基本週期型態的隨機實驗顯示，EBP 型態在低產能率、多產品數的組合下，能獲得比使用 CC 型態低約 6.07% 的平均總成本。

關鍵字：經濟批量、檢驗策略、搜尋演算法、共同週期、延伸基本週期。

The Study of Economic Lot, Inspection and Scheduling Problem for the Imperfect Production Systems

Student: Shih-Chieh Chen

Advisor: Prof. Ming-Jong Yao,

Associate Prof. Tsueng-Yao Tseng

Department of Industrial Engineering and Enterprise Information
Tunghai University

ABSTRACT

In this study, we are interested in formulating mathematical models and proposing solution methodology for solving the Economic Lot, Inspection and Scheduling Problem (ELISP) for imperfect production and inventory systems. Inspection plays an important role in an imperfect production system since it saves the cost from producing and restoring defective items though it also incurs extra inspection cost at the same time. In this dissertation, we apply two different types of replenishment cycle, the common cycle (CC) approach and the extended basic period (EBP) approach in the ELISP model, and adopt a consensus inspection policy. The focus of this study is to determine the optimal cycle time and an optimal production and inspection schedule that minimize the total cost per unit time. We formulate a mathematical model in which we take into accounts both the production capacity and inspection capacity constraints. We conduct full theoretical analysis and propose an effective search algorithm for solving an optimal solution as using CC approach. Then we design a hybrid genetic algorithm (HGA) to get the optimal solution when applying EBP approach, which is equipped with a search algorithm that not only seeks to improve solution quality, but also assuring a feasible solution. Our numerical experiments demonstrate the effectiveness of the proposed search algorithm and HGA to solve the ELISP. Also, we can get about 6.07% the lower average total cost by using EBP approach than CC approach as the number of items is larger and the utilization rate is lower.

Keywords: economic lot, inspection strategy, search algorithm, common cycle approach, extended basic period approach.

目錄

摘要.....	I
ABSTRACT.....	II
致 謝.....	III
目錄.....	IV
圖目錄.....	VII
表目錄.....	VIII
第一章 緒論.....	1
1.1 研究動機與背景.....	1
1.2 研究範圍與限制.....	3
1.3 研究目的.....	4
1.4 論文架構.....	5
第二章 文獻探討.....	7
2.1 會產出瑕疵品的生產系統的相關研究.....	7
2.2 經濟批量排程問題相關研究.....	9
2.2.1 經濟批量排程問題的求解.....	9
2.2.1.1 共同週期法 (Common Cycle approach, CC).....	10
2.2.1.2 基本週期法 (Basic Period approach, BP).....	11
2.2.1.3 延伸基本週期法 (Extend Basic Period approach, EBP).....	13
2.2.2 經濟批量排程問題相關延伸研究.....	13
2.3 批量問題考慮檢驗情況之相關研究.....	15
2.4 文獻評論與研究主題確立.....	17
第三章 共同週期法求解經濟批量、檢驗與排程問題.....	19
3.1 前言.....	19
3.2 模式建構.....	19
3.2.1 模式符號定義.....	19

3.2.2 模式假設.....	20
3.2.3 模式建立.....	21
3.3 最佳解理論性質分析.....	28
3.3.1 目標函數式性質分析.....	28
3.3.2 接合點與凹性函數曲線的性質.....	28
3.4 搜尋演算法.....	32
3.4.1 搜尋演算法的下界.....	33
3.4.2 接合點搜尋過程.....	34
3.4.3 搜尋演算法的上界.....	35
3.4.4 搜尋演算法的程序.....	39
3.5 數據實驗.....	41
3.5.1 數值範例.....	41
3.5.2 隨機實驗.....	43
3.5.3 效能分析.....	45
3.6 結論與討論.....	46
第四章 延伸基本週期法求解經濟批量、檢驗與排程問題.....	47
4.1 前言.....	47
4.2 模式建構.....	47
4.2.1 模式符號定義.....	47
4.2.2 模式假設.....	48
4.2.3 模式建構.....	48
4.3 以基因演算法結合搜尋演算法求解的相關說明.....	53
4.3.1 基因演算法的建置.....	53
4.3.1.1 染色體的編碼與初始母體.....	54
4.3.1.2 適應函數的計算與選擇機制.....	56
4.3.1.3 基因運算子.....	58
4.3.1.4 參數設定與終止條件.....	59
4.3.2 搜尋演算法的程序.....	59
4.3.2.1 搜尋法的下界.....	60
4.3.2.2 接合點搜尋過程.....	61

4.3.2.3 搜尋演算法的上界	62
4.3.3 為候選解產生可行的生產與檢驗排程	64
4.3.3.1 可行性測試程序.....	64
4.3.3.2 二元搜尋演算法.....	66
4.3.4 混合基因演算法虛擬程式碼	68
4.4 數據實驗	69
4.4.1 數值範例.....	69
4.4.2 隨機實驗.....	70
4.4.3 效能分析.....	73
4.5 結論與討論	73
第五章 結論與建議.....	74
5.1 研究結論	74
5.2 未來研究方向	75
參考文獻.....	76
附錄 A.....	82
A1. 產品 <i>i</i> 之一個生產批量的不良品的期望數推導.....	82
A2. 存貨三角形面積的高之推導說明	83
A3. 式(3.10) 的推導過程	84
附錄 B.....	86
B.1 x_i 的上界 $x_i^{UB}(B)$ 之推導	86
B.2 排程初始程序 (PROC IS) 說明	87
B.3 排程撫平程序 (PROC SS) 說明.....	90
B.4 B 值的下界 B_i^{PCL} 與 B_i^{ICL} 推導.....	95

圖目錄

圖 1.1 本論文研究的流程.....	6
圖 2.1 三種產品的共同週期法存貨圖	10
圖 2.2 三種產品的基本週期法存貨圖	11
圖 3.1 使用 CC 法之生產與檢驗存貨圖	22
圖 3.2 不同 m_i 下的成本曲線 $TC_i(m_i, T)$	29
圖 3.3 $TC_i(T)$ 的成本曲線	30
圖 4.1 使用 EBP 法之生產與檢驗存貨圖	49
圖 4.2 染色體的編碼方式.....	54
圖 4.3 三種交配運算子說明圖	58
圖 4.4 W_p 與 W_l 的排程結構圖	65
圖 A.1 圖 3.1 之下半部鋸齒三角形部分圖	84

表目錄

表 3.1 單一產品 <i>i</i> 的參數設定	28
表 3.2 數值範例 10 個產品的參數資料	41
表 3.2 數值範例 10 個產品的參數資料 (續).....	42
表 3.3 數值範例的結果.....	42
表 3.4 隨機實驗的參數組合.....	44
表 3.5 以 CC 法之不同水準 $\sum \rho_i$ 的實驗結果.....	44
表 3.5 以 CC 法之不同水準 $\sum \rho_i$ 的實驗結果 (續).....	45
表 4.1 使用 GA 與 Proc FT 建立的可行排程.....	69
表 4.2 隨機實驗的參數組合.....	70
表 4.3 不同水準 $\sum \rho_i$ 的實驗結果	71
表 4.3 不同水準 $\sum \rho_i$ 的實驗結果 (續)	72
表 B-1 Proc IS 範例產品的基本資料.....	87
表 B-2 產品 2 加入排程 (Proc IS 範例).....	88
表 B-3 產品 1 加入排程 (Proc IS 範例).....	88
表 B-4 產品 3 加入排程 (Proc IS 範例).....	89
表 B-5 產品 4 加入排程 (Proc IS 範例).....	89
表 B-6 移出程序(改善前)範例	90
表 B-7 移出程序(改善後)範例	91
表 B-8 成對交換程序(改善前)範例	92
表 B-9 成對交換程序(改善後)範例	92
表 B-10 2 對 1 互換程序(改善前)範例	93
表 B-11 2 對 1 互換程序(改善後)範例	94

第一章 緒論

1.1 研究動機與背景

生產與存貨管理，長久以來一直是製造產業所關心的重要課題。對企業來說，一個有效的生產計畫與排程，除了可以降低生產成本、避免過多存貨或是缺貨情況發生外，還可以提升產能、增加產品附加價值與提升顧客的滿意度。

一般而言，生產的過程中不免都會有存貨產生，但如何能有效的管理存貨，實際上並非是件容易的事。如果存貨水準設定過高，會造成企業資金積壓與利息損失；反之，若存貨水準設定過低，則需增加訂貨次數以滿足顧客需要，其使得總成本上升；再者，存貨水準過低易發生缺貨情況，可能會造成訂單或顧客流失。因此，對企業而言，設定適當的存貨水準以尋求最低總成本是非常重要的議題。

早期有學者 Harris (1913) 提出經濟生產批量 (Economic Production Quantity, EPQ) 數量分析模式來解決生產與存貨的問題，其模式假設是在完美的生產存貨系統 (perfect production inventory system) 環境下進行生產。換言之，生產過程中不會產生任何的不良品或瑕疵品 (defective item)。但如此的假設，並不符合真實的生產狀況。

生產作業的執行，不管是過去以人工為主，或是現代以高度自動化為主的系統，不良品的產生是必然的情況。早期的不良品，可能是因為生產過程中人為操作疏失所引起；而在自動化時期所發生的不良品，則可能是因為設備使用方式不當或使用年限等因素造成。一旦這些有瑕疵的產品未經品質檢驗 (inspection) 而被售出，會對企業獲利與商譽造成莫大的損失與傷害。因此，在生產與存貨系統下，除了生產問題，產品檢驗工作是非常重要的。

生產系統要減少不良品有許多的方式，可以實施例如：製程分析、統計製程管制 (Statistical Process Control, SPC) 與6個標準差的使用等都是希望能長期維持生產線的品質穩定。而一旦進入生產過程，不良品如何被找出，則須仰賴產品檢驗工作。

實務上，生產線產品的檢驗工作，會依照產品特性與生產流程有不同的做法：一則可以透過線上檢驗 (on-line inspection) 方式在生產過程中立即發現不良品，或者是當生產部分批量後，藉由送至品管單位檢查後找出瑕疵品。被發現的不良品，可透過修護 (repair) 或重製 (rework) 的方式，恢復至顧客所要求的水準，後續依然可依約出貨。有關不良品修護或重製的時間與成本問題，在早期傳統的 EPQ 模式的研究中未見被納入考量。但近數十年來，已有許多學者針對此情況的存貨模式陸續進行探討。如：Lee & Rosenblatt (1987)，Porteus (1986) 與 Shih (1980)。

再者，生產批量 (lot-sizing) 在生產管理中也是相當重要的課題。生產的批量大小，除了會直接影響生產計畫實行的成本績效外，同時也會受到生產資源限制的影響。例如：在一個產品種類繁多、但個別種類需求批量不大的工廠裡，生產眾多產品種類是足以應付需求的(其不超出資源限制)；但如果工廠設備有限，而個別種類需求的批量又很大時，若此時生產作業機台無法負荷，排程計畫可能會行不通。因此，生產批量的大小在生產排程研究中必須事前先加以限制。

經濟批量排程問題 (Economic Lot-sizing Scheduling Problem, ELSP) 在過去數十年中，一直受到許多學者的重視。其主要解決在單一機台生產多種產品時，產品之生產排程與每一次批量大小的問題。ELSP 可藉由生產週期的調整，尋求合理可行且平均總成本最低的生產排程解決方案。經濟批量排程問題在製造業的應用非常廣泛，例如：製罐、汽車、鋼鐵、製藥及金屬成型等產業。由於這些產業的生產設置成本及存貨成本都很高，所以，批量問題自然成為生產排程中非常重要的考量因素。而 ELSP 的研究，正好可協助決策者尋找適當的生產批量。再者，不單是針對製造業，其實服務業也可以利用經濟批量排程概念來規劃商品的採購數量。

早期 Elmarghraby (1978) 針對單機 (single machine) 生產的經濟批量排程問題求解，提出了兩種主要方法：分析式解法 (analytical approach) 與啟發式解法 (heuristic approach)。其說明以分析式解法求解 ELSP 時，可以明確找到存貨模式的最佳解 (optimal solution)，但缺點是當需要求解超過 10 個產品項的問題時，會耗費極多的演算時間。其被使用的相關方法有：整數規劃法 (integer programming) 與動態規劃法 (dynamic programming)

等；而啟發式解法在求解 ELSP 時，會易於找到區域最佳解 (local optimal solution)，此種解法的缺點是不能保證找到全域的最佳解 (global optimal solution)。近年來，學者針對 ELSP 提出的啟發式解法不勝枚舉，包括有：貪婪式搜尋法 (Park & Yun, 1985)、基因演算法 (Khouja & Mehrez, 1994)等。

在過去經濟批量排程問題的相關研究中，多數沒有假設產能限制 (production capacity)的問題。但在實際的生產系統中，工廠設備是有其限制的；特別是同一週期時間內要生產眾多產品以應付需求時，唯有考慮生產機器或檢驗機台產能的限制，才能找出真正合理且可行的排程，使產品能在總成本最低的情形下，分批完成以滿足顧客的需求。

綜合以上闡述生產與存貨管理、生產批量、檢驗策略、產能限制與 ELSP 等相關問題，本論文將探討 ELSP 的創新延伸問題：把產品檢驗納入考量。換言之，本研究將針對單一生產設備生產多項產品，藉由生產週期、生產排程與檢驗排程的調整，並考量生產產能與檢驗產能 (inspection capacity) 的限制，尋找出最低平均總成本下，可行的生產與檢驗排程。

1.2 研究範圍與限制

本研究的研究範圍將設定單機之經濟批量排程問題並考慮生產檢驗。研究限制主要分為四項，說明如下：

首先，本研究限定為單機生產之經濟批量排程問題研究。雖然瞭解實務上工廠為多機的生產型態，但在本研究主題上，多機問題需考慮到更複雜的生產與檢驗排程模式，而本論文為 ELSP 考量檢驗相關問題的首創研究，故先考慮較單純的單機生產型態作為初探。

再者，本研究採用每次生產固定批量後，將批量送至檢驗機台進行全數品質檢驗的假設。在實務上，根據生產程序與產品特性，可延伸非常多種的檢驗策略；而目前多數採用較有效率的方式為線上即時檢驗。在本研究所實行的檢驗策略，雖針對特定產業的產品(如：電路板)實行，但仍可做為其他產業實務應用的價值。

第三，本研究限制產品生產為不發生缺貨 (shortage) 情況。由於瞭解實際生產系統處理缺貨的情形屬於常態，而本研究不發生缺貨的假設實為

強烈。但考慮到當發生缺貨時，其衍生的缺貨成本與其他問題，將會造成模式更為複雜。因此，本論文在初始研究中暫不考慮缺貨情形，只將其列為未來後續研究之用。

第四，本研究為確定性之經濟批量排程問題研究。由於瞭解實務上生產率 (production rate) 或是產品需求 (product demand) 是屬於非穩定與不確定的狀態。一般而言，造成生產率變動的原因包括有：機器損壞、人力安排問題、上游原物料或半成品供給未及等因素；而產品需求的變動則與經濟景氣變化、市場競爭有關。因此，生產率與產品需求是屬於不確定性的因素。在本研究中，會假設上述兩項因素與其他等相關參數為確定性，是因確定性之經濟批量排程問題，過去已被證明為 NP-hard 的問題 (Hsu, 1983)。本研究如再考慮非確定性因素，會增加模擬 (simulation) 求解的困難度。再者，即使在早期學術研究中，常見文獻會以隨機過程表示不確定性需求，例如：卜瓦松過程。但此學理上的隨機過程是否真的貼近於實際發生的情況，此答案至今仍待查證。因此，本研究雖瞭解確定性的假設未必恰當，但考量到先簡化研究問題，以獲得初始結論。所以，仍採用確定性方式探討主要研究問題。

1.3 研究目的

根據 1.1 節研究背景與動機的討論，本論文首先將針對會產出瑕疵品的生產系統、批量生產、檢驗策略與 ELSP 過去相關延伸研究的文獻進行探討，瞭解所設定的研究主題，在過去是否有學者進行相關研究過；同時，本研究會針對文獻中求解 ELSP 常用的解法進行整理。

本論文之主要目的，是將 ELSP 與檢驗策略結合，發展出新的研究主題，以求解在最低平均總成本下，合理的最佳生產排程與檢驗排程。本研究將依據生產與檢驗過程中所可能產生的成本項，設計數學研究模式(模式新命名為『經濟批量、檢驗與排程問題』)，並同時考量生產機台與檢驗機台的產能，予以適當之限制條件 (constraint)。

至於模式的最佳可行解求解，本論文將分別採用啟發式解法中的搜尋演算法及混合基因演算法(基因演算法結合搜尋演算法、合理解測試演算法

與二元搜尋演算法)，以相同或不同生產週期生產的類型方式，尋求合理最佳解。有關不同週期類型的求解，將在第三章以後詳細介紹。

1.4 論文架構

本論文的研究架構說明如下：

第二章文獻探討中，本研究首先介紹會產出瑕疵品之生產系統的相關研究，說明其與無產出瑕疵品之生產系統的差異以及過去重要的文獻。再者，解釋經濟批量排程問題的定義與應用，介紹常用的三種週期設定方式：共同週期法、基本週期法以及延伸基本週期法，以及其過去相關的研究。同時，本章將瀏覽經濟批量排程問題各類延伸問題的論文。在此章最後，本研究將針對批量生產與檢驗策略的主題，進行相關文獻的討論。

第三章將介紹使用共同週期法的經濟批量、檢驗與排程問題模型。本研究設計一搜尋演算法，來求解最低平均總成本下，合理可行的生產與檢驗排程。為了確定最佳解為合理可行，設計模式將加上生產產能與檢驗產能的限制條件。此章將就搜尋演算法進行相關的理論分析，探討其接合點與凹性曲線的特性，並介紹演算法搜尋之上、下界的推導，以及接合點搜尋過程。本章研究以一數值範例驗證模式，並有隨機實驗說明在不同產品數與不同產能使用率下，產生最低平均總成本與執行時間的結果。此章最後，將根據數據結果與貢獻作一結論。

第四章內容介紹求解經濟批量、檢驗與排程問題的另一種週期型態，使用延伸基本週期法。本章將以基因演算法結合第三章使用之搜尋演算法，尋求平均總成本的最低解。相較於第三章生產週期使用共同週期法，延伸基本週期法可避免因共同週期法造成產能過度閒置的問題，其可藉由個別產品生產週期長短的彈性調整，找到一組較有效率且平均總成本較低的解。由於期望同樣能獲得合理可行的解，此章研究將利用合理解測試程序及二元搜尋演算法，來確保並判斷決策變數所形成的解集合，在該模式之產能資源限制條件下合理可行，亦或是指出無任何可行解。本章研究從數值範例與隨機實驗的結果中，顯示使用延伸基本週期法可獲得相較於使用共同週期法更低的平均總成本。由於所設計之混合基因演算法結合了眾多演算法，模式演算程序較為複雜，因此，實驗結果產生花費較多的執行時間。

在最後第五章，本研究提出了論文的綜合結論與未來研究方向。此章首先針對第一個研究子題以使用共同週期法求解，歸納研究結果與建議。接續，說明第二個研究子題以使用延伸基本週期法求解的研究結果與建議。在此章結尾，本研究將討論未來相關可能的研究方向。

本論文之研究架構流程請參考圖 1.1。

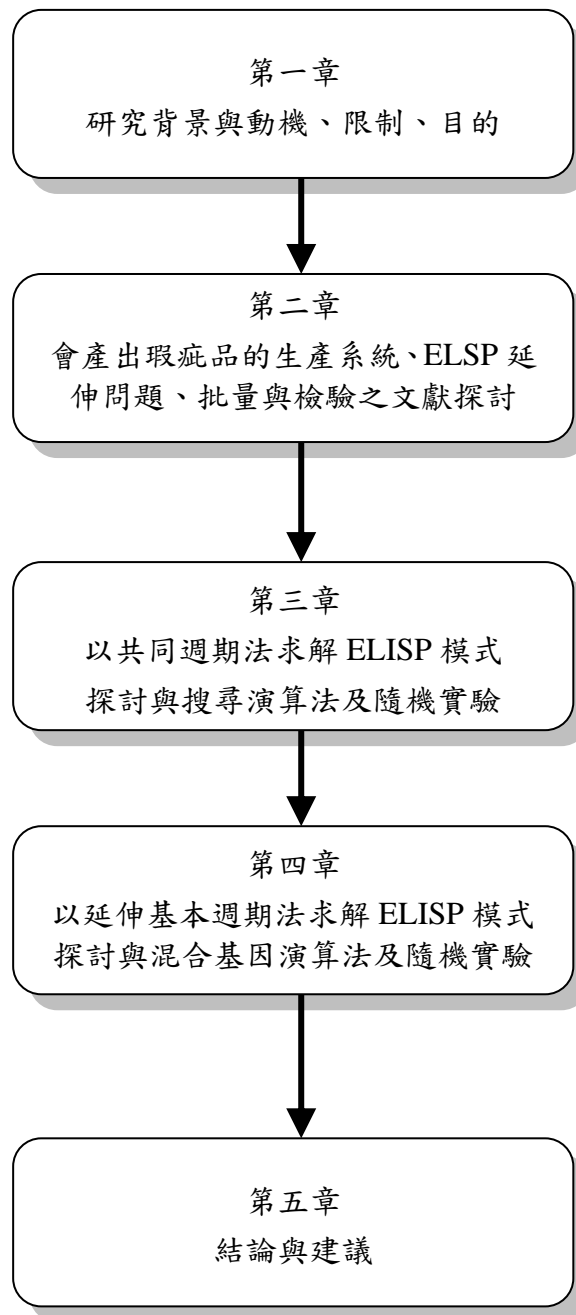


圖 1.1 本論文研究的流程

第二章 文獻探討

在此章節，本研究將針對經濟批量排程問題與檢驗策略過去的相關研究作深入的探討。由於本論文研究考量生產過程會有不良品產生，是屬於會產出瑕疵品的生產系統。因此，本章開始首先針對會產出瑕疵品的生產存貨系統研究的演進與發展歷程作說明，接著再探討傳統的經濟批量排程問題；本章將介紹經濟批量排程問題過去的背景與發展應用，並就其主要三種週期類型求解方式：共同週期法、基本週期法與延伸基本週期法做解說。然後，會針對早期經濟批量排程問題所衍生的相關延伸研究進行探討。

由於「檢驗策略」是本研究論文設定探討經濟批量排程問題延伸研究的主題，所以此章將會詳加討論過去批量與檢驗策略的相關研究，並做系統性的整理與分析。本章的最後，將針對文獻探討的內容，提出本論文研究的問題重點。

2.1 會產出瑕疵品的生產系統的相關研究

Harris (1913) 提出經濟生產批量 (Economic Production Quantity, EPQ) 模式，其假設生產系統在製程過程中不會有失效的情況發生，且生產出來的產品皆為品質完美的產品，此稱之為完美生產系統 (perfect production system) 或無產出瑕疵品的生產系統。在此之後，有許多學者以此完美生產系統模式假設來解決存貨的相關問題。例如：Osteryound, McCarty & Reinhart (1988), Nahmias (1997) 與 Silver (1998)。

然而實際的生產過程並不同上述所言。一般來說，製程起先會在可控制 (正常) 的狀態下 (in-control state) 開始生產，其會生產出符合顧客品質要求的產品。但隨著製程時間的增加，生產系統本身可能會因為使用年限、機械零件磨損等因素影響，產生有持續性的衰退現象 (Yeh *et al.*, 2000)，若此時沒有採取任何修復行動，生產系統可能會轉變成不可控制(非正常)的狀態 (out-of-control state)，並生產出不符合品質要求的瑕疵品，此過程可稱之為不完美的生產 (imperfect production) 或是會產出瑕疵品的生產。實務上，眾所皆知生產系統的衰退現象，是不可避免的事實。因此，生產過程通常是處於可控制與不可控制的狀態兩者之一。

有鑑於上述實際的生產情況，有學者開始針對會產出瑕疵品的生產系統模式進行研究。過去文獻關於生產系統「非正常狀態」的發生有許多不同假設。Porteus (1986) 以及 Rosenblatt & Lee (1986) 是最早在研究中探討不良品對 EPQ 模式影響的論文。Porteus (1986) 假設每當製程生產出一個產品後，都會存在一個固定的機率使得製程由正常狀態轉變為非正常狀態；Rosenblatt & Lee (1986) 則假設製程由正常狀態轉變為非正常狀態的時間，是一隨機變數且其符合指數分配 (exponential distribution)。Lee & Rosenblatt (1987) 與 Sung & Ock (1992) 分別在其生產週期與檢驗排程的 EPQ 模式研究以及單機生產單項產品模式的研究中，都假設製程轉變成非正常狀態是服從一指數分配。Ben-Daya & Hariga (2000) 及 Moon *et al.* (2002) 在其探討經濟訂購批量問題模式的論文中，假設生產系統會產生一固定比例的瑕疵品，而系統轉變成非正常狀態是服從一指數分配。另外，Lin, Chen & Kroll (2003) 針對單一產品生產之可能製造瑕疵品的系統研究中，討論整合性的 EPQ 模型並發展出一演算法，其系統非正常狀態的假設也是服從指數分配。

過去也有不同學者針對系統非正常狀態採用其他的分配，例如：Braglia (1995) 在其訂單式生產研究中，假設生產系統由正常狀態轉變為非正常狀態是服從卜瓦松分配 (poisson distribution)；Makis & Fung (1998) 與 Tseng *et al.* (1998) 在探討生產系統檢驗與維修策略的研究中，都假設系統轉變為非正常狀態是服從一般性的轉換分配 (general distribution)；Lin (1999) 探討原物料資源限制的會產出瑕疵品系統，其假設系統衰退為一個服從給定平均數之指數分配的隨機變數；Wang & Sheu (2003) 針對連續型之生產系統研究，假設其衰退轉變為服從幾何存活分配 (geometric survival distribution)。

綜觀上述來說，會產出瑕疵品的生產系統相較於無產出瑕疵品的生產系統，概念上更貼近實務。近幾年來，學者所研究的生產貨模式大部分都以會產出瑕疵品的生產系統環境下來進行探討，而其對於系統非正常狀態的假設，多數傾向使用指數分配來詮釋，並假定在系統轉變成非正常狀態時，會生產一定比例的不良品。

2.2 經濟批量排程問題相關研究

經濟批量排程問題 (Economic lot scheduling problem, ELSP) 最早是由 Roger (1958) 所提出。其生產環境是單一設備或是單一生產線可以生產多種不同的產品，並設定生產速率與需求速率是已知且固定的。ELSP 主要在探討產品批量大小、生產次數及週期時間的問題，目的是希望藉由調整產品週期性的生產排程使生產計劃可行，以滿足顧客長期的需求，並且使平均總成本（包含整備成本及存貨持有成本）能達到最低。實務上，ELSP 可支援生產管理人員做出適當批量大小 (lot size) 與加工順序 (job sequence) 的生產決策，其在製造業的應用相當廣泛，包括：汽車、油漆、飲料、紡織品及地毯工業等 (Chang & Yao, 2009)。

2.2.1 經濟批量排程問題的求解

Hsu (1983) 確認 ELSP 的求解是一個非多項式時間演算法可解 (NP-hard) 的存貨問題。Elmarghraby (1978) 將求解單機生產之 ELSP 方法區分為分析式解法 (analytical approach) 與啟發式解法 (heuristic approach) 兩種：分析式解法的作法是在某一種限制 (restricted version) 下求得最佳解，常用模式如：動態規劃 (Dynamic Programming, DP)、整數非線性規劃 (Integer Nonlinear Programming, INLP) 等；其利用窮舉法 (enumeration) 或分支界線法 (Branch and Bound, B&B) 的方式求取最佳解。而啟發式解法大部分是利用反覆式 (iterative) 搜尋的程序，求得批量問題較好的解。比較上述兩者，分析式解法通常可以得到最佳解或近似最佳解，但缺點是需耗費大量的執行時間；而啟發式解法可耗費較少執行時間求得其解，但求解過程有可能只是找到的解是區域最佳解 (local minimum)，因此不能保證所得到的解為全域最佳解。而在本論文研究考量 ELSP 多產品的生產模式及決策時間因素，將採取啟發式解法來求解研究主題。

在過去相關文獻中，學者在求解 ELSP 考量其生產週期問題時，會使用四種主要模式型態：獨立解法 (Independent Solution, IS)、共同週期法 (Common Cycle approach, CC)、基本週期法 (Basic Period approach, BP) 與延伸基本週期法 (Extended Basic Period approach, EBP)。獨立解法為當中概念最為簡單的模式型態，作法是每個產品只考慮本身的最佳解，然後加總

各產品的平均最低成本以獲得最後總成本；獨立解法並不能保證最佳解的排程是合理可行的，但獲得的結果通常可作為 ELSP 求解時可行解的下界 (Lower Bound)。而本論文研究主要與後面三種模式型態相關，用其來求解問題。詳細說明如下：

2.2.1.1 共同週期法 (Common Cycle approach, CC)

共同週期法是由 Hanssamann (1962) 所提出。其假設每個週期內每一種產品都只生產一次，而且所有產品的生產週期都相同。共同週期法的作法是把所有產品的排程都納入在一個共同的循環時間內，因此週期必須夠大足以容納所有產品的生產，而每個產品的生產都以此週期時間重覆循環。圖 2.1 說明生產三種產品的共同週期法存貨圖。圖中 s_i 代表產品 i 的前置時間 (setup time)； $\rho_i T$ 則表示產品 i 的生產時間 (production time)。

