

東 海 大 學

工業工程與經營資訊研究所

碩士論文

多目標與限制驅導式零工式生產排程之探討

A Study of Multi-objective and Drum-Buffer-Rope

Job Shop Scheduling



研究生：阮亦群

指導教授：張炳騰 教授

中華民國九十六年六月

A Study of Multi-objective and Drum-Buffer-Rope Job Shop Scheduling

By
Yi-Chun Juan

Advisor : Prof. Ping-Teng Chang

A Thesis
Submitted to the Institute of Industrial Engineering and
Enterprise Information at Tunghai University
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in
Industrial Engineering and Enterprise Information

June 2007
Taichung , Taiwan , Republic of China

多目標與限制驅導式零工式生產排程之探討

學生：阮亦群

指導教授：張炳騰 教授

東海大學工業工程與經營資訊研究所

摘要

本研究以零工式排程型態為基礎，提出一具混合式遺傳演算法機制的限制驅導排程技術，以質性與量性瓶頸資源需求導向為優先，針對多目標營運策略訴求，求解具穩健性的排程規劃。

由於企業追求多項營運目標，常受限於不同型態的瓶頸資源，管理者因受到多瓶頸資源干擾，而在執行排程規劃時變得困難。然而傳統排程研究多僅對量性瓶頸資源做規劃，忽略如潛在獲利、顧客歷史交易紀錄之類的質性瓶頸資源影響。另外，亦缺乏考量系統可能遭受意外中斷而停擺，造成企業損失之負面影響發生，這類規劃結果不具穩健性，不適用於實際環境。

本研究利用一改良式限制驅導排程方法，針對有限資源進行最大化利用，於企業多目標營運策略下，考量訂單交期達交率、排程製距、瓶頸工作站使用率，並建構合適的緩衝時間長度，保護瓶頸作業穩定產出，同時規劃結果具有彈性、可吸收一定波動異常現象干擾，使其可應用於實務界的限制驅導零工式生產排程系統。

關鍵字詞：限制驅導排程技術、零工式生產排程、多目標、多瓶頸資源、穩健排程

A Study of Multi-objective and Drum-Buffer-Rope Job Shop Scheduling

Student : Yi-Chun Juan

Advisor : Dr. Ping-Teng Chang

Institute of Industrial Engineering and Enterprise Information
Tunghai University

ABSTRACT

In this research, a scheduling methodology is proposed that Drum-Buffer-Rope with Hybrid Genetic Algorithm. This methodology addresses the utilization of quantitative and qualitative constrained resources and multiple-objectives are concerned. We present a DBR scheduling method with Hybrid Genetic Algorithm to find a robust and flexible schedule (solution).

As a context for solving multi-objective scheduling problems, the job shop model is highly complicated. It is restricted by different types of the constrained resources for enterprises pursuing the business achievements. The scheduling problems become more complex by multiple-bottleneck resources disturbed. In tradition, most of the researches however have only concerned the qualitative constrained resources, and ignored the quantitative constrained resources which were included potential profit and historical trade records. Besides, these studies also lack of considering idle condition in production systems when the unexpected events or disruptions occur.

This scheduling method can gain the maximal utilization of constrained resources, construct the appropriate time buffer length, and keep the throughput of bottleneck operations from starvation due to manufacturing fluctuations. The results will be identified to support the effectiveness of DBR scheduling system within a complex manufacturing environment in real factory conditions.

Keywords: Drum-Buffer-Rope; Job shop scheduling; Multiple objectives; Multiple-bottleneck resources; Robust scheduling

誌謝

看著隨風灑落的鳳凰花瓣，我知道我做到了一完成現階段任務。回想這兩年來的研究路程，首先要感謝指導教授張炳騰博士，不僅在學業上從旁教導，鼓勵我們思考、創新，且在處事上給予我諸多指引，讓我能夠重新對相同事物有別的想法、更勇於競爭。由衷感謝老師的教導。而口試期間，承蒙洪堯勳博士、白炳豐博士、陳琨太博士撥空審閱論文，提供寶貴意見與指導，使論文內容更臻完善，在此致上最深的謝意。

就讀研究所期間，感謝國禎、國平、欣怡、金鍊、志昇、龍廷、舜麟學長姐在課業上的指導，而學長姊們嚴謹態度也令我深深佩服，研究室伙伴正明、季詮、敬予、曉嬋與學弟妹哲慎、維揚、堅元、明燕、竣翔的真情相挺，一起挑燈夜戰抱佛腳、一起為報告頭痛、一起為年會忙碌，強烈的革命情感令人難忘。另外其他研究室同學與學弟妹也是這兩年來的親密戰友，很高興有你們。

最後特別要感謝家人及女友佳穎的關懷與照顧，使我能在無憂無慮的環境下，順利完成學業。感謝所有曾經關心及幫助我的人，謝謝。

阮亦群 謹誌於

東海大學工業工程與經營資訊所

智慧知識系統研究室

民國九十六年七月

目錄

摘要.....	I
ABSTRACT.....	II
誌謝.....	III
目錄.....	IV
圖目錄.....	VI
表目錄.....	VII
第一章 緒論.....	1
1.1 研究背景與動機	1
1.2 研究目的	2
1.3 研究假設與範圍	3
1.4 研究方法與步驟	3
1.5 論文架構	4
第二章 文獻探討	5
2.1 排程問題之描述	5
2.1.2 多目標排程	6
2.2 限制理論與限制驅導排程技術	10
2.2.1 限制理論	10
2.2.2 限制驅導排程技術	13
2.3 穩健排程	16
2.4 遺傳演算法	18
2.4.1 編碼	20
2.4.2 運算子	23
2.4.3 遺傳演算法於排程問題之應用	26
2.5 禁忌搜尋法	28
2.5.1 禁忌搜尋法簡介	28
2.5.2 禁忌搜尋法在排程問題之應用	29
第三章 多目標與限制驅導零工式生產排程架構	30
3.1 零工式生產、多瓶頸資源與多目標模式	31
3.1.1 多瓶頸資源與零工式生產架構	32

3.1.2 質性瓶頸－懲罰函數模式架構	33
3.1.3 量性瓶頸－多目標函數模式架構	36
3.2 限制驅導排程演算法	41
3.2.1 限制驅導排程技術的運作	41
3.2.2 限制驅導排程技術結合遺傳演算法機制的運作架構	46
3.2.3 解碼法則	54
3.2.4 混合式遺傳演算法(HGA)	58
3.3 中斷測試與穩健性	62
3.3.1 中斷測試架構：	62
第四章 系統實證	66
4.1 Job shop 排程實證一	66
4.1.1 Job shop 排程實證一之求解結果	72
4.1.2 實證一結果探討與分析	76
4.2 Job shop 排程實證二	79
4.2.1 Job shop 排程實證二之求解結果	85
4.2.2 實證二結果探討與分析	87
4.3 中斷測試與穩健排程	90
4.3.1 中斷測試：機台故障	90
4.3.2 中斷測試：物料短缺	93
4.3.3 中斷測試：緊急插單	95
4.4 質性與量性瓶頸資源之敏度度分析	105
4.5 本章小結	108
第五章 結論與未來研究方向	109
5.1 結論	109
5.2 未來研究方向	110
參考文獻	111

圖目錄

圖 2.3 一般遺傳演算法運作流程	18
圖 2.4 分支圖.....	22
圖 3.1 多目標限制驅導零工式生產排程規劃架構圖	30
圖 3.2 多目標限制適應函數架構圖	32
圖 3.3 交期之模糊隸屬函數	38
圖 3.4 配合度之模糊隸屬函數	40
圖 3.5 修正型 DBR 架構示意圖	42
圖 3.6 緩衝設置架構示意圖	43
圖 3.7 總緩衝時間長度示意圖	43
圖 3.8 廢墟圖.....	44
圖 3.9 廢墟推平後之甘特圖	44
圖 3.10 投料 Rope 制定之甘特圖	45
圖 3.11 投料時程修正後之排程甘特圖	46
圖 3.12 限制驅導排程技術 DBR 結合遺傳演算法 GA 之架構	47
圖 3.13 上半部基因編碼表示	48
圖 3.14 下半部基因編碼表示	48
圖 3.15 上半部基因初始族群示意圖	49
圖 3.16 下半部基因初始族群示意圖	49
圖 3.17 上半部基因交配示意圖	53
圖 3.18 上半部突變示意圖	54
圖 3.19 解碼流程圖	55
圖 3.20 混合式遺傳演算法(HGA)運作流程圖	61
圖 4.1 瓶頸工作站之排程甘特圖	73
圖 4.2 實證一之 DBR_GA 適應函數值趨勢圖	75
圖 4.3 實證一之 DBR_HGA 適應函數值趨勢圖	76
圖 4.4 實證二之 DBR_HGA 適應函數值趨勢圖	87

表目錄

表 2.2 多目標排程文獻整理	8
表 2.3 行軍隊伍與生產系統比喻關係表	12
表 2.4 限制驅導排程技術文獻整理表	15
表 2.5 穩健排程文獻整理表	17
表 2.6 遺傳演算法之特性與優缺點歸納表	27
表 3.1 隨機指標表	34
表 3.2 AHP 評估尺度	35
表 3.3 各因素之權重制訂	35
表 3.4 各方案之評估值	36
表 3.5 各方案之加權排序	36
表 3.6 各訂單之加權值	50
表 3.7 訂單平均基因解碼順序	50
表 3.8 加權後差異平方總和之計算	51
表 3.9 加權後最大差異平方總和之計算	51
表 3.10 各訂單作業之基因	59
表 3.11 S1 的鄰域	60
表 3.12 S2 的鄰域	60
表 3.13 S3 的鄰域	60
表 3.14 S4 的鄰域	61
表 4.1 情境一的訂單資料	66
表 4.2 各工作站產能負荷表	67
表 4.3 作業分群表	67
表 4.4 質性限制之權重值	68
表 4.5 實證一的質性限制(一)—訂單利潤	68
表 4.6 實證一的質性限制(二)—歷史交易	69
表 4.7 實證一的質性限制(三)—市場因素	69
表 4.8 實證一的質性限制(四)—潛在訂單	69
表 4.9 實證一的質性限制之訂單順序	69
表 4.10 量性限制之權重值	70