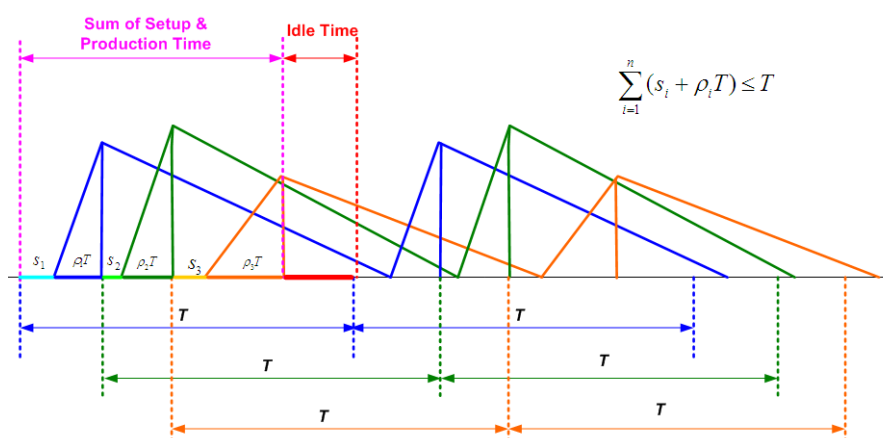


圖 2.1 三種產品的共同週期法存貨圖

假若產品的項目為 $i=1,2,\dots,n$ ；產品 i 的生產速率 (production rate) 是 p_i ；產品 i 的需求速率 (demand rate) 是 d_i ；產品 i 的設置成本 (setup cost) 是 A_i ；產品 i 的單位時間存貨成本 (holding cost) 是 h_i ； T 是產品生產週期 (production cycle time)，則共同週期法的基本數學模式為

$$\min TC(T) = \sum_{i=1}^n \frac{A_i}{T} + \frac{T}{2} \sum_{i=1}^n h_i d_i (1 - \rho_i)$$

$$s.t. \quad T \geq \frac{\sum_{i=1}^n S_i}{1 - \sum_{i=1}^n \rho_i}$$

ρ_i 代表產品 i 的產能利用率， $\rho_i = d_i / p_i$ 且 $\rho_i < 1$

由於共同週期法具有簡單與容易操作的特性，所以有很多學者常以此法當作基礎進行研究。例如：Elhafsi & Bai (1997) 利用共同週期法研究兩種產品生產速度的問題。

2.2.1.2 基本週期法 (Basic Period approach, BP)

基本週期法是由 Bomberger (1966) 所提出來的。其主要是使用動態規劃法 (DP) 來求解 ELSP。不同於前一節的共同週期法，BP 法假設不同產品有不同的生產週期時間，而每個產品 i 的週期時間必須是某一個基本時間 (basic period) 單位 B 的整倍數，即 $T_i = k_i B$ 。總和所有產品的生產循環週期就是所有產品的週期乘數 k_i 的最小公倍數 (LCM) 乘上 B 的值。

基本週期法假設所有的產品於第一週期都要生產，之後的週期是否要生產，則視每個產品的週期乘數 k_i 而定。再者，若當每個產品的 k_i 皆為 1 時，則其結果就等於是共同週期法的解。圖 2.2 為生產三種產品的基本週期法存貨圖。

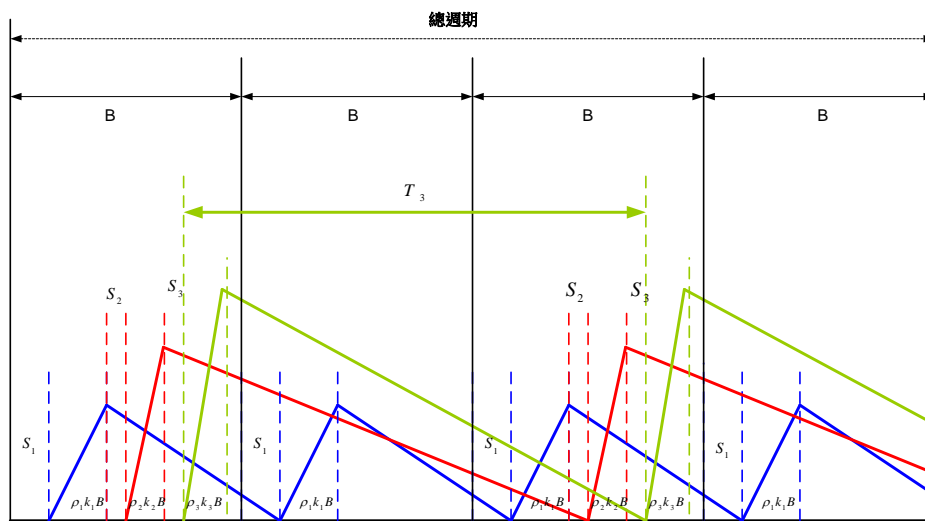


圖 2.2 三種產品的基本週期法存貨圖

基本週期法的數學模式表示如下：

$$\min TC(B, \{k_i\}) = \sum_{i=1}^n \frac{A_i}{k_i B} + \sum_{i=1}^n \frac{h_i}{2} d_i (1 - \rho_i) k_i B$$

$$\text{s.t. } \sum_{i=1}^n (s_i + \rho_i k_i B) \leq B \quad k_i \in \{1, 2, 3, \dots\}$$

由於基本週期法改善了共同週期法限定每個週期內每個產品皆要生產的缺點，因此，其所獲得的解會比共同週期法的結果來得好（平均總成本較低）。但是，基本週期法也有其困難點：因為每個基本週期負載的產品狀況不同，所以算法上難以保證可以收斂到可行解；同時，基本週期法所產生的排程除了週期一的機器負荷較重外，其他週期可能發生機器空閒的時間 (idle time) 較多，因此，很容易造成設備產能浪費及喪失找尋最佳解的機會。

過去研究中，Granar & Riggle (1997) 以基本週期法提出解決單一機台 ELSP 的演算法，其演算法主要針對兩個部分：第一部份是確定基本週期 B 的最佳值的範圍；第二部份是確定週期乘數組合 \mathbf{k} 的上下界，最後以窮舉法搜尋找到最佳解。

Yao & Elmaghraby (2001) 的 ELSP 研究中，以基本週期法為基礎產生模式，並採取二冪策略 (Power of Two policy, PoT) 將 k_i 表示成 $1, 2, 4, \dots, 2^p$ 等 2 的次方倍。他們採用二冪策略的目的是為進行可行解測試 (Proc FT) 時，可以方便求算所有產品生產週期之最小公倍數。

Soman *et al.* (2004) 在其考量產品保存期限 (shelf life) 因素的經濟批量排程問題研究中，也使用了基本週期法，並提出類似分支界線法的演算法來求取 k_i 的最佳解。Cookie (2004) 針對基本週期法不一定獲得可行排程的問題進行研究，其利用整數規劃的模式來具體求出產品應於何時開始生產、且應該生產多少。

由上說明，由於在基本週期法中的週期乘數 k_i 必須是整數，若加上模式本身的複雜性，求解上會變得困難。目前主要研究方向大部分集中在提出有效的演算法來求得問題的解。但是，即便是得到 B 和 k_i 的值，排程是否合理可行 (feasible) 仍須再加以確認。

2.2.1.3 延伸基本週期法 (Extend Basic Period approach, EBP)

延伸基本週期法可以讓每一個產品依週期乘數 k_i 自由調整其生產排程。不同於前段之基本週期法，延伸基本週期法沒有限制所有產品於第一週期都要生產；只要被安排到每個週期內的產品集合生產時間的總和小於基本週期時間 B ，即表示此生產排程是可行的；EBP法建立的生產排程是可以調整的，因此相較於基本週期法，EBP更具有彈性，可以有機會找到更低的平均總生產成本。

Haesseler & Hogue (1976) 是最早使用延伸基本週期法來檢視 ELSP 問題的可行性。他們運用一整數規劃模式來描述 ELSP 模式中排程的可行解判斷問題。而後，Elmaghraby (1978) 與 Davis(1990) 分別提出了修正模式，其兩者的模式比 Haesseler & Hogue (1976) 的整數規劃模式更加正確與嚴謹。

Yao(2000) 在其 ELSP 的可行性問題的研究中，以 EBP 法為基礎提出可行性檢測演算法 Proc FT，其可協助判斷機台 j 的某組解 $(B_j, \{k_i\})$ 是否為可行解。

由上述 2.2.1.1 至 2.2.1.3 節得知：共同週期法簡單且容易操作，其可確保模式所得解可行，一般以其做為研究問題的基礎；基本週期法彈性讓模式內每一項產品的生產週期長度不同，其可找到平均總成本比使用共同週期法更低的最佳解。但是基本週期法有兩項主要缺點：一為每一項產品第一週期皆要生產，造成後面週期會有過多的閒置時間；另一項則是無法保證結果為可行解。2.2.1.3 節的延伸基本週期法消除了基本週期法的第一項缺點，其無限制所有產品於第一週期都要生產，比基本週期法更具彈性。根據過去的相關文獻，模式使用延伸基本週期法可以找到比基本週期法平均總成本更低的最佳解 (Chang & Yao, 2009)。

2.2.2 經濟批量排程問題相關延伸研究

在過去數十年，有許多的學者針對傳統的 ELSP 問題做不同的延伸研究，當中有不少的延伸研究是以 CC 法來求解。有學者關注於 ELSP 模式下設置時間與生產順序依賴關係 (sequence-dependent setups) 的問題。例如：Dobson (1992) 與 Wagner & Davis (2002)。Dobson 針對 ELSP 的設置時間與生產順序

依賴關係提出 ELSP 的延伸模式，其透過轉換 (transformations) 與鬆弛 (relaxations) 的技術來求解。傳統 ELSP 模式是假設各產品之間，其彼此的設置時間、生產順序是相互獨立的關係，但實際的生產情況是彼此相互影響的關係存在，Dobson 以此問題提出 ELSP 的延伸研究。而後 Wagner & Davis 針對相同的問題，發展一搜尋啟發式演算法 (search algorithm) 來求解。不同於過去研究提供單一最佳解的方式，Wagner & Davis 的演算法提供一組的解供決策者選擇，以應付現實中多變且動態的生產環境。而根據數據實驗結果顯示，Wagner & Davis 設計的演算法優於 Dobson。上述兩篇文章都利用 CC 法求解。

Faaland *et al.* (2004) 針對 ELSP 考慮創造高利潤但允許銷售損失的情況提出模式，其設計一搜尋演算法求取最佳解。他們在隨機實驗中以 100 至 1000 種產品進行比較，結果發現模式的運算時間 (computation time) 會隨著產品數目的增加呈現平方次的成長。Salviotti & Smith (2008) 以整數規劃與 column generation 技術，求解 ELSP 考慮產品價格之利潤最大化的問題，其實驗結果說明此技術能有效率找到最佳近似解。Somon *et al.* (2004) 延伸了 ELSP，研究有關產品之保存期限問題。他們根據 Haessler's (1979)，求解相同問題並建立模式，其週期的類型採用基本週期法，求解方法為發展類似於分支界線法的啟發式演算法。實驗結果顯示其優於共同週期法所獲得的結果。Chang *et al.* (2006) 的 ELSP 延伸研究中，考慮不確定與變動需求 (uncertain or fluctuating demand) 的問題建立模式。他們的模式是以延伸基本週期法、二冪策略以及三角模糊 (triangular fuzzy) 需求為基礎來建立，並以基因演算法進行求解。

Tang & Teunter (2006) 提出考慮有退貨 (returns) 狀況的 ELSP 模式。他們假設所有的作業都在單一的生產線上進行。後續 Teunter *et al.* (2008) 根據 Tang & Teunter 的模式作修正，發展出多產品生產且有退貨狀況的 ELSP 模式。不同於前 Tang & Teunter 的模式，Teunter *et al.* 的模式中假設生產線可以進行重製工作。之後，Ouyang & Zhu (2008) 將 ELSP 問題延伸考慮在單一相同生產線上的製造與重製問題，其模式以基本週期法為基礎並限制製造與重製的產能。Chang & Yao (2009) 提出了 ELSP 考慮重製問題的模式，其設計在共同生產週期下，每個週期內，每一項產品僅有一生產批與一重製批。他們發展一混合基因演算法 (hybrid genetic algorithm)，求解最低總成本下最佳的生產排程與每項產品的重製批量。

Moon *et al.* (2006) 以一啟發式演算法與應用混和基因演算法，解決考慮群體技術 (group technology) 的 ELSP 問題 (GT-ELSP)，他們模式的建立是以共同週期法型態與時間變動批量 (time varying lot sizing) 為基礎。GT-ELSP 模式可降低產品的前置時間與成本，並由數值實驗結果顯示其設計的演算法優於 Ham *et al.* (1985) 與 Kuo & Inman (1990) 所提的演算法。

除了上述探討 ELSP 不同的延伸主題外，在求解的方法上，本研究也發現某些研究嘗試使用基因演算法(GA)來求解 ELSP。Khouja *et al.* (1998) 使用 GA 來求解 ELSP，其利用 Bomberger (1966) 的產品資料作數據實驗，得到總成本比以動態規劃法(DP)所得的結果來得低；主要原因是使用 BP 法來安排產品的生產週期。Moon *et al.* (2002) 基於批量可以隨著時間變動大小，應用混合式 GA 來解 ELSP。Sarker & Newton (2002) 使用 GA 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。本節前面介紹 Chang *et al.* (2006) 求解不確定與變動需求的 ELSP 問題、Moon *et al.* (2006) 的 GT-ELSP 和 Chang & Yao (2009) 的 ELSP 考慮重製問題，也都是應用 GA 來進行計算。

從上述的文獻中，本研究發現 CC 法是 ELSP 延伸研究中，較常被使用的週期型態。同時，也未見有學者在 ELSP 主題中考慮到產品檢驗問題。因此，本論文就以 ELSP 考慮檢驗狀況作為研究主題。而有關檢驗狀況相關的文獻討論見 2.3 節。

2.3 批量問題考慮檢驗情況之相關研究

在會產出瑕疵品的生產系統下，為避免不良品產生時無法即時發現或處理，實行產品檢驗機制是非常重要的。若生產系統無檢驗機制，一旦生產過多不符合顧客需求的不良品，會對企業造成包括原物料、重製作業與修護工作等成本的增加。再者，若是不良品被售出，可能會造成廠商拒收或保固服務成本的負擔。因此，對企業而言，生產過程中施行產品檢驗作業固然會增加成本(檢驗人員的時間成本等)，但卻可以降低不良品過多的產生與買家拒收的風險。

過去二十年間，有許多的學者針對生產系統中，產品的最佳檢驗策略(決定生產週期內產品最佳的檢驗次數)發展出不同的研究模式。

Porteus(1986) 及 Rosenblatt & Lee (1986) 是在此研究領域的先導者。他們提出製程可透過檢驗的方式來進行監測，以得知是處於正常或非正常狀態；若是處於非正常狀態中的製程，可透過修復將其轉變為正常狀態。

Lee & Rosenblatt (1987, 1989) 針對會產出瑕疵品的生產，提出了在一個生產批量內，如何透過檢驗策略來監控制程的模式。該模式比較適用於大型批量生產的問題，原因是若生產批量小，檢驗次數增加需付出更多的費用來進行製程檢驗，其對總生產成本會造成負擔。Lee & Rosenblatt 並指出最佳的檢驗間距 (inspection interval) 是採用等距 (equally spaced) 的方式，換言之，每次檢驗同一種產品，其檢驗的批量大小是一樣的。之後，Lee & Park (1991) 在其生產週期與檢驗間距的研究中，將 Lee & Rosenblatt (1987) 的模式進行修正，並加入了重製成本 (reworking cost) 以及保固成本 (warranty cost) 兩項成本因素。Lin *et al.* (1991) 則針對 Lee & Rosenblatt (1987) 的模式，提出了一個判斷最佳檢驗策略的模式，其假設製程轉變為非正常狀態的機率是一遞增函數 (increasing failure rate, IFR)。Tseng (1996) 則指出若透過完整的事前維修策略 (a perfect preventive maintenance policy)，可取代在製程中進行檢驗的策略，進而改進會產出瑕疵品製程的可靠度。Tseng 的結果證實，若透過事前維修策略所得到的效果，將會比 Lin *et al.* (1991) 透過製程檢驗所得到的效果要來得好。Kim *et al.* (2001) 延伸 Lee & Rosenblatt (1987) 的研究，找到最佳的生產週期與檢驗排程。

Wang & Sheu (2001) 歸納 Porteus (1986) 的模式而發展出新的產品檢驗策略。Wang & Sheu 認為原料是整個生產系統中的主體，原料的經濟訂購量應該與生產的批量同時決定。對於有隨機失敗之會產出瑕疵品系統下，他們主張必需經由頻繁的檢驗，才能避免重製成本帶來的損失。Wang & Sheu 的設計主要是將原料因素放入會產出瑕疵品製程中，期望獲致總成本最小化，所以模式目的在決定製程檢驗時程和原料訂購次數。

Kim & Hong (1999) 則是假設不良品的檢驗是發生在製程結束後。後續 Wang (2005) 延伸 Kim & Hong (1999) 的研究，也是以生產結束後才進行產品檢驗的策略取代生產過程中的全面檢驗方式 (consensus inspection)。

過去針對會產出瑕疵品製程的研究中，有許多學者期望透過檢驗動作，修正製程非正常狀態所帶來的影響。然而，執行檢驗後，檢驗的結果

也可能會發生誤差，對製程的品質造成影響。Liou *et al.* (1994)，Makis & Fung (1998) 以及 Makis (1998) 都在其會產出瑕疵品系統之 EPQ 模式的論文中，指出檢驗誤差 (inspection errors) 對其製程所帶來的影響。

綜觀上述文獻，雖然大部分學者在產品檢驗的討論，多數傾向於生產過程中進行檢驗，但本研究也發現實務上檢驗策略的採用與產業別有重要的關聯，有些產業會因製程或產品特性，需要採取其他不同的檢驗方式。本研究將就特定產業所選擇的檢驗策略，試圖找到其最佳的排程來進行討論。

再者，除了上述討論到產品檢驗保固成本，以及檢驗後的重製或維修成本外，若是每次進行檢驗時，檢驗機台需重新校正調整，其會衍生檢驗相關的設置成本(inspection setup cost)，這是在建立模式時，另外會衍生的成本項。

2.4 文獻評論與研究主題確立

綜觀上述文獻探討，發現在 ELSP 的延伸研究中，未見有檢驗相關問題的討論；同時也注意到檢驗策略問題在其他討論生產與存貨管理模式的文獻中受到重視。因此，本研究將 ELSP 加入檢驗策略的討論作為本論文的研究主軸，此舉將是創新的研究主題。實務上，產品生產與其半成品/成品的檢驗原是息息相關，檢驗工作所帶給生產系統的相關成本，對於生產的排程有決定的影響。

從 2.2 節中，本研究知道傳統 ELSP 主要討論最佳的生產排程與週期，但其實生產產能的限制，才決定了排程是否合理可行。因此，在此設定考慮檢驗策略的 ELSP 延伸研究中，除了將會加入產品檢驗排程的決定外，也會將檢驗機台的產能一併列入考慮，以確保檢驗作業合理可行。此項假設條件在過去的文獻中也未見被討論。

關於本研究設定的檢驗策略，雖然在 2.3 節見有過去文獻討論不同的檢驗策略，但如同 1.1 節說明檢驗策略的決定需視製程與產品特性。而本論文研究將採取產線生產部分批量後，移至檢驗機台進行批量全數檢驗的策略。此檢驗方式在電腦面板的生產已有實際案例可循。

再者，因假設為會產出瑕疵品系統下產生不良品的檢驗問題，本論文研究不良品的發生(非正常狀態)將採用固定機率。雖然，在 2.1 節中說明過去的文獻採用不同的機率分配來表示會產出瑕疵品的生產狀態，但有鑑於本研究為初始研究，為求模式求解過程不確定性問題降低，決定將生產轉變非正常狀態的機率先以確定性做為參數。

第三章 共同週期法求解經濟批量、檢驗與排程問題

3.1 前言

本章研究主要提出一個在會產出瑕疵品生產系統下，考慮有檢驗狀況的經濟批量排程問題模式。3.2 節將詳細介紹此模式的符號定義、假設與建構方法。本章目標在求取最小時間單位總成本下之最佳生產週期與各產品的最佳檢驗次數。建構的數學模式會將生產產能與檢驗產能因素列入限制條件，目的為求取合理且可行的解。由於本章為本論文研究主題的初次探討，故週期排程採用簡單與確保可行之共同週期法(CC)類型，使所有產品共有相同長度的生產週期時間；而產品檢驗策略採行全數檢驗 (consensus inspection) 的假設。本章設計一搜尋演算法 (search algorithm) 來求解上述條件的模式，在 3.3 節將介紹此搜尋演算法求取最佳解的重要理論性質；3.4 節就搜尋演算法的搜尋上、下界與搜尋過程做詳盡說明；本章在 3.5 節會以一個 10 項產品的數值範例，說明本模式為一個有效求解經濟批量與檢驗排程問題的演算法；而後執行相關的隨機實驗，以瞭解產品項目數與產能率兩項因素的改變，對平均總成本與模式執行時間的影響；然後再進行此模式的效能分析。本章的最後將針對此模式做研究結論與討論。

3.2 模式建構

3.2.1 模式符號定義

本章研究模式建立所使用的數學符號，說明如下：

- n ：產品項的數目。
- d_i ：產品 i 每單位時間(天)的需求率。
- p_i ：產品 i 每單位時間(天)的生產率。
- y_i ：產品 i 每單位時間(天)的檢驗率。
- s_i ：產品 i 的生產設置時間。
- z_i ：產品 i 的檢驗設置時間。
- A_i ：產品 i 的生產設置成本。

C_i : 產品 i 的檢驗設置成本。

v_i : 產品 i 每單位的檢驗成本。

r_i : 產品 i 每單位的修復成本。

π_i : 產品 i 每單位的不良品成本。

h_i : 單位時間內持有一單位產品 i 的存貨持有成本。

α_i : 產品 i 的每個檢驗批量在製造過程中，由「正常狀態」轉變為「非正常狀態」的機率；一旦該檢驗批量的製程為非正常狀態，就只能生產出不良品。

本章研究模式有兩個決策變數：

T : 每個批量的生產週期。

m_i : 產品 i 在一次生產週期中的檢驗次數。

$TC(T, \mathbf{m})$: 所有產品在單位時間內的總成本，其中 $\mathbf{m} = (m_1, m_2, \dots, m_n)$ 代表每一項產品 i 在一生產週期內檢驗次數的組合(multiplier)。

3.2.2 模式假設

本模式共有十一項假設說明：

1. 在任何時間點內，所有產品的需求率、生產率、整備時間、單位檢驗時間、整備成本、單位檢驗成本和存貨持有成本等皆是已知且不隨時間而改變。本研究假設每一項產品的生產率大於檢驗率且大於需求率。換言之， $p_i > y_i > d_i$ 。
2. 所有產品皆由同一部生產機台進行生產，且週期排程是以共同週期法(CC approach)為基礎。
3. 在任何一個時間點上，生產機台只能生產一種產品。
4. 生產與檢驗之整備成本及兩者的整備時間，只與被生產出來的產品本身有關，與生產的順序、檢驗的順序及批量大小無關。
5. 某項產品在一生產週期中，其每一送檢驗的批量的大小皆為相同。在一生產週期中的生產數量，將被分成整數批的批量送至單一特定檢驗機台進行全數檢驗 (consensus inspection)。

6. 直到檢驗機台完成檢驗目前生產機台剛完成製造之檢驗批量，生產機台才會開始下一批該產品之檢驗批量或下一批產品的第一批檢驗批量。
7. 當生產機台開始生產每個檢驗批量，其製程皆處於正常狀態。當製造程序開始產生不良品時，因無法透過檢驗得知生產機台而會持續製造直到整個檢驗批量生產完成。
8. 一旦被檢驗機台檢驗出的不良品將會送至修復部門 (reworking department) 進行修復。修復部門將負責不良品的完整修復，使其修復後的產品能因應顧客的需求，不會發生需求短缺。同時，修復部門也需負責修復後產品的檢驗工作。因此，修復工作並不會佔用到生產機台及檢驗機台的產能。
9. 缺貨(shortage) 狀況不被允許發生。
10. 假設生產部門、檢驗部門與修復部門三者之間的產品運送時間非常短，其對模式影響不顯著。
11. 本研究假設不良品發生的狀況為輕微 (minor) 瑕疵，其所需耗費修復時間非常短，因而本研究假定修復時間可以被忽略 (negligible)。

在第二章文獻探討中，發現過去的文獻中並未有同時探討檢驗時間與檢驗產能限制問題。因此，在本研究的模式中將列入生產產能與產能產能的限制式，使其合理並貼近實務的生產狀況。

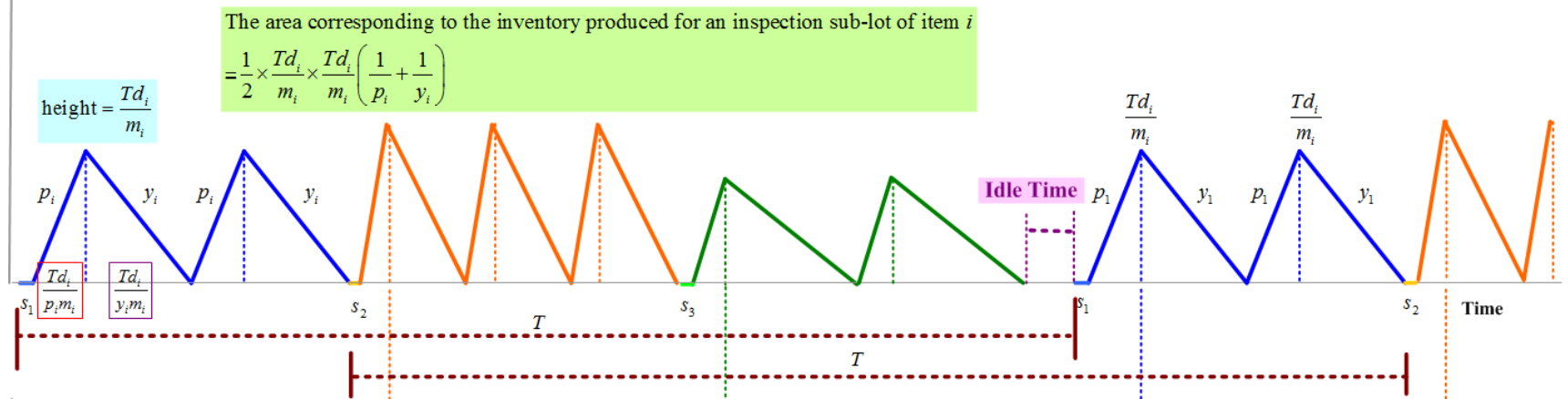
3.2.3 模式建立

建構模式前，本研究先就目標函數可能發生的成本項進行討論，其中包含五項成本：設置成本、檢驗成本、存貨持有成本、不良品所帶來的期望修復成本以及期望產能損失成本。本研究希望藉由對各項成本的個別分析，設計出研究主題的目標函數式。

本章研究主要探討在考量製程可能發生不良品的前提下，如何透過檢驗次數的安排與生產週期的控制，使得在單一生產機台生產 n 種不同產品的生產系統，能達到單位時間內最低的平均總成本。

首先針對生產單一項產品 i 的生產批量為 Td_i 時，分析所發生的相關成本，然後再討論生產 n 項產品的平均總成本。

The inventory level of the items produced for inspection



The inventory level of the items that have finished their inspection (and repair, if necessary)

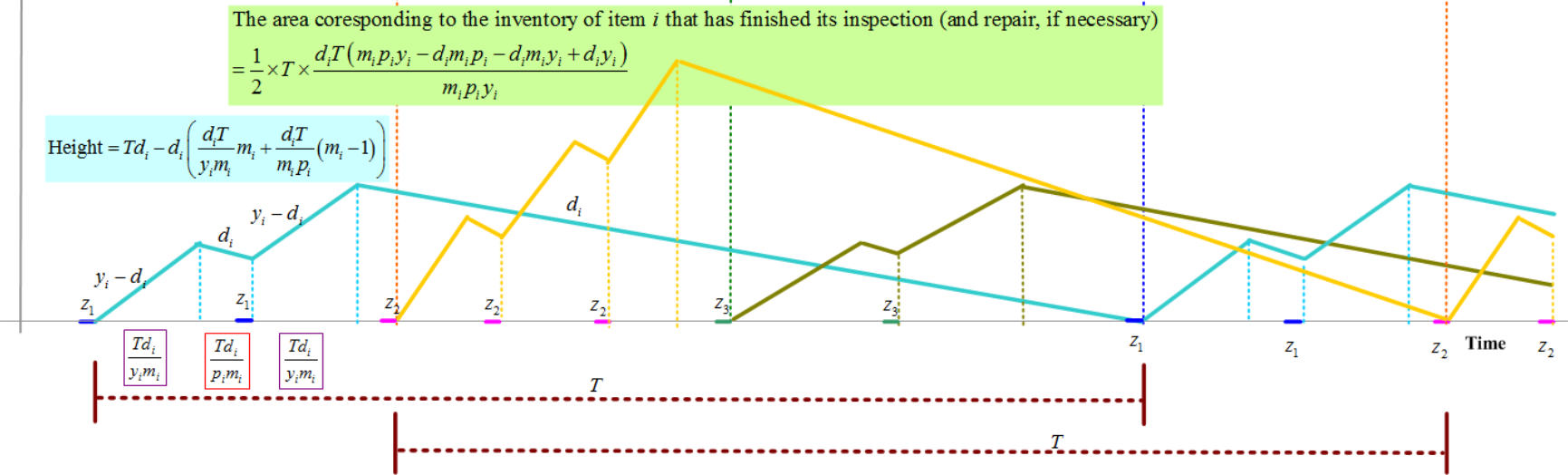


圖 3.1 使用 CC 法之生產與檢驗存貨

下面就上述所提的五項成本進行個別的分析：

1. 生產/檢驗設置成本 (production/ inspection setup cost)

當生產產品 i 時，需先進行生產線（單一生產機台）的設置工作。本研究假設生產線設置需花費的成本為 A_i ，同時每次生產皆要滿足一個生產批量 Td_i 。因此，當生產產品 i 一個生產批量 Td_i 時，其生產設置成本即為 A_i 。

同上段所述，當要檢驗產品 i 時，需先進行檢驗機台的設置工作。本研究假設當檢驗產品 i 時，每次的檢驗批量需花費設置成本為 C_i 。

有關生產設置成本與檢驗設置成本的發生，請見圖 3.1 使用 CC 法之生產與檢驗存貨圖。

2. 檢驗成本 (inspection cost)

Lee & Rosenblatt (1987) 提出，進行檢驗與修復所耗費的時間是可以忽略。本章研究修正其檢驗耗費時間可忽略的假設，考慮了單位檢驗時間，並於模式中加入檢驗產能的限制式。此限制式主要是觀察週期內所有產品檢驗所耗費的時間有無超過實際檢驗機台的負載，所得排程是否合理可行 (feasible)。因此，衡量檢驗成本時，除考慮檢驗次數所花費的固定成本（見前項檢驗設置成本 C_i 說明）外，還需考慮到檢驗時間（單位數）所帶來的成本（此為檢驗之變動成本）。

已知產品 i 在檢驗機台每次檢驗批量將發生固定成本 C_i ，而產品 i 每單位檢驗成本為 v_i 。若一個生產批量 Td_i 下將進行 m_i 次檢驗，則每一次的檢驗批量大小為 Td_i / m_i ，生產批量 Td_i 下的檢驗固定成本為 $C_i m_i$ ；而變動成本為 $Td_i v_i$ 。

3. 期望修復成本 (expected restoration cost)

根據 3.2.2 節模式假設第 9 項，經檢驗後的不良品將會移至修復部門進行修復，本研究並將不涵蓋修復部門產能的探討，也不考慮不良品之修復時間（此相同於 Lee & Rosenblatt (1987) 修復所耗費的時間是可以被忽略的假設），但會考慮到修復不良品的期望修護成本。

依據 Porteus (1986, 1990) 的研究，本章研究推導產品 i 生產批量 Td_i 之期望不良品數目，說明如下：

產品 i 每隔 Td_i / m_i 的數量進行檢驗 (m_i 次等距檢驗)，根據 Porteus (1986)，可以推導出當生產檢驗批量 Td_i / m_i 個產品，會發生不良品的期望數為 $Td_i / m_i - [(1 - \alpha_i)(e^{(Td_i/m_i)\ln(1-\alpha_i)} / \alpha_i)]$ (見附錄 A.1)。Porteus 提出當生產機台轉變為非正常狀態的機率 α_i 趨近於 0 時，每一個檢驗批量 Td_i / m_i 的期望不良品數目為 $E[N_i | (Td_i / m_i)] = T^2 d_i^2 \alpha_i / 2m_i^2$ (N_i 是產品 i 的不良品數目)。因此，Porteus (1990) 指出產品 i 一個生產批量的期望修護成本為 $(T^2 d_i^2 \alpha_i / 2m_i^2) \times m_i \times r_i = T^2 d_i^2 \alpha_i r_i / 2m_i$ 。由上瞭解，本研究只適用於 α_i 很小的製造系統。

4. 期望產能損失成本 (expected opportunity cost)

當製程處於非正常狀態，所發生的不良品會引起額外的產能損失成本，造成總成本增加。當求取不良品的產能期望產能損失成本時，需先知道不良品的期望數目與生產一個不良品所造成的產能損失成本。換言之，產品 i 的不良品期望產能損失成本，即在一個生產批量 Td_i 下，所產出不良品的期望數乘上生產一個不良品所需的產能損失成本。

因本研究採取檢驗批量內產品的全數檢驗，前項所討論期望修護成本之期望不良品數目應與本項所求期望產能損失之期望不良品數目相等。因此，前述每一檢驗批量 Td_i / m_i 的期望不良品數目為 $E[N_i | (Td_i / m_i)] = T^2 d_i^2 \alpha_i / 2m_i^2$ ，若每單位所需要的產能損失成本為 π_i ，則在一個生產批量 Td_i 下的期望產能損失成本為

$$(T^2 d_i^2 \alpha_i / 2m_i^2) \times \pi_i \times m_i = T^2 d_i^2 \alpha_i \pi_i / 2m_i$$

5. 存貨持有成本 (inventory holding cost)

產品 i 的存貨持有成本分成兩個部分：第一個部分是生產產品 i 之 m_i 次檢驗批量的存貨持有成本(發生於生產機台的存貨變化)；第二個部分則是產品 i 完成檢驗工作後的存貨持有成本(發生於檢驗機台的存貨變化)。兩項存貨持有成本的示意圖請見圖 3.1 使用 CC 法之生產與檢驗存貨圖，推導詳細說明如下：

- (1) 生產產品*i*之 m_i 次檢驗批量的生產存貨面積：根據圖 3.1 使用 CC 法之生產與檢驗存貨圖的上半部圖形，當產品*i*生產所需檢驗批量結束後，將批量送至檢驗機台進行檢驗，直到全部數量送至檢驗機台後，才開始進行下一項產品的生產。

因此，產品*i*生產一個檢驗批量的存貨面積(最左邊的一個鋸齒狀三角形)是

$$\frac{1}{2} \times \frac{Td_i}{m_i} \times \frac{Td_i}{m_i} \left(\frac{1}{p_i} + \frac{1}{y_i} \right)$$

當中生產產品*i*一個檢驗批量耗費時間為 $\frac{Td_i}{p_i m_i}$ ，將檢驗批量全部

送至檢驗機台完成所耗費時間為 $\frac{Td_i}{y_i m_i}$ 。

所以，當生產出 m_i 個檢驗批量，產品*i*的存貨成本則是

$$\frac{1}{2} \times \left(\frac{Td_i}{m_i} \right)^2 \left(\frac{p_i + y_i}{p_i y_i} \right) \times m_i \times h_i = \frac{T^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i}$$

- (2) 產品*i*完成檢驗工作後的生產存貨持有成本面積：根據圖 3.1 使用 CC 法之生產與檢驗存貨圖的下半部圖形，說明檢驗機台接收到生產機台送來產品*i*的檢驗批量進行檢驗的存貨變化情形。

所以，產品*i*之 m_i 個檢驗批量的存貨面積(最左邊的一個鋸齒三角形)計算如下：

存貨三角形的底= T

存貨三角形的高(詳細說明請見附錄 A.2)

$$= Td_i - d_i \left(\frac{d_i T m_i}{m_i y_i} + \frac{d_i T (m_i - 1)}{m_i p_i} \right) = Td_i - d_i \left(\frac{d_i T (m_i p_i + m_i y_i - y_i)}{m_i p_i y_i} \right)$$

$$= Td_i - d_i \left(\frac{d_i T (m_i p_i + m_i y_i - y_i)}{m_i p_i y_i} \right)$$

$$= \frac{d_i T (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{m_i p_i y_i}$$

所以，一個檢驗後存貨鋸齒三角形的面積

$$\begin{aligned}
&= \frac{1}{2} \times T \times \frac{d_i T (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{m_i p_i y_i} \\
&= \frac{d_i T^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{2 m_i p_i y_i}
\end{aligned}$$

因此，產品*i*完成*m_i*個檢驗批量，存貨成本是

$$\frac{d_i T^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{2 m_i p_i y_i} \times h_i$$

綜合上述兩項存貨面積，本研究可得一個生產週期內，產品*i*的存貨總成本為

$$\frac{T^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i T^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i}$$

根據上述五項成本說明，當生產批量為*Td_i*時，單一產品*i*在一生產週期*T*內的設置成本、檢驗成本、期望修復成本、存貨持有成本以及不良品所帶來的期望機會成本的總和為：

$$\begin{aligned}
&A_i + c_i m_i + d_i v_i T + \frac{T^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i T^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i} + \\
&\frac{\alpha_i d_i^2 T^2}{2 m_i} (\pi_i + r_i), i = 1, 2, \dots, n
\end{aligned}$$

若在生產週期*T*內生產*n*種產品時，其總成本為：

$$\sum_{i=1}^n \left\{ A_i + c_i m_i + d_i v_i T + \frac{T^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i T^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i} + \frac{\alpha_i d_i^2 T^2}{2 m_i} (\pi_i + r_i) \right\}$$

而計算單位時間內，生產*n*種產品的平均總成本則為：

$$\sum_{i=1}^n \left\{ \frac{A_i + c_i m_i}{T} + d_i v_i + \frac{T d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i T (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i} + \frac{\alpha_i d_i^2 T}{2 m_i} (\pi_i + r_i) \right\}$$

由於本研究檢驗策略假設進行全數檢驗，所以每項產品檢驗成本*d_iv_i*為常數項，其並不會對目標函數式的最佳化過程產生影響。因此，本論文設計之目標函數式將不列入此項。

本研究以上述所討論之成本項，建構有考慮檢驗狀況的經濟批量排程問

題模式，並新命名為『經濟批量、檢驗與排程問題』(Economic Lot, Inspection and Scheduling Problem, ELISP)。此 ELISP 模式將可以協助決策者決定生產系統的最佳生產批量與檢驗排程，其數學模式表示如下：

$$\min TC(T, \mathbf{m}) = \sum_{i=1}^n \left\{ \frac{A_i + c_i m_i}{T} + \frac{T d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i T (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i} + \frac{\alpha_i d_i^2 T}{2 m_i} (\pi_i + r_i) \right\} \quad (3.1)$$

Subject to

$$\sum_{i=1}^n \left(s_i + \frac{T d_i}{p_i} + \frac{T d_i}{y_i} \right) \leq T \quad (3.2)$$

$$\sum_{i=1}^n \left(z_i + \frac{d_i T}{y_i} + \frac{d_i T (m_i - 1)}{m_i p_i} \right) \leq T \quad (3.3)$$

$$z_i \leq \frac{T d_i}{m_i p_i}, \quad i = 1, 2, \dots, n \quad (3.4)$$

$$m_i \leq T d_i, \quad i = 1, 2, \dots, n \quad (3.5)$$

本章研究以式(3.1)為目標函數，求取每單位時間最低的總成本。成本項包括了設置成本、存貨成本、不良品機會成本與不良品修護成本。本模式有兩個產能的限制條件：式(3.2)與(3.3)。式(3.2)為生產產能的限制式，表示所有被安排至生產設備上生產的產品項，其有關生產活動與檢驗活動時間加總不能超過循環週期 T 的長度(本研究採用的生產產能限制與傳統只限制所有生產活動時間的加總有所不同，式(3.2)中有考慮檢驗活動時間)；而每一項產品 i 佔用生產設備包含了生產設置時間(s_i)、生產所需時間($T d_i / p_i$)與批量全數檢驗被消耗的所需時間($T d_i / y_i$)。式(3.3)為檢驗產能的限制式，說明所有被安排至檢驗設備上之檢驗活動，其時間加總不能超過循環週期 T 的長度；而每一項產品 i 佔用檢驗設備包含了檢驗設置時間(z_i)、 m_i 個檢驗批量的總時間長度(每一個檢驗批量時間是 $T d_i / m_i y_i$)，以及製造 $(m_i - 1)$ 檢驗批量的總時間長度(製造每一個檢驗批量時間是 $T d_i / m_i p_i$)。有關式(3.2)與(3.3)的情況可參考圖 3.1。不等式(3.4)解釋每一項產品 i 檢驗設置時間的長度，須不能超過其檢驗批量的生產時間長度。最後，不等式(3.5)說明各項產品的檢驗次數必須小於其生產批量的大小。

3.3 最佳解理論性質分析

本節將針對目標函數式的性質提供相關的理論分析，並深入探討模式中接合點與凹性曲線的性質，以作為後續搜尋最佳解演算法的理論基礎。

3.3.1 目標函數式性質分析

本研究發現目標函數式的成本項可以拆解成兩部分：一部份成本項含有決策變數 m_i ，另一部份則不含有 m_i 。其目標函數式可重新整理為：

$$\sum_{i=1}^n \left\{ \frac{c_i m_i}{T} + \frac{T d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i^2 h_i T}{2 m_i p_i} + \frac{\alpha_i d_i^2 T (\pi_i + r_i)}{2 m_i} \right\} + \sum_{i=1}^n \left\{ \frac{A_i}{T} + \frac{T d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} \right\} \quad (3.6)$$

為了方便分析，我們定義 $TC_i(m_i, T)$ 為含有 m_i 的成本項，表示如下：

$$TC_i(m_i, T) = \frac{c_i m_i}{T} + \frac{T d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i^2 h_i T}{2 m_i p_i} + \frac{\alpha_i d_i^2 T (\pi_i + r_i)}{2 m_i}, \quad i = 1, 2, \dots, n. \quad (3.7)$$

因此，目標函數式可寫為：

$$TC(\mathbf{m}, T) = \sum_{i=1}^n TC_i(m_i, T) + \sum_{i=1}^n \left\{ \frac{A_i}{T} + \frac{T d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} \right\} \quad (3.8)$$

3.3.2 接合點與凹性函數曲線的性質

本節以式(3.8)進行接合點與凹性函數曲線性質的相關探討。本研究首先以單一項產品的函數式做理論分析，然後再針對多項產品的性質做說明。單一項產品的相關參數設定請見表 3.1 產品 i 的參數值，本研究以其畫出在不同檢驗次數 m_i 值下的 $TC_i(m_i, T)$ 曲線圖(見圖 3.2)。

表 3.1 單一產品 i 的參數設定

C_i	r_i	α_i	d_i	ρ_i	h_i	π_i
120	0.0002	0.1	200	0.01	0.00011	0.0015

假定 $\underline{TC}_i(T)$ 是產品 i 在 T 函數下的最低成本函數(取 T 軸上所有的 T 值對應到 m_i ，取最低值的 $TC_i(m_i, T)$)，換言之

$$\underline{TC}_i(T) = \min_{m_i \in \mathbb{N}^+} \{TC_i(m_i, T)\} \quad (3.9)$$

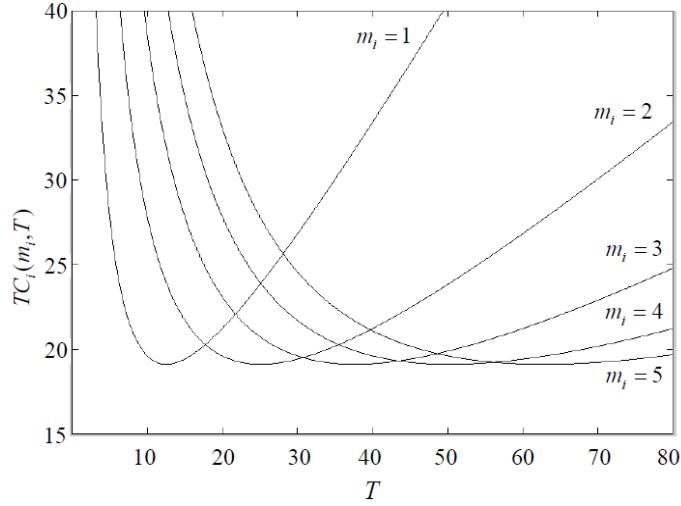


圖 3.2 不同 m_i 下的成本曲線 $TC_i(m_i, T)$

本研究定義 $\underline{TC}_i(T)$ 為沿著 T 軸最低的 $TC_i(m_i, T)$ 連結起來的曲線(見圖 3.3)。從圖 3.2 中，可以觀察到在不同 m_i 下，曲線交錯間有接合點 (Junction Point) 的存在。在此，本研究定義「結合點」為任兩個緊鄰的 $TC_i(m_i, T)$ 凹性曲線之交點。而這些由凹性曲線形成的接合點，決定了當產品 i 由檢驗次數 m_i 轉變為 $m_i + 1$ 時，所對應 T 的值將是多少。另外，任兩個緊鄰接合點間的曲線，為一條凹型連續曲線且存在一局部最小解 (local minimum)，其為 $TC_i(m_i, T)$ 的最低值。

因此，當令 $TC_i(m_i, T) = TC_i(m_i + 1, T)$ 時，可得到產品 i 接合點的時間封閉解 (closed-form) 為：

$$w_i(m_i) = T = \frac{1}{d_i} \sqrt{\frac{2c_i m_i (m_i + 1)}{h_i/y_i + 2h_i/p_i + \alpha_i(\pi_i + r_i)}}, \quad i = 1, 2, \dots, n. \quad (3.10)$$

詳細推導過程見附錄 A.3。

定理 3.1 若在一給定的生產週期 T 下，則產品 i 的最佳檢測次數為

$$m_i^*(T) = \begin{cases} 1, & T \in \left(0, \frac{2}{d_i} \sqrt{\frac{c_i}{h_i/y_i + 2h_i/p_i + \alpha_i(\pi_i + r_i)}} \right] \\ x, & T \in \left(\frac{1}{d_i} \sqrt{\frac{2c_i x(x-1)}{h_i/y_i + 2h_i/p_i + \alpha_i(\pi_i + r_i)}}, \frac{1}{d_i} \sqrt{\frac{2c_i x(x+1)}{h_i/y_i + 2h_i/p_i + \alpha_i(\pi_i + r_i)}} \right] \\ & \text{where } x \geq 2. \end{cases} \quad (3.11)$$

定理 3.1 提供本研究沿著 T 軸搜尋到產品 i 最佳的檢驗次數；而推論 3.2 則針對 $\underline{TC}_i(T)$ 的性質做更深入探討。

推論 3.2 令 $m_i^{(L)}$ 與 $m_i^{(R)}$ 分別表示產品 i 在凹性連續曲線 $\underline{TC}_i(T)$ 圖形上位於接合點左右兩邊曲線的檢驗次數，則產品 i 在接合點右邊的檢驗次數 $m_i^{(R)} = m_i^{(L)} + 1$ 。

[證明]：

假設 T_1 與 T_2 是位在某一特定接合點的 $w_i(m_i)$ 的左邊與右邊(換言之， $T_1 < w_i(m_i) < T_2$)，從定理 3.1 得知， T_1 與 T_2 的檢驗次數分別是 $m_i^{(L)} = x$ 與 $m_i^{(R)} = x + 1$ 。因此，對於正整數 x 來說，接合點右邊的檢驗次數等於接合點右邊的檢驗次數加上一，也就是 $m_i^{(R)} = m_i^{(L)} + 1$ 。■

圖 3.3 為 $\underline{TC}_i(T)$ 曲線的繪製。從圖形中可以發現對所有的產品項而言， $\underline{TC}_i(T)$ 是 T 的凹性曲線。

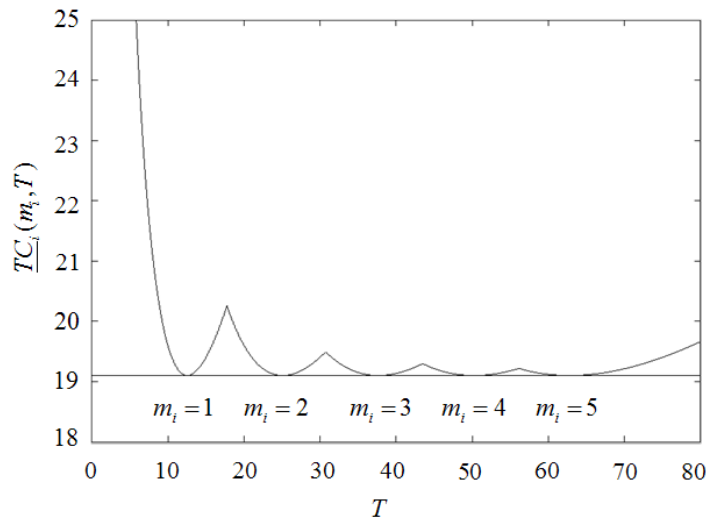


圖 3.3 $\underline{TC}_i(T)$ 的成本曲線

定理 3.3 $\underline{TC}_i(T)$ 是 T 的片段凹性 (piece-wise convex) 連續函數曲線。對產品 i 每一檢驗次數 m_i ， $\underline{TC}_i(T)$ 的局部最小值發生位置是在 $\tilde{T}_i(m_i) = (m_i/d_i) \sqrt{2p_i y_i c_i / [h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)]}$ ，其獲得的局部最小值 (即最低總成本) 為 $2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)} / \sqrt{2p_i y_i c_i}$ 。

[證明]：

假設給定一特定值 $m_i = \kappa$ ，且 $T \in [w_i(\kappa-1), w_i(\kappa)]$ ，從定理 3.1 得知，當 $T \in [w_i(\kappa-1), w_i(\kappa)]$ 時， $\underline{TC}_i(T) = TC_i(T, \kappa)$ 。從式(3.8)之 $TC_i(T, m_i)$ ，若是以 $TC_i(T, \kappa)$ 對 T 進行一階導數與二階導數，得到算式結果如下：

$$\frac{\partial}{\partial T} TC_i(T, \kappa) = -\frac{A_i + c_i \kappa}{T^2} + \frac{d_i h_i(p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \frac{d_i h_i(d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 \kappa p_i y_i}$$

$$\frac{\partial^2 TC_i(\kappa, T)}{\partial T^2} = \frac{2 \kappa c_i}{T^3} \quad (3.12)$$

從上得知，由於二階導數結果所得的值大於零，所以任意給定一 m_i 值，其 $TC_i(T, m_i)$ 是恆為開口向上的凹性曲線。

當 $TC_i(T, \kappa)$ 對 T 進行一階導數結果等於零時，即 $\partial TC_i(m_i, T) / \partial T = 0$ ，可以求得 $TC_i(T, m_i)$ 的局部最小值發生在

$$\tilde{T}_i(m_i) = (m_i / d_i) \sqrt{2 p_i y_i c_i / [h_i(p_i + 2 y_i) + \alpha_i p_i y_i (\pi_i + r_i)]} \quad (3.13)$$

■

本研究定義 $\underline{TC}(T)$ 是目標函數式 $TC(T, \mathbf{m})$ 沿著 T 搜尋的最佳解。因此，

$$\underline{TC}(T) = \sum_{i=1}^n \underline{TC}_i(T) + \sum_{i=1}^n \left\{ \frac{A_i}{T} + \frac{T d_i h_i(p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} \right\} \quad (3.14)$$

推論 3.4 將詳細說明 $\underline{TC}(T)$ 函數式的特性。

推論 3.4 $\underline{TC}(T)$ 是 T 的片段凹性連續函數曲線。

[證明]：

首先定義

$$g_i(T) = A_i / T + T d_i h_i(p_i y_i - d_i p_i - d_i y_i) / 2 p_i y_i \quad (3.15)$$

$$\text{然後由式(3.14)得知 } \underline{TC}(T) = \sum_{i=1}^n \underline{TC}_i(T) + \sum_{i=1}^n g_i(T)$$

從式(3.15)得知由於 $g_i(T)$ 的對 T 的二次導數是正數，因為若 $A_i, T > 0$ ，則

$$\frac{d^2 g_i(T)}{dT^2} = \frac{2 A_i}{T^3} > 0。因此，可以推論 $g_i(T)$ 是 T 的凹性函數。$$

由定理 3.3 已得知 $\underline{TC}_i(T)$ 是 T 的片段凹性曲線。

事實上 $\underline{TC}(T) = \sum_{i=1}^n \underline{TC}_i(T) + \sum_{i=1}^n g_i(T)$ ，若是加總所有 n 項產品的函數（包含每一項產品的 $g_i(T)$ ）以及所有 n 項產品的片段凹性曲線（包含每一項產品的 $\underline{TC}_i(T)$ ），可以推論 $\underline{TC}(T)$ 是 T 的片段凹性函數曲線片段凹性曲線。■

推論 3.5 對某產品 i 之每一個所有接合點都會承接在曲線 $\underline{TC}(T)$ 上成為接合點。

[證明]：

可以個別觀察 $\underline{TC}(T)$ 的函數，在不會失去維度的狀態下，假設 w 是產品 i 的接合點，但不是其餘 $(n-1)$ 個項目的結合點。所以，在下列的狀況下必存在 $\varepsilon > 0$ 。

1. 因為每一個 $\underline{TC}_j(T)$ 在區間 $[w-\varepsilon, w+\varepsilon]$ ，且 $j \neq i$ 是具有凹性性質。對於 $\sum_{j \neq i} \underline{TC}_j(T)$ 的曲線是具凹性性質。
2. 在區間 $[w-\varepsilon, w]$ 與 $[w, w+\varepsilon]$ ， $\underline{TC}_i(T)$ 是具有凹性性質。
3. 在區間 $[w-\varepsilon, w]$ 與 $[w, w+\varepsilon]$ ， $\sum_{i=1}^n g_i(T)$ 是具有凹性性質。

由於 $\underline{TC}(T) = \sum_{j \neq i} \underline{TC}_j(T) + \underline{TC}_i(T) + \sum_{i=1}^n g_i(T)$ ， $\underline{TC}(T)$ 在區間 $[w-\varepsilon, w]$ 與 $[w, w+\varepsilon]$ 之間是有凹性性質。因此， w 是 $\underline{TC}(T)$ 曲線上的接合點。■

3.4 搜尋演算法

本節設計一搜尋演算法，其中將考慮生產產能與檢驗產能的限制，以其求解 ELISP 的最佳解。3.3.1 與 3.3.2 節顯示，我們可沿著時間軸 T 搜尋到最佳解。

前節接合點與凹性曲線的理論分析，幫助本章研究藉由搜尋時間 T 軸，求得式(3.1)至(3.5)模式的解。但為了節省在時間軸上搜尋的時間，需要訂定搜尋時的範圍。換言之，需要定義搜尋時間 T 軸的下界 (low bound) 與上界 (upper bound)。再者，在此要計算出在搜尋範圍內 $\underline{TC}(T)$ 上每一條凹性曲線的區域最小值。而目標函數之全域最佳值即落在搜尋上下界之

間，也就是發生於所有區域最小值當中的最小值。在後續求取搜尋上下界的過程中，本研究將分別予以證明上述的主張。

在接下來的小節中，首先介紹搜尋演算法下界的推演，然後說明如何利用接合點進行搜尋，以及搜尋法的上界如何在搜尋過程中持續修正；最後詳細解說整個搜尋演算法的步驟。

3.4.1 搜尋演算法的下界

本章研究在推導搜尋下界 T_{LB} 時，需同時考量三個情況：(1)共同週期法 (CC approach) 所產生的最小週期時間，即令 $m_i = 1, i = 1, 2, \dots, n$ ；(2)使產品生產維持為可行解所需的最小週期時間(參考式 3.2)；以及(3)使產品檢驗維持為可行解所需的最小週期時間(參考式 3.3)。符號說明如下： T_{CC} 代表由共同週期法獲得之最佳生產週期； T_{PC} 是使產品生產維持為合理可行的最佳生產週期； T_{IC} 是使產品檢驗維持為合理可行的最佳生產週期。為滿足上述三種狀況皆能成立，搜尋起始下界 T_{LB} 將為上述三項最佳生產週期當中的最大值。

$$\begin{aligned}
 T_{LB} &= \max \{T_{CC}, T_{PC}, T_{IC}\} \\
 &= \max \left\{ \sqrt{\frac{\sum_{i=1}^n (A_i + c_i)}{\sum_{i=1}^n \left\{ \frac{d_i h_i (d_i + p_i) + \alpha_i d_i^2 p_i (\pi_i + r_i)}{2 p_i} \right\}}}, \frac{\sum_{i=1}^n s_i}{\left(1 - \sum_{i=1}^n \frac{d_i}{p_i} - \sum_{i=1}^n \frac{d_i}{y_i}\right)}, \frac{\sum_{i=1}^n z_i}{1 - \sum_{i=1}^n \frac{d_i}{y_i}} \right\}
 \end{aligned} \tag{3.16}$$

推論 3.6 對 $\underline{TC}(T)$ 函數而言，在 $T < T_{LB}$ 的條件下，不會存在可行的區域最佳解。

[證明]：

推論 3.4 說明 $\underline{TC}(T)$ 是片段凹性曲線。其說明全域最佳解必然是眾多區域最佳解當中之。對任何一組檢驗組合乘數 $\mathbf{m} = (m_1, m_2, \dots, m_n)$ 而言，可以此計算所有產品的區域最佳解 $\check{T}(m_1, m_2, \dots, m_n)$ 。在 T 軸上本研究將目標函數式對 T 做一次導數且設定結果等於零，其計算如下：

$$\begin{aligned}\frac{\partial}{\partial T}TC(T, \mathbf{m}) &= \sum_{i=1}^n \left\{ -\frac{A_i + c_i m_i}{T^2} + \frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \frac{d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 m_i p_i y_i} \right\} \\ 0 &= \sum_{i=1}^n \left\{ -\frac{A_i + c_i m_i}{T^2} + \frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \frac{d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 m_i p_i y_i} \right\} \\ \frac{1}{T^2} \sum_{i=1}^n (A_i + c_i m_i) &= \sum_{i=1}^n \left\{ \frac{d_i h_i [d_i p_i + 2 d_i y_i + m_i (p_i y_i - d_i p_i - d_i y_i)] + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 m_i p_i y_i} \right\}\end{aligned}$$