表 4.11 質性與量性限制之權重值	70
表 4.12 遺傳演算法(GA)中參數設定值	71
表 4.13 DBR_GA 與 DBR_HGA 參數設定值	71
表 4.14 實證一的最適排程結果	72
表 4.15 實證一的訂單達交結果	73
表 4.16 實證一的非瓶頸資源配合度結果	73
表 4.17 全部作業解碼順序	74
表 4.18 各訂單的作業解碼順序平均與排序	74
表 4.19 加權後差異平方總和之計算	75
表 4.20 實證一以三種方法求解之數據表	76
表 4.21 實證二的訂單資料	79
表 4.22 實證二之作業分群資訊	79
表 4.23 實證二的質性限制(一)－訂單利潤之權重值	81
表 4.24 實證二的質性限制(二)－歷史交易	82
表 4.25 實證二的質性限制(三)－市場因素	83
表 4.26 實證二的質性限制(四)－潛在訂單	84
表 4.27 實證二的質性限制之訂單順序	85
表 4.28 實證二之最適排程結果	86
表 4.29 實證二以三種方法求解之數據表	87
表 4.30 實證一與實證二之故障資訊	90
表 4.31 實證一的機器故障中斷測試干擾結果	91
表 4.32 實證二的機器故障中斷測試干擾結果	92
表 4.33 兩實證問題之物料短缺資訊	93
表 4.34 實證一的物料短缺中斷干擾測試結果	94
表 4.35 實證二的物料短缺中斷干擾測試結果	95
表 4.36 實證一之緊急訂單資訊	96
表 4.37 實證一的質性限制(一)－訂單利潤	96
表 4.38 實證一的質性限制(二)－歷史交易	96
表 4.39 實證一的質性限制(三)－市場因素	97
表 4.40 實證一的質性限制(四)－潛在訂單	97

表 4.41 實證一的質性限制之訂單順序	97
表 4.42 實證二之緊急訂單資訊	98
表 4.43 實證二的質性限制(一)－訂單利潤之權重值	98
表 4.44 實證二的質性限制(二)－歷史交易	99
表 4.45 實證二的質性限制(三)－市場因素	100
表 4.46 實證二的質性限制(四)－潛在訂單	101
表 4.47 實證二的質性限制之訂單順序	102
表 4.48 實證一緊急插單中斷干擾測試結果	103
表 4.49 實證二緊急插單中斷干擾測試結果	104
表 4.50：實證一以權重組合 1 之實驗測試結果	105
表 4.51：實證二以權重組合 1 之實驗測試結果	105
表 4.52：實證一以權重組合 2 之實驗測試結果	106
表 4.53：實證二以權重組合 2 之實驗測試結果	106
表 4.54：實證一以權重組合 3 之實驗測試結果	107
表 4.55：實證二以權重組合 3 之實驗測試結果	107

第一章 緒論

1.1 研究背景與動機

市場競爭日趨激烈，企業永續經營必須仰賴足夠競爭力與應變能力，而排程規劃是生產製造應變能力的關鍵所在。實務上的排程規劃受限於不同限制資源 (Constrained resources)，如市場需求、訂單交期要求、機器產能負荷等，而不同限制資源更交互影響企業追求多目標策略。然而傳統的排程研究多數僅考慮量性限制資源 (Quantitative constrained resources)，針對現場量性瓶頸資源做嚴密管控規劃，而忽略質性限制資源 (Qualitative constrained resources) 對排程規劃造成的影響，使得所制定之排程規劃並非整體績效最佳。

零工式生產 (Job Shop) 為排程規劃問題最複雜的一種，由於各訂單的加工途程並非完全相同，機台產能負荷隨訂單增加使規劃者更難平衡現場產能，同時因機台產能負荷之不平衡，在不同加工限制下，造成工廠量性瓶頸 (Quantitative Bottleneck) 隨規劃週期而漂移不定；另一方面，在接單式生產 (Make-To-Order) 的製造策略下，重要訂單優先考量的規劃政策亦成為現場的質性瓶頸 (Qualitative Bottleneck) 之來源。製造工廠除了有量性瓶頸資源影響下，同時亦必須考量質性瓶頸資源，如此一來，如何同步考量兩種瓶頸資源，在有限資源下做最佳的限制資源分配規劃成為當務之急。

一般排程規劃以派工法則作為排程方法，忽略系統瓶頸需求，造成規劃結果不佳。而限制驅導排程技術 (Drum-Buffer-Rope, DBR) 以系統限制導向為先，透過投料繩 (Rope) 與緩衝 (Buffer) 機制保護系統平順產出，排程規劃具穩健性，可有效適用於市場變動快速的競爭環境。然而 DBR 雖可提供一完善流程於系統瓶頸改善，但對細部的瓶頸生產計劃 (Drum) 制訂、緩衝時間長度設定、多瓶頸相互影響、投料機制缺乏考慮非瓶頸搶奪有限產能資源等問題，實缺一套可遵循之法則。

另外，關於排程問題的相關研究繁多，但要將其應用至實務界有所困難，因其作法仍以簡單派工法則為主，無法同時考量互相衝突的多重目標與多種瓶頸干擾排程績效的影響，且於規劃過程忽略系統中斷情事發生，使規劃結果一旦受中斷干擾後，勢必得重新規劃，而系統停擺期間已造成

企業莫大的損失。為了更契合現實世界情境，必須考量生產製造時可能發生中斷事件，而在預先排程計劃增加排程規劃彈性，使其規劃結果具有容忍力，達到穩健效應。

由於零工式排程問題為一 NP-hard 問題，求解困難度極高，而且進行排程規劃時面臨的限制種類繁多，所追求多項目標之間亦是相互衝突，如何有效分配限制資源，達成多目標訴求，來提升工廠整體營運績效，是相當值得探討。

1.2 研究目的

依據上述之研究背景與動機，將現有排程規劃問題彙總如下：

- 零工式排程問題為一 NP-hard 問題，求解困難度高，傳統數理規劃方法或一般派工法則已不適用。
- 企業以達成多項目標績效為訴求，目標之間相互衝突，且各績效目標所需資源亦不同，使排程規劃更為困難。
- 實務排程規劃受限於多種限制資源，而質性限制資源常被忽略，多重瓶頸資源限制企業追求更高整體績效。
- 現場瓶頸資源隨不同訂單增加，造成短暫或長期瓶頸漂移發生，管理者不易確認實際瓶頸所在。
- DBR 尚缺一套可遵循法則來處理細部相關排程問題。
- 排程研究無法應用至實務界，規劃過程忽略中斷干擾，排程結果不具穩健性。

本研究欲針對零工式排程問題，擬提出一結合限制驅導排程技術、遺傳演算法、禁忌搜尋法、及模糊理論的排程方法，同步考量質性與量性的限制資源與多目標訴求，制訂一具有穩健性之排程規劃，使生產製造及企業營運整體績效最佳化。本研究將排程規劃分為預先規劃排程及穩健排程測試兩階段。期望發展一應用於實務界的限制驅導零工式排程系統。

本研究重點目的如下：

1. 考量企業及工廠運作的量性限制資源與質性限制資源之影響，建構多目標零工排程規劃。
2. 利用限制驅導排程技術針對限制資源進行優先規劃考量。

3. 將傳統限制驅導排程技術的不足之處，以混合式遺傳演算法作為補強方法，強化限制驅導排程技術的規劃能力，使其排程方法更實用於現實環境。
4. 發展具有對於中斷事件有足夠容忍力或彈性的穩健排程，以降低中斷事件或干擾因素對於生產規劃的負面影響。

1.3 研究假設與範圍

在本研究中，於實體限制部份僅針對機台產能負荷最大之限制進行考量，並不討論其他實體限制影響有效產出之情形，將該機台資源稱為產能受限資源（Capacity Constrained Resource, CCR）。本研究排程模式為零工式生產問題，基本假設如下：

- 一、僅對零工式生產作討論。
- 二、忽略機台整備時間(Set-up time)及各工件在製造過程的移動搬運時間。
- 三、假設所有產出皆為良品。
- 四、不考慮作業人員因作業數量增加而產生的工作學習率。
- 五、不考量限制資源的替代產能方案。
- 六、排程時間開始時，所有訂單皆已準備就緒。

1.4 研究方法與步驟

本研究主要是透過電腦實驗的方式進行研究，實驗資料為兩組假設零工式排程問題。以此兩種實證問題對本研究所提的限制驅導排程方法，在多目標衡量下進行實驗，最後再依據實驗數據與結果作分析討論。本研究主要可分為三個步驟：

1. 確認排程問題的質性與量性瓶頸資源所在，針對多瓶頸與多重目標訴求，並設計欲求解之多目標適應函數。
2. 建構一結合限制驅導技術與遺傳演算法、禁忌搜尋法、及模糊理論的排程演算法。
3. 利用不同的中斷測試，發展具穩健性的排程規劃。

4. 以電腦進行實驗，根據上步驟各方法進行系統驗證。

1.5 論文架構

本研究論文的內容共分為五章：第一章說明本研究之動機、目的、範圍、假設、步驟等相關內容；第二章則針對本論文內容所涉及方法之相關文獻加以探討。包括限制理論、限制驅導排程技術、遺傳演算法、禁忌搜尋法、多目標問題、穩健排程之相關文獻；第三章則根據本論文之目的與文獻探討結果，提出本研究中多目標限制驅導排程系統架構與穩健排程之求解方式；第四章為本研究系統之實驗結果與分析；第五章則根據本研究所得之結果，說明研究結論與未來發展方向。本論文的進行流程如圖 1.1。

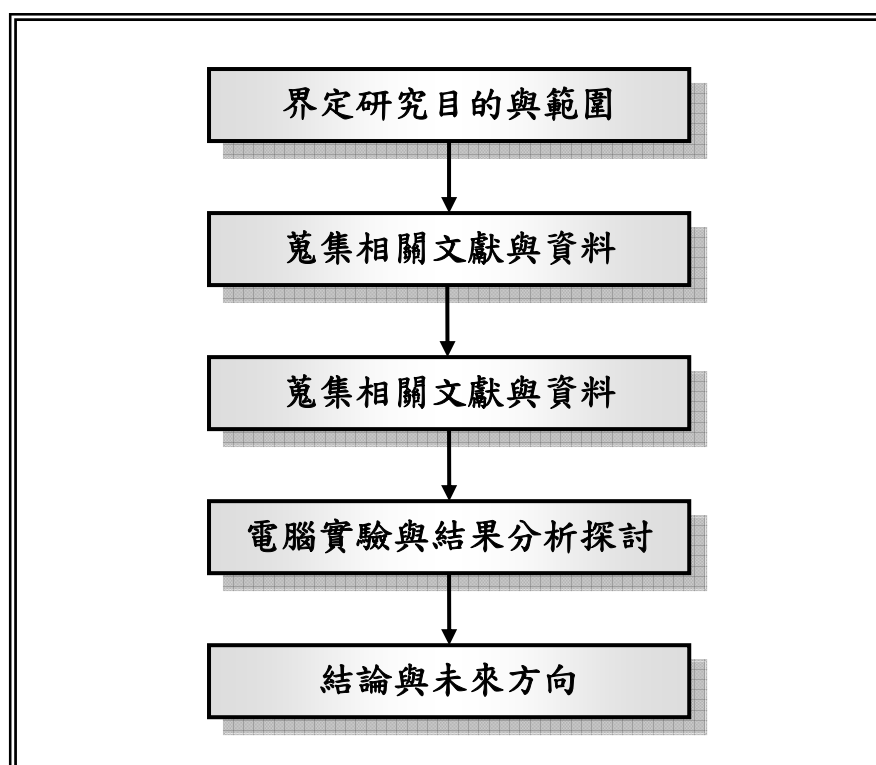


圖 1.1 研究流程圖

第二章 文獻探討

本章分為五個部份，第一節探討排程問題之分類與多目標排程問題；第二節說明限制理論與限制驅導排程技術（DBR）並整理相關研究；第三節說明穩健排程意義與其相關研究；第四節與第五節則分別說明遺傳演算法與禁忌搜尋法，並簡介該方法於排程上的應用。