因此，本研究可以得到封閉解 $\tilde{T}(m_1, m_2, \dots, m_n)$ 如下：

$$\tilde{T} = \sqrt{\frac{\sum_{i=1}^n (A_i + c_i m_i)}{\sum_{i=1}^n \left\{ \frac{d_i h_i [d_i p_i + 2 d_i y_i + m_i (p_i y_i - d_i p_i - d_i y_i)] + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 m_i p_i y_i} \right\}}} \quad (3.17)$$

由上式，本研究可以得知對所有的 i 來說，當 $m_i \geq 1$ 時， $\tilde{T}(m_1, m_2, \dots, m_n) \geq T_{CC}$ 。換言之，在 $T < T_{CC}$ 不存在有區域最佳解。

推論 3.4 說明本研究可以 T_{CC} 作為搜尋的起始點，從 T_{CC} 開始往 T 較大值的方向開始搜尋直到觸及到上界(T_{UB})才能停止。但是，另外必須考慮到最佳的下界 T_{LB} 必須是合理可行，換言之，必須同時符合生產產能的限制式

$$\sum_{i=1}^n \left(s_i + \frac{T d_i}{p_i} + \frac{T d_i}{y_i} \right) \leq T \text{ 與 檢驗產能的限制式 } \sum_{i=1}^n \left(z_i + \frac{d_i T}{y_i} + \frac{d_i T (m_i - 1)}{m_i p_i} \right) \leq T \text{ 條}$$

$$\text{件。因此，本研究藉由 } \sum_{i=1}^n \left(s_i + \frac{T d_i}{p_i} + \frac{T d_i}{y_i} \right) = T \text{ 與 } \sum_{i=1}^n \left(z_i + \frac{d_i T}{y_i} + \frac{d_i T (m_i - 1)}{m_i p_i} \right) = T$$

來求得 T_{PC} 與 T_{IC} ，而合理可行的最佳下界則是取 T_{CC} 、 T_{PC} 與 T_{IC} 三者的最大值。所以，在 $T < T_{LB}$ 的條件下，不存在可行的區域最佳解。■

3.4.2 接合點搜尋過程

本小節本研究將詳細說明利用接合點排序搜尋的過程。從推論 3.2 與推論 3.5 中，得知沿著 T 搜尋過程中，每碰到一個接合點 $w_i(m_i)$ 代表產品 i 的最佳檢驗次數將從 m_i 變成 $m_i + 1$ ，目的是為了尋找 $TC(T)$ 函數的最佳值。在沿著 T 軸搜尋的過程中，需要一個 n 維度的檢驗次數向量 $(w_1(m_1^* + 1), w_2(m_2^* + 1), \dots, w_n(m_n^* + 1))$ ，而每一個 $w_i(w_i^* + 1)$ 代表當產品 i 必須改變最佳檢驗次數時之下一個接合點的位置。換言之，當搜尋過程中隨著 T 值

增加時，會選擇維度中 w_i 最小值之 i 品項，將其檢驗次數加上一，也就是 $w_i(m_i^* + 1)$ 產生檢驗次數變動的解。自此之後，要必須更新維度以產生新的最佳檢驗次數。詳細說明如下：令 T_c 代表搜尋時間軸 T 目前的所在位置。 δ 代表產品 i 在 $w_i(m_i^* + 1)$ 擁有最小值的產品項目代號，所以

$$\delta = \arg \min_i \{w_i(m_i^* + 1) > T_c\} \quad (3.18)$$

本研究以 $\mathbf{m}(T_c)$ 代表在 T_c 位置所有產品最佳檢驗次數的組合，所以 $\mathbf{m}(T_c) = (m_1^*(T_c), m_2^*(T_c), \dots, m_n^*(T_c))$ 。再者，每一種產品的檢驗次數在搜尋過程中，皆不應當超過檢驗批量的大小。換言之， $m_i \leq Td_i$, $i = 1, 2, \dots, n$ 。因此，當 $m_i = Td_i$ 時，不能再繼續搜尋並改變產品 i 的 m_i ，因為此時 m_i 的次數已經是產品 i 生產一個，就執行一次檢驗的狀況。

若是 $m_i^*(T_c) \leq Td_i$, $i = 1, 2, \dots, n$ 時，本研究藉由推論 3.2 從 T_c 開始搜尋。在搜尋過程中，必須由式(3.19)更新在 $w_\delta(m_\delta^* + 1)$ 位置上所有最佳檢驗次數的組合。

$$\mathbf{m}(w_\delta(m_\delta^* + 1)) \equiv (\mathbf{m}(T_c) \setminus \{m_\delta^*\}) \cup \{m_\delta^* + 1\} \quad (3.19)$$

“\” 代表組合中去除的部分。

令 $\{w_j\}$ 代表搜尋法中找到的所有接合點並按照順序排列的組合， $w_0 \equiv T_{LB}$ 表示搜尋的起始點與下界， w_0 不一定為接合點。而搜尋的過程是以遞增方式(ascending order)搜尋接合點位置，因此 $w_{j+1} \geq w_j$, $j = 0, 1, 2, \dots$ 。從定理 3.1 中，本研究藉由 $\underline{TC}(T)$ 中 w_j 與 w_{j+1} 的位置，去計算出所有產品的最佳檢驗次數組合，換言之， $\mathbf{m}(w_j)$ 是在區間 $[w_j, w_{j+1})$ 下最佳產品檢驗次數的組合。再者，本研究定義 $\tilde{T}(\mathbf{m}(w_j))$ 為產品檢驗次數組合 $\mathbf{m}(w_j)$ 在時間軸 T 上區域最佳解的位置。

3.4.3 搜尋演算法的上界

搜尋進行過程中，當搜尋超過上界時即停止。所以，上界即為本研究所設計演算法的停止條件。假設 $\mathbf{m}(w_j)$ 是在兩個相鄰接合點 $[w_j, w_{j+1})$ 間的最佳檢驗次數的組合。根據 3.4.1 節，本研究可以得到其區域最佳解在 T 軸上

的位置 $\check{T}(\mathbf{m}(w_j))$ 。當在獲得第一個區域最佳值位置 $\check{T}(\mathbf{m}(w_0))$ 後，即可以計算出起始上界 T_{UB} 的位置。而在隨後進行搜尋過程中，當每計算出一個 $\check{T}(\mathbf{m}(w_j))$ 時，必須同步更新上界的值。

根據定理 3.7，在 $T > \beta$ 的情形下，沒有任何一個目標函數的值小於 $TC(\mathbf{m}(\check{T}), \check{T})$ ，則 β 可被當作此搜尋演算法的上界。以下說明了上界 β 的封閉解。

定理 3.7 若局部最小解發生在 \check{T} ，則搜尋演算法的上界 β 在

$$\beta = \frac{X + \sqrt{X^2 - 2 \left(\sum_{i=1}^n \frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i} \right) \left(\sum_{i=1}^n A_i \right)}}{\sum_{i=1}^n \left(\frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i} \right)} \quad (3.20)$$

其中

$$X = \sum_{i=1}^n \frac{A_i}{\check{T}} + \sum_{i=1}^n \frac{\check{T} d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \sum_{i=1}^n \phi_i(m_i^*(\check{T}), \check{T}) \quad (3.21)$$

以及

$$\phi_i(m_i^*(\check{T}), \check{T}) = \begin{cases} \frac{c_i}{\check{T}} + \frac{\check{T} d_i^2 [(p_i + y_i) h_i + h_i y_i + \alpha_i p_i y_i (\pi_i + r_i)]}{2 p_i y_i} - \frac{2 c_i d_i \sqrt{h_i (p_i + 2 y_i) + \alpha_i p_i y_i (\pi_i + r_i)}}{\sqrt{2 p_i y_i c_i}}, & m_i^*(\check{T}) = 1 \\ \frac{2 c_i d_i \sqrt{h_i (p_i + 2 y_i) + \alpha_i p_i y_i (\pi_i + r_i)} [(m_i^* + 1) - \sqrt{m_i^* (m_i^* + 1)}]}{\sqrt{2 c_i p_i y_i m_i^* (m_i^* + 1)}}, & m_i^*(\check{T}) \geq 2 \end{cases} \quad (3.22)$$

[證明]：

在推導演算法上界 β 前，本研究觀察到目標函數式中有兩項成本項會隨著搜尋中週期 T 的增加(由 \check{T} 增加至 T)而造成成本增加，說明如下：

1. 單位時間的存貨成本項會隨著週期 T 時間的增加而增加。當週期時間由 \check{T} 增加至 T 時，單位時間的存貨成本的增加量為

$$\sum_{i=1}^n \frac{T d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} - \sum_{i=1}^n \frac{\check{T} d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} \quad (3.23)$$

2. 單位時間的設置成本項會隨著週期 T 時間的增加而減少。當週期時間由 \bar{T} 增加至 T 時，單位時間的設置成本的減少量為

$$\sum_{i=1}^n \frac{A_i}{T} - \sum_{i=1}^n \frac{A_i}{\bar{T}} \quad (3.24)$$

由上面兩項成本可以得知，在沿著週期時間 T 軸搜尋過程中，目標函數式當中的成本項有的會隨之增加，有的會隨之減少。因此，本研究要先分析當週期時間由 \bar{T} 增加至 T 時，單項產品 i 的總成本 $TC_i(m_i, T)$ 變化情形。在此，首先定義 $\phi_i(m_i^*(\bar{T}), \bar{T})$ 為產品 i 由 \bar{T} 增加至 T 時總成本的改變量：

$$\phi_i(m_i^*(\bar{T}), \bar{T}) = TC_i(m_i^*(\bar{T}), \bar{T}) - TC_i(m_i^*(T), T) \quad (3.25)$$

$\phi_i(m_i^*(\bar{T}), \bar{T})$ 指的是從 \bar{T} 增加至 T 時，產品 i 總成本 $TC_i(m_i, T)$ 最大的改變量 (the maximum magnitude of decrement)。而在討論區域最小值 \bar{T} 時， $\phi_i(m_i^*(\bar{T}), \bar{T})$ 有兩個關於最佳檢驗次數的狀況需要分析，說明如下：

1. 當產品 i 最佳檢驗次數為1時，即 $m_i^*(\bar{T})=1$ ：從定理3.3，得知 $TC_i(m_i, T)$ 的最小值是 $2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)} / \sqrt{2p_i y_i c_i}$ ，可以由下式計算出 $\phi_i(m_i^*(\bar{T}), \bar{T})$ 。

$$\begin{aligned} \phi_i(m_i^*(\bar{T})=1, \bar{T}) &= \frac{c_i}{\bar{T}} + \frac{\bar{T} d_i^2 (p_i + y_i) h_i}{2p_i y_i} + \frac{d_i^2 h_i \bar{T}}{2p_i} + \frac{\alpha_i d_i^2 \bar{T} (\pi_i + r_i)}{2} - \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)}}{\sqrt{2p_i y_i c_i}} \\ &= \frac{c_i}{\bar{T}} + \frac{\bar{T} d_i^2 [(p_i + y_i) h_i + h_i y_i + \alpha_i p_i y_i (\pi_i + r_i)]}{2p_i y_i} - \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)}}{\sqrt{2p_i y_i c_i}} \end{aligned} \quad (3.26)$$

2. 當產品 i 最佳檢驗次數大於1時，即 $m_i^*(\bar{T}) > 1$ ：由圖3.3與推論3.2，得知 $TC_i(m_i, T)$ 是凹性曲線，其局部最小總成本是 $TC_i(m_i^*(\bar{T}), \bar{T})$ ，其具備有以下的性質：

$$TC_i(m_i^*(\bar{T}), \bar{T}) \leq \max \{ TC_i(m_i^*(\bar{T}), w_i(m_i^*(\bar{T}))), TC_i(m_i^*(\bar{T}), w_i(m_i^*(\bar{T}) + 1)) \} \quad (3.27)$$

同時，可以得知

$$TC_i(m_i^*(\bar{T}), w_i(m_i^*(\bar{T}))) \geq TC_i(m_i^*(\bar{T}), w_i(m_i^*(\bar{T}) + 1)) \quad (3.28)$$

換言之， $TC_i(m_i, T)$ 之左邊接合點的總成本會高於右邊接合點的總成本。

因此， $\phi_i(m_i^*(\bar{T}), \bar{T})$ 值的計算如下：

$$\begin{aligned}
& \phi_i(m_i^*(\bar{T}), \bar{T}) \\
&= TC_i(m_i^*(\bar{T}), \bar{T}) - \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)}}{\sqrt{2p_i y_i c_i}} \\
&= \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)} \left[(m_i^* + 1) - \sqrt{m_i^* (m_i + 1)} \right]}{\sqrt{2c_i p_i y_i m_i^* (m_i + 1)}}
\end{aligned} \tag{3.29}$$

綜合上述兩式，本研究在式(3.22)說明 $\phi_i(m_i^*(\bar{T}), \bar{T})$ 針對不同檢驗次數的應用。當沿著時間軸 T 搜尋超越上界 β 後，若是單位時間存貨持有成本增加量

$\left(\sum_{i=1}^n \frac{(T - \bar{T}) d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} \right)$ 大於單位時間設置成本
 $\left(\sum_{i=1}^n \frac{A_i}{T} - \sum_{i=1}^n \frac{A_i}{\bar{T}} \right)$ 減少量加上 $\phi_i(m_i^*(\bar{T}), \bar{T})$ 改變量的和，即表示可以停止繼續往後搜尋。因為此時目標函數式的總成本只會增加，同時也不會產生最佳的局部最小值。因此，當式(3.20)成立，可以進行推導：

$$\sum_{i=1}^n (T - \bar{T}) \frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} \geq \left(\sum_{i=1}^n \frac{A_i}{T} - \sum_{i=1}^n \frac{A_i}{\bar{T}} \right) + \sum_{i=1}^n \phi_i(m_i^*(\bar{T}), \bar{T})$$

令 $\beta = T$ ，所以

$$\begin{aligned}
& \sum_{i=1}^n \frac{\beta d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} - \sum_{i=1}^n \frac{\bar{T} d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} \geq \\
& \sum_{i=1}^n \frac{A_i}{\bar{T}} - \sum_{i=1}^n \frac{A_i}{\beta} + \sum_{i=1}^n \phi_i(m_i^*(\bar{T}), \bar{T})
\end{aligned}$$

透過移項與簡化

$$\sum_{i=1}^n \frac{\beta d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} + \sum_{i=1}^n \frac{A_i}{\beta} \geq \sum_{i=1}^n \frac{A_i}{\bar{T}} + \sum_{i=1}^n \frac{\bar{T} d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} + \sum_{i=1}^n \phi_i(m_i^*(\bar{T}), \bar{T})$$

同時，令

$$X = \sum_{i=1}^n \frac{A_i}{\bar{T}} + \sum_{i=1}^n \frac{\bar{T} d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2p_i y_i} + \sum_{i=1}^n \phi_i(m_i^*(\bar{T}), \bar{T})$$

$$\sum_{i=1}^n \frac{\beta d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \sum_{i=1}^n \frac{A_i}{\beta} \geq X$$

可以得到

$$\beta = \frac{X + \sqrt{X^2 - 2 \sum_{i=1}^n \frac{\beta^2 d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i} \sum_{i=1}^n A_i}}{\sum_{i=1}^n \frac{\beta^2 d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i}}$$

■

所以在搜尋過程中，當找到一個新的區域最佳解，本研究要藉由 $T_{UB} = \min\{T_{UB}, \beta\}$ 更新上界 T_{UB} 。

3.4.4 搜尋演算法的程序

本節將詳細介紹搜尋演算法的搜尋步驟。在前節介紹的搜尋過程中，本研究從搜尋法下界 T_{LB} 開始，以增加 T 值的方式進行搜尋，同時以排序過的接合點 $\{w_j\}$ 當作架構，並確保 $\underline{TC}(T)$ 函數式所有的區域最小解將介於 $[T_{LB}, T_{UB}]$ 間。本研究令 (T^*, \mathbf{m}^*) 為搜尋至目前的最佳解， T^* 與 \mathbf{m}^* 分別代表目前最佳週期與最佳檢驗次數組合。

在搜尋過程中，當在區間 $[w_j, w_{j+1})$ 內獲得新的 $\tilde{T}(\mathbf{m}(w_j))$ ，將更新目前手邊 (best-on-hand) 最佳解的資訊，包括：最佳平均總成本、循環週期長度與檢驗次數組合以及搜尋上界。詳細步驟說明如下：

1. 藉由檢查每個產品的最佳檢驗次數不超過其檢驗批量數量大小，測試解的可行性。若為可行解，執行步驟 2；否則固定 $m_i^* = u_i$ 。假若在搜尋過程中產品 i 之 $m_i^*(w_j) > u_i$ ，則執行步驟 2。
2. 藉由產品的生產產能與檢驗產能的限制式，測試 $(\mathbf{m}(w_j), \tilde{T}(\mathbf{m}(w_j)))$ 結果的可行性。如果合理可行，執行步驟 3；否則在符合生產與檢驗兩項限制式以確保可行性的前提下，選擇接合點中最小的 T 值，求得 $\tilde{T}(\mathbf{m}(w_j))$ 來做取代，然後執行步驟 3。
3. 檢查目標函數所得的總成本可行解是否比目前手邊最佳總成本解低，若是，則以所得的目標函數總成本取代成為目前手邊最佳成本解

(\mathbf{m}^*, T^*) ，同時更新手邊的最佳生產週期與檢驗次數組合。

4. 藉由式(3.20)的 β 來更新搜尋上界 T_{UB} ，以縮短搜尋範圍。

本研究將所設計之搜尋演算法的虛擬程式碼詳細說明如下：

1. 起始過程：

- (a) 以式(3.16)計算搜尋下界當作搜尋起始點，並令上界為 $T_{UB} = \infty$ 。
- (b) 以式(3.11)計算 $\mathbf{m}^* = \mathbf{m}(T_{LB})$ ；以式(3.10)獲得 $\{w_i(m_i(T_{LB})) | i = 1, \dots, n\}$ 與確保 $m_i^*(w_i) \leq u_i$ 。
- (c) 當 $T_c = T_{LB}$ ，令 $\delta = \arg \min_i \{w_i(m_i) > T_c\}$ 。
- (d) 計算 $\tilde{T}(\mathbf{m}(T_{LB}))$ ，令 $\mathbf{m}^* = \mathbf{m}(T_{LB})$ 與 $T^* = \tilde{T}(\mathbf{m}(T_{LB}))$ 。檢查 (\mathbf{m}^*, T^*) 的可行性。如果必要，更新 T^* 確保為可行，然後計算 $TC(\mathbf{m}^*, T^*)$ 。
- (e) 藉由式(3.20)求取搜尋上界 $T_{UB} = \beta$ 。以式(3.10)求取 $[T_{LB}, T_{UB})$ 的所有接合點並按遞增順序排列 $\{w_i\}$ 。否則，令 $TC(\mathbf{m}^*, T^*) = \infty$ 。
- (f) 令 $j = 1$ 與 $T_c = w_j$ 。

2. 搜尋過程：

- (a) 藉由 $\mathbf{m}(w_j) \equiv (\mathbf{m}(T_c) \setminus \{m_\delta\}) \cup \{m_\delta + 1\}$ 獲得 $\mathbf{m}(w_j)$ ，然後計算 w_{j+1} 的 $\delta = \arg_i \{w_i(m_i + 1) = w_{j+1}\}$
- (b) 檢查 $(\mathbf{m}(w_j), \tilde{T}(\mathbf{m}(w_j)))$ 是否可行；如果必要，更新 T 的值與確保可行並計算 $TC(\mathbf{m}(w_j), \tilde{T}(\mathbf{m}(w_j)))$ ，然後更新目前手邊的最佳上界 T_{UB} 。
- (c) 假若 $TC(\mathbf{m}(w_j), \tilde{T}(\mathbf{m}(w_j))) < TC(\mathbf{m}^*, T^*)$ ，更新 \mathbf{m}^* 與 T^* ，並且 $T^* = \tilde{T}(\mathbf{m}(w_j))$ 與 $\mathbf{m}^* = \mathbf{m}(w_j)$

3. 搜尋演算法的終止條件：假若 $w_{j+1} > T_{UB}$ ，輸出 \mathbf{m}^* 、 T^* 以及 $TC(\mathbf{m}^*, T^*)$ ，然後終止執行；否則，令 $j = j + 1$ 與 $T_c = w_j$ ，然後到步驟2。

3.5 數據實驗

在本節分成兩個部分：3.5.1 節首先以一數值範例 (numerical example) 介紹搜尋演算法的實際操作情況；3.5.2 節以隨機實驗 (random experiment) 說明所設計之搜尋演算法的有效情形；最後 3.5.3 節將做模式的效能分析。

3.5.1 數值範例

本節的數值範例主要為說明本研究所設計的搜尋演算法的實際操作情形。表 3.2 為產生一組 10 種產品項的參數 (表中設置成本 A_i 與單位存貨成本 h_i 參考 Fujita (1978) 之範例 1)。

表 3.2 數值範例 10 個產品的參數資料

項目	A_i	c_i	v_i	r_i	h_i	π_i
產品項	單位：仟元					
1	15	120	50	0.00002	0.00001	0.0015
2	20	50	50	0.00015	0.00007	0.0002
3	30	35	10	0.00018	0.00005	0.0011
4	10	45	260	0.00001	0.00004	0.0015
5	110	70	70	0.00025	0.00116	0.0008
6	50	60	160	0.00003	0.00011	0.0002
7	310	120	30	0.00014	0.00063	0.0025
8	130	24	40	0.00005	0.00246	0.0028
9	200	33	30	0.00001	0.00038	0.0022
10	5	250	20	0.00016	0.00002	0.0016

表 3.2 數值範例 10 個產品的參數資料 (續)

項目	d_i	P_i	y_i	s_i	z_i
產品項	單位/天			天/批量	天/檢驗批量
1	50	35000	33000	0.0125	0.0009
2	200	20000	18000	0.00125	0.0009
3	150	18000	16000	0.0025	0.0018
4	100	10000	7000	0.00125	0.0023
5	400	3500	3400	0.005	0.0022
6	400	6500	6300	0.0025	0.0030
7	150	3000	2900	0.001	0.0009
8	150	5000	4500	0.005	0.0010
9	150	6500	6400	0.0075	0.0027
10	150	16000	15000	0.00125	0.0014

從表 3.3 中，得知本範例最佳的生產週期為 $\bar{T} = 44.6327$ 天；最佳檢驗次數組合是 $\mathbf{m} = \{1, 2, 6, 2, 11, 6, 4, 7, 4, 3\}$ ；其目標函數之最低平均總成本為 \$181.929 仟元。每一產品之檢驗批量大小為 $\mathbf{u} = \{2231, 4463, 1116, 2232, 1623, 2976, 1674, 956, 1674, 2232\}$ 。

表 3.3 數值範例的結果

$(w_j, w_{j+1}]$	\mathbf{m}	\bar{T}_j	$TC(\mathbf{m}, \bar{T}_j)$
(13.509, 14.009]	(1,1,2,1,4,2,1,2,1,1)	19.795	\$203.691
...
(40.304, 42.413]	(1,2,5,2,10,6,4,7,4,3)	44.437	\$182.103
(42.413, 42.692]	(1,2,5,2,11,6,4,7,4,3)	44.232	\$181.995
(42.692, 45.179]	(1,2,6,2,11,6,4,7,4,3)	44.633	\$181.929
(45.179, 46.255]	(1,2,6,2,11,6,4,8,4,3)	44.894	\$181.939
(46.255, 46.462]	(1,3,6,2,11,6,4,8,4,3)	45.431	\$181.991
...
(304.038, 304.788]	(9,16,39,13,75,41,26,50,27,19)	180.469	\$248.032

3.5.2 隨機實驗

本節研究以一隨機實驗來說明本研究所設計之搜尋演算法的有效性 (effectiveness)。表 3.4 為實驗的參數組合，隨機實驗採用一致分配 (uniform distributions) 來產生實驗樣本。表中的 A_i 、 s_i 、 d_i 與 h_i 為參考 Carreno (1990) 的參數設定，由於本研究模式為創新探討，所以其餘所使用的參數則藉由實驗獲得。本研究實驗設備是採用 1G 記憶體，Centrino 1.7GHz CPU 的個人電腦。

根據表 3.4 的參數條件，本研究以不同水準的產能率 (utilization rate，在此定義的產能率 $\sum \rho_i$ 並非是 $\sum (d_i / p_i)$ ，而是指所有產品項的總生產時間加上檢驗批量的總消耗時間後除上循環週期 T) 與不同水準的產品項目數來觀察搜尋演算法的有效性。本研究隨機產生八種不同水準的產能率 (0.5 至 0.9) 以及八種不同水準的產品項目數 (5 至 40)，每一種水準組合產生 50 個隨機樣本。

本研究以平均總成本 $TC(\mathbf{m}^{LB}, T_{LB})$ 當作是成本的上界 (以 \overline{TC}_{UB} 來表示)。 T_{LB} 代表目標函數保證能產生可行解的最小週期時間。首先將 $TC_i(m_i, T_{LB})$ 對 m_i 做一次微分，得到在 T_{LB} 下的最佳檢驗次數 $m_i^{LB} = d_i T_{LB} \sqrt{\alpha_i (\pi_i + r_i) / 2}$ 。當得到上述 T_{LB} 與 \mathbf{m}^{LB} 兩個變數後，可以計算出上界 \overline{TC}_{UB} 如下：

$$\overline{TC}_{UB}(\mathbf{m}^{LB}, T_{LB}) = \sum_{i=1}^n \left\{ \frac{A_i + c_i m_i^{LB}}{T_{LB}} + \frac{T_{LB} d_i^2 (p_i + y_i) h_i}{2 m_i^{LB} p_i y_i} + \frac{d_i T_{LB} (m_i^{LB} p_i y_i - d_i m_i^{LB} p_i - d_i m_i^{LB} y_i + d_i y_i) h_i}{2 m_i^{LB} p_i y_i} + \frac{\alpha_i d_i^2 m_i^{LB} T_{LB} (\pi_i + r_i)}{2} \right\} \quad (3.30)$$

\overline{TC}_{UB} 是簡單與可行解的值，其可當作目標函數之最佳解的上界。令 TC^* 為本設計之演算法，以一以特定個案獲得之最佳成本。可用式(3.31)的 Δ 計算目標函數值的變異情況。須注意到的是：產能利用率會影響到所設計搜尋法求解的品質，因為本研究將生產與檢驗產能放入決策的考量因素中。

$$\Delta = \left(\overline{TC}_{UB} - TC^* \right) / \overline{TC}_{UB} \cdot 100\% \quad (3.31)$$

表 3.5 中，Avg. (%) Δ 則表示 50 個隨機案例之平均總成本的平均減少百分比；Avg. Run Time 則為 50 個隨機案例的平均執行時間。

表 3.4 隨機實驗的參數組合

參數	分配	參數	分配
A_i	UNIF(5,300)	d_i	UNIF(500,2000)
v_i	UNIF(5,400)	p_i	UNIF(15000,35000)
h_i	UNIF(0.0005,0.0025)	s_i	UNIF(0.002,0.005)
π_i	UNIF(0.0001,0.003)	z_i	UNIF(0.00015, 0.0003)
α_i	UNIF(0.01,0.05)	y_i	UNIF(3500,12000)
r_i	UNIF(0.00002,0.00025)	c_i	UNIF(20,250)

表 3.5 以 CC 法之不同水準 $\sum \rho_i$ 的實驗結果

產品數	$\sum \rho_i$	[0.50, 0.55)	[0.55, 0.60)	[0.60, 0.65)	[0.65, 0.70)
5	Avg. (%) Δ	13.58 %	12.72 %	9.33 %	8.81 %
	Avg. Run Time [†]	1.453	1.455	1.441	1.436
10	Avg. (%) Δ	12.76 %	10.93 %	8.15 %	7.80 %
	Avg. Run Time	1.461	1.457	1.467	1.445
15	Avg. (%) Δ	11.96 %	9.71 %	7.74 %	7.27 %
	Avg. Run Time	1.464	1.461	1.456	1.450
20	Avg. (%) Δ	10.68 %	8.65 %	7.36%	6.52 %
	Avg. Run Time	1.470	1.467	1.463	1.460
25	Avg. (%) Δ	9.75 %	7.82 %	6.44 %	5.28 %
	Avg. Run Time	1.477	1.474	1.471	1.466
30	Avg. (%) Δ	9.69 %	7.73 %	6.37 %	4.57 %
	Avg. Run Time	1.482	1.481	1.477	1.474
35	Avg. (%) Δ	9.22 %	7.87 %	6.18 %	4.33 %
	Avg. Run Time	1.516	1.503	1.471	1.483
40	Avg. (%) Δ	8.49 %	7.20 %	5.61 %	4.16 %
	Avg. Run Time	1.564	1.525	1.508	1.498

[†] 單位：秒

表 3.5 以 CC 法之不同水準 $\sum \rho_i$ 的實驗結果 (續)