2.1 排程問題之描述

Pinedo 及 Chao (1999) 指出排程規劃是一種有限資源分配的決策制定過程，常用於製造業的採購、生產、轉運、配銷等企業相關規劃活動。Leung (2004) 定義排程活動是決定所有工作在機器上加工起始時間以及加工順序，同時考慮資源限制和執行限制之最佳化問題。經由較佳的排程規劃可以讓企業資源利用最佳化，並達成公司目標或經營績效。進行排程規劃不僅可以協調製程活動，也可評估專案的完工時程，顯示作業關鍵性、及製造程序相關作業之完整時程，可清楚地讓現場管理人員瞭解目前營運情況。

2.1.1 排程問題之分類

Pinedo 及 Chao (1999) 指出生產現場的排程問題若以機器數目配合工作劃分，可將排程問題分成數種不同複雜程度的問題，依序分類如下：

1. 單機 (Single-machine) 問題：各訂單只有一道加工途程，且現場只有一台加工機器。
2. 平行機台 (Parallel-machine) 問題：各訂單只有一道加工途程，且現場只有多部加工相同功能機器。平行機台問題可細分為等效平行機台問題與非等效平行機台問題。等效平行機台 (Identical) 是指該加工站的機器的處理加工效能相同，非等效平行機台 (Non-Identical) 則表示處理加工效能存有差異，同一作業會因為指派到不同機台加工而導致加工時間不同，或良率不同。
3. 流程式生產 (Flow Shop) 問題：各訂單有數道加工途程，所有各訂單遵循相同的加工途程順序。
4. 零工式生產 (Job Shop) 問題：各訂單有數道加工途程，所有各訂單有其各自的加工途程順序。

5. 開放式生產 (Open Shop) 問題：各訂單有數道加工途程，但加工途程順序不受限制。

其中以零工式生產 (Job Shop) 問題最為複雜。由於各訂單製造途程皆不完全相同、工作站數量、機台數量與產能負荷上限、及排程績效衡量指標等限制條件下，排程問題更加複雜與困難。

Garey 等學者 (1976)、Pinedo 及 Chao (1999)、Leung (2004) 指出在 1970 年代開始，進行排程規劃時，當問題規模過大時，如：訂單數、機器數、加工途程數等變數規模的增加，排程規劃問題被認定為 NP-hard 問題。而 NP-hard 問題，表示該問題困難度極高，無法用多項式表示其求解空間，即無法以多項式時間演算法 (Polynomial time algorithms) 求解出最佳解，若是以傳統的數理規劃方式求解，則需耗費大量的時間；另外，在現實環境中，由於市場競爭日趨嚴重，企業須滿足顧客需求、快速反應市場變化，將資訊回饋到研發、製造生產部門，追求產品多樣化、客製化，並使其生產規劃具有彈性來因應變化發生，才能強化自身企業的核心競爭力。在追求上述目標下，多數企業將製造策略設定為以接單生產 (Make-to-order) 策略，而工廠生產製造則以零工式生產 (Job shop) 模式配合企業策略，達到產品少量多樣化、客製化，藉以滿足顧客需求。因此，在競爭激烈的環境下，零工式生產排程問題愈趨重要。

2.1.2 多目標排程

實務上的生產排程規劃並非僅追求單一目標，往往同時考慮多個目標，Pinedo 及 Chao (1999) 指出在實務上的排程問題通常包含多個目標。Mellor (1966) 摘要了 27 項常見排程規劃指標，如表 2.1，一般所考慮的量化目標有：交期、機器使用率、製距，...等。現實環境中還存有另外的質性因素，例如：公司策略因素、顧客的歷史交易、或是該訂單所需物料的供貨情形等。而目前多數關於排程研究中，相關文獻多著重於僅考慮量化方面目標，很少提及質性因素，亦忽略其質性限制資源，所發展的排程演算法多是以單一目標作為排程績效衡量的基礎，進行排程規劃，在追求單一績效最佳化的同時，無法兼顧其他績效考量。

表 2.1 常見排程績效指標

1. 設備閒置時間最小化(minimum idle facility investment)
2. 在製品數量最小化(minimum work-in-process inventory)
3. 設備整備成本最小化(minimum facility set-up cost)
4. 平穩化勞動量(day to day stability of work force)
5. 交期滿足(adherence to promised shipping date)
6. 產出率最大化(maximum output(product rate))
7. 物料持有成本最小化(minimum materials-handling cost)
8. 訂單優先次序的滿足(adherence to arbitrary job priorities), such as arise in dealing with preferred customers, emergency repair parts, etc.
9. 加工合理性(technological feasibility)
10. 變換生產時的敏感度(sensitivity to possible production change)
11. 彈性(general flexibility)
12. (non-dependence on unreliable process)
13. 保留生產力予緊急插單(reverse capacity for rush order)
14. 廠內運送排程最佳化(optimal in-plant transportation schedule)
15. 運送成本最小化(minimum shipping cost)
16. 預期成本最小化(minimum total expected costs, primarily in theoretical investigations)
17. 設備使用率最大化(maximum weighted facility utilization)
18. 人力使用率最大化(maximum utilization of manpower)
19. 勞工績效(optimal assignment of various labor grades)
20. 原物料存貨最小化(minimum raw material in inventory)
21. 最終成品最小化(minimum finished product inventory)
22. 存貨投資最小化(minimum investment in inventories)
23. (minimum obsolescence and deterioration of product)
24. 特定產品製距最小(shortest make-span for certain products)
25. 製距最小化(minimum overall fabrication span)
26. 損失風險最小化(minimum risk of excessive losses)
27. 預防價格的改變(anticipated changes in price)

資料來源：Mellor (1966)

關於多目標排程的文獻中，Pinedo 及 Chao (1999) 在實務上的排程問題通常包含多個目標，並利用權重法來結合數種目標績效，而各指標的權重值因排程問題所考量的規劃時程長短而更改。Brown (1997) 則提出在整個多目標的環境中，對於在規劃生產排程時應考量市場 (Marketing) 與生產 (Production) 兩大類型的因素。

本研究整理多目標排程相關文獻，其包含應用的排程問題形態、求解方法、考量目標、權重製定，整理如下表 2.2：

表 2.2 多目標排程文獻整理

作者	排程型態	求解方法	考量之目標	權重制訂
Itoh et al.(1993)	零工式排程	TLAS	最小化平均延遲時間 最小化平均流程時間	設定為 0.5
Daniels(1994)	單機排程	派工法則	最小化總流程時間 最小化最大延遲 最小化延遲數目	依照各目標 成本制訂
Murata et al. (1996)	流程式排程	遺傳演算法	最小化製距 最小化總延遲 最小化總流程時間	隨機方式決定
Kim et al. (1998) Min et al. (1998)	彈性製造系統	競爭式類神經訓練機器、工件、儲位以及搬運車選擇之派工法則搭配組合	最小化總流程時間 最小化平均延遲時間、最小化平均流程時間、加工機器平均使用率、搬運車平均使用率、最小化製距、系統平均工單數、平均在製品	依照各單一目標設定水準值，再予以評估。
Chou and Lee(1999)	流程式排程	啟發式演算法	加權總流程時間 製距	自訂權重， 相加為 1
Rajendran and Ziegler(1999)	流程式排程	啟發式演算法	加權總流程時間 加權延遲工作	直接加總
Chang and Lo (2001)	零工式排程	GA/TS	作業順序 (job-sequence) 最小化製程時間 交期滿足 機器利用率	層級分析法 (AHP) 訂定 權重
Lee(2001)	多機雙階製造系統	雙階段禁忌搜尋法；單階段禁忌搜尋法；遺傳演算法；模擬退火法；鄰近空間搜尋法	交期懲罰函數 存貨成本 加工成本	權重設定為 1：1：1
Kacem et al.(2002)	彈性零工式排程	AL/CGA	最大機器利用率 總機器利用率 製距	權重設定為 1：1：1
Esquivel et al.(2002)	零工式排程	提升轉移演算法 (Enhance devolutionary algorithms)	製距 提早完成時間 加權完成時間	優先順序列表決定
Chang et al.(2002)	流程式排程	GA	製距 總流程時間 總延遲 最大延遲	運用 gradual-priority weighting 制訂權重

作者	排程型態	求解方法	考量之目標	權重制訂
Framinan et al.(2002)	流程式排程	NEH 啟發式解法	製距 流程時間	以 posteriori heuristic 制訂權重
Ghrayeb (2003)	零工式排程	遺傳演算法	模糊製距的不確定性，模糊製距的積分值	自訂權重
Ishibuchi et al.(2003)	流程式排程	局部搜尋多目標遺傳演算法、多目標演化尋優法	製距 最大延遲	pseudoweight vector
Allahverdi (2003)	流程式排程	啟發式演算法	加權製距和平均流程時間	設定為 0.1,0.9
Lin and Liao(2003)	流程式排程	啟發式演算法	加權最大延遲	根據各成本設定權重
Allahverdi (2004)	流程式排程	DCH/SAH/MNEH /PH	製距 最大延遲	實驗三組： 0.25 : 0.75 0.5 : 0.5 0.75 : 0.25
Birman and Mosheiov (2004)	流程式排程	Johnson 演算法	最大提早完成 最大延遲 交期成本	自訂權重
Ravindran et al. (2005)	流程式排程	CR 法為基的啟發式演算法	總流程時間 製距	測試不同 主次要關係
Gupta and Sivakumar (2005)	單機排程	間斷事件模擬方法	平均週期時間 平均延遲時間 機器使用率	自訂權重
Loukil et al. (2005)	單機排程、平行機、流程式排程	模擬退火法	平均權重完工時間 平均延遲時間、平均提早完成時間、製距 最大延遲時間、最大提早完成時間、延遲 訂單數	自訂權重
Varadharajan et al. (2005)	流程式排程	模擬退火法	製距 總流程時間	製距： 0.1666 總流程時間： 0.8334
Low et al. (2006)	彈性製造系統	啟發式演算法	平均流程時間 平均延遲數目 平均機器延遲時間	權重為 1 : 1 : 1
Martin Josef Geiger (2007)	流程式排程	鄰域搜尋法	製距、總完成時間、 最大延遲時間、總延遲時間、延遲 訂單數 總機器延遲時間	測試 12 組 績效指標組合，每種組合內權重相等

資料來源：本研究整理

2.2 限制理論與限制驅導排程技術

限制理論近年來廣泛的應用到各種不同的領域中，是一種新的組織管理的思維方法，對組織或系統運作，針對最弱的環節提供持續性的改善，讓系統組織績效可整體提升，而非追求局部最佳。

在 2.1.2 小節中敘述企業營運追求多種目標績效，將傳統的排程規劃問題轉變成多目標排程問題。而根據限制理論概念，系統組織中對於每一種目標有其相對應的關鍵企業資源，例：追求製距最小化，則關鍵資源便是工廠中的可用產能 (Available capacity)，其中產能負荷最重的工作中心站，則稱瓶頸資源 (Bottleneck resources)。因此進行多目標排程規劃時，無論是量性績效指標或質性績效指標，皆會受限於不同的企業資源，因此，執行排程規劃前須先確認瓶頸資源，設法將該限制資源利用率達到最大化，才能確保系統組織的整體績效最佳。

在多目標排程問題架構下，確認影響排程問題目標的瓶頸資源所在，如何將限制理論提及組織管理的思維方法轉為實際方法，並應用到多目標排程問題，則需仰賴一套有效方法，針對有限資源做妥善分配，嚴加管控瓶頸資源，使瓶頸資源利用達到最大化，讓系統整體產出績效更高。而限制驅導排程技術 (Drum-Buffer-Rope, DBR) 的概念與此相符，利用嚴密生產排程計畫 Drum，其他資源全力配合的機制下，達成生產系統穩定產出。

關於限制理論與限制驅導排程技術的詳細架構於後續小節說明。2.2.1 小節說明限制理論；2.2.2 小節說明限制驅導排程技術。

2.2.1 限制理論

限制理論 (Theory of Constraint, TOC) 為 Goldratt, E. M. 博士於 1980 年代所提出的一種新的管理思維方法。認為任何系統或組織都應該有目標，影響系統追求更高績效或目標的障礙即為限制 (Constraint)，系統組織因為受到障礙限制，而造成系統產出受阻，無法提升系統績效。限制理論著重於找到影響系統產出的限制，並嘗試管理此限制，讓系統製造穩定，達到系統產出最大化。

限制理論提及的限制種分為兩類：一、政策限制 (Policy Constraint)：為無形的障礙，如公司政策及制度、績效指標、組織文化、法令規章等；

二、實體限制 (Physical Constraint)：為有實體的限制，例如工廠機具設備產能不足、勞工人力限制、原物料供應量等。因為一個系統組織的績效須取決於最弱的一環，此最弱的環即是所謂的限制瓶頸 (Bottleneck)，因此若能有效改善此最弱的環，則能有效的提升這個組織的整體績效，反之，若追求非限制因素的改善，系統產出仍是受限於原先限制因素，無助於組織績效改善。

Goldratt 及 Cox (1986) 提出執行 TOC 的五個步驟，作為改善限制，提升系統限制，循序改善系統整體產出績效的流程：

步驟 1. 確認 (Identify) 系統的限制—最弱的環：

組織系統的整體績效受到限制資源影響，執行組織改善前，須先確認瓶頸資源所在，管理者須嚴密監控，以求組織績效持續性的改善。

步驟 2. 決定如何充分利用 (Exploit) 系統的限制：

找出系統限制的瓶頸資源後，制定一個完善計畫，以其最有利或是瓶頸資源利用最大化之方案供組織運作，進而追求系統整理績效最佳。

步驟 3. 系統全力配合 (Subordinate) 步驟二的決定：

為保護瓶頸資源的運作計畫不受中斷干擾，其它組織資源須慎防打破瓶頸資源的運作計畫，以配合瓶頸資源為最先考量，另一方面，管理者不需刻意追求非瓶頸資源 (Non-bottleneck resources) 最大利用率，避免其他問題產生。

步驟 4. 提升 (Elevate) 系統的限制：

瓶頸資源限制系統績效，管理者須提升瓶頸資源的可用率，獲得更高的系統績效。但若是提高瓶頸資源的可用率，可能使系統組織的結構改變，原先的瓶頸資源不再是系統組織中最弱環節，導致新的瓶頸資源產生。

步驟 5. 如果限制在步驟四被打破，則回步驟一，不讓惰性成為系統限制：

限制理論講求的是系統組織中限制的持續改善，一旦新的瓶頸資源產生，則回到步驟一持續改善新的系統組織的限制。

限制理論已被應用至許多領域，如生產規劃問題、行政流程改善、專案管理、決策問題、績效衡量指標探討等。利用上述的改善邏輯，先確認影響問題的關鍵或限制，配合問題限制，尋求突破現有狀況，讓系統組織

運作更有效率與效果，達到組織績效最大化。

Goldratt 其著作：【目標】一書中，以童子軍隊伍行軍故事為例，為控制隊伍行進速度，且避免脫隊，必須平衡整體行進的速度，設定適當間隔距離來避免隊伍擁塞或拖長，達到隊伍同步前進的效果。工廠製造生產系統具有製程相依關係，這與隊伍行軍相當類似。吳，李（1999）指出一個比喻關係，如表 2.3。

表 2.3 行軍隊伍與生產系統比喻關係表

行軍隊伍	生產系統
隊伍成員	工廠機台
行軍路徑	機台加工的工件
隊員的依序關係	生產加工途程限制
隊伍速度	工廠的產出率
隊伍所拖的長度	工廠的在製品數
排與排的距離	站與站間的在製品數
隊伍的第一排	工廠的投料站
隊伍方向	訂單
目的：速度最快、隊伍整齊	目的：產出率最大、加工正確、在製品數最低

資料來源：吳，李（1999）

由表 2.3 可知，行軍隊伍可視為一個簡單的生產系統，兩者追求強化最弱的環節，平衡整個系統產出，求取系統整體同步產出最佳化，而非追求局部最佳。在生產系統中，管理者希望能將機台利用最大化，但是機台加工受到加工件的製程相依性限制，而無法任意加工，造成系統會發生機台等待加工件的閒置情況發生；另一方面，機台的可用產能與產出良率也影響生產績效，若有良好的良率與可用產能，則系統產出將會加快。從前述中可知工件的製程相依、機台的可用產能、產出良率皆是影響生產績效的限制，如何將這限制納入排程規劃考量，達到整體績效最佳化，實是生管排程規劃者的一大難題。

2.2.2 限制驅導排程技術

限制驅導排程技術(Drum-Buffer-Rope, DBR),是 Goldratt 博士於 1986 年所提出的一種製造生產排程規劃方法, DBR 以限制理論的思維為基礎,進行生產製造規劃時,以系統限制或瓶頸需求導向為先,有效的充分利用(Exploit)且系統全力配合(Subordinate)的有限資源分配排程技術。

任何系統或組織都存有限制,影響著系統追求更高績效或目標。Morris (1994)、Wu 及 Yeh (2006) 指出製造工廠的限制來自市場需求、工廠產能、物料短缺等。而相關學者 Ronen 及 Starr (1990)、Wu et al. (1994)、Simons et al. (1999)、Satya (2001)、Sirikrai 及 Yenradee (2006) 等研究中驗證出 DBR 為一有效管理限制資源產出和平衡生產流量的現場排程規劃機制,可使限制資源利用最大化。

Morris (1994) 指出 DBR 是利用一連串的規則來執行 TOC 的前三步驟,而 Schragenheim 及 Ronen (1990) 根據 TOC 思維,提出限制驅導排程技術執行步驟:

1. 確認系統實體限制: 在限制驅導式排程技術中,先將所有可用資源區分為限制資源或瓶頸(Bottleneck)與非限制資源(Non-Bottleneck)兩類,於規劃先前先確定瓶頸資源,稱之產能受限資源(Capacity Constrained Resource, CCR)。Ronen 及 Starr (1990) 定義 CCR 是小於或等於市場需求的資源; Morris (1994) 指出 CCR 是阻礙系統達到更高績效的限制所在; Sirikrai 及 Yenradee (2006) 定義 CCR 是限制製造系統產出的任何資源,在其研究中以產能小於或等於需求的工作站,設定為 CCR。
2. 決定系統緩衝(Buffer)大小: 緩衝(Buffer)可以保護瓶頸資源不受到變動因素作用而影響產出,所以選定在瓶頸資源前設置一時間緩衝,用來吸收系統波動或是無法預期的中斷事件發生,以維持生產系統穩定、高效率產出,提高排程規劃的穩健度。Morris (1994)、Satya (2001)、Satya 及 Atwater (2005)、Wu 及 Yeh (2006)、Sirikrai 及 Yenradee (2006) 指出緩衝機制的設立是為保護瓶頸資源可在生產系統中維持高效率穩定產出,不受到影響產出變動因素作用,故選定在瓶頸資源前設置一緩衝(Buffer),稱之產能受限資源緩衝(Capacity Constrained Resource Buffer),以時間緩衝吸收統計波動和異常狀況

的影響。Morris (1994) 認為有效的緩衝機制可以讓系統更有彈性的處理波動發生；Ronen 及 Starr (1990) 採用經驗法則，建議緩衝時間長度為前置期總和的四分之一；Schragenheim 與 Ronen (1990) 依據經驗與真實世界情況來調整緩衝時間長度，設定為限制資源的前置期三倍；Wu et al. (1994) 建議緩衝時間長度為製造前置期的 1.5 倍；Daniel 及 Guide (1997) 定義一般製造環境為物料釋放前置期的三倍，而在再製造 (Re-manufacture) 環境中環境的緩衝長度為過去的最小累積加工時間乘上某一固定常數。

3. 限制驅導節奏 (Drum) 之設計：Ronen 及 Starr (1990)、Morris (1994)、Satya (2001)、Sirikrai 及 Yenradee (2006) 定義鼓 (Drum) 為排定限制資源或瓶頸的細部生產排程計畫。而良好的 Drum 可控制整個系統有效產出的速度，使整體生產系統可依循細部排程節奏來維持系統最大產出。DBR 針對瓶頸資源做嚴密管控與規劃，來提高系統有效產出，關於 Drum 的細部排程制定，Goldratt (1986)、Wu 及 Yeh (2006) 提出制定 Drum 的程序，如下：

步驟 1、產生廢墟 (Generating the ruins)：廢墟 (ruins) 是表示不考慮瓶頸站產能下，所制定的理想瓶頸工作站生產排程甘特圖，由於狀似廢墟，故稱之為廢墟圖。廢墟的產生是藉由交期減去出貨緩衝而得到瓶頸的完工時程，再回推開工時程與瓶頸緩衝，得到投料時程。

步驟 2、推平廢墟 (Levelling the ruins)：由於廢墟表示為一個理想卻不可行 (infeasible) 排程規劃，瓶頸站無法同時製造多個作業，所以必須將重疊的時程部份推平，或是合理化，如此一來便可以得到合理的瓶頸生產排程規劃 (Drum)。