產品數	$\sum \rho_i$	[0.70, 0.75)	[0.75, 0.80)	[0.80, 0.85)	[0.85, 0.90)
5	Avg. (%) Δ	7.27 %	4.63 %	3.62 %	2.16 %
	Avg. Run Time [†]	1.412	1.407	1.402	1.368
10	Avg. (%) Δ	5.41 %	4.83 %	2.32 %	1.86 %
	Avg. Run Time	1.437	1.433	1.430	1.374
15	Avg. (%) Δ	5.54 %	3.89 %	1.83 %	0.72 %
	Avg. Run Time	1.447	1.443	1.442	1.417
20	Avg. (%) Δ	4.29 %	2.85 %	1.57 %	0.62 %
	Avg. Run Time	1.458	1.451	1.451	1.450
25	Avg. (%) Δ	4.15 %	2.48 %	1.24 %	0.38 %
	Avg. Run Time	1.458	1.457	1.454	1.454
30	Avg. (%) Δ	3.55 %	2.16 %	1.05 %	0.19 %
	Avg. Run Time	1.469	1.467	1.462	1.458
35	Avg. (%) Δ	3.10 %	1.72 %	0.98 %	0.16 %
	Avg. Run Time	1.475	1.471	1.467	1.465
40	Avg. (%) Δ	2.75 %	1.52 %	0.70 %	0.12 %
	Avg. Run Time	1.487	1.525	1.476	1.468

[†] 單位：秒

3.5.3 效能分析

從表 3.5 中所顯示的結果，平均執行時間隨著問題 (產品的項目數) 的變大而增加。本研究依據 Neter, Wasserman 與 Kutnur (1986) 書中第九章多項式迴歸 (polynomial regression) 之多項式階次測驗程序 (polynomial order testing procedure) 計算。曲線配適 (curve-fitting) 顯示表 3.5 中執行時間針對問題大小呈顯現二次增長 (grows quadratically)。由於，上述分析說明本研究的所設計的演算法呈現是一個 $O(n^2)$ 的演算法，所以，本搜尋演算法是 CC 模式求解 ELISP 有效率的方式。

3.6 結論與討論

本研究主要針對有檢驗狀況的經濟批量排程問題進行探討。其藉由基本週期法週期模式並考量生產與檢驗產能限制，發展出搜尋演算法以求得在最小平均單位總成本下的最佳生產週期與最佳檢驗排程。本章的研究有三項主要貢獻，說明如下：

第一，根據第二章相關文獻的探討(參見 1.2 節)，本章的主題為經濟批量排程問題創新的延伸主題。所以，相關理論性質探討，可做為其他批量問題考量檢驗問題的重要參考。第二，根據第二章文獻探討，本文為近期主要針對考量多項產品檢驗的批量問題進行討論的文章，其對於實務上多項產品生產檢驗排程的安排，可提供決策的參考。第三，數據實驗結果顯示，本研究設計之搜尋演算法，可做為生產管理決策者求解此問題最佳可行解有效可行的工具。

第四章 延伸基本週期法求解經濟批量、檢驗與排程問題

4.1 前言

前章研究以共同週期法(CC)求解經濟批量、檢驗與排程問題，並設計一搜尋演算法來求得最佳解。根據第二章 2.1 節，文獻中有相關研究指出以延伸基本週期法(EBP)求解傳統的 ELSP，可獲得比以 CC 法求解相同問題更低的總成本。因此，本章研究的生產週期類型，將以 EBP 法配合使用二冪策略 (Power of Two, PoT) 來求解 ELISP；至於在求得多組候選解後，其模式可行性的驗證，則搭配合理解測試程序 (Proc FT) 來進行。本章研究目的除了求取最佳可行的生產與檢驗排程外，同時也想瞭解可否能獲得優於 CC 法的可行解，以及獲知其改善的幅度為何。

本章 4.2 節將介紹符號定義、假設與建構方法。研究目標與前章相同，主要在求取最小時間單位總成本下，最佳生產週期與各產品之的最佳檢驗次數，限制條件為生產產能與檢驗產能；而生產後產品的檢驗策略同前章，採行全數檢驗方式。本章研究將設計一混合基因演算法 (Hybrid Genetic Algorithm, HGA)，搭配前章之搜尋演算法產生多組候選解 (candidate solutions)。4.3 節將詳細說明此結合基因演算法、搜尋演算法以及合理解測試程序所形成的 HGA 方法。4.4 節以一個 10 項產品的數值範例說明本演算法可以有效求解經濟批量與檢驗排程問題模式；同時，在本章研究也會執行隨機實驗，其結果將與前章 CC 法求得的最佳可行解進行比較。隨機實驗將探討產品數與產能率兩項因素的改變對最佳平均總成本與執行時間的影響；在 4.4 節結尾將針對此模式進行效能分析。本章的最後 4.5 節將針對以此 EBP 法求解 ELISP 模式，做出本章研究的結論與討論。

4.2 模式建構

4.2.1 模式符號定義

本章研究模式所使用的數學符號與前章大致相同(請參考 3.2.1 節)，至於有改變與增加的參數，請見下列說明。

在前章的 CC 模式中，所有的產品項 n 共用相同的生產週期 T 。在本章使用 EBP 法放寬了這個限制，不同產品 i 的生產週期 T_i 要等同於 $k_i B$ ；換言

之， $T_i = k_i B$ 。 k_i 是一個正整數的乘數 (multiplier)，而 B 則代表一個生產排程的基本週期 (basic period)。因此，本章研究會有三個決策變數，其數學符號說明如下：

B ：生產的基本週期。

k_i ：產品 i 的基本週期乘數。

m_i ：產品 i 在一次生產週期中的檢驗次數。

$TC(B, \mathbf{k}, \mathbf{m})$ ：所有產品在單位時間內的總成本，其中 $\mathbf{k} = (k_1, k_2, \dots, k_n)$ 代表每一項產品 i 的基本週期乘數組合； $\mathbf{m} = (m_1, m_2, \dots, m_n)$ 則代表每一項產品 i 在一生產週期內檢驗次數組合。

在本章後段 4.3 節介紹混合基因演算法的求解步驟中，將會使用到不同意義的基本週期參數 B 。為方便分辨起見，其符號說明如下：

\bar{B} ：基本週期的區域最小值。

\tilde{B} ：藉由二元搜尋法 (Binary Search) 所獲得的基本週期。

B^* ：目前手邊 (best-on-hand) 擁有最小平均總成本的基本週期。

4.2.2 模式假設

本章模式的假設說明與前章 3.2.2 節假設內容大致相同。除了第 2 項假設中，週期排程是將共同週期法改為是以延伸基本週期法 (extended basic period approach, EBP) 為基礎之外。其餘假設說明皆與 3.2.2 節的假設相同。

4.2.3 模式建構

本章目標函數發生的成本項與前章相同，包含有：設置成本、檢驗成本、存貨持有成本、不良品所帶來的期望修復成本以及期望產能損失成本。各項成本的分析與推導與 3.2.3 節類似，但當中必須將原本決策變數共同週期 T 改成由基本週期 B 乘上產品的週期乘數 k_i 。

各項相關成本的分析同 3.2.3 節，下列將針對 EBP 模式下的五項成本進行說明。

The inventory level of the items produced for inspection

Item 1 (in Blue) $k_1 = 1 ; m_1 = 2$

Item 2 (in Orange) $k_2 = 2 ; m_2 = 3$

Item 3 (in Green) $k_3 = 2 ; m_3 = 2$

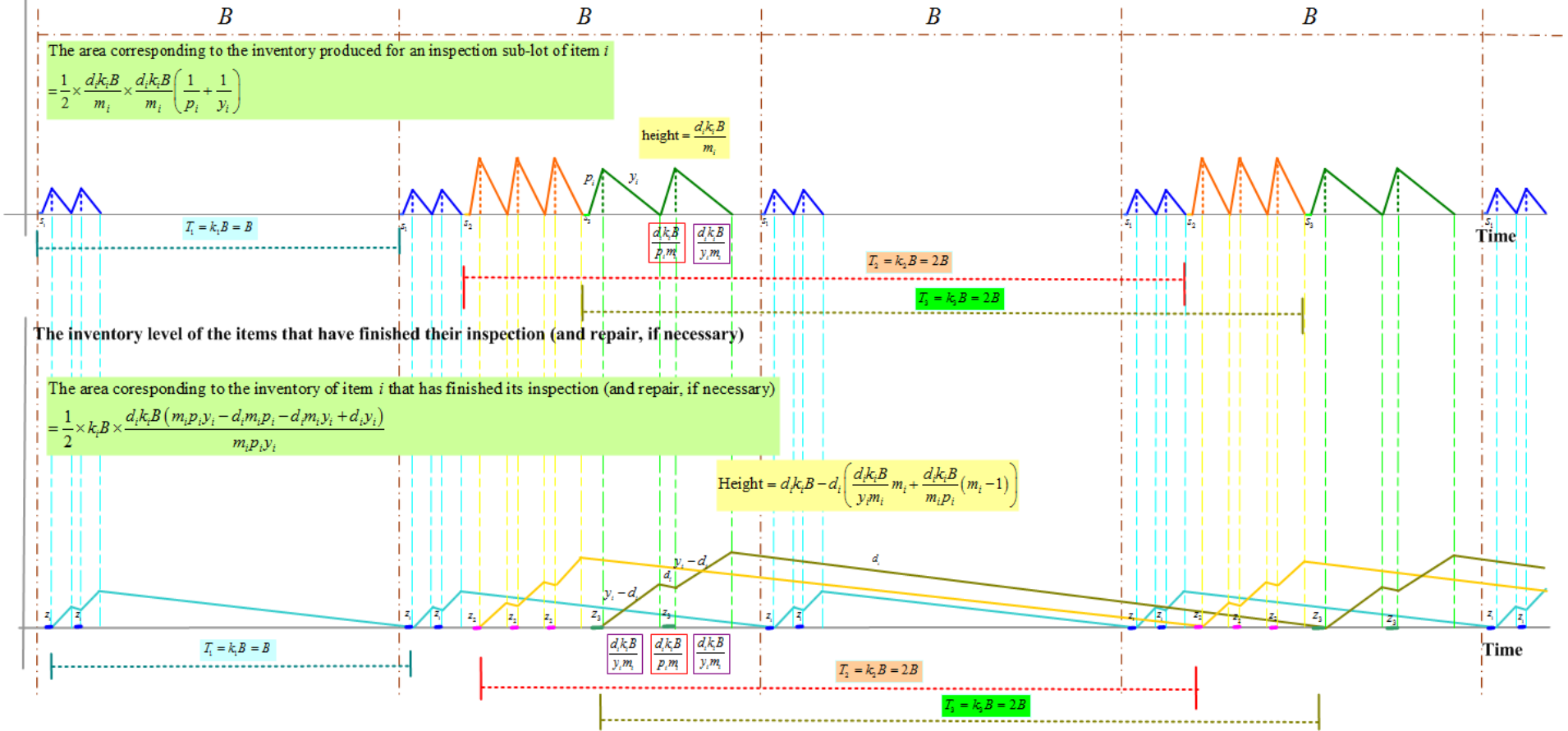


圖 4.1 使用 EBP 法之生產與檢驗存貨

1. 生產/檢驗設置成本：

每次生產要滿足一個生產批量 $k_i B d_i$ ，而其所需的生產設置成本為 A_i ；本研究假設當檢驗產品 i 時，每次的檢驗批量需花費檢驗機台設置成本為 C_i 。有關生產設置成本與檢驗設置成本發生的示意圖，請見圖 4.1 使用 EBP 法之生產與檢驗存貨圖。

2. 檢驗成本 (inspection cost)

已知產品 i 每次檢驗批量將發生固定成本 C_i ，而產品 i 每單位檢驗成本為 v_i 。若在一個生產批量 $k_i B d_i$ 下將進行 m_i 次檢驗，每一次檢驗的批量數目為 $k_i B d_i / m_i$ 。而生產批量 $k_i B d_i$ 下的檢驗固定成本為 $C_i m_i$ ，變動成本為 $k_i B d_i v_i$ 。

3. 期望修復成本 (expected restoration cost)

依據 Porteus (1986, 1990) 的研究，針對一產品的批量 $k_i B d_i$ 之期望不良品數目，詳細推演如 3.2.3 節所述，說明如下：

每一個檢驗批量的期望不良品數目為 $E[N_i | (k_i B d_i / m_i)] = k_i^2 B^2 d_i^2 \alpha_i / 2m_i^2$ 。因此，產品 i 一整個生產批量的期望修護成本為 $(k_i^2 B^2 d_i^2 \alpha_i / 2m_i^2) \times m_i \times r_i = k_i^2 B^2 d_i^2 \alpha_i r_i / 2m_i$ 。

4. 期望產能損失成本 (expected opportunity cost)

此成本為在一個生產批量 $k_i B d_i$ 下，所生產出不良品的期望數乘上生產一個不良品所需的產能損失成本。每一檢驗批量 $k_i B d_i / m_i$ 的期望不良品數目為 $E[N_i | (k_i B d_i / m_i)] = k_i^2 B^2 d_i^2 \alpha_i / 2m_i^2$ ，若每單位所需要的產能損失成本為 π_i ，則在一個生產批量 $k_i B d_i$ 下的期望產能損失成本為 $(k_i^2 B^2 d_i^2 \alpha_i / 2m_i^2) \times \pi_i \times m_i = k_i^2 B^2 d_i^2 \alpha_i \pi_i / 2m$ 。

5. 存貨持有成本 (inventory holding cost)

產品 i 的存貨持有水準包含兩個部分：第一個部分是生產產品 i 之 m_i 次檢驗批量的生產存貨持有成本；第二個部分則是產品 i 完成檢驗工作後的生產存貨持有成本。兩項存貨持有成本的表示，請見圖 4.1 使用 EBP 法之生產與檢驗存貨圖，詳細說明如下：

- (1) 生產產品 i 之 m_i 次檢驗批量的生產存貨面積：參考圖 4.1 使用 EBP 法之生產與檢驗存貨圖的上半部圖形，生產出 m_i 個生產批量，產品 i 的存貨成本是

$$\frac{1}{2} \times \left(\frac{k_i B d_i}{m_i} \right)^2 \left(\frac{p_i + y_i}{p_i y_i} \right) \times m_i \times h_i = \frac{k_i^2 B^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i}$$

- (2) 產品 i 完成檢驗工作後的生產存貨持有成本：根據圖 4.1 使用 EBP 法之生產與檢驗存貨圖的下半部圖形，產品 i 之 m_i 個檢驗批量的存貨面積（最左邊的三角形）計算如下：

一個檢驗後存貨三角形的面積

$$\begin{aligned} &= \frac{1}{2} \times k_i B \times \frac{d_i k_i B (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{m_i p_i y_i} \\ &= \frac{d_i k_i^2 B^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i)}{2 m_i p_i y_i} \end{aligned}$$

其存貨成本是

$$\frac{d_i k_i^2 B^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i}$$

綜合上述兩項計算存貨面積，可得產品 i 的存貨總成本為

$$\frac{k_i^2 B^2 d_i^2 (p_i + y_i) h_i}{2 m_i p_i y_i} + \frac{d_i k_i^2 B^2 (m_i p_i y_i - d_i m_i p_i - d_i m_i y_i + d_i y_i) h_i}{2 m_i p_i y_i}$$

根據過去 EBP 相關文獻，求取基本週期乘數 k_i 通常有兩種方式：一般整數 (General-Integer, GI) 與二冪 (PoT) 策略。在 GI 策略下每一種產品的 k_i 必須是正整數；然而 PoT 策略下，除正整數條件外，必須增加額外的限制，即對於所有產品 i ， $k_i = 2^x$ ， $x \geq 0$ 且為整數。近年來，PoT 策略常被應用在處理批量問題上。(Jacksons *et al.*, 1985; Roundy, 1989; Federgruen & Zheng, 1993)。將此兩種策略相比，PoT 策略比 GI 策略在處理可行性測試 (feasibility test) 上較有效率，原因是 $lcm\{k_i\} = \max\{k_i\}$ ，其 k_i 的最小公倍數在 PoT 策略下比 GI 策略容易求得。再者，從過去的文獻發現，PoT 策略下最差結果與 GI 策略所得結果不會差距太大，例如：Jackson *et al.* (1985) 推

演出 6% 的差距；Roundy (1989) 與 Federgruen & Zheng (1993) 提出了 2% 的差距。因此，本章 EBP 模式的研究針對基本週期將採用 PoT 策略。

此以 EBP 使用 PoT 策略求解 ELISP，數學模式如下所示：

$$\begin{aligned} \text{Min } TC(B, \mathbf{k}, \mathbf{m}) = & \\ \sum_{i=1}^n \sum_{j=0}^{O_i} \sum_{l=1}^{2^j} & \left\{ \frac{A_i + C_i m_i}{k_i B} + \frac{k_i B d_i (m_i p_i y_i - (m_i - 1) d_i p_i - (m_i - 1) d_i y_i + d_i y_i) h_i}{2 m_i p_i y_i} + \right. \\ & \left. \frac{k_i B \alpha_i d_i^2 (\pi_i + r_i)}{2 m_i} \right\} w_{ij\varphi(j,l)} \end{aligned} \quad (4.1)$$

Subject to

$$\sum_{i=1}^n \sum_{j=0}^{O_i} \left(s_i + \frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\varphi(j,t)} \leq B, \text{ for } t = 1, \dots, 2^{\max\{O_i\}} \quad (4.2)$$

$$\sum_{i=1}^n \sum_{j=0}^{O_i} \left(z_i + \frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\varphi(j,t)} \leq B, \text{ for } t = 1, \dots, 2^{\max\{O_i\}} \quad (4.3)$$

$$\sum_{j=0}^{O_i} \sum_{l=1}^{2^j} w_{ij\varphi(j,l)} = 1, \text{ for all } i = 1, \dots, n \quad (4.4)$$

$$\varphi(j,t) = \begin{cases} t \bmod 2^j & \text{if } t \neq \gamma 2^j, \gamma \in N \\ 2^j & \text{if } t = \gamma 2^j, \gamma \in N \end{cases} \quad (4.5)$$

$$w_{ij\varphi(j,l)} \in \{0,1\} \text{ for all } i, j, l \quad (4.6)$$

本章研究式(4.1)為 EBP 模式下的目標函數式，求取每單位時間最小的總成本。成本項之五項成本類型與第三章相同。再者，本模式有兩個產能的限制式，式(4.2)與(4.3)。式(4.2)為生產產能的限制式，其表示所有被安排至生產設備上之產品，其生產與檢驗活動的時間加總不能超過基本週期 B 的長度。式(4.3)為檢驗產能的限制式，其表示所有被安排至檢驗設備上之檢驗活動，其時間加總不能超過基本週期 B 的長度。兩項成本的詳細說明可參考 3.2.3 節之式(3.2)與(3.3)。式(4.4)依據 PoT 策略，調整產品 i 的生產時間長度必須被安排在前 k_i 個基本週期內。式(4.5)說明一旦產品 i 被安排至時間週期 t 內進行生產，生產週期將以 $t + \gamma 2^j$, $\gamma = 1, 2, 3, \dots$ 週期長度重複循

環。最後，式(4.6)以二元方式表示基本週期排程內有成本發生時為 1，無成本發生時為 0。

4.3 以基因演算法結合搜尋演算法求解的相關說明

本節將設計一個結合前章的搜尋演算法及合理解測試程序之混合基因演算法 (HGA)，求解使用 EBP 法下的 ELISP。本章基因演算法的主要目的為產生及演化不同組合的基本週期乘數組合 \mathbf{k} ；搜尋演算法的功能則是為在 B 與 \mathbf{k} 已知情形下，藉由個別產品檢驗次數的改變，搜尋最佳檢驗次數組合 \mathbf{m} ；而合理解測試程序與二元搜尋法主要是在生產與檢驗產能的限制下，求取三個決策變數的合理可行解。

本節將詳細介紹求解的程序與各項公式推導。4.3.1 節首先介紹基因演算法的建置；4.3.2 節說明如何利用前一章之搜尋演算法，協助每個染色體 (chromosome) 在演化的程序中搜尋到較好的檢驗次數組合 \mathbf{m} 。4.3.3 節介紹檢驗決策變數組合為可行的演算法—可行性程序測試 (Proc FT)。倘若以 Proc FT 找不到合理可行的生產與檢驗排程，4.3.4 節將進一步說明可藉由二元搜尋法 (binary search) 以修正基本週期的方式尋找可行解；最後，4.3.5 節將詳細說明 HGA 完整的進行步驟。

4.3.1 基因演算法的建置

本節基因演算法的建置，主要目的是藉由代數演化的概念，產生不同的基本週期乘數組合 \mathbf{k} ，並尋找最佳可行的決策變數組合 \mathbf{k} 。本節中，演算法的作法如同傳統基因演算法 (Holland, 1975) 的方式，需處理下列四項工作：(1) 染色體編碼 (chromosome representation) 與初始母體 (initial population)、(2) 適應函數計算與選擇 (fitness evaluation and selection)、(3) 基因運算子 (genetic operators) 設定，當中包括：複製 (reproduction)、交配 (crossover) 與突變 (mutation) 相關運作流程、(4) 參數設定 (parameter settings) 與終止條件 (termination conditions)。本章研究將針對上述此基因演算法之過程，詳細說明如下：

4.3.1.1 染色體的編碼與初始母體

在本章之混合基因演算法下，週期乘數組合 \mathbf{k} 中每一個產品的基本週期乘數 k_i ，本研究採用編碼方法所產生的位元字串 (bits string)，依序連接形成一個完整的位元字串，其稱之為染色體 (chromosome) 或個體 (individual)。舉例來說，定義產品 1 基本週期的乘數 k_1 值是以染色體的前面 u_1 位元來進行編碼，產品 2 的基本週期乘數 k_2 則是從第 $(u_1 + 1)^{\text{th}}$ 位元至第 $(u_1 + u_2)^{\text{th}}$ 位元來計算；後續產品的編碼則以此類推(見圖 4.2)。

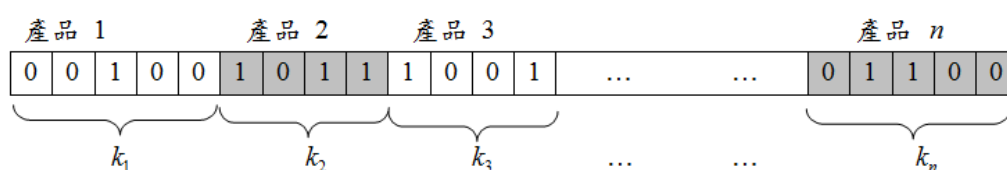


圖 4.2 染色體的編碼方式

因為本章基本週期的決定是使用二冪策略，每一個 k_i 是一個以 2 為底的整數，換言之， $k_i = 2^{x_i}$ ， x_i 是非負的正整數。在此 HGA 中，本研究以 k_i 之 2 的整數值次方 x_i 來做為染色體編碼。假若使用 $u_1 = 3$ 來表示所有 k_i ，然後 x_i 會有 $2^{u_1} = 2^3 = 8$ 可能的值 $\{0, 1, 2, \dots, 7\}$ ，其用二進位編碼如下： $(0,0,0)$, $(0,0,1)$, ..., $(1,1,1)$ 。

為了能表示每一項產品 i 的可能有的 k_i 值，本研究需要計算 k_i 的上限 (upper bound) 來表示在 HGA 中染色體個體編碼的情況。在此，本研究以 ELISP 目標函數式的下界與上界來推導 k_i 值的上界。首先， TC_i^{IS} 表示產品 i 使用獨立解法 (Independent Solution, IS) 所得的平均總成本，其當作下界。因此， $TC^{IS} = \sum_{i=1}^n TC_i^{IS}$ 表示 ELISP 目標函數式的下界。另一方面，本研究以 CC 法求解的結果 TC^{CC} 當作是上界。當給定任一 B 值，可以推演 x_i 的上界 $x_i^{UB}(B)$ 如下：(式(4.7)的詳細推導請見附錄 B.1)

$$x_i^{UB}(B) = \left\lceil \log_2 \left(\frac{TC^{CC} - \sum_{j \neq i} TC_j^{IS} + \sqrt{p_i y_i \left(TC^{CC} - \sum_{j \neq i} TC_j^{IS} \right)^2 - 2\Psi_i(A_i + C_i)}}{B\Psi_i} \right) \right\rceil$$

當中 $\Psi_i = d_i h_i (p_i y_i + d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)$

(4.7)

推導 k_i 時， B 值的下界以 B_L 來表示。由於基本週期的時間長度不能少於所有產品項目生產活動耗費的總時間；同時基本週期的時間長度也不能低於所有產品項目檢驗活動所耗費的總時間。因此，當 $B < \max_i (\max_i (s_i + k_i B d_i / p_i + k_i B d_i / y_i, z_i + k_i B d_i / y_i + k_i B d_i (m_i - 1) / m_i p_i))$ 時，不可能有可行的排程存在，否則擁有最大總生產時間或總檢驗時間的產品項將無法在基本週期時間內完成。 B 值下界 B_L 的公式如下：

$$B_L = \max_i \left(\max_i \left(s_i + \frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i}, z_i + \frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) \right) \quad (4.8)$$

因為 B_L 是 B 值的下界，而有 k_i 的上界 $2^{x_i^{UB}(B_L)}$ ，所以

$$x_i^{UB}(B_L) = \left\lceil \log_2 \left(\frac{TC^{CC} - \sum_{j \neq i} TC_i^{IS} + \sqrt{p_i y_i (TC^{CC} - \sum_{j \neq i} TC_i^{IS})^2 - 2\Psi_i (A_i + C_i)}}{B_L \Psi_i} \right) \right\rceil \quad (4.9)$$

再者，在 PoT 的策略下， k_i 下界的值是 1。因此，每一個 x_i 在下界值時是 $x_i^{LB} = 0$ 。

因為本研究藉由整數次方之二元向量來編碼 k_i 的值，因此必須建立每一個二元向量與整數間的對應關係。實際上，需要 u_i 個位元的二進位字串來編碼，而 u_i 最小的整數是 $2^{u_i} \geq x_i^{UB}(B_L)$ 。本研究可利用公式(4.10)來表示二進位字串與 k_i 值的對應關係。

$$\langle b_{u_i} b_{u_i-1} \dots b_1 \rangle = \left(\sum_{j=1}^{u_i} b_j 2^{j-1} \right) \quad (4.10)$$

如果發生對應整數的值大於 $x_i^{UB}(B_L)$ ，本研究使用補數計算(compliment computation) 的方式來確保其不會大於 $x_i^{UB}(B_L)$ 。總之，在 HGA 中，染色體個體的總長度是 $\sum_{i=1}^n u_i$ 個位元。

本章研究採以隨機的方式產生 $10n$ 組的基因個體當作初始母體，換言之，在演化過程中，設定母體大小即為 $10n$ 。

在取得起始母體前，本研究首先以 $k_i = 1, m_i = 1, \forall i$ 獲得特殊例子(particular case) 的可行解，此解將作為此 HGA 第一個手邊最佳解

(best-on-hand solution)；同時，此做法能藉由式(4.10)計算最佳 B 值（未考慮限制式的情況下）：

$$\tilde{B}(\mathbf{1}, \mathbf{1}) = \sqrt{\frac{\sum_{i=1}^n (A_i + C_i m_i)}{\sum_{i=1}^n \{d_i h_i [d_i p_i + 2d_i y_i + (p_i y_i - d_i p_i - d_i y_i)] + \alpha_i d_i^2 p_i y_i (\pi_i + r_i) / 2 p_i y_i\}}} \quad (4.11)$$

本研究將確認 $(\tilde{B}(\mathbf{1}, \mathbf{1}), \mathbf{1}, \mathbf{1})$ 是否合理可行；若不可行，則將另搜尋一組合理可行的解，下列說明初始母體可行性測試程序的虛擬程式碼。

初始母體可行性測試 (*Proc Initial*)

1. 令 $k_i = 1, m_i = 1, i = 1, 2, \dots, n$ ，以式(4.11)計算 $(\tilde{B}(\mathbf{1}, \mathbf{1}), \mathbf{1}, \mathbf{1})$ 。
2. 藉由 Proc FT（見 4.3.3 節）檢查 $(\tilde{B}(\mathbf{1}, \mathbf{1}), \mathbf{1}, \mathbf{1})$ 是否合理可行：假若可行，計算成本 $TC(\tilde{B}(\mathbf{1}, \mathbf{1}), \mathbf{1}, \mathbf{1})$ 與紀錄 $TC(B^*, \mathbf{k}^*, \mathbf{m}^*) = TC(\tilde{B}(\mathbf{1}, \mathbf{1}), \mathbf{1}, \mathbf{1})$ 當作起始手邊最佳解；若不可行，以二元搜尋法方式（見 4.3.4 節）尋找可行的 \tilde{B} ：若能獲得 \tilde{B} ，紀錄 $TC(B^*, \mathbf{k}^*, \mathbf{m}^*) = TC(\tilde{B}, \mathbf{1}, \mathbf{1})$ 當作起始手邊最佳解；再不然，指出參數設定不存在任何的可行解並終止執行 HGA。
3. 以隨機的方式產生 $10n$ 組的基因個體當作初始母體。

4.3.1.2 適應函數的計算與選擇機制

適應函數的目的是用來評估 GA 的效能函數，判斷某一組變數值之集合的適應程度；其能將最佳的函數多次複製，最差的函數快速淘汰。當要採用選擇機制時，必須先計算母體中每一個染色體的適應值。

在評估每一個染色體（一組基本週期乘數組合 \mathbf{k} ）的效能之前，本研究以前章所提的搜尋演算法（詳見 4.3.2 節），搜尋獲得一組最佳可行的生產與檢驗排程解（包含 B ， \mathbf{k} 與 \mathbf{m} ）。然後根據帶入目標函數式後所計算出來的總成本結果，做為評估染色體的適應性函數（fitness function）。