4. 規劃投料時程 (Rope)：確定系統瓶頸所在、瓶頸資源排程與時間緩衝的機制後，利用投料繩 (Rope) 從瓶頸機台前的緩衝區長度藉由時間逆推方式而得到原物料適當的投料時間點與品項數量，即為投料時程制定。Morris (1994)、Wu 及 Yeh (2006)、Sirikrai 及 Yenradee (2006) 指出有效的投料繩機制可使原物料投放品項、數量、投料時程正確性更高，避免在製品存貨提早發生。

關於限制驅導排程技術 DBR 的文獻整理如下表 2.4：

表 2.4 限制驅導排程技術文獻整理表

作者(年代)	排程型態	瓶頸資源設定	緩衝設定	效應
Wu et al. (1994)	零工式 生產	產能負荷最重設 定為瓶頸機台	1/2 的前置期	利用軟體模擬家俱 產業，得到更短的 製距與較高的瓶頸 使用率
Satya (2001)	零工式 生產	依產能負荷，不同 產品族有不同瓶 頸。	依照模擬情境 不同，設定不 同常數	利用 DBR 控制機 制與投料機制，獲 得較低的總成本和 存貨成本
Gupta et al.(2002)	零工式 生產	產能負荷最重設 定為瓶頸機台	未設定	利用 ARENA 軟體 模擬 DBR 與 TOC 的步驟，提出詳細 的 DBR 應用方法
Chakravorty and Atwater (2005)	零工式 生產	交期緊急的訂單	1/2 的前置 期，分成限制 緩衝跟出貨緩 衝	Free goods 增加 DBR 系統額外產 出
Sirikrai and Yenradee (2006)	流程式 生產	產能負荷最重設 定為瓶頸機台	每個作業加工 時間的標準差	產出、較低的在製 品數量、較短的週 期時間、改善產品 準時達交績效

資料來源：本研究整理

由前述的文獻整理可知，DBR 將生產系統的資源分為瓶頸與非瓶頸資源，針對瓶頸資源加以管控，制定最佳的生產排程計畫 Drum，而非受限資源則必須透過緩衝機制 Buffer 與投料機制 Rope 來配合 Drum 計畫進行，維持系統平順，如此一來，可有效監控原物料投料、加工流程和交期控制，同時緩衝機制亦有助生產系統吸收波動影響，不受中斷干擾，使生產系統的產出績效更為穩健。

DBR 雖可提供一完善流程於系統瓶頸的改善，但對於細部的 Drum 制訂與其排程規劃方法、Buffer 緩衝時間長度設定等問題尚缺一套完善手法，而多數的文獻中指出利用 DBR 技術進行規劃時，常以經驗法則或平均值作為評估依據，對於新產品或是新的製程，則無法以經驗法則來制定合宜的計畫與緩衝長度。Ronen 及 Starr (1990) 提及現實世界中，因前置期、不良率、加工時間的波動現象，若是規劃時以平均值做為評估依據，無法正確反應現實的現象，則規劃結果失去可信度；Simons et al. (1999) 指出傳

統 DBR 機制在多種產能瓶頸的生產環境中，因多瓶頸資源互相交互作用，其規劃績效無法達到良好績效；Schaefers et al. (2004) 說明市場限制下，導致系統易產生局部最佳解；Sirikrai 及 Yenradee(2006)指出傳統型的 Rope 機制，造成修正時間緩衝長度過長或緊密的缺點，同時投料時程機制缺乏考慮非瓶頸作業搶奪有限產能資源的情況；Wu 及 Yeh (2006) 指出在半導體工廠中，因製造過程中會出現多個瓶頸作業，如何排定瓶頸生產節奏 Drum 為一關鍵問題，影響整體排程劃績效。由上可知傳統的 DBR 的缺點如下幾點：

1. 無法有效考量多瓶頸的相互影響；
2. 缺乏 Drum 制訂的完整排程規劃方法；
3. 沒有一套制定 Buffer 緩衝時間長度的標準；
4. 投料時程機制缺乏考慮非瓶頸作業搶奪有限產能資源。

2.3 穩健排程

Ronen 及 Starr (1990) 指出在真實世界的製造環境中，因為前置期、不良率、加工時間等因素使得系統不穩定，而有統計波動現象存在，造成生產系統的中斷發生，而降低生產營運績效；Mikkel (2003) 指出具有穩健及彈性的排程規劃，比發生中斷再作重排程規劃更為重要。因為重新排定排程規劃需要時間，在重新規劃期間內，生產系統處於閒置，降低生產績效。因此如何在預先規劃階段考量生產排程的穩健性與彈性亦是排程人員需面對的問題。

穩健排程的主要目的是在於，讓排程遭遇變異或中斷事件時，仍然能維持好的排程系統績效，使一個具有穩健性或彈性的排程能夠吸收變異或排程中斷造成的負面影響。有關於排程穩健性(Robustness)的定義，Dooley and Mahmoodi (1992)、Jensen (2003) 等學者的研究中定義排程穩健性為系統遭遇變異時仍可運作良好的能力。Velagapudi (1992) 定義排程穩健性為排程可消除 (eliminate) 或最小化 (minimize) 變異的能力。Sevaux 及 Sørensen (2002) 指出穩健性相似於彈性，一具有彈性的排程應該較容易修正，如將初始排程依變異情況重排修正成為一個新的高品質排程。Sørensen (2002) 定義排程品質的穩健性為排程目標函數值的敏感度。舉例來說，

若生產活動發生若干干擾或變異時，排程仍然能維持高的生產績效(目標函數值)，則稱此排程具有穩健性。Artigues et al. (2005) 指出排程系統的穩健性可定義為其可吸收預期中斷事件的能力。本文中，關於穩健排程建立方法與文獻，整理如表 2.5。

表 2.5 穩健排程文獻整理表

文獻	方法與貢獻
Leon et al.(1994)	利用GA演算法產生穩健的排程，最小化預期的延遲(delay)或製距(makespan)。
Byeon et al. (1998)	將一零工式排程問題分解成數個子問題，再以啟發式演算法分別求解各個部份排程(partial schedule)。
Wu et al. (1999)	使用branch and bound演算法處理部分排程(partial schedule)相關的disjunctive graph。
Mehta & Uzsoy (1998)	使用shifting bottleneck 演算法，建立預測式排程，在其間插入閒置時間(idle time)以減小排程中斷的衝擊。
Herrmann (1999)	利用GA演算法最佳化最差(worst-case)的績效生產排程。
Shafaei & Brunn (1999)	研究在動態的零工式工廠建立強健排程的法則(rules)。
Sevaux & Sörensen (2002)	於單機排程問題中，以GA演算法求解穩健排程，其中各作業所需零件的到廠日是變動的。
Sörensen (2002)	定義評估穩健排程的目標函數，以禁忌搜尋法求解。
Jensen (2003)	於零工式排程問題中，定義穩健性衡量指標，考量排程的彈性與穩健性，以 GA 演算法求解低製距的穩健排程。

資料來源：潘 (2006)

由上述文獻探討可知，穩健排程可使排程規劃具有彈性與吸收波動現象的能力，利用機器閒置時間來吸收部份中斷干擾，使系統維持運作，並維持良好的生產績效；而限制驅導排程技術中的緩衝機制 (Buffer) 亦具有同樣的功能，利用時間緩衝來吸收系統波動，保護瓶頸計畫 (Drum)，使系統不受中斷干擾，維持系統整體平順產出。當排程系統規劃設立適當緩衝機制來容忍中斷干擾，則有助於維持系統穩健性，讓生產系統產出計畫，不受波動現象影響。現實世界的生產系統容易受到許多干擾發生而中斷運作，若能在預先規劃階段先考量中斷干擾，制定具有較高穩健性的排程計畫，則有助於維持系統運作，避免中斷而導致機器閒置、績效降低等問題。

2.4 遺傳演算法

遺傳演算法（Genetic algorithm，GA）為 Holland 於 1975 年所發表的一種應用於適應性系統（adaptive system）的搜尋機制，其應用原理是利用計算機的計算速度優勢，仿效自然生態運作「物競天擇、適者生存」的方式，藉由演算法的汰換機制，淘汰較差的族群與部分群體間交配、突變來求解最佳化問題。一般遺傳演算法運作的流程大致如圖 2.3。

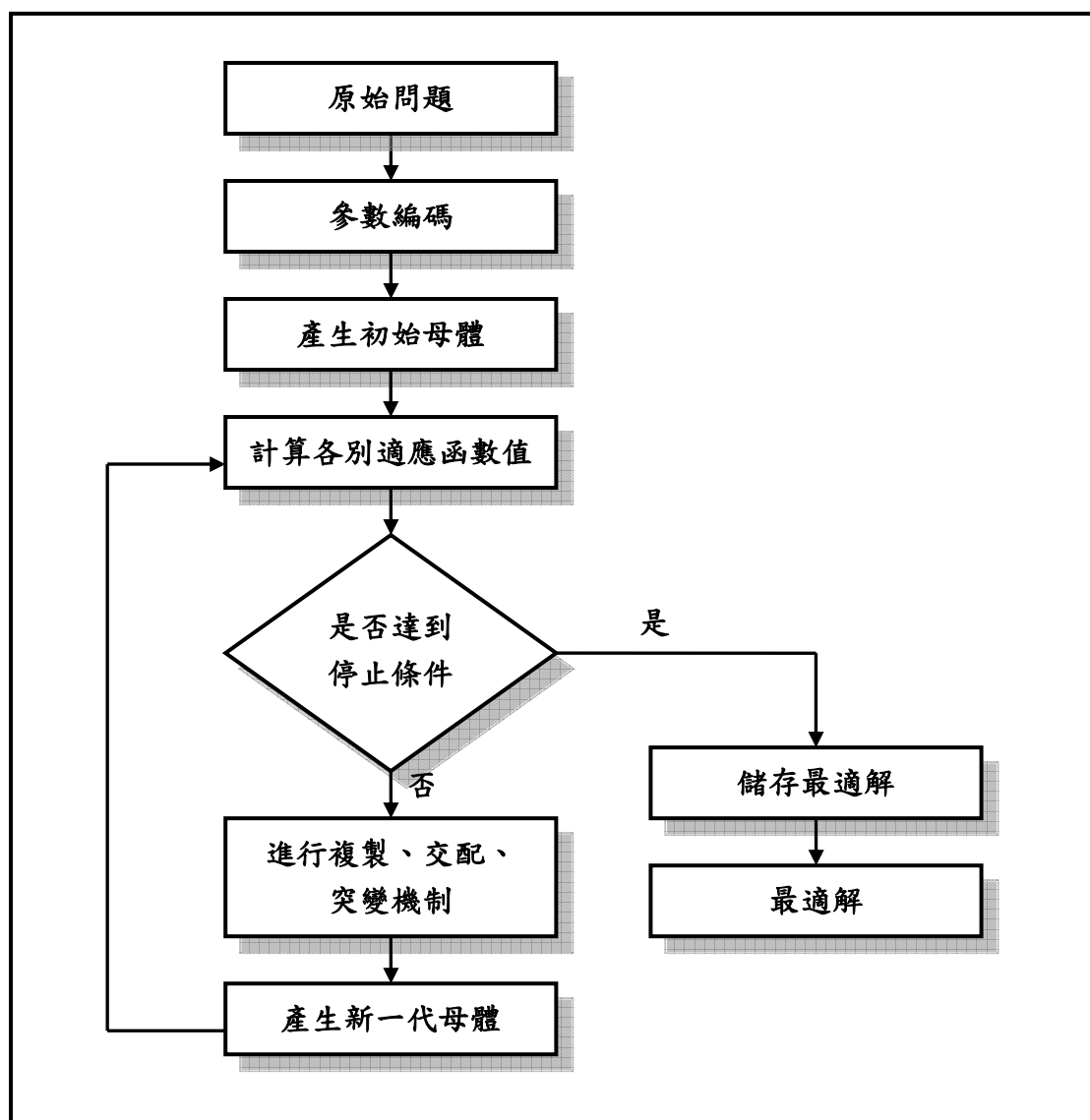


圖 2.3 一般遺傳演算法運作流程

Michalewicz (1994) 整理出運作遺傳演算法的五個基礎步驟：

1、以基因型態表示問題的特徵或解答：遺傳演算法對問題的代表方式是將

不同的問題特徵（或變數）分別以一個或一組基因表示，其中一基因是二元（binary）數字，而如同一個體是由數個基因所構成，問題的解答也就是由不同的問題特徵所組成。

- 2、創造任意數目的初始解答：在遺傳演算法開始運作之前，需先產生一些初始的解答做為初始的狀態，亦即讓電腦創造一些「數位個體」，形成原始族群（initial population）再進行演算搜尋。至於產生這些個體的方式，分為隨機或是特定的方式，Forgaty（1989）則提出產生初始狀態的方法會影響演算法的搜尋績效。
- 3、評估功能的建立-適應性函數（fitness function）的設計：若一個體的適應能力愈高，代表此一個體在環境中存活下來的機會較高，則愈有可能繁殖下一代。相對於適應函數的是在解決問題時之目標函數，藉由目標函數來評判解答接近預設目標的程度，若目標函數值愈高則代表此解答愈接近目標，也愈有機會讓遺傳演算法進一步搜尋到更好的解答個體。
- 4、使用基因運算子（genetic operator）產生子代：最常見的基因運算子有以下三個。
 - (i)複製（reproduction）：此運算子的功能決定哪些個體可以存活至下一代，而根據「適者生存」的原則，適應函數值高的個體應具有較高的機率被選中複製而存留至下一族群。
 - (ii)交配（crossover）：此運算子的功能是透過交換個體間的基因，以重組個體的基因組合，來擴展搜尋空間。Murat（1994）以 10 種不同的交配方式對於流程式排程問題進行電腦模擬測試，以決定不同交換方式的績效優劣。
 - (iii)突變（mutation）：此運算子的功能是藉由隨機改變個體內的基因，引入新的個體型式，增加新的搜尋空間。突變的發生是隨機的，以使得在求解的過程中能夠搜尋新的領域，避免掉入局部最佳解（local optimal）。Murata（1994）以 4 種不同的突變方式來進行試驗。
- 5、參數的設定：在遺傳演算法的運算過程中，有許多的參數必須事先設定，如交配機率、突變機率、個體數及族群數等。參數的設定會影響搜尋的績效，Forgaty（1989）曾對不同的突變機率與遺傳演算的績效進行評估，在特定的原始族群中，不同的突變機率可以改善搜尋績效。

2.4.1 編碼

在 Cheng (1997) 的研究報告中，曾調查使用遺傳演算法求解 JSP 時，常見的表示方法。說明如下：

1. Operation-based representation

此類型的編碼方式是將一個排程編碼成作業的順序，而染色體中的每一個基因代表一作業。亦即以一個 m 部機器與 n 個訂單的 Job shop 排程問題 (Job-shop Scheduling Problem, JSP) 而言，它的每一染色體包含有 $m \times n$ 個基因。但是，由於訂單本身的作業有一定的加工先後順序限制，經過交配後所產生的基因順序不能保證能獲得一可行的排程，所以這種編碼方式必須加入修正調整，研究中的修正方式是將同一訂單中的作業在染色體中皆以同一個代號表示。

在這個表示法中，如果問題大小是 n 個訂單與 m 部機器，染色體會含有 $n \times m$ 個基因，而染色體中每個訂單只會出現 m 次，每個基因是表示工作內的相對步驟，如此一來使得每一染色體所產生的排列皆是可行解。以一 3×3 的問題為例，如果染色體中的基因排列是 $[2\ 1\ 1\ 1\ 2\ 3\ 2\ 3\ 3]$ ，經過解碼轉換後為 $[O_{21}\ O_{11}\ O_{12}\ O_{13}\ O_{22}\ O_{31}\ O_{23}\ O_{32}\ O_{33}]$ ， O_{ij} 表示第 i 張訂單的第 j 個作業，最後在排程時依照作業加工順序來進行排程規劃。

2. Job-based representation

此類型只有將訂單做編碼，得到一個訂單的優先順序，排程時是先安排訂單優先次序為 1 的訂單之所有作業，之後，依訂單的優先順序完成整個排程。此法的原則是先決定訂單優先順序，再按其作業程序來安排所有作業。但是，這種方式與 shifting bottleneck heuristic 有類似的假設：當每一機器的排程是最佳化的時候整個排程即是最佳解，所以這種編碼方式並不能保證可以包含最佳解於其解答空間之內。

3. Preference list-based representation

以一個 m 部機器與 n 個訂單的 JSP 而言，此法的表達方式：每一染色體由 m 個子染色體 (subchromosome) 所構成，每一子染色體是一個長度為 n 的符號字串，每一個符號代表一個機器的相關作業。子染色體並不是該機器上作業的加工順序，而是該機器的喜好表列 (preference list)。整個排程是先找出每部機器的第一喜好作業，再依據每一訂單作業順序限制，判斷機器的第一喜好作業是否可以排入排程之中，若不能則保留。否則找

出機器的下一喜好作業，持續至完成所有作業。這種方法只能排出非延遲 (non-delay) 的排程，可能無法將最佳解含入其中。

4. Job pair relation-based representation

此法是利用一個二元矩陣來，該矩陣是表示二個訂單在機器上的先後關係。其二元的關係定義如下：

$X_{ijm} = 1$ ；如果在機器 m 上，訂單 i 處理時間比訂單 j 早。
 0 ；其它。

因此一個 3×3 的問題染色體如果是 $[0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0]$ 其二元矩陣如下：

訂單(1,2)在機器(m_1, m_2, m_3): $X_{121} \quad X_{122} \quad X_{123} = 0 \quad 1 \quad 0$

訂單(1,3)在機器(m_1, m_2, m_3): $X_{131} \quad X_{132} \quad X_{133} = 1 \quad 0 \quad 1$

訂單(2,3)在機器(m_1, m_2, m_3): $X_{231} \quad X_{232} \quad X_{233} = 1 \quad 1 \quad 0$

然後再依照這個二元矩陣完成整個排程。這個方法最大的缺點是隨著問題變大，會產生愈多不合理的染色體，必須加入其它的調整法則來修正。

5. Priority rule-based representation

這種方法中每一染色體代表派工法則的順序，基因演算法是用來尋求出一個較好的派工法則之順序。以一個 m 部機器與 n 個訂單的 JSP 為例，一個染色體是一個 $n \times m$ 的字串 $(p_1, p_2, \dots, p_{nm})$ ， p_i 表示第 i 個循環所使用的派工法則，整個步驟如下：首先列出每個訂單的第一個作業做為可選擇作業，選取加工時間最少的作業，若加工此作業的機器還需要加工這循環中的其它作業，則使用該循環的派工法則決定何者應被選取，進入下一循環。下一循環開始時須將上一循環中加工被選取的訂單之下一作業納入本循環的可選擇作業。每一循環選取一個作業，直到完成整個排程。使用 Priority rule-based 編碼方式所得的排程解，其缺點是品質很不穩定。

6. Disjunctive graph-based representation

這個方法也可以視為是 Job pair relation-based representation 的一種。排程的問題可以利用分支圖 (如圖 2.4) 來表達， $G = (N, A, E)$: N 代表節點來表示所有的作業， A 代表用來連結同一訂單相連的作業 (實線部分)， E 用來連結同一機器上的作業 (虛線部分)。圖中箭頭方向代表作業之間的優先次序， N 與 A 都是確定的，而為了確保機器上的作業不會有加工順序衝突的現象發生，每個機器的作業節點不可形成一循環狀 (acyclic)，用 e_{ij}

來表示 E 中的每一個箭頭。

$e_{ij}=1$ ，當箭頭的方向是由節點 i 至節點 j，亦即先加工作業 i 再加工作業 j。

$e_{ij}=0$ ，當箭頭的方向是由節點 j 至節點 i，亦即先加工作業 j 再加工作業 i。

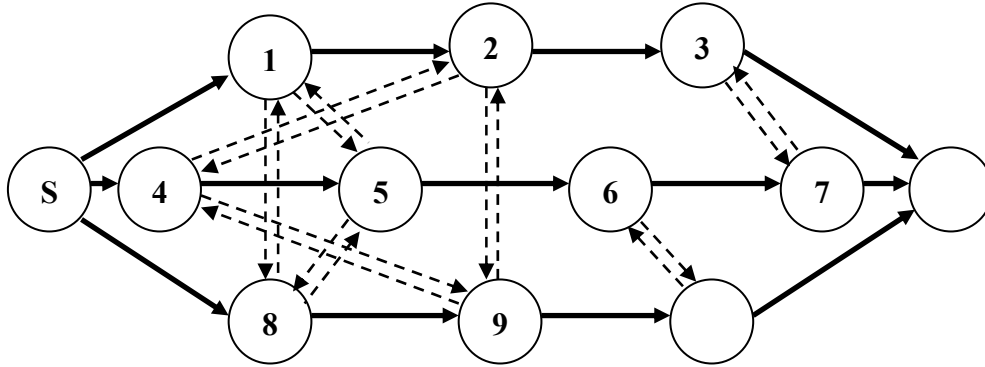


圖 2.4 分支圖

7. Random key representation

在 Random key representation 中，所有的解答被編碼為一串隨機數 (random key)，用這些數字作為排序的依據。如果問題是 n 個訂單和 m 個機器，每個基因 (隨機數) 被視為兩個部分，一個是整數部分，其集合為 $\{1, 2, \dots, m\}$ (代表在哪一個機器執行，因此數字 1 至 m 的出現次數一定是 n 個)，另一部份是介於 0~1 的小數，在解碼的時候要將同機器數字來做比較，後面小數部分的大小代表作業的優先次序。假設在一 3×3 的問題中有一染色體如下：