關於 GA 選擇機制，本研究先採用適應性函數正規化（fitness normalization）的程序來轉換適應函數，其正規化程序可以使得較佳的解，

在演化程序中能有較高的機率被挑選到 (Hunter, 1998)；而對於較差的染色體，在演化過程仍有機會存下來 (Boesel *et al.*, 2003)。

一般而言，適應值正規化可以分成三種形式：(i)逆向正規化 (inversion normalization) (ii)線性排序正規化 (linear ranking normalization) 與 (iii)非線性正規化 (nonlinear normalization) (Marzouk & Moselhi, 2003)。逆向正規化是簡單將所有染色體的適應函數，直接倒數做其正規適應函數；線性排序正規化是將母體內所有染色體先根據其適應函數值大小排序，然後依值高低給於相對的排名，其選擇壓力值 (Selection Pressure, SP) 可設定[1.0, 2.0] 的範圍內；至於非線性正規化的處理，其母體內所有染色體的排名方式與線性排序作法相同，但其選擇壓力值的設定範圍是[1.0, $i_{temp} - 2.0$](i_{temp} 代表排序後的名次)。

本章研究測試了前兩種方式：逆向正規化與線性排序正規化。根據實驗數據結果顯示，以線性排序正規化所得到的基本週期乘數指派結果稍優於使用逆向正規化所得的結果。

染色體 i_{temp} 線性正規化後的適應值 $eval_{i_{temp}}$ 的計算，說明如下：

$$eval_{i_{temp}} = 2 - SP + \frac{2(SP - 1)(i_{temp} - 1)}{PS - 1} \quad (4.12)$$

設定 $SP=1.5$

式(4.12)中，PS 代表母體大小 (population size)。 i_{temp} 代表排序後的名次，換言之，適應值最佳的排名最高 $i_{temp} = PS$ ；適應值最差的排名最低 $i_{temp} = 1$ 。

當適應性函數值經過正規化後，本研究使用輪盤式選擇機制 (roulette wheel mechanism) 來選擇需複製 (reproduction) 的染色體。每一個染色體被複製的機率與正規化適應性函數值的大小成正比關係。其複製機率的計算說明如下：

$$P_{i_{temp}} = eval_{i_{temp}} / \sum_{i_{temp}=1}^{PS} eval_{i_{temp}} \quad (4.13)$$

從上式得知，較高機率(等同於較高的適應性函數值)的染色體，在下一代的演化過程將有較高的機會被複製。

4.3.1.3 基因運算子

經由前節選擇機制存活下來的染色體，接下來將歷經兩個運算子：交配(crossover) 與突變 (mutation) 以產生子代。

交配的目的是藉由兩條染色體(雙親)的合併程序，產生具有雙親部分特性的子代；其希望能產生出更好適應性函數值的子代染色體。交配運算子的作法是隨機選擇兩條染色體，其在預定的交配機率(Crossover Rate, CR) 下交換當中的基因 (genes)。交配結果並不能保證一定會產生更好的子代，但是好的子代會繼續繁衍；而較差的子代會逐漸被淘汰。

常見的交配方式有單點 (single-point)、雙點 (two-point)和均勻 (uniform) 交配法 (圖 4.3)。單點交配法是以隨機選出一個切點，交換此切點後 (右邊) 的所有基因，產生新的個體；但切點前 (左邊) 的基因則維持不變。兩點交配法的做法和單點交配法類似，不同的地方是要隨機選出兩個切點，然後交換這兩個交配點之間的所有基因，而切點外其餘的基因則維持原狀。均勻交配法是對於位元字串中的每一個基因，以類似於擲銅板的方式來決定是否要交換。交換的機率為 0.5，而執行均勻交配前，針對每個基因隨機產生 0 或 1 的數值 (稱之為 Mask) 來決定是否要進行交換 (0 代表不用交換，1 代表要交換)。本章研究測試了雙點交配法和均勻交配法。

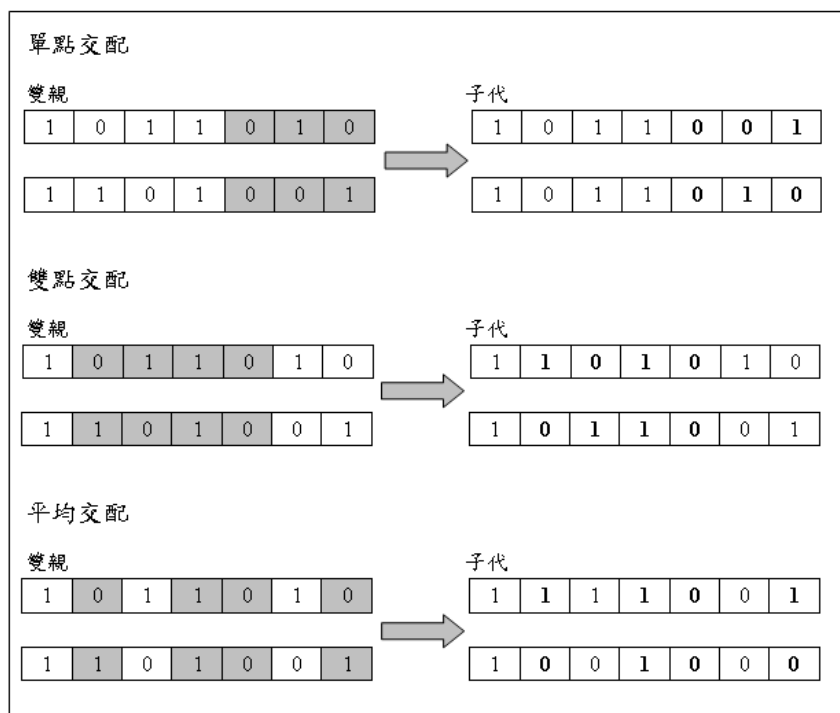


圖 4.3 三種交配運算子說明圖

倘若基因演算法僅執行上述的交配程序，過程中易於掉入區域最佳解。因此，在執行交配運算子之後，接著要對母體執行突變運算。突變運算子的目的主要是藉由突變率 (Mutation Rate, MR) 隨機改變染色體中的基因，使結果避免掉入區域最佳解，並同時增快系統演化的速度。本章採用的突變設定見下面 4.3.1.4 節的說明。

4.3.1.4 參數設定與終止條件

本章研究中 HGA 有四個主要的參數：演化的代數 (the number of generations)、交配率、突變率與母體大小。我們知道好的參數設定可以幫助 GA 在合理的執行時間下找到近似最佳解 (close-to-optimal solution)。

根據數據實驗，本研究建議 GA 的母數大小 (Population Size, PS) 設定為 $10 \cdot n$ (n 為產品項的數目)。而關於交配率與突變率的搭配，有兩組不同的設定：第一組是將交配率與突變率固定，設定交配率為 0.6 與突變率是 $1/n$ (交配率與突變率的設定是根據實驗結果得到最好的組合)；第二組設定是交配率與突變率在演化過程中以線性做改變。換言之，在演化的初期，先設定交配率在高的水準 0.9，突變率在低的水準 0.05；隨著演化過程，每增加一代交配率降低 0.001，而突變率則在每歷經 100 代時增加 0.01。交配率與突變率以此線性方式隨演化過程變動，直達到特定水準(如交配率與突變率皆為 0.2 時)時才停止。本研究期望此種線性改變能在交配率降低時產生相類似的染色體；並也希望藉由突變率的增加，讓在演化的末期仍然能找到不同的母體。

本章 GA 演化程序的終止條件有兩種選擇：當 GA 的演化程序歷經 50 代未見手邊最佳解有改善時，或是演化程序達到 1000 代時即停止演化。經由實驗設計的分析，結果發現兩種情況所得結果並無顯著的不同。因此，本章採用演化程序達到 1000 代數則停止運算作為終止條件。

4.3.2 搜尋演算法的程序

本節所使用搜尋演算法的搜尋過程類似於前章的作法，主要目的是為每個染色體找尋檢驗組合的最佳解 (前章我們假設 $k_i = 1, i = 1, 2, 3, \dots, n$)。在本節，本研究允許的 k_i 是在 PoT 策略下，從染色體解碼的整數；當已知基

本週期乘數組合 \mathbf{k} 時，本研究可運用搜尋演算法尋求最佳的 B 與 \mathbf{m} ，以獲得可行的生產與檢驗排程。

在詳細介紹本節使用的搜尋演算法前，首先簡單概要說明其作法。本搜尋法將沿著 B 軸進行搜尋，同前章一樣，要先定義 B 軸搜尋的上、下界。本研究將尋找 B 軸上每一個接合點的位置，然後藉由兩兩相鄰的接合點來找出最佳的 B 與 \mathbf{m} 。對任何一組被找到的解，其不僅應該是目標函數式的區域最小解，還必須確認是生產與檢驗排程的可行解。當沿著 B 軸在搜尋範圍間(上、下界之間)找到所有可行的區域最小候選解後，本章研究將取其中目標函數式總成本最低值者做為是研究的最佳解。

在接下的小節中，4.3.2.1 節首先說明搜尋的下界；4.3.2.2 節介紹搜尋過程中接合點位置與凹性曲線的相關推導公式；4.3.2.3 節解說搜尋的上界；最後 4.3.2.4 詳列搜尋法的搜尋步驟。

4.3.2.1 搜尋法的下界

本章搜尋法的下界 B_{LB} 與前章 3.4.1 節相似，需同時考量三個情況：(1) EBP 所產生的最小週期時間 B_{m1} ，其令 $m_i = 1, i = 1, 2, \dots, n$ ；(2) 使產品生產維持為可行解所需的最小週期時間 B_{PC} ；以及(3) 使產品檢驗維持為可行解所需的最小週期時間 B_{IC} 。為滿足上述三種狀況皆能成立，搜尋下界 B_{LB} 將為上述三項最佳生產週期的最大值。公式如下：

$$B_{LB} = \max \{ B_{m1}, B_{PC}, B_{IC} \} \quad (4.14)$$

$$\text{而 } B_{m1} = \sqrt{\frac{\sum_{i=1}^n \left(\frac{A_i + C_i}{k_i} \right)}{\sum_{i=1}^n \left\{ \frac{k_i d_i h_i [d_i p_i + 2d_i y_i + (p_i y_i - d_i p_i - d_i y_i)] + k_i \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}{2 p_i y_i} \right\}}}$$

$$B_{PC} = \frac{\sum_{i=1}^n s_i}{\left(1 - \sum_{i=1}^n \frac{k_i d_i}{p_i} - \sum_{i=1}^n \frac{k_i d_i}{y_i} \right)} \quad \text{與} \quad B_{IC} = \frac{\sum_{i=1}^n z_i}{\left(1 - \sum_{i=1}^n \frac{k_i d_i (m_i - 1)}{m_i p_i} - \sum_{i=1}^n \frac{k_i d_i}{y_i} \right)}$$

下界的推導過程可參考 3.4.1 節的做法。

4.3.2.2 接合點搜尋過程

當由基因演算法產生一組 \mathbf{k} ，可以利用 3.4.2 節中藉由沿著 B 軸上接合點的變化，找到最佳的檢驗次數組合 \mathbf{m} 。須注意到的是：當由 GA 獲得 $\underline{TC}(B, \mathbf{k}_i)$ 的最佳值，此節接合點提供產品 i 的最佳檢驗次數由 m_i 變成 $m_i + 1$ 的資訊。因此，當給予一組週期乘數 \mathbf{k} ，並找到在 B 軸上所有接合點的位置，可以藉由相鄰兩接合點，幫助找到最佳的檢驗次數組合 \mathbf{m} 。在兩相鄰的接合點之間，本研究尋找 \mathbf{k} 與 \mathbf{m} 組合下的區域最佳值，其結果不僅為兩點間目標函數式的最小值，也必須是可以產生可行 (feasible) 的生產與檢驗排程。關於在接合點上的搜尋，本研究首先找出產品 i 的檢驗次數 m_i 時，其在 B 軸上位置的封閉解 (closed-form)：

$$w_i(m_i) = \frac{1}{k_i d_i} \sqrt{\frac{2c_i m_i (m_i + 1)}{\frac{h_i}{y_i} + \frac{2h_i}{p_i} + \alpha_i (\pi_i + r_i)}} \quad (4.15)$$

從式(4.15)，可獲得當 B 已知，產品 i 的最佳檢驗次數，如式(4.16)所示。

$$m_i(B) = \begin{cases} 1, & B \in \left(0, \frac{2}{k_i d_i} \sqrt{\frac{c_i}{\frac{h_i}{y_i} + \frac{2h_i}{p_i} + \alpha_i (\pi_i + r_i)}} \right) \\ x, & B \in \left(\frac{1}{k_i d_i} \sqrt{\frac{2c_i x(x-1)}{\frac{h_i}{y_i} + \frac{2h_i}{p_i} + \alpha_i (\pi_i + r_i)}}, \frac{1}{k_i d_i} \sqrt{\frac{2c_i x(x+1)}{\frac{h_i}{y_i} + \frac{2h_i}{p_i} + \alpha_i (\pi_i + r_i)}} \right) \text{ where } x \geq 2. \end{cases} \quad (4.16)$$

同前章 3.4.2 節的做法，本研究的搜尋過程需要一個 n 維度的向量 $(w_1(m_1 + 1), w_2(m_2 + 1), \dots, w_n(m_n + 1))$ ，而當中每一個 $w_i(m_i + 1)$ 代表當產品 i 必須改變最佳檢驗次數時之下一個接合點的位置。進行搜尋過程中， B 值增加時，本研究會選擇維度中最小值 w_i 之產品 i ，將其檢驗次數加上一，也就是 $w_i(m_i^* + 1)$ 產生檢驗次數變動。令 B_c 代表搜尋時間軸 B 目前所在的位置。 δ 代表產品 i 在 $w_i(m_i^* + 1)$ 擁有最小值的項目代號，則

$$\delta = \arg \min_i \{w_i(m_i^* + 1) > B_c\} \quad (4.17)$$

本研究以 $\mathbf{m}(B_c)$ 代表所有產品最佳檢驗次數的組合，而每一種產品的檢驗次數在搜尋過程中不應當超過檢驗批量的大小。所以 $m_i \leq Bk_i d_i$, $i=1,2,\dots,n$ 。

參考推論 3.2，從 B_c 開始搜尋。在搜尋過程中，本研究可藉由式(4.18)更新在 $w_\delta(m_\delta^*+1)$ 位置上所有最佳檢驗次數的組合。

$$\mathbf{m}(w_\delta(m_\delta^*+1)) \equiv (\mathbf{m}(B_c) \setminus \{m_\delta^*\}) \cup \{m_\delta^*+1\} \quad (4.18)$$

“ \setminus ” 代表組合中去除的部分。

整個搜尋過程可參考 3.4.2 節的做法。當給定一組的 \mathbf{k} 與 \mathbf{m} 組合時，很明顯目標函數式是以 B 為導數的凹性曲線。本研究可藉由目標函數式對 B 一次導數等於 0 時來求得區域最小值的位置 $\tilde{B}(\mathbf{k}, \mathbf{m})$ 。公式如下：

$$\tilde{B}(\mathbf{k}, \mathbf{m}) = \sqrt{\frac{\sum_{i=1}^n (A_i + C_i m_i) / k_i}{\sum_{i=1}^n \{k_i d_i h_i [d_i p_i + 2d_i y_i + m_i (p_i y_i - d_i p_i - d_i y_i)] + k_i \alpha_i d_i^2 p_i y_i (\pi_i + r_i) / 2m_i p_i y_i\}}} \quad (4.19)$$

4.3.2.3 搜尋演算法的上界

當搜尋超過上界時演算法即立即停止，在此，本研究以 B_{UB} 表示上界的值。本章上界 β 的推導請參考定理 3.7 作法。在搜尋過程中，第一個區域最小值的位置是在 $\tilde{B}(\mathbf{m}(w_0))$ ，爾後每計算出一個 $\tilde{B}(\mathbf{m}(w_j))$ 時，必須同步更新上界的值。

若局部最小解發生在 \tilde{B} ，則搜尋演算法的上界 β 在

$$\beta = \frac{X + \sqrt{X^2 - 2 \left(\sum_{i=1}^n \frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i} \right) \left(\sum_{i=1}^n A_i \right)}}{\sum_{i=1}^n \left(\frac{d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{p_i y_i} \right)} \quad (4.20)$$

其中

$$X = \sum_{i=1}^n \frac{A_i}{\tilde{B} k_i} + \sum_{i=1}^n \frac{\tilde{B} k_i d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \sum_{i=1}^n \phi_i(m_i^*(\tilde{B}), \tilde{B}) \quad (4.21)$$

以及

$$\phi_i(m_i^*(\tilde{B}), \tilde{B}) = \begin{cases} \frac{c_i}{\tilde{B}k_i} + \frac{\tilde{B}k_i d_i^2 [(p_i + y_i)h_i + h_i y_i + \alpha_i p_i y_i (\pi_i + r_i)]}{2p_i y_i} - \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)}}{\sqrt{2p_i y_i c_i}}, & m_i^*(\tilde{B}) = 1 \\ \frac{2c_i d_i \sqrt{h_i(p_i + 2y_i) + \alpha_i p_i y_i (\pi_i + r_i)} [(m_i^* + 1) - \sqrt{m_i^*(m_i + 1)}]}{\sqrt{2c_i p_i y_i m_i^*(m_i + 1)}}, & m_i^*(\tilde{B}) \geq 2 \end{cases} \quad (4.22)$$

搜尋過程中，當找到一個新的區域最佳解，本研究要藉由 $B_{UB} = \min\{B_{UB}, \beta\}$ 更新上界 B_{UB} 。請見 4.3.5 節虛擬程式碼 (Pseudo Code) 的詳細說明。

綜合上述 4.3.1 與 4.3.2 節內容，本研究將所設計混合基因演算法 (HGA) 的主要運作架構，說明如下：

HGA

{

以 *Proc Initial* 程序產生起始母體

以搜尋演算法 (Algor Search(**k**)) 評估起始母體中每一條染色體

當 (尚未終止)

{

選擇機制 (Selection)

交配 (Crossover)

突變 (Mutation)

以搜尋演算法 (Algor Search(**k**)) 評估新母體中每一條染色體

}

}

4.3.3 為候選解產生可行的生產與檢驗排程

在 HGA 中，本研究必須為每一個染色體（即一組的基本週期乘數組合 \mathbf{k} ）求得最佳的檢驗次數組合 \mathbf{m} 與基本週期 B ；其所產生的生產與檢驗排程，在評估是否為最佳解之前，必須先確保為合理可行解。本節將介紹可行性測試程序與二元搜尋演算法來協助本研究找尋到可行的解。

4.3.3.1 可行性測試程序

針對任何產生的候選解 $(B, \mathbf{k}, \mathbf{m})$ ，本研究以一演算法名為「『可行性測試程序』 (The Feasibility Testing Procedure, Proc FT)，來協助產生可行的生產與檢驗排程。

對一組候選解 $(B, \mathbf{k}, \mathbf{m})$ ，當給定一乘數 k_i 時，本研究測試程序關心的是：指派產品 i 的生產批量(檢驗批量)至前 k_i 個基本週期中，並安排於適當的基本週期位置。其任一個基本週期內，被指派到產品項之總生產時間(總檢驗時間)皆不可以超過一個基本週期的時間長度(此概念以式(4.2)與(4.3)表示)。本研究以 $w_{ij\varphi(j,t)}$ 二元變數的值來決定生產與檢驗的空間限制。

在此，本章研究所使用的 Proc FT 是根據 Yao *et al.* (2003) 修正後類似的版本。Yao *et al.* 原先的版本主要是針對傳統的 ELSP 模式中當給定一組 (B, \mathbf{k}) 時，可以有效測試其解的可行性。而本研究所設計的 Proc FT 作法類似，不同點是需多考慮到檢驗排程的可行性。作法是首先以一初始排程程序 (Initial Schedule Procedure, Proc IS) 來產生一組初始的生產排程與檢驗排程(見附錄 B.2 說明)，其分別以 W_p 與 W_l 表示從 Proc IS 產生的一種可能的生產與檢驗排程。說明如下：

$$W_p = \sum_{i=1}^n \sum_{j=0}^{O_i} (s_i + k\bar{B}_i d_i / p_i + k_i \bar{B} d_i / y_i) w_{ij\varphi(j,t)} \quad \text{for } t = 1, \dots, 2^{\max\{o_i\}} \quad (4.23)$$

$$W_l = \sum_{i=1}^n \sum_{j=0}^{O_i} (z_i + k_i \bar{B} d_i / y_i + k_i \bar{B} d_i (m_i - 1) / m_i p_i) w_{ij\varphi(j,t)} \quad \text{for } t = 1, \dots, 2^{\max\{o_i\}} \quad (4.24)$$

W_p 與 W_l 的結構(圖 4.4)可分別看作是兩個大小一樣的矩陣，矩陣的大小是由產品數與基本週期乘數 k_i (取 $lcm\{k_i\}$) 來決定。兩排程中週期與產品

對應的數字分別是機台的生產時間 $s_i + k_i \bar{B}d_i / p_i + k_i \bar{B}d_i / y_i$ 與機台的檢驗時間 $z_i + k_i \bar{B}d_i / y_i + k_i \bar{B}d_i (m_i - 1) / m_i p_i$ ，需注意的是：根據產品項的週期乘數 k_i 與其起始生產的週期， W_p 內發生生產時間與 W_l 內發生檢驗時間在矩陣內對應的位置是相同的。

W_l	產品 1	產品 2	產品 3	...	產品 n	W_p	產品 1	產品 2	產品 3	...	產品 n
週期 1	檢驗時間			...		週期 1	生產時間			...	
週期 2				...		週期 2				...	
週期 3				...		週期 3				...	
⋮	⋮	⋮	⋮	...	⋮	⋮	⋮	⋮	⋮	...	⋮
週期 K-1				...		週期 K-1				...	
週期 K				...		週期 K				...	

圖 4.4 W_p 與 W_l 的排程結構圖

令 $L(W) \equiv \max(W_p, W_l)$ ， W 代表取 W_p 與 W_l 兩排程間最大工作負載量的排程， $L(W)$ 則是 W 排程的最大工作負載量(見附錄 B.3)。根據 Yao *et al.* (2003) Proc FT 的運作原理：本研究令 L^* 代表至目前為止，手邊最佳排程的最大工作負載量； W^* 代表手邊最大工作負載量的排程(假若 W 是從 Proc IS 產生起始的生產/檢驗排程，則令 $L^* = L(W)$ 與 $W^* = W$)。當 $L^* \leq B$ ，最小公倍數 $LCM\{k_i\}$ 所求得所有週期中，任何一個週期最大的負載量皆不超過基本週期的時間長度，就可獲得可行的生產與檢驗週期。在此，本研究於 Proc FT 中定義一指標 ϕ ，若是生產與檢驗排程可行，則為 $\phi = 1$ ；否則 $\phi = 0$ 。

假若以 Proc IS 產生起始的生產排程是不可行(換言之， $\phi = 0$)，則 Proc FT 使用排程撫平程序(The Schedule Smoothing procedure, Proc SS) 三種子程序來修正某些產品項 $w_{ij\phi(j,t)}$ 的值，此三個子程序分別為：移出程序(Smooth-Out routine)、成對互換程序(Pair-Exchange routine) 與 2 對 1 的互換程序(Two-to-One exchange routine)，它們可以被用來降低產品在基本週期內的最大負載量，使其產生合理可行的排程。三種子程序的詳細介紹請參閱 Yao *et al.* (2003) 與附錄 B.3 說明。本節僅就其概念簡述說明如下：

(1) 移出程序：

從排程 W 中，將擁有最大負載量之某產品 i (其 k_i 大於 1 之產品)，將其生產/檢驗時間(W_p 與 W_l 中需同時移動生產時間與檢驗時間)從原本的週期移出，並指派此產品 i 的生產/檢驗時間至其他生產/檢驗時間

較小的週期內。移動過後， W_p 與 W_l 中每個週期的總生產時間及總檢驗時間皆必須小於或等於 B ，是則為可行解；否則要進入成對互換程序。

(2) 成對互換程序：

此程序藉由在 W 中的最大負載週期 τ ，將其產品 i (其 k_i 需大於 1) 的生產/檢驗時間與另一項擁有較小負載生產的產品 (其 k_i 需大於 1) 進行交換 (此擁有較小負載的產品，交換前在週期 τ 中並不生產)。若進行此成對互換後， W_p 與 W_l 中每個週期的總生產時間與總檢驗時間需同時皆小於等於 B ，則為可行解；否則要進入下一項 2 對 1 互換的程序。

(3) 2 對 1 的互換程序：

此程序將 W 中，兩項 k_i 大於 1，且基本週期中非最大機器負載 (以 σ_1 與 σ_2 表示其負載) 的產品，與基本週期 τ 中擁有最大機器負載之產品 (以 σ_3 表示負載) 進行週期互換 ($\sigma_1 + \sigma_2 < \sigma_3$ ，同時此兩項非最大機器負載的產品，交換前在週期 τ 中並不生產)，以上程序降低排程 W 中的最大負載。若 2 對 1 互換後，每個基本週期的總生產時間與總檢驗時間皆小於等於 B ，是則為可行解；否則當判斷是否已達到連續無法改善的次數時，停止 Proc FT 的程序。

本研究利用上述三項子程序調整排程的最大工作負載量 L^* ，並計算直到排程是可行的 $\phi = 1$ ；或者說明在三個子程序後， L^* 無法產生可行的結果。

假若上述程序 L^* 無法產生可行解並被改善，接下來可藉由一個重新 (re-start) 修正 B 的程序，來使目前手邊的生產/檢驗排程可以跳脫陷於區域最佳解的困境，試圖尋找到可行的解。此做法稱之為『二元搜尋演算法』 (Binary Search heuristic)，本研究將在下一節進行說明。

4.3.3.2 二元搜尋演算法

當給定任何候選解 $(B, \mathbf{k}, \mathbf{m})$ ，若能以前一節 Proc FT 產生一組可行的生產與檢驗排程，則可以確定其解是使用 EBP 法下 ELISP 模式的可行解。但若是 Proc FT 無法針對 $(B, \mathbf{k}, \mathbf{m})$ 產生可行的生產與檢驗排程，則可藉由修正產生新的基本週期值 \tilde{B} ，以得到新的解 $(\tilde{B}, \mathbf{k}, \mathbf{m})$ ，使其不僅獲取可行的生產與檢驗排程，同時能確保 \mathbf{k} 與 \mathbf{m} 組合下的總成本為最低。

本研究以下說明二元搜尋演算法如何處理 Proc FT 無法產生可行解的問題。首先需設定二元搜尋演算法搜尋的範圍，本研究定義搜尋上下界的符號分別為 B^l 與 B^u 。根據公式(4.3)，本研究可以推導基本週期 t 在其生產產能限制下， B 值的下界 B_t^{PCL} ，說明如下：(詳細說明見附錄 B.4)

$$B_t^{PCL} = \sum_{i=1}^n \sum_{j=0}^{O_i} (s_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\phi(j,t)} \right] \right. \quad (4.25)$$

相同的，本研究可以推導基本週期 t 在其檢驗產能限制下， B 值的下界 B_t^{ICL} ，說明如下：

$$B_t^{ICL} = \sum_{i=1}^n \sum_{j=0}^{O_i} (z_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\phi(j,t)} \right] \right. \quad (4.26)$$

事實上，當 $B > T_{CC}^*$ (T_{CC}^* 是 ELISP 模式在 CC 模式下獲得的最佳週期長度, Yao *et al.*, 2011) 是沒有區域最佳解存在的可能性。因此，可以設定二元搜尋演算法搜尋的範圍如下式：

$$(B^l, B^u) \equiv (\max_t (B_t^{PCL}, B_t^{ICL}), T_{CC}^*) \quad (4.27)$$

當設定好搜尋範圍 (B^l, B^u) ，本研究以二元搜尋程序 (Cormen *et al.*, 1993) 尋找在使用 Proc FT 下產生可行生產與檢驗排程最小的 B 值。當設定二元搜尋的起始點在上界 B^u ，因為若是 $(B^u, \mathbf{k}, \mathbf{m})$ 是不可行解，則認定範圍內所有的 B 值是不會產生可行的生產與檢驗週期；應立即停止二元搜尋並可做出「此 \mathbf{k} 與 \mathbf{m} 組合下無法產生任何可行的排程」之結論。若是發生上述狀況，本研究將試著重新隨機產生一條新的染色體以取代原本不可行的染色體，同時要確保新產生的染色體是可行解。倘若 $(B^u, \mathbf{k}, \mathbf{m})$ 是可行的，則測試帶入下界 B^l 之 $(B^l, \mathbf{k}, \mathbf{m})$ 的可行性；若是 $(B^l, \mathbf{k}, \mathbf{m})$ 也為可行，則停止二元搜尋，因為 $(B^l, \mathbf{k}, \mathbf{m})$ 是在此 \mathbf{k} 組合下能夠獲得最小的目標函數值。若 $(B^l, \mathbf{k}, \mathbf{m})$ 不可行，針對 $\tilde{B} = (B^l + B^u) / 2$ 進行可行性測試，以尋求較好目標函數值且為可行解的機會。當 \tilde{B} 為可行時，則以其更新上界 B^u 。本研究將持續執行二元搜尋直達到特定的容許誤差範圍為內。根據數值實驗，二元搜尋法將 B 值設定為 10^{-2} 的容許誤差是有效的。

4.3.4 混合基因演算法虛擬程式碼

本節以虛擬程式碼來說明混合基因演算法的詳細步驟：

搜尋演算法(\mathbf{k}):