[1.34 1.09 1.88 2.66 2.91 2.01 3.23 3.21 3.44]

在機器 1 上的小數部分按照小到大排列順序為 [2, 1, 3]，而在機器 2 上的順序為 [3, 1, 2]，機器 3 上的順序為 [2, 1, 3]，因此將這個染色體轉換成實際機器順序下：

[O_{21} O_{11} O_{31} O_{32} O_{12} O_{22} O_{23} O_{13} O_{33}]

使用這種方式來轉換的排程很容易違反訂單加工途程的限制，必須搭配一些特別的解碼方式。另外，在產生初始族群、交配、突變時所產生的染色體整數部分必須符合數目的限制。

上述之表達法可以分為二個類型：直接式與間接式。所謂的直接式是將一個排程解編碼成一染色體，在透過基因演算法的運算，得到一較佳的

排程。而間接式的染色體則是以優先順序法則做為編碼依據，如 Priority rule-based representation 中，其是要透過基因演算法找出較佳的派工法則之順序。

2.4.2 運算子

使用遺傳演算法求解時，依據問題特性的不同，必須設計適當運算子，讓整個演算過程能順利的進行。在解決排序問題時，一個染色體是由許多“不重覆”的數字基因所構成的字串。在這個限制之下，如果以傳統簡易遺傳演算法中所提的運算子來搜尋，很有可能會產生一個個體中有兩個相同的數字基因。為了避免發生重覆的情況，必須要設計新的運算子，來產生合理的子代。在 Fang (1994) 提及關於遺傳演算法中交配運算子可以分為以下幾種類型：

1. 單點交配 (one-point crossover)：根據育種選擇策略選擇的二個母體，隨機產生二個切點進行交配。

父代 1：[7 3|7 6 1 3] → 子代 1：[7 3|4 5 2 2]

父代 2：[1 7|4 5 2 2] → 子代 2：[1 7|7 6 1 3]

子代 1 中第一部份的基因是繼承父代 1 基因而來，第二部分繼承父代 2，子代 2 則是繼承父代 1 第二部份的基因與父代 2 第一部份的基因。

2. 雙點交配 (two-point crossover)：隨機產生二個切點，交換父代彼此的基因。

父代 1：[7 3|7 6|1 3] → 子代 1：[7 3|4 5|1 3]

父代 2：[1 7|4 5|2 2] → 子代 2：[1 7|7 6|2 2]

3. N 點交配 (N-point crossover)：隨機產生 3 至 n 個切點數，決定父代交換奇數或是偶數部分的基因。以下為例：產生 3 個切點 (1、2、4) 交換奇數部分的基因，產生二子代。

父代 1：[7|3|7 6|1 3] → 子代 1：[7|7|7 6|2 2]

父代 2：[1|7|4 5|2 2] → 子代 2：[1|3|4 5|1 3]

4. 均於交配 (uniform crossover)：每一基因的位置皆產生一個 0~1 的亂數，假如亂數 > 0.5，則該位置的基因不交換。舉列而言，在下列中的中產生的亂數依序為 0.2, 0.7, 0.9, 0.4, 0.6, 0.1，產生的子代如下：

父代 1: [7 3 7 6 1 3] → 子代 1: [7 7 4 6 2 3]

父代 2: [1 7 4 5 2 2] → 子代 2: [1 3 7 5 1 2]

在吳信儀 (1996) 的研究中，將一些常用在排序問題的運算子，整理如下。

1. PMX (Partially Matched Crossover)

(1) 隨機產生兩個切點。

個體 A: [9 8 4 | 5 7 6 | 1 3 2]

個體 B: [8 7 1 | 2 3 6 | 9 5 4]

(2) 將個體 A 與個體 B 在兩切點中的基因互調。

個體 A: [9 8 4 | 2 3 6 | 1 3 2]

個體 B: [8 7 1 | 5 7 6 | 9 5 4]

(3) 將個體 A 位於切點之外重覆基因與個體 B 位於切點之外重覆基因互調。

個體 A: [9 8 4 | 2 3 6 | 1 7 5]

個體 B: [8 3 1 | 5 7 6 | 9 2 4]

2. LOX (Linear Order Crossover)

(1) 隨機產生兩個切點。

個體 A: [9 8 4 | 5 7 6 | 1 3 2]

個體 B: [8 7 1 | 2 3 6 | 9 5 4]

(2) 將個體 A 切點中的所有位元，在個體 B 中以 * 代替；B 者亦然。

個體 A: [9 8 4 | 5 7 * | 1 * *]

個體 B: [8 * 1 | 2 3 * | 9 * 4]

(3) 將 * 往中間移動，使得兩個切點中的位元皆為 *。

個體 A: [9 8 4 | * * * | 5 7 1]

個體 B: [8 1 2 | * * * | 3 9 4]

(4) 將原本個體 A、B 切點中的位元互調。

個體 A: [9 8 4 | 2 3 6 | 5 7 1]

個體 B: [8 1 2 | 5 6 7 | 3 9 4]

3. SX (Simple Crossover)

(1)隨機產生 1 個切點。

個體 A : [9 8 4 | 5 7 6 1 3 2]

個體 B : [8 7 1 | 2 3 6 9 5 4]

(2)保留切點左邊的基因，切點右邊的位元以其在另一個體的順序填入。

個體 A : [9 8 4 | 7 1 2 3 6 5]

個體 B : [8 7 1 | 9 4 5 6 3 2]

4. RX (Random Crossover)

(1)隨機產生兩個切點。

個體 A : [9 8 4 | 5 7 6 | 1 3 2]

個體 B : [8 7 1 | 2 3 6 | 9 5 4]

(2)切點兩旁的基因保留不變，兩切點中的基因以隨機產生。

個體 A : [9 8 4 | 6 5 7 | 1 3 2]

個體 B : [8 7 1 | 3 6 2 | 9 5 4]

5. CX (Cycle Crossover)

(1)在個體 A 中任選一個基因，假設選到 9，其相對在個體 B 為 1，將其標示起來。

個體 A : [9 8 2 1 7 4 5 6 3]

個體 B : [1 2 3 4 5 6 7 8 9]

(2)個體 A 中基因為 1 的位元在個體 B 中是 4，將其標示起來。

個體 A : [9 8 2 1 7 4 5 6 3]

個體 B : [1 2 3 4 5 6 7 8 9]

(3)個體 A 中基因為 4 的位元在個體 B 中是 6，將其標示起來。

個體 A : [9 8 2 1 7 4 5 6 3]

個體 B : [1 2 3 4 5 6 7 8 9]

(4)個體 A 中基因為 6 的位元在個體 B 中是 8，將其標示起來。

個體 A : [9 8 2 1 7 4 5 6 3]

個體 B : [1 2 3 4 5 6 7 8 9]

(5)重覆以上的步驟，直到最後的標示回到 9。

個體 A : [9 8 2 1 7 4 5 6 3]

個體 B : [1 2 3 4 5 6 7 8 9]

(6)將個體 A、B 中沒有被標示的基因互換。

個體 A : [9 8 2 1 5 4 5 6 3]

個體 B : [1 2 3 4 7 6 7 8 9]

6. OM (Order-based Mutation)

任選兩個基因，將其互調即可。

[9 8 4 5 7 6 1 3 2] → [9 8 4 1 7 6 5 3 2]

7. PM (Position based Mutation)

任選兩個基因，假設為 5、1，將 7、6、1 往前移，再將 5 填入原本基因 7 的位置即可。

[9 8 4 5 7 6 1 3 2] → [9 8 4 7 6 1 5 3 2]

2.4.3 遺傳演算法於排程問題之應用

排程問題在生產管理是相當重要的研究領域。大都利用啟發式演算法解決問題，達到最適或次佳解。Petty et al. (1987) 提出分散式遺傳演算法，將一族群分割成較小的族群，再分別進行 GA 運作，以提高求解效率；Kim et al. (1994) 以 GA 為基礎發展一啟發式搜尋技術解決 Job shop 的排程問題；Pinedo et al. (1995) 指出，遺傳演算法可以被應用在結構未知的問題上；Wellman et al. (1995) 提到遺傳演算法已經成功地應用在最佳化求解的問題，如排程、運輸問題、旅行者問題等。Kumar 及 Srinivasan (1996) 依據不同情境下，並且從七種不同法則中採用不同派工法則進行排程；Murata et al. (1996) 以製距最小為目標，評估 GA 分別結合局部搜尋法、模擬退火法在 Flow Shop 排程問題上的表現；Norman 及 Bean (1999) 利用隨機法則，發展出一套方法，並應用在汽車工業。Liaw (2000) 利用混合式遺傳演算法(HGA)，即傳統遺傳演算法結合禁忌搜尋法，求解開放式 (open shop) 排程；Park et al. (2003) 分別利用單一遺傳演算法以及平行式遺傳演算法，並與傳統遺傳演算法比較。

基本上遺傳演算法與其他搜尋方式比較起來，能夠以較短的時間搜尋到不錯的解答，對問題變數的多寡影響相對的比較小，有其他搜尋演算法

所缺乏的優點。但是在實際應用時，遺傳演算法在不同的問題上仍有一些缺點，尚待克服。本研究將遺傳演算法的特徵及其優缺點整理如表 2.6，而根據所整理出的缺點，本研究擬以禁忌搜尋法來強化原先遺傳演算法不足之處。

表 2.6 遺傳演算法之特性與優缺點歸納表

<p>特 性</p>	<p>1. 決策變數以染色體形式表示 將決策變數以染色體方式表示之，因此求解問題不受問題目標函數的連續性或可微分性之限制，有別於傳統數學規劃受限之處。</p> <p>2. 適應性函數評估 遺傳演算法的運作過程只使用適應函數的資訊而不需要其它輔助的資訊，如：微分。</p> <p>3. 天擇、演化機制 在每代族群中，弱勢族群易被取代(天擇)，族群個體須不斷提升適合度(演化)，提高存活率，經此演化過程，各族群探索求解空間，形成多方向探索，而非傳統數理方式的既定方向探索，如梯度(gradient direction)搜尋。</p>
<p>優 點</p>	<p>1. 多點同步搜尋 遺傳演算法同時考慮搜尋空間上多個點而不是單一個點，從族群的方向探索求解空間，因此可以較快地獲得整體最佳解(global optimum)，同時也可以避免陷入區域最佳解(local optimum)的機會，此項特性乃是基因演算法的最大優點。</p> <p>2. 使用適應函數 遺傳演算法的運作過程只使用適應函數的資訊而不需要其它輔助的資訊，且用編碼方式來求解，因此可以使用各種型態的適應函數，並可節省計算資源避免繁複的數學運算。</p> <p>3. 機率式的搜尋 遺傳演算法是使用隨機規則(stochastic)的方式去引導搜尋方向，而不是用明確(deterministic)的規則，因此較能符合各種不同類型的最佳化問題。</p>
<p>缺 點</p>	<p>1. 運算子的設計問題 根據不同的問題，需要設計不同功能的運算子，以提高搜尋效率。若單純使用簡易基因演算法的運算子，搜尋速度會受到影響。</p> <p>2. 缺乏數學理論模型 遺傳演算法雖成功應用到許多領域，尤其是最佳化問題，但其本身尚缺數學理論模型來驗證其數理的可行性。</p> <p>3. 重複搜尋的問題。 因為遺傳演算法並沒有記憶功能，且其運作過程只與適應函數相關，因此往往在搜尋的過程中，重複搜尋到相同的點，增加系統搜尋的時間。</p>