1. 使用式(4.14)計算法下界 B_{LB} 並令上界 $B_{UB} = \infty$ 。
2. 使用式(4.16)計算 $\mathbf{m}=\mathbf{m}(B_{LB})$ 並獲得 $\{w_i(m_i(B_{LB})) | i=1, \dots, n\}$ 且確認 $m_i(w_i)$ 每一項產品的檢驗次數勿超過檢驗批量大小。令 $B_c = B_{LB}$ 且得到 $\delta = \arg \min_i \{w_i(m_i) > B_c\}$ 。
3. 藉由式(4.19)獲得 $\tilde{B}(\mathbf{m}(B_{LB}))$ ，假若 $TC(\tilde{B}(\mathbf{m}(B_{LB})), \mathbf{k}, \mathbf{m}(B_{LB})) < TC(B^*, \mathbf{k}^*, \mathbf{m}^*)$ 以 *Proc FT* 檢查 $(\tilde{B}(\mathbf{m}(B_{LB})), \mathbf{k}, \mathbf{m}(B_{LB}))$ 的可行性：假如 $(\tilde{B}(\mathbf{m}(B_{LB})), \mathbf{k}, \mathbf{m}(B_{LB}))$ 是可行的，更新 B^* ， \mathbf{k}^* 與 \mathbf{m}^* ，儲存 $(\tilde{B}(\mathbf{m}(B_{LB})), \mathbf{k}, \mathbf{m}(B_{LB}))$ 與 $TC(\tilde{B}, \mathbf{k}, \mathbf{m})$ ，跳至步驟4；否則，使用二元搜尋法(見 4.3.2 節)搜尋 \tilde{B} 以試圖尋找有可行的生產與檢驗排程的最小值的目標函數，至步驟4。
4. 以式(4.20)計算手邊的最佳上界 $B_{UB} = \beta$ ，使用式(4.15)獲得在 $[B_{LB}, B_{UB})$ 間所有的接合點，並以從小到大的順序排成 $\{w_i\}$ 。令 $j=1$ 與 $B_c = w_j$ 。
5. 藉由 $\mathbf{m}(w_j) \equiv (\mathbf{m}(B_c) \setminus \{m_\delta\}) \cup \{m_\delta + 1\}$ ，我們獲得檢驗次數組合 $\mathbf{m}(w_j)$ 與 $\delta = \arg_i \{w_i(\hat{m}_i + 1) = w_{j+1}\}$ 。以式(4.19)計算區域最小解 $\tilde{B}(\mathbf{m}(w_j))$ ，假若 $TC(\tilde{B}(\mathbf{m}(w_j)), \mathbf{k}, \mathbf{m}(w_j)) < TC(B^*, \mathbf{k}^*, \mathbf{m}^*)$ ，以 *Proc FT* 檢查 $(\tilde{B}(\mathbf{m}(B_{LB})), \mathbf{k}, \mathbf{m}(B_{LB}))$ 的可行性：假如 $(\tilde{B}(\mathbf{m}(w_j)), \mathbf{k}, \mathbf{m}(w_j))$ 為可行，更新 $B^* = \tilde{B}(\mathbf{m}(w_j))$ 、 $\mathbf{k}^* = \mathbf{k}$ 與 $\mathbf{m}^* = \mathbf{m}(w_j)$ ，儲存 $(\tilde{B}(\mathbf{m}(w_j)), \mathbf{k}, \mathbf{m}(w_j))$ 與 $TC((\tilde{B}(\mathbf{m}(w_j)), \mathbf{k}, \mathbf{m}(w_j)))$ ，跳至步驟6；否則使用二元搜尋法(見 4.3.2 節)搜尋 \tilde{B} 以試圖尋找有可行的生產與檢驗排程的最小值的目標函數。假若以二元搜尋法可以找到一可行解，跳至步驟6，否則以隨機方式產生一組新的染色體後回到步驟1。
6. 若需要，藉由式(4.20)更新 β 與上界 $B_{UB} = \min\{B_{UB}, \beta\}$ 。令 $j = j+1$ 與

$B_c = w_j$ ，假若 $w_j > B_{UB}$ ，輸出 \mathbf{m}^* 、 B^* 與 $TC(B^*, \mathbf{k}^*, \mathbf{m}^*)$ ，終止演算法計算；否則跳至步驟 5。

4.4 數據實驗

本節實驗分成兩個部分：4.4.1 節首先以一數值範例 (numerical example) 介紹在 EBP 下混合基因演算法的實際操作情況；4.4.2 節以隨機實驗 (random experiment) 說明所設計之 HGA 的有效情形；最後 4.4.3 節將做模式的效能分析。

4.4.1 數值範例

本節主要說明本章研究所設計之 HGA 的實際操作情形，其所使用的 10 個產品數值範例與前章 3.5.1 節中表 3.2 的數值範例參數設定資料皆為相同。而表 4.1 說明以表 3.2 的數值範例參數設定，藉由 HGA 找到最佳可行的解。

表 4.1 使用 GA 與 Proc FT 建立的可行排程

項目	1	2	3	4	5	6	7	8	9	10	說明
k_i	16	4	4	4	1	1	4	1	2	1	週期乘數
m_i	13	6	15	5	7	4	10	5	5	2	檢驗次數
B	29.0945										單位：天
TC	177.418										單位：仟元

從表 4.1 得知，10 項產品的最佳基本週期乘數組合為 $\mathbf{k}^* = \{16, 4, 4, 4, 1, 1, 4, 1, 2, 1\}$ ；最佳檢驗次數組合為 $\mathbf{m}^* = \{13, 6, 15, 5, 7, 4, 10, 5, 5, 2\}$ ；最佳的基本週期為 29.045 天；而最佳平均總成本為 \$177.418 仟元。此結果是在考量生產產能與檢驗產能限制下所獲得的可行最佳解。相較於前章 CC 法求解 ELISP 模式所得最佳平均總成本 \$181.929 仟元，本章可獲得更低的平均總成本。

4.4.2 隨機實驗

本節以一隨機實驗來說明本混合基因演算法的有效性 (effectiveness)。表 4.2 為產品的參數組合，本隨機實驗以一致分配 (uniform distributions) 產生實驗例子。本章的實驗設備是採用 i7-860 2.8GHz CPU 與 2G 記憶體的个人電腦。

表 4.2 隨機實驗的參數組合

參數	分配	參數	分配
A_i	UNIF(5,300)	d_i	UNIF(500,2000)
v_i	UNIF(5,400)	p_i	UNIF(15000,35000)
h_i	UNIF(0.0005,0.0025)	s_i	UNIF(0.002,0.005)
π_i	UNIF(0.0001,0.003)	z_i	UNIF(0.00015, 0.0003)
α_i	UNIF(0.01,0.05)	y_i	UNIF(3500,14500)
r_i	UNIF(0.00002,0.00025)	c_i	UNIF(20,250)

根據 Elmaghraby (1978) 指出生產設備較大的產能利用率將會較難找到可行的排程解。本章研究觀察所設計的 HGA 在不同水準的產能率 (utilization rate，在此定義的產能率 $\sum \rho_i$ 並非是 $\sum (d_i / p_i)$ ，而是指所有產品項的總生產時間加上檢驗批量的總消耗時間後除上循環週期 $k_i B$) 與不同水準的產品項目數下的有效性。本研究隨機產生八種不同水準的產能率(0.5 至 0.9)以及八種不同水準的產品項目數(5 至 40)，每一種組合情況將產生 50 個樣本。

在本章隨機實驗以第三章 CC 法所得到的最佳平均總成本當作是標竿 (benchmark)，計算其與本章 EBP 法所得結果比較後兩者間的差距。本研究以 TC_{CC} 當作是在 CC 法下得到最佳可行的平均總成本； TC^* 表示是由 HGA 某一種組合中的樣本所得到最佳的平均總成本。本研究以式(4.28)的 Δ 去計算目標函數的變異情況。

$$\Delta = (TC_{CC} - TC^*) / TC_{CC} \cdot 100\% \quad (4.28)$$

因為本研究所設計的 HGA 通常可以得到比 CC 法更好的解，因此可以視 Δ 的值為求解 ELISP 品質改善的幅度。值得注意的是，產能率的高低會影響到 HGA 的求解品質，原因是本研究的決策假設加入了生產與檢驗產能的限制條件。表 4.3 顯示將隨機實驗各種組合的結果。Avg. (%) Δ 則表示 50 個隨機案例之平均總成本的平均減少百分比；Avg. Run Time 則為 50 個隨機案例的平均執行時間。

表 4.3 不同水準 $\sum \rho_i$ 的實驗結果

產品數	$\sum \rho_i$	[0.50, 0.55)	[0.55, 0.60)	[0.60, 0.65)	[0.65, 0.70)
5	Avg. (%) Δ	1.91 %	1.81 %	1.66 %	1.57 %
	Avg. Run Time [†]	69.26	78.10	80.43	81.19
10	Avg. (%) Δ	2.46 %	2.58 %	2.40 %	2.33 %
	Avg. Run Time	83.59	93.85	98.34	102.22
15	Avg. (%) Δ	2.85 %	2.72 %	2.42 %	2.40 %
	Avg. Run Time	108.51	122.25	144.08	171.58
20	Avg. (%) Δ	3.52 %	3.45 %	2.97 %	2.93 %
	Avg. Run Time	162.69	181.33	213.98	228.62
25	Avg. (%) Δ	3.88 %	3.83 %	3.32 %	3.22 %
	Avg. Run Time	217.30	256.82	274.22	296.19
30	Avg. (%) Δ	4.23 %	4.11 %	3.86 %	3.49 %
	Avg. Run Time	285.96	313.78	321.16	330.03
35	Avg. (%) Δ	5.76 %	4.69 %	4.55 %	4.28 %
	Avg. Run Time	302.50	331.81	349.67	377.69
40	Avg. (%) Δ	6.07 %	5.88 %	5.33 %	5.02 %
	Avg. Run Time [†]	337.26	380.04	3405.42	417.27

[†] 單位：秒

表 4.3 不同水準 $\sum \rho_i$ 的實驗結果 (續)

產品數	$\sum \rho_i$	[0.70, 0.75)	[0.75, 0.80)	[0.80, 0.85)	[0.85, 0.90)
5	Avg. (%) Δ	1.43 %	1.21 %	0.95 %	0.63 %
	Avg. Run Time [†]	97.60	103.02	107.98	125.35
10	Avg. (%) Δ	2.17 %	1.80 %	1.72 %	1.52 %
	Avg. Run Time	114.78	127.78	147.79	179.20
15	Avg. (%) Δ	2.35 %	2.06 %	1.92 %	1.68 %
	Avg. Run Time	188.89	231.84	238.88	262.80
20	Avg. (%) Δ	2.77 %	2.47 %	2.37 %	2.29 %
	Avg. Run Time	227.23	251.80	254.42	282.01
25	Avg. (%) Δ	3.16 %	2.93 %	2.78 %	2.55 %
	Avg. Run Time	302.53	326.50	349.63	357.97
30	Avg. (%) Δ	3.36 %	3.18 %	3.07 %	2.87 %
	Avg. Run Time	320.18	361.86	390.46	392.83
35	Avg. (%) Δ	4.19 %	3.67 %	3.46 %	3.40 %
	Avg. Run Time	394.46	432.59	448.76	473.16
40	Avg. (%) Δ	4.59 %	4.66 %	4.38 %	4.08 %
	Avg. Run Time [†]	403.27	437.35	459.98	490.86

[†] 單位：秒

從表 4.3 實驗結果顯示：在特定水準的產品數下，較高的產能率會產生較小值的 Avg. (%) Δ 。其與本研究的直覺相符合：當產能率變大時，生產設備呈現滿載 (full occupied) 情況，其很難滿足生產批量大之產品項的需求，而將其安排於一基本週期內。此種狀況易導致每一個基本週期內皆必須生產一個批量—恰如 CC 法的生產方式。再者，當產品數增加，對任何水準的產能率而言，Avg. (%) Δ 總成本改善幅度呈現上升的情況。可解釋此現象是因為當產品數增加時，其生產系統有較多的機會調整批量大小以及生產與檢驗排程。

4.4.3 效能分析

從表 4.3 顯示所設計的演算法其平均執行時間隨著問題（產品的項目數）的大小而呈現溫和（mild）增加。在 95%信心水準下，本研究設計的 HGA 數據執行時間 T 檢定中，假設檢定不拒絕（fails to reject the null hypothesis）二次係數等於零的情形。同前章根據 Neter, Wasserman 與 Kutnur 書中第九章 polynomial regression 之二項式測驗程序計算，本研究的所設計的 HGA 是隨問題內容變大，執行時間呈現線性（linear order）成長，以 EBP 法求解 ELISP 有效的演算法。

4.5 結論與討論

本章研究主要以延伸基本週期法週期類型處理有檢驗狀況的經濟批量排程問題。除了與前章相同，採取了產品全面檢驗策略及考量生產與檢驗產能限制外，本章發展出混合演算法結合前章的搜尋演算法試圖尋找產品項不同的週期乘數組合與檢驗次數組合以改善求解的品質。同時，也試著藉由染色體的演化程序以產生候選解。本章研究貢獻如下：

第一，經由 4.4 節的數值範例與隨機實驗，證明本章研究以延伸基本週期法求解 ELISP 模式可以獲得比前章以 CC 法求解更低的平均總成本。第二，根據隨機實驗，當生產系統非滿載或當產品項目數較多時，本研究所設計的 HGA 所呈現出的改善效果會比較明顯。第三，與 CC 法下的 ELISP 模式相比，本章所設計之 HGA 在低產能利用率與多產品項生產的情形，對於生產決策者而言是一項好的決策工具。

第五章 結論與建議

5.1 研究結論

本論文主要是針對會產出瑕疵品之生產系統下，經濟批量排程問題的延伸主題進行探討。本研究將多產品的檢驗排程問題納入 ELSP 模式中，並考慮到生產與檢驗的產能限制條件，以求得合理可行的解。在第三章與第四章的內容中，本研究分別以不同的生產週期類型與設計不同的演算法來求取最低平均總成本下合理可行的最佳生產週期與檢驗排程，期望其實務上能為生產管理者提供相關決策的參考。

在第三章中，本主題研究先採取共同週期法的週期類型，因為無論是學術研究或實務上，共同週期法的週期型態可做為創新問題的初探，其概念最為簡單並可確保最佳解可行，同時結果可被用於測試其他演算法解答的品質。在此章中，本研究根據成本曲線之接合點與凹性曲線理論的探討，設計了搜尋演算法來尋求最佳解，並以數值範例與隨機實驗說明了演算法的有效性。在效能分析結果，搜尋演算法可以短時間內有效率找到 ELISP 模式的最佳可行解，所以建議業界之管理決策者，可以此演算法做為相關生產排程問題的工具。

在第四章中，本研究將模式的週期類型改為延伸基本週期法，因為其不需要每項產品在每個週期都需要生產，所以降低了設置的次數(設置成本)。而過去的相關文獻也指出使用延伸基本週期法可獲得比使用共同週期法更低的平均總成本，因此本章設計了基因演算法 (GA) 結合搜尋演算法，來尋求最佳基本週期、各產品週期乘數與其檢驗次數；至於獲得之候選解可行性的驗證與修正基本週期程序，則交由 Proc FT 與二元搜尋法處理。數值範例與隨機實驗除驗證演算法的有效性外，並比較使用共同週期法的結果說明改善幅度。雖然，使用延伸基本週期法求算可行之最佳基本週期、週期乘數與檢驗次數組合，可以獲得比 CC 法更低的平均總成本，但由於採用了三個決策變數與演算法複雜程序帶來的影響，執行時間耗費較長，時效上較不理想。

在學術上，本論文是經濟批量排程問題考慮檢驗策略的初始研究，也是近期主要討論多產品生產與檢驗排程的報告。因為本研究結合實務上製造過程中產品檢驗的問題，所以隨機實驗中以不同產品數與產能利用率產生多種組合的結果，可以提供管理決策者後續生產排程安排的參考。

5.2 未來研究方向

本論文在研究與撰寫過程中，發現有許多的項目與延伸主題可做為未來研究討論的方向，其說明如下：

1. 動態需求率(stochastic demand rate)設定：本論文研究假設產品需求率是固定，其與實際市場需求狀況未盡符合，所以未來可以採用動態的需求率來進行模式探討。
2. 不良率 (defect rate) 設定：本論文以均勻分配作為在可能製造瑕疵品之系統下產品不良率的假設，未來研究可以考慮採用其他的分配，例如：指數分配等。
3. 檢驗策略設定：本論文所使用的全面檢驗策略，僅針對部分相關產業適用。未來可以針對產品與生產線的特性設定不同的檢驗方式進行探討，例如：固定比率抽樣檢驗、線上立即檢驗 (in-line inspection)。
4. ELISP 延伸主題研究：未來可以針對所發展的 ELISP 模式，加入其他相關的考慮因素進行研究。例如：不良品的重製 (reworking)、倉庫空間限制 (warehouse limitation) 與定價策略 (pricing) 等。
5. 基因演算法設定：第四章關於基因演算法內運算子的設定，未來可針對染色體複製、交配與突變，進行實驗並採用不同的方式搭配，以了解在求解上是否能獲得更好的結果。
6. 演算法設計：本論文研究主要設計以搜尋演算法及基因演算法的求解技術，來計算目標函數的最佳解。未來可以設計其他的啟發式演算法，例如：模擬退火法 (Simulation Annealing, SA)、塔布搜尋法 (Tabu Search, TS) 等來求解模式。

本論文將傳統的經濟批量問題與實務的檢驗策略議題結合發展模式，是在學術研究上一個新的研究方向，其未來可以結合相當多的生產實務議題進行討論。由於本研究為求得合理可行的結果，同時考慮到生產產能與檢驗產能的限制問題，以至於增加求解複雜度與造成執行時間的增長。關於此部分，在未來的研究裡可以思考與設計不同演算法，來尋求更好的改善。

參考文獻

- Ben-Daya, M. & Hariga, M., “Economic lot scheduling problem with imperfect production processes,” *Journal of Operational Research Society*, 851–881 (2000).
- Bomberger, E., “A dynamic programming approach to a lot size scheduling problem,” *Management Science*, **12**, 778–784 (1966).
- Boesel, J., Nelson, B. L., & Ishii, N., “A framework for simulation-optimization software,” *IIE Transactions*, **35**, 221–229 (2003).
- Braglia, M., “Influence of the limited service capacity of the inspection station on the control policy of an automatized production system,” *International Transactions in Operational Research*, **2(3)**, 273–285 (1995).
- Carreno, J., “Economic lot scheduling for multiple products on parallel identical processors,” *Management Science*, **36 (3)**, 348–358 (1990).
- Chang, P. T., Yao, M. J., Haung, S. F. & Chen, C. T., “A genetic algorithm for solving a fuzzy economic lot-size scheduling problem,” *International Journal of Production Economics*, **102**, 265–288 (2006).
- Chang, Y. J., & Yao, M. J. “A genetic algorithm for solving the economic lot scheduling problem with reworks,” *Journal of the Chinese Institute of Industrial Engineers*, **26 (5)**, 411–425 (2009).
- Cookie, D. L., “Finding effective schedules for the economic lot scheduling problem, a simple mixed integer programming approach,” *International Journal of Production Research*, **42(1)**, 21–36 (2004).
- Cormmen, T. H., Leiserson, C. S., & Rivest, R. L., *Introduction to algorithms*, New York City: McGraw-Hill (1993).
- Davis, S. G., “Scheduling economic lot size production runs,” *Management Science*, **36**, 985–998 (1990).
- Dobson, G., “The economic lot-scheduling problem: achieving feasibility using time-varying lot size,” *Management Science*, **35 (5)**, 764–771 (1992).
- Elhafsi, M. & Bai, S. X., “The common cycle economic lot scheduling problem with backorders: benefits of controllable production rates,” *Journal of Global Optimization*, **10(3)**, 283–303 (1997).

- Elmaghraby, S., “The economic lot scheduling problem (ELSP): Review and extension,” *Management Science*, **24**, 587–598 (1978).
- Faaland, B. H., Schmitt, T. G. & Arreola-Risa, A. “Economic lot scheduling with lost sales and setup times,” *IIE Transactions*, **36(7)**, 629–640 (2004).
- Federgruen, A. & Zheng, Y. S., “Optimal power-of-two replenishment strategies in capacitated general production/distribution networks,” *Management Sciences*, **39**, 710–727 (1993).
- Fujita, S., “The application of marginal analysis to the economic lot size scheduling problem,” *AIIE Transactions*, **10**, 354–361 (1978).
- Granar, J. & Riggle, C., “An optimal algorithm for the basic period approach to the economic lot scheduling problem,” *Omega –International Journal of Management Science*, **25(3)**, 355–364 (1997).
- Haessler, R. & Hogue, S., “A note on the single machine multi-product lot scheduling problem,” *Management Science*, **22**, 909–912 (1976).
- Haessler, R. W., “An improved extended basic period procedure for solving the economic lot scheduling problem,” *AIIE Transactions*, **11(4)**, 336–340 (1979).
- Ham, I., Hitomy, K. & Yoshida, T., *Group Technology: Applications to Production Management*, Kluwer–Nijhoff: Boston, MA (1985).
- Hanssmann, F., *Operations Research in Production Planning and Control*. New York: John Wiley (1962).
- Harris, F., “How many parts to make at once,” *The Magazine of Management*, **10**, 135–152 (1913).
- Holland, J. H., *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press (1975).
- Hsu, W. L., “On the general feasibility of scheduling lot sizes of several products on one machine,” *Management Science*, **29**, 93–105 (1983).
- Hunter, A., “Crossing over genetic algorithms: the Sugal generalized GA,” *Journal of Heuristics*, **4**, 179–192 (1998).
- Jackson, P. L., Maxwell, W. L., & Muckstadt, J. A., “The joint replenishment problem with a power-of-two restriction,” *IIE Transactions*, **17**, 25–32 (1985).

- Khouja, M. & Mehrez, A., "Economic production lot size model with variable production rate and imperfect quality," *Journal of the Operational Research Society*, **45**, 1405–1417 (1994).
- Khouja, M., Mehrez, A. & Wilmot, M., "The use of genetic algorithm to solve the economic lot size scheduling problem," *European Journal of Operational Research*, **110**, 509–524 (1998).
- Kim, C. H. & Hong, Y., "An optimal production run length in deteriorating production processes," *International Journal of Production Economics*, **58**, 183–189 (1999).
- Kim C. H., Hong, Y. & Chang, S. Y., "Optimal production run length and inspection schedules in a deteriorating production process," *IIE Transactions*, **33**, 421–426 (2001).
- Kuo, H. & Inman, R., "A practical heuristic for the group technology economic lot scheduling problem," *International Journal of Production Research*, **28**, 709–722 (1990).
- Lee, J.S. & Park, K.S. "Joint determination of production cycle and inspection intervals in a deteriorating production system," *Journal of the Operational Research Society*, **42**, 775–783 (1991).
- Lee, H. & Rosenblatt, M. "Simultaneous determination of production cycle and inspection schedules in a production system," *Management Science*, **33**, 1125–1136 (1987).
- Lee, H. L. & Rosenblatt M. "A production and maintenance planning model with restoration cost dependent on detection delay," *IIE Transactions*, **21**, 368–375 (1989).
- Lin, C. S., "Integrated production-inventory models with imperfect production processes and a limited capacity for raw materials," *Mathematical and Computer Modeling*, **29**, 81–89 (1999).
- Lin, C. S., Chen, C. H. & Kroll, D. E., "Integrated production-inventory models for imperfect production processes under inspection schedules," *Computers and Industrial Engineering*, **44**, 633–650 (2003).
- Lin, T. M., Tseng, S. T. & Liou, M. J., "Optimal inspection schedule in the imperfect production system under general shift distribution," *Journal of the Chinese Institute of Industrial Engineers*, **8(2)**, 73–81 (1991).

- Liou, M.J., Tseng, S.T. & Lin, T.M. "The effects of inspection errors to the imperfect EMQ model," *IIE Transactions*, **26**, 42–51 (1994).
- Makis, V. "Optimal lot sizing and inspection policy for an EMQ model with imperfect inspections," *Naval Research Logistics*, **45**, 165–186 (1998).
- Makis, V. & Fung, J. "An EMQ model with inspection and random machine failures," *Journal of the Operational Research Society*, **49**, 66–76 (1998).
- Marzouk, M. & Moselhi, O. "Constraint-based genetic algorithm for earthmoving fleet selection," *Canadian Journal of Civil Engineering*, **30(4)**, 673–683 (2003).
- Moon, I., Giri, B. C. & Choi, K. "Economic lot scheduling problem with imperfect production processes and setup time," *Journal of the Operational Research Society*, **53**, 620–629 (2002).
- Moon, I. K., Cha, B. C. & Bae, H. C. "Hybrid genetic algorithm for group technology economic lot scheduling problem," *International Journal of Production Research*, **44(21)**, 4551–4568 (2006).
- Nahmias, S. *Production and Operations Analysis*, McGraw Hill, Singapore (1997).
- Neter, J. W. Wasserman and M. Kutner, *Applied Linear Statistical Models*, Irvine, Chicago (1986).
- Osteryoung, J. S., McCarty, D. E. & Reinhart, W. J. "Use of EOQ model for inventory analysis," *Production and Inventory Management Journal*, 3rd Q, 39–45 (1988).
- Ouyang, H. & Zhu, X. "A economic lot scheduling problem for manufacturing and remanufacturing," *2008 IEEE International Conference on Cybernetics and Intelligent Systems*.
- Park, K.S. & Yun, D.K., "Optimal Scheduling of Periodic Activities," *Operations Research*, **33**, 690–695, (1985).
- Porteus, E. L., "Optimal lot size, process quality improvement and setup cost reduction," *Operations Research*, **34(1)**, 137–144 (1986).
- Porteus, E. L., "The impact of inspection delay on process and inspection lot sizing," *Management Science*, **36**, 999–1007 (1990).
- Rogers, J., "A computational approach to the economic lot scheduling problem," *Management Science*, **4**, 264–291 (1958).

- Rosenblatt, M. J. & Lee, H. L., "Economic production cycles with imperfect processes," *IIE Transactions*, **21**, 48–55 (1986).
- Roundy, R., "Rounding off to powers of two in continuous relaxations of capacitated lot sizing problem," *Management Science*, **35**, 1433–1442 (1989).
- Salvietti, L. & Smith, N. R., "A profit-maximizing economic lot scheduling problem with price optimization," *European Journal of Operational Research*, **184**, 900–914, (2008).
- Sarker, R., & Newton, C., "A genetic algorithm for solving economic lot size scheduling problem," *Computers & Industrial Engineering*, **42**, 189–198 (2002).
- Shih, W., "Optimal inventory policies when stockouts result from defective products," *International Journal of Production Research*, **18(6)**, 677–686 (1980).
- Silver, E. A., Pyke, F. D. & Peterson, R. *Inventory Management and Production Planning and Scheduling*. Wiley, New York (1998).
- Soman C. A., "A basic period approach to the economic lot scheduling problem with shelf life considerations," *International Journal of Production Research*, **42(8)**, 1379–1393 (2004).
- Sung, C. S & Ock, Y. S. "Optimal production policy for a single-product single-machine problem with intermediate machine inspection allowed," *International Journal of Production Economics*, **28(1)**, 85–94 (1992).
- Tang, O. & Teunter, R. H. "Economic lot scheduling problem with returns", *Production and Operations Management*, **15 (4)**, 488–497 (2006).
- Teunter, R., Kaparis, K. & Tang, O., "Multi-product economic lot scheduling problem with separate production lines for manufacturing and remanufacturing," *European Journal of Operational Research*, **191**, 1241–1253 (2008).
- Tseng, S. H., Yeh, R. H. & Ho, W. T., "Imperfect maintenance policies for deteriorating production systems," *International Journal of Production Economics*, **67**, 253–267 (1998).
- Tseng, S.T., "Optimal preventive maintenance policy for deteriorating production systems," *IIE Transactions*, **28**, 687–694 (1996).

- Wagner, B. J. & Davis, D. J., "A search heuristic for the sequence-dependent economic lot scheduling problem," *European Journal of Operational Research*, **141**, 133–146 (2002).
- Wagner, H. M. & Whitin, T. M., "Dynamic version of the economic lot-size model," *Management Science*, **5**, 89–96 (1958).
- Wang C. H., Sheu S. H., "Simultaneous determination of the optimal production–inventory and product inspection policies for a deteriorating production system," *Computers and Operations Research*, **28**, 1093–110 (2001).
- Wang, C. H., "Integrated production and product inspection policy for a deteriorating production system," *International Journal of Production Economics*, **95**, 123–134 (2005).
- Wang, C. H. & Sheu, S. H. "Optimal lot sizing for products sold under free-repair warranty," *European Journal of Operational Research*, **149**, 131–141 (2003).
- Yao, M. J. "On the feasibility testing problem for the economic lot scheduling problem," *Proceeding of the 8th Bellman Continuum Conference*, Hsinchu, Taiwan, 307–315 (2000).
- Yao, M. J. & Elmaghraby, S. E. "The economic lot scheduling problem under power-of-two policy," *Computers and Mathematics with applications*, **41(10-11)**, 1379–1393 (2001).
- Yao, M. J., Elmaghraby, S. E. & Chen, I. C., "On the feasibility testing of the economic lot scheduling problem using the extended basic period approach," *Journal of the Chinese Institute of Industrial Engineering*, **20**, 435–448 (2003).
- Yao, M. J., Chen, S. C. & Chang, Y. J. "A common cycle approach for solving the economic lot and inspection scheduling problem," to appear in *Journal of Industrial and Management Optimization* (2011).
- Yeh, R. H., Ho, W. T. & Tseng, S. T. "Optimal production run length for products sold with warranty," *European Journal of Operational Research*, **120**, 575–582 (2000).

附錄 A

A1. 產品*i*之一個生產批量的不良品的期望數推導

Porteus(1986) 提出，生產批量為 n 與生產批量為 $n-1$ 時，兩個批量之間期望不良品數目差為 $e_n - e_{n-1}$ ，關係式如下：

$$e_n - e_{n-1} = q(n - e_{n-1}) \quad (\text{A.1})$$

e_n ：產品之生產批量為 n ，且生產週期為 T 時，產品的期望不良品數。

e_{n-1} ：產品之生產批量為 $n-1$ ，且生產週期為 T 時，產品的期望不良品數。

q ：製程由正常狀態轉變為不正常狀態的機率。

$n - e_{n-1}$ ：生產批量為 n 時，當生產到第 $n-1$ 個產品時，製程中依然是在正常狀態加工的產品數目。

$q(n - e_{n-1})$ ：當繼續生產第 n 個產品時，製程會將 $n - e_{n-1}$ 個產品，由正常狀態轉變為非正常狀態的不良品期望數目。

另外 Porteus(1986) 表示，當生產批量為 n 時，在此一生產批量中，所生產出的期望不良品數 e_n 為：(當中 $\bar{q} = 1 - q$)

$$\begin{aligned} e_n &= qn + \bar{q}e_{n-1} \quad (\text{此項透過式(A.1)移項求得}) \\ &= qn \sum_{i=0}^{n-1} \bar{q}^i - q \sum_{i=0}^{n-1} i \bar{q}^i \quad (\text{此項透過遞迴方式求得}) \\ &= qn \frac{(1 - \bar{q}^n)}{q} - q \frac{\bar{q}(1 + (n-1)\bar{q}^n - n\bar{q}^{n-1})}{q^2} \quad (\text{此項透過幾何級數求得}) \\ &= n - \frac{\bar{q}(1 - \bar{q}^n)}{q} \end{aligned} \quad (\text{A.2})$$

所以，若生產批量為 Q_i (同式(A.2)中的 n)，製程由正常狀態轉變為非正常狀態的機率為 α_i (同式(A.2)中的 q)，且 $\beta_i = 1 - \alpha_i$ ，將其代入式(A.2)時，得到不良品的期望個數 $E[N_i | Q_i]$ 為

$$E[N_i | Q_i] = Q_i - \frac{\beta_i(1 - \beta_i^{Q_i})}{\alpha_i} = Q_i - \frac{\beta_i(1 - e^{Q_i \ln(\beta_i)})}{\alpha_i} \quad (\text{A.3})$$

Porteus 指出，當 $\alpha_i \rightarrow 0$ 時，

$$E[N_i | Q_i] \approx \frac{\alpha_i Q_i^2}{2}, \beta_i = 1 - \alpha_i \quad (\text{A.4})$$

(將式(A.3)中的 $e^{Q_i \ln(\beta_i)}$ 透過泰勒展開式以及利用 $\ln q \cong -q/\bar{q}$)

所以，若生產一檢驗批量 $Q_i = Td_i / m_i$ ，其不良品的期望數為

$$E[N_i | Td_i / m_i] = \frac{\alpha_i (Td_i / m_i)^2}{2} = \frac{\alpha_i (Td_i)^2}{2m_i^2} = \frac{\alpha_i T^2 d_i^2}{2m_i^2} \quad (\text{A.5})$$

而產品 i 一個生產批量的不良品的期望數為

$$\frac{\alpha_i T^2 d_i^2}{2m_i^2} \times m_i = \frac{\alpha_i T^2 d_i^2}{2m_i} \quad (\text{A.6})$$

A2. 存貨三角形面積的高之推導說明

見圖 3.1 使用 CC 法之生產與檢驗存貨圖的下半部的鋸齒三角形與圖 A.1。產品 i 在一次生產週期的批量大小為 Td_i ，進行 m_i 次檢驗。鋸齒三角形左邊面積：

$$m_i \text{ 個檢驗批量的檢驗時間總長度為 } \frac{Td_i}{y_i m_i} \times m_i,$$

$$\text{不執行檢驗，等待生產 } m_i \text{ 個檢驗批量耗費時間總長度為 } \frac{Td_i}{p_i m_i} \times (m_i - 1)$$

(首批檢驗批量生產不列入計算，所以 $m_i - 1$)。

當產品 i 生產週期內批量大小 Td_i 結束時，檢驗機台產品批量的變化應為 $d_i \left(\frac{d_i T (m_i p_i + m_i y_i - y_i)}{m_i p_i y_i} \right)$ ，而此時檢驗存貨的數量應為

$$Td_i - d_i \left(\frac{d_i T m_i}{m_i y_i} + \frac{d_i T (m_i - 1)}{m_i p_i} \right)。$$

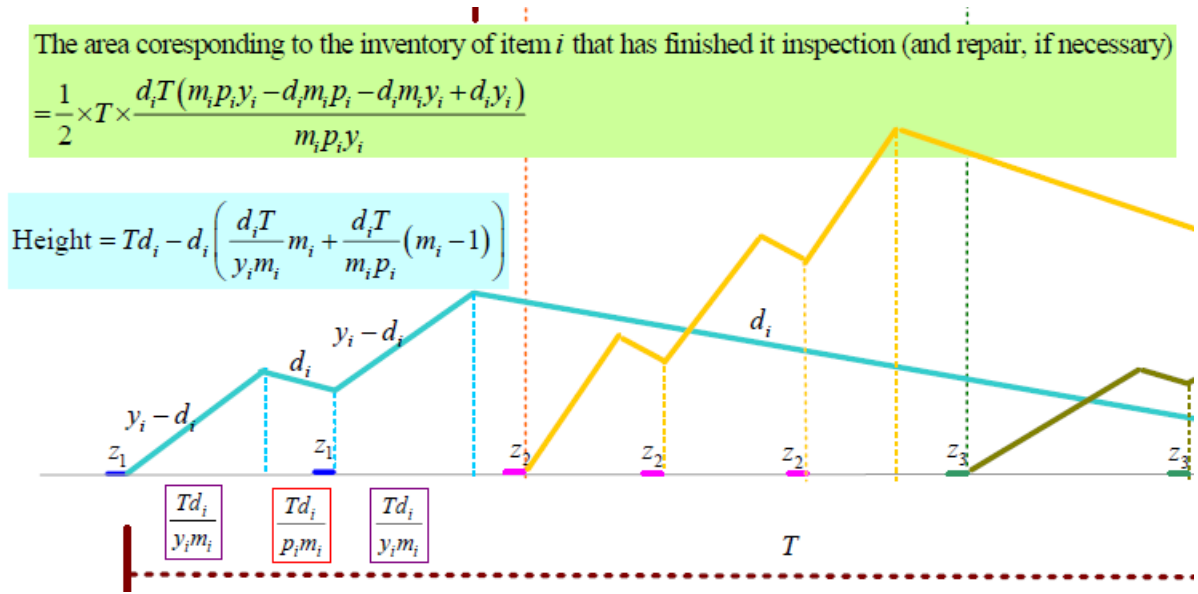


圖 A.1 圖 3.1 之下半部鋸齒三角形部分圖

A3. 式(3.10) 的推導過程

令當 $TC(T, m_i) = TC(T, m_i + 1)$ ，所以

$$\frac{A_i + c_i m_i}{T} + \frac{T d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \frac{T [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)]}{2 m_i p_i y_i} = \frac{A_i + c_i (m_i + 1)}{T} + \frac{T d_i h_i (p_i y_i - d_i p_i - d_i y_i)}{2 p_i y_i} + \frac{T [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)]}{2 (m_i + 1) p_i y_i} \quad (\text{A.7})$$

式(A.7)等式兩邊的第二項與第三項合併後

$$\frac{A_i + c_i m_i}{T} + \frac{T [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)]}{2 m_i p_i y_i} = \frac{A_i + c_i (m_i + 1)}{T} + \frac{T [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)]}{2 (m_i + 1) p_i y_i} \quad (\text{A.8})$$

式(A.8)移項合併後

$$\frac{T [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)]}{2 m_i (m_i + 1) p_i y_i} = \frac{c_i}{T}$$

$$T^2 [d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)] = 2 m_i (m_i + 1) c_i p_i y_i$$

所以

$$T = \sqrt{\frac{2 m_i (m_i + 1) c_i p_i y_i}{d_i h_i (d_i p_i + 2 d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)}} \quad (\text{A.9})$$

式(A.9)根號內分子、分母項同除以 $p_i y_i$ ，得下式

$$T = \sqrt{\frac{\frac{2m_i(m_i+1)c_i p_i y_i}{p_i y_i}}{\frac{d_i^2 h_i (p_i + 2y_i)}{p_i y_i} + \frac{d_i^2 \alpha_i p_i y_i (\pi_i + r_i)}{p_i y_i}}}$$

化簡後

$$T = \frac{1}{d_i} \sqrt{\frac{2c_i m_i (m_i + 1)}{\frac{h_i}{y_i} + \frac{2h_i}{p_i} + \alpha_i (\pi_i + r_i)}}$$

附錄 B

B.1 x_i 的上界 $x_i^{UB}(B)$ 之推導

我們令 TC_i^{CC} 表示產品 i 使用共同週期法所得的平均成本，而 B_L 則表示基本週期的下界。由於每個產品 i 的共同週期必定要大於或等於其生產整備時間，同時共同週期也要大於或等於其檢驗整備時間。再者，基本週期必須同時滿足所有 κ 個排程週期中生產機台與檢驗機台的負載情形。所以，基本週期的下界 B_L 可藉由個別產品當共同週期等於生產整備時間及共同週期等於檢驗整備時間兩種情況取其最大值後，再取所有 κ 個排程週期中的最大機台負載值（換言之，取生產機台與檢驗機台負載之最大值）來表示，即下式(B.1)所示。

$$B_L = \max_i \left(\max_i \left(s_i + \frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i}, z_i + \frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) \right) \quad \text{for } t = 1 \dots \kappa = \text{lcm}\{k_i\} \quad (\text{B.1})$$

k_i 的乘數上界 x_i^{UB} 可以藉由單一產品代入 B_L 至目標函數式後，所得的平均成本必定小於或等於使用共同週期法所得的平均成本來決定，見不等式(B.2)所示

$$\frac{A_i + C_i}{k_i B_L} + \frac{k_i B_L d_i h_i (p_i y_i + d_i y_i)}{2 p_i y_i} + \frac{k_i B_L \alpha_i d_i^2 (\pi_i + r_i)}{2} \leq TC^{CC} - \sum_{j \neq i} TC_j^{IS} \quad (\text{B.2})$$

而每個產品之 k_i 的乘數上界 x_i^{UB} ，可以由上式(B.2)求得，表示如下式(B.3)。

$$k_i = \frac{TC^{CC} - \sum_{j \neq i} TC_j^{IS} + \sqrt{p_i y_i \left(TC^{CC} - \sum_{j \neq i} TC_j^{IS} \right)^2 - 2 \Psi_i (A_i + C_i)}}{B_L \Psi_i} \quad (\text{B.3})$$

$$\Psi_i = d_i h_i (p_i y_i + d_i y_i) + \alpha_i d_i^2 p_i y_i (\pi_i + r_i)$$

在 PoT 策略下 $k_i = 2^{x_i^{UB}(B_L)}$ ，所以 k_i 的乘數上界 x_i^{UB} 為

$$x_i^{UB}(B_L) = \left\lceil \log_2 \left(\frac{TC^{CC} - \sum_{j \neq i} TC_j^{IS} + \sqrt{p_i y_i \left(TC^{CC} - \sum_{j \neq i} TC_j^{IS} \right)^2 - 2 \Psi_i (A_i + C_i)}}{B_L \Psi_i} \right) \right\rceil \quad (\text{B.4})$$

B.2 排程初始程序 (Proc IS) 說明

產品的初始排程程序目的是要產生一組初始的生產排程與檢驗排程，將產品指派至 W_p 與 W_l 完成其建構。主要的程序步驟請參考 Yao *et al.* (2003)，此附錄 B.2 以簡單的範例來說明本研究 Proc IS 的演算程序。

假設產品項的相關資料如表 B-1 的前四列：

表 B-1 Proc IS 範例產品的基本資料

產品 i	產品 1	產品 2	產品 3	產品 4
k_i	2	1	4	8
生產時間	5	9	7	2
檢驗時間	8	6	3	6
最大值	8	9	7	6
源自於	檢驗時間	生產時間	生產時間	檢驗時間
排序	2	1	3	4

首先，先求取個別產品項之生產時間與檢驗時間的最大值，換言之，計算 $Max(s_i + k_i \bar{B}_i d_i / p_i + k_i \bar{B}_i d_i / y_i, z_i + k_i \bar{B}_i d_i / y_i + k_i \bar{B}_i d_i (m_i - 1) / m_i p_i)$ 。結果如表 B-1 第四、五列。然後產品項依此結果的大小，做為進入初始排程的順序（表 B-1 第六列）。因為 $k_i = 1$ 的產品項在每一個基期都需要生產，所以優先指派，因此產品 2 首先進入排序。產品 2 加入生產與檢驗排程後結果如表 B-2。表 B-2 中的第二列總週期數的決定由最小公倍數 $LCM\{k_i\} = 8$ 算出。

接著，按照各產品所得最大值（表 B-1 第六列）的大小做降冪排列 $\{1, 3, 4\}$ ，先取排名順序第一的產品 1 加入排程。由於產品 1 的 $k_i = 2$ ，所以排至基期 1 或 2 作為起始皆可，其兩種方式產生負載結果相同，因此先將其排入基期 1。產品 1 加入生產與檢驗排程後結果如表 B-3。

表 B-2 產品 2 加入排程 (Proc IS 範例)

W_p	產品 2				總生產時間
k 值	1				
基期 1	9				9
基期 2	9				9
基期 3	9				9
基期 4	9				9
基期 5	9				9
基期 6	9				9
基期 7	9				9
基期 8	9				9

W_l	產品 2				總檢驗時間
k 值	1				
基期 1	6				6
基期 2	6				6
基期 3	6				6
基期 4	6				6
基期 5	6				6
基期 6	6				6
基期 7	6				6
基期 8	6				6

表 B-3 產品 1 加入排程 (Proc IS 範例)

W_p	產品 2	產品 1			總生產時間
k 值	1	2			
基期 1	9	5			14
基期 2	9	0			9
基期 3	9	5			14
基期 4	9	0			9
基期 5	9	5			14
基期 6	9	0			9
基期 7	9	5			14
基期 8	9	0			9

W_l	產品 2	產品 1			總檢驗時間
k 值	1	2			
基期 1	6	8			14
基期 2	6	0			6
基期 3	6	8			14
基期 4	6	0			6
基期 5	6	8			14
基期 6	6	0			6
基期 7	6	8			14
基期 8	6	0			6

接下來，依據降冪排列將排名順序第二的產品 3 加入排程，因為其 $k_i=4$ ，可以從基期 1, 2, 3 或 4 開始排列，但是產生負載的效果不同。因此，將分別將產品 3 的生產時間與檢驗時間放入 W_p 與 W_l ，並尋找從四個不同基期開始，當加入產品 3 後能產生總時間較低負載的週期，其即為設定產品 3 開始的位置。經測試後，產品 3 要從基期 4 開始排起。產品 3 加入生產與檢驗排程後結果如表 B-4。

表 B-4 產品 3 加入排程 (Proc IS 範例)

W_p	產品 2	產品 1	產品 3		總生產時間
k 值	1	2	4		
基期 1	9	5	0		14
基期 2	9	0	0		9
基期 3	9	5	0		14
基期 4	9	0	7		16
基期 5	9	5	0		14
基期 6	9	0	0		9
基期 7	9	5	0		14
基期 8	9	0	7		16

W_l	產品 2	產品 1	產品 3		總檢驗時間
k 值	1	2	4		
基期 1	6	8	0		14
基期 2	6	0	0		6
基期 3	6	8	0		14
基期 4	6	0	3		9
基期 5	6	8	0		14
基期 6	6	0	0		6
基期 7	6	8	0		14
基期 8	6	0	3		9

最後，將降冪排列中排名順序第三的產品 4 加入排程，因為其 $k_i=8$ ，所以從基期 1,2,...8 開始排列，然後分別將其生產時間與檢驗時間放入 W_p 與 W_l ，並尋找從八個不同基期開始，加入產品 4 後能產生總時間較低負載的週期位置，即為開始的位置。經測試後，產品 4 要從基期 2 開始排起。產品 4 加入生產與檢驗排程後結果如表 B-5。

表 B-5 產品 4 加入排程 (Proc IS 範例)

W_p	產品 2	產品 1	產品 3	產品 4	總生產時間
k 值	1	2	4	8	
基期 1	9	5	0	0	14
基期 2	9	0	0	2	11
基期 3	9	5	0	0	14
基期 4	9	0	7	0	16
基期 5	9	5	0	0	14
基期 6	9	0	0	0	9
基期 7	9	5	0	0	14
基期 8	9	0	7	0	16

W_l	產品 2	產品 1	產品 3	產品 4	總檢驗時間
k 值	1	2	4	8	
基期 1	6	8	0	0	14
基期 2	6	0	0	6	12
基期 3	6	8	0	0	14
基期 4	6	0	3	0	9
基期 5	6	8	0	0	11
基期 6	6	0	0	0	6
基期 7	6	8	0	0	14
基期 8	6	0	3	0	9

Proc IS 的做法即將每個產品依上述方法加入生產排程與檢驗排程，直到所有的產品排完，即可完成初始排程。

B.3 排程撫平程序 (Proc SS) 說明

排程撫平程序主要是藉由調整生產/檢驗排程(W_p 與 W_l) 中基期之機台的負載, 使得原本不可行的生產/檢驗排程變成合理可行(每一個基期的總生產時間與總檢驗時間皆不超出基本週期時間的長度)。本部分主要參考 Yao *et al.* (2003) 排程撫平方法中的三個主要子程序: 移出程序、成對交換程序與 2 對 1 互換程序, 經過本研究按照 ELISP 模式修正後, 在此以下列簡單範例說明三個子程序的處理做法。

一、移出程序 (Remove Routine)

移出程序的作法是從排程 W 中, 將擁有最大負載量之某產品 i (其 k_i 大於 1 之產品), 將其生產/檢驗時間從原本基期位置移至其他基期, 並指派此產品 i 的生產/檢驗時間至其他會造成總生產/總檢驗時間較小的基期內。

表 B-6 移出程序(改善前)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	8	4	2	2	2	$B=25$
基期 1	0	0	7	0	5	12
基期 2	0	10	0	8	0	18
基期 3	0	0	7	0	5	12
基期 4	0	0	0	8	0	8
基期 5	16	0	7	0	5	28
基期 6	0	10	0	8	0	18
基期 7	0	0	7	0	5	12
基期 8	0	0	0	8	0	8

W_l	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	8	4	2	2	2	$B=25$
基期 1	0	0	10	0	3	13
基期 2	0	12	0	5	0	17
基期 3	0	0	10	0	3	13
基期 4	0	0	0	5	0	5
基期 5	10	0	10	0	3	23
基期 6	0	12	0	5	0	17
基期 7	0	0	10	0	3	13
基期 8	0	0	0	5	0	5

假設 $W \equiv \max(W_p, W_l)$ 時, 可得到最大負載量之某產品 i 來自於檢驗排程 W_p , 即 $W \equiv \max(W_p, W_l) \equiv W_p$ (見表 B-6 中改善前之 W_p 與 W_l)。在調整前 W_p 中最大總生產時間為 28 (基期 5), 其大於 B (假設得到最佳的 $B=25$), 所以不是可行的解。本程序首先將最大機台工作負載的基本週期 5 上生產的產品(其 k_i 大於 1 的產品)找出, 然後按照其生產時間的長短, 以降冪方式排序可以得到 {2,3,4}。然後以生產時間最長之產品 2 先由基本週期 5 移出重新

排程，此時，相對應新的檢驗排程 W_I 中的產品2也要執行相同的動作將檢驗時間移出。由於產品2的 $k_i=8$ ，所以可以將生產時間調整至其他可能的七個基期，尋找可以產生較小負載的結果。在調整的同時，相對應的檢驗排程 W_I 內的檢驗時間也要做一致的位置調整（表 B-7）。

改善後 W_p 要檢查每一個基期的總生產時間皆要小於 B ，同時也要檢查 W_I 在經過相同調整後是否每一個基期的總檢驗時間皆要小於 B 。當調整後的 W_p 與 W_I 內總生產時間與總檢驗時間皆小於 B （改善後最大總檢驗時間為24），才算是合理可行的解，否則為不可行解。當為不可能解時，恢復未執行此程序前狀態，跳至成對交換程序。

表 B-7 移出程序(改善後)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	8	4	2	2	2	$B=25$
基期 1	0	0	7	0	5	12
基期 2	0	10	0	8	0	18
基期 3	0	0	7	0	5	12
基期 4	0	0	0	8	0	8
基期 5	0	0	7	0	5	12
基期 6	0	10	0	8	0	18
基期 7	0	0	7	0	5	12
基期 8	16	0	0	8	0	24

W_I	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	8	4	2	2	2	$B=25$
基期 1	0	0	10	0	3	13
基期 2	0	12	0	5	0	17
基期 3	0	0	10	0	3	13
基期 4	0	0	0	5	0	5
基期 5	0	0	10	0	3	13
基期 6	0	12	0	5	0	17
基期 7	0	0	10	0	3	13
基期 8	10	0	0	5	0	15

二、成對交換程序 (Exchange Routine)

成對交換程序是將 W 中的最大負載週期 τ ，將其產品 i (其 k_i 需大於1的產品)的生產/檢驗時間與另一項擁有較短生產/檢驗時間長度的產品(其 k_i 需大於1的產品)進行交換(此擁有較短生產/檢驗時間長度的產品，在週期 τ 中並不生產)。假設 $W \equiv \max(W_p, W_I)$ ，本程序得到擁有最大負載量之某產品 i 來自於生產排程 W_p ，即 $W \equiv \max(W_p, W_I) \equiv W_p$ (見表 B-8 中改善前之 W_p 與 W_I)。在調整前 W_p 中最大總生產時間為38(基期2)，其大於 B (假設得到最佳的 $B=35$)，所以是不可行的解(最大總檢驗時間為36，大於 B)。

表 B-8 成對交換程序(改善前)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	1	4	2	2	2	$B=35$
基期 1	16	0	0	0	3	19
基期 2	16	10	5	7	0	38
基期 3	16	0	0	0	3	19
基期 4	16	0	5	7	0	28

W_l	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	1	4	2	2	2	$B=35$
基期 1	10	0	0	0	2	12
基期 2	10	12	8	6	0	36
基期 3	10	0	0	0	2	12
基期 4	10	0	8	6	0	24

本程序首先將 W_p 基期 2 上各產品的生產時間依降冪方式排序可以得到產品集合{1,5,3}(其中 $k_i=1$ 與此基期無生產時間者的產品不需列入)，然後將未在基期 2 上生產但 $k_i > 1$ 的產品集合，按照生產時間依降冪方式排序可以得到{4}。因此，將產品 4 從原本基期 1 開始生產移至基期 2 開始生產，產品 1 則是從原本基期 2 開始生產移至基期 1 開始生產。在調整的同時，相對應的檢驗排程 W_l 內的檢驗時間也要做一致的位置調整 (表 B-9)。

表 B-9 成對交換程序(改善後)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	1	4	2	2	2	$B=35$
基期 1	16	10	0	0	0	26
基期 2	16	0	5	7	3	31
基期 3	16	0	0	0	0	16
基期 4	16	0	5	7	3	31

W_l	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	1	4	2	2	2	$B=35$
基期 1	10	12	0	0	0	22
基期 2	10	0	8	6	2	26
基期 3	10	0	0	0	0	10
基期 4	10	0	8	6	2	26

改善後的 W_p 每一個基期皆要小於 B ，同時也要檢查 W_l 在經過相同調整後是否每個基期皆要小於 B (表 B-9)。當調整後的生產與檢驗排程內總時間皆小於 B (調整後最大總生產時間為 31；最大總檢驗時間為 26)，才算是合理可行的解，否則為不可行解。當為不可行解，恢復未執行此程序前狀態，跳至 2 對 1 互換程序。

三、2 對 1 互換程序 (Two-One Routine)

2 對 1 互換程序將 W 中，兩項 k_i 大於 1，且基本週期中生產/檢驗總時間長度非最大機器負載(σ_1 與 σ_2)的產品，與基本週期 τ 中擁有最大機器負載之產品(σ_3)進行互換(此兩項非最大機器負載的產品，在週期 τ 中並不生產)，以此將降低排程 W 中的最大負載。

假設 $W \equiv \max(W_p, W_l)$ ，本程序得到擁有最大負載量之某產品 i 來自於檢驗排程 W_l ，即 $W \equiv \max(W_p, W_l) \equiv W_l$ (見表 B-10 中改善前之 W_p 與 W_l)。在調整前 W_l 中最大總檢驗時間為 35(基本週期 1)，其大於 B (假設得到最佳的 $B=33$)，所以是不可行的解。

表 B-10 2 對 1 互換程序(改善前)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	1	4	2	2	2	$B=33$
基期 1	16	10	6	0	0	32
基期 2	16	0	0	3	2	21
基期 3	16	0	6	0	0	22
基期 4	16	0	0	3	2	21

W_l	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	1	4	2	2	2	$B=33$
基期 1	10	15	10	0	0	35
基期 2	10	0	0	6	5	21
基期 3	10	0	10	0	0	20
基期 4	10	0	0	6	5	21

本程序首先利用位在 W_l 中最大機台工作負載的基本週期 1 上生產的產品(其 k_i 大於 1 的產品)找出，然後按照其檢驗時間的長短，以降冪方式排序可以得到 $\{1,3\}$ 。而未安排在基本週期 1 生產但是 k_i 大於 1 的產品，本程序將其找出，然後按照其檢驗時間以升冪方式排序可以得到 $\{4,5\}$ 。要進入 2 對 1 互換程序，所找到產品集中要選取兩項產品其檢驗時間(本範例產品 4 的檢驗時間為 5，產品 5 的檢驗時間為 6)相加不超過要進行交換產品的檢驗時間(產品 1 的檢驗時間為 15)。再者，選取的兩項產品需在同一基期有檢驗時間發生(本範例產品 4 與 5 的檢驗時間發生在基期 2)。

所以，本程序將產品 4 與產品 5 移至基期 1 開始進行檢驗，而產品 1 則從基本週期 1 中移出至產品 4 與產品 5。改善後的每一個檢驗基本週期皆要小於 B ，同時也要檢查 W_p 在經過相同調整後是否每一個生產基本週期

皆要小於 B (表 B-11)。當調整後的生產與檢驗排程內總時間皆小於 B (調整後最大總檢驗時間為 31，小於 B)，才算是合理可行的解。否則，其無合理可行的解。

表 B-11 2 對 1 互換程序(改善後)範例

W_p	產品 2	產品 1	產品 3	產品 5	產品 4	總生產時間
k 值	1	4	2	2	2	$B=33$
基期 1	16	0	6	3	2	27
基期 2	16	10	0	0	0	26
基期 3	16	0	6	3	2	27
基期 4	16	0	0	0	0	16

W_l	產品 2	產品 1	產品 3	產品 5	產品 4	總檢驗時間
k 值	1	4	2	2	2	$B=33$
基期 1	10	0	10	6	5	31
基期 2	10	15	0	0	0	25
基期 3	10	0	10	6	5	31
基期 4	10	0	0	0	0	10

B.4 B 值的下界 B_t^{PCL} 與 B_t^{ICL} 推導

一、下界 B_t^{PCL} 的推導

從式(4.3)中得知生產產能限制式

$$\sum_{i=1}^n \sum_{j=0}^{O_i} \left(s_i + \frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\phi(j,t)} \leq B \quad (\text{B.5})$$

式(B.5)利用移項求取 B

$$\begin{aligned} \sum_{i=1}^n \sum_{j=0}^{O_i} (s_i) w_{ij\phi(j,t)} &= B - B \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\phi(j,t)} \\ B &= \sum_{i=1}^n \sum_{j=0}^{O_i} (s_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\phi(j,t)} \right] \right. \end{aligned}$$

所以，下界 B_t^{PCL} 為

$$B_t^{PCL} = \sum_{i=1}^n \sum_{j=0}^{O_i} (s_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{p_i} + \frac{k_i B d_i}{y_i} \right) w_{ij\phi(j,t)} \right] \right. \quad (\text{B.6})$$

二、下界 B_t^{ICL} 的推導

從式(4.4)中得知生產產能限制式

$$\sum_{i=1}^n \sum_{j=0}^{O_i} \left(z_i + \frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\phi(j,t)} \leq B \quad (\text{B.7})$$

式(B.7)利用移項求取 B

$$\begin{aligned} \sum_{i=1}^n \sum_{j=0}^{O_i} (z_i) w_{ij\phi(j,t)} &= B - B \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\phi(j,t)} \\ B &= \sum_{i=1}^n \sum_{j=0}^{O_i} (z_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\phi(j,t)} \right] \right. \end{aligned}$$

所以，下界 B_t^{ICL} 為

$$B_t^{ICL} = \sum_{i=1}^n \sum_{j=0}^{O_i} (z_i) w_{ij\phi(j,t)} \left/ \left[1 - \sum_{i=1}^n \sum_{j=0}^{O_i} \left(\frac{k_i B d_i}{y_i} + \frac{k_i B d_i (m_i - 1)}{m_i p_i} \right) w_{ij\phi(j,t)} \right] \right. \quad (\text{B.8})$$