資料來源：本研究整理

2.5 禁忌搜尋法

2.5.1 禁忌搜尋法簡介

禁忌搜尋法為 Glover 於 1977 年所提出，是一種可以用來處理組合爆炸問題的演算法。Glover (1989) 指出禁忌搜尋法具有三個主要特性：(1) 使用彈性記憶結構 (flexible memory structure)，較其它固定記憶 (如分枝界限法) 或無記憶結構 (如模擬退火法) 更能充分利用歷史搜尋資訊。(2) 應用記憶彈性結構的一個關聯控制機制，透過對條件的限制與釋放，來控制搜尋過程。(3) 從短期到長期，合併不同時間間距的記憶功能，以實行強化與多變的搜尋策略。

Glover (1990) 中，指出禁忌搜尋法主要的組成要素有：

1. 起始解 (starting solution)：由最原始的可行解中，產生一啟發解作為搜尋的起始點。
2. 鄰域解 (neighbor solution)：不屬於記憶架構內的紀錄，並符合問題限制之可行解。而鄰域解的移動取法有兩種交換式變動 (swap move) 與插入式變動 (insert move)，因為一個插入式變動是由兩次交換式變動所造成，所以插入式變動對目前的解所造成的變動程度較交換式高，但必須視問題的型態作最佳的選取。
3. 移動路徑 (move)：在目前的鄰域解中，選取評估值最高者作為改善之移動路徑，若此鄰域解已在禁忌串列中，且又不符合凌駕準則選取次佳之鄰域解作為改善移動的路徑；若此鄰域解符合凌駕準則，則其不受禁忌串列的限制。
4. 記憶架構 (memory structure)：記憶結構亦可稱為禁忌串列 (tabu list)，記憶結構又可分為「短期記憶」與「長期記憶」，短期的架構只記錄該次搜尋前的數次記憶，長期記憶則是記錄從開始運作以來所有的搜尋過程。記憶期數的大小必須依照問題的特性和複雜度決定，Glover (1990) 建議可使用魔術數字 7 作為禁忌串列的尺寸。
5. 凌駕準則 (Aspiration criterion)：考慮一鄰域解，若此一移動所得之鄰域解之評估值比截至目前為止搜尋過的最佳解還要好，若其恰巧在禁忌串列的限制中，此時應破除禁忌串列的限制，接受此一移動。
6. 停止條件 (stop criterion)：終止搜尋進行的條件，一般最常是以一預設

最大循環次數，當到搜尋循環達此一次數及終止搜尋。

在長期階段使用強化性 (intensification) 與多樣性 (diversification) 將搜尋帶入新的區域以求出更佳的解。所謂的強化性與多樣性乃是指當搜尋終止後，由選取最佳解或是最後一次的可行解，清除所有禁忌串列重新開始進行搜尋，其目的是為了增加搜尋時跳脫區域最佳解的機會。短期記憶的功能是避免下一次移動時，發生重複選取的現象，而長期記憶是記錄所有移動決策及其發生過的次數，計算其頻率的記憶 (frequency base memory)，使其能提供資訊來補足短期記憶的不足。

2.5.2 禁忌搜尋法在排程問題之應用

Wesley (1995) 以 14 種派工法則求出起始解，再利用禁忌搜尋法求解 Job shop 排程問題。Adenso (1996) 利用禁忌搜尋法結合模擬退火法求解流程式排程問題，利用模擬退火法中接受較差解的機制，避免求解過程中掉入局部最佳解的情形。Liaw (2000) 的研究中利用禁忌搜尋法結合遺傳演算法求解流程式排程問題，以最小化製距為目標，然後利用 Tabu 搜尋法對每一個體解其關鍵路徑 (critical path) 進行鄰域搜尋修正改善。Scrichet al. (2004) 在彈性零工式生產排程問題上，以最小延遲時間為目標，先以派工法則先求出一初始解，再以禁忌搜尋法進行鄰域搜尋，並驗證出此求解機制具有高效率的求解品質。Shi 及 Pan (2005) 以禁忌搜尋法為基礎，針對零工式生產問題設計出更具效率的演算方法。Saidi (2007) 利用禁忌搜尋法在彈性零工式生產排程問題中，以最短製距為績效考量，成功求解出最佳的作業排序與作業加工機器指派。

第三章 多目標與限制驅導零工式生產排程架構

本研究主要於零工式生產環境下，建構生產預先規劃排程的方法與穩健排程的求解。第一階段，排程規劃同步考量質性與量性瓶頸限制並以多目標限制適應函數為評估績效，以限制驅導技術結合混合式遺傳演算法作為排程演算法，求解預先規劃排程。第二階段為穩健排程階段，加入中斷測試事件，並觀測系統瓶頸資源漂移情況，以右移法調整受到波動影響的作業，得到穩健排程解，讓規劃結果增加排程彈性與穩健性，提升排程規劃對於中斷事件的容忍程度。多目標限制驅導零工式生產排程規劃架構如圖 3.1：

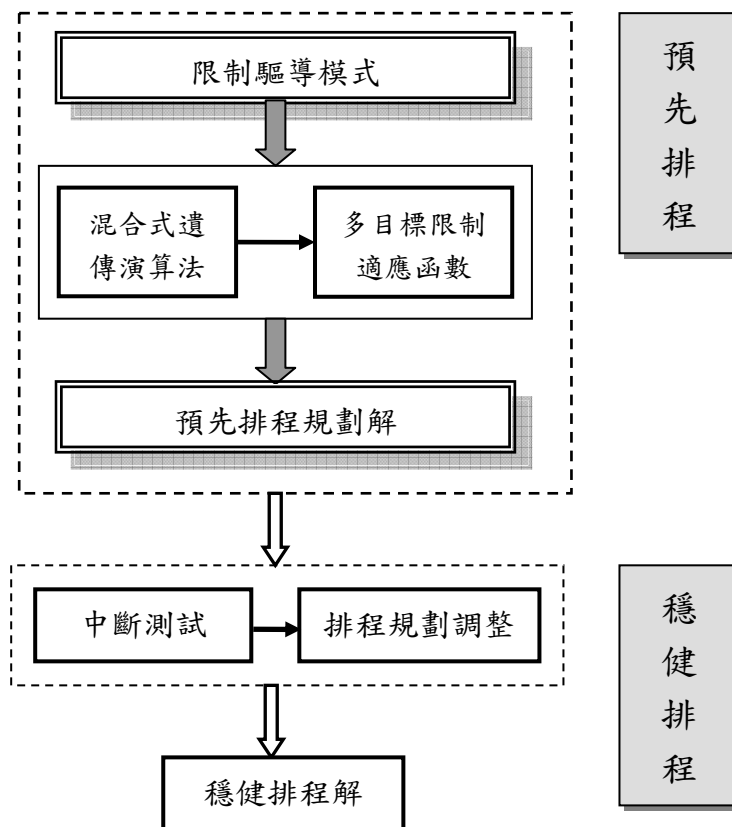


圖 3.1 多目標限制驅導零工式生產排程規劃架構圖

本研究各項詳細流程與方法於本章各節說明。3.1 節介紹本研究的基本環境，說明多瓶頸資源環境下，考量量性瓶頸資源與質性瓶頸資源，建構多目標限制驅導零工式生產排程之架構；3.2 節說明以限制驅導排程技術為基礎，結合遺傳式演算法、禁忌搜尋法之排程技術方法、及其求解步驟流

程；3.3 節說明加入測試性干擾機制，如：機台故障、物料短缺與緊急插單等中斷事件，測試排程結果對於中斷事件的容忍程度；另一方面檢測生產系統瓶頸漂移程度，獲得更具穩健性的預先排程解。

3.1 零工式生產、多瓶頸資源與多目標模式

本節於 3.1.1 小節介紹本研究的多目標零工式生產模式架構，包含質性與量性瓶頸資源考量，於 3.1.2 小節說明質性瓶頸資源架構；於 3.1.3 小節說明量性瓶頸資源架構。

實務排程規劃追求的目標並非單一績效最佳化，而是期望多項指標皆有不錯的績效。企業營運追求多項目標，受限於許多資源，如：如市場需求、訂單交期、機器產能負荷等諸多資源的限制，稱之：多限制資源(Multiple constrained resources)，而不同的限制資源，交互影響企業追求更高績效。這些限制資源可分為質性限制資源與量性限制資源，其中最關鍵資源稱之瓶頸資源(Bottleneck resources)。一般所考慮的量性限制資源有：機台產能負荷、人力資源等，其為實體與非實體限制，其中多數研究將產能負荷最高之工作站設定為量性瓶頸。現實環境中還存有質性限制資源，例如：市場需求、潛在獲利、顧客的歷史交易紀錄等，則多為非實體限制(政策限制)，而多數的排程研究中，忽略此質性限制資源所在，本研究將其納入考量，並確認質性瓶頸。排程規劃受到多限制資源影響，如何將多種瓶頸資源進行妥善分配，讓受限資源發揮到最大，並從多項衝突目標中，折衷得求取一配適排程解，則需仰賴限制驅導排程技術 DBR。

本研究於排程目標上，考量量性與質性瓶頸資源。將產能負荷最高之工作站設定為量性瓶頸(Quantitative Bottleneck)，而受到量性瓶頸所影響的績效：製距、瓶頸工作站使用率、交期滿足度、非受限資源配合度，利用層級分析法(Antalytical Hierarchy Process, AHP)算出相關權重，制訂本研究之量性多目標函數。再以質性限制資源(非實體限制)評估各訂單間相對重要性，同樣以 AHP 法則算出相關權重，決定訂單優先順序，所制定的理想訂單規畫順序則設定為質性瓶頸(Qualitative Bottleneck)，並依此順序訂出違背此排程順序的質性限制懲罰函數。依據此兩方面函數決定本研究之多目標限制適應函數。本研究之多目標限制適應函數架構如圖 3.2：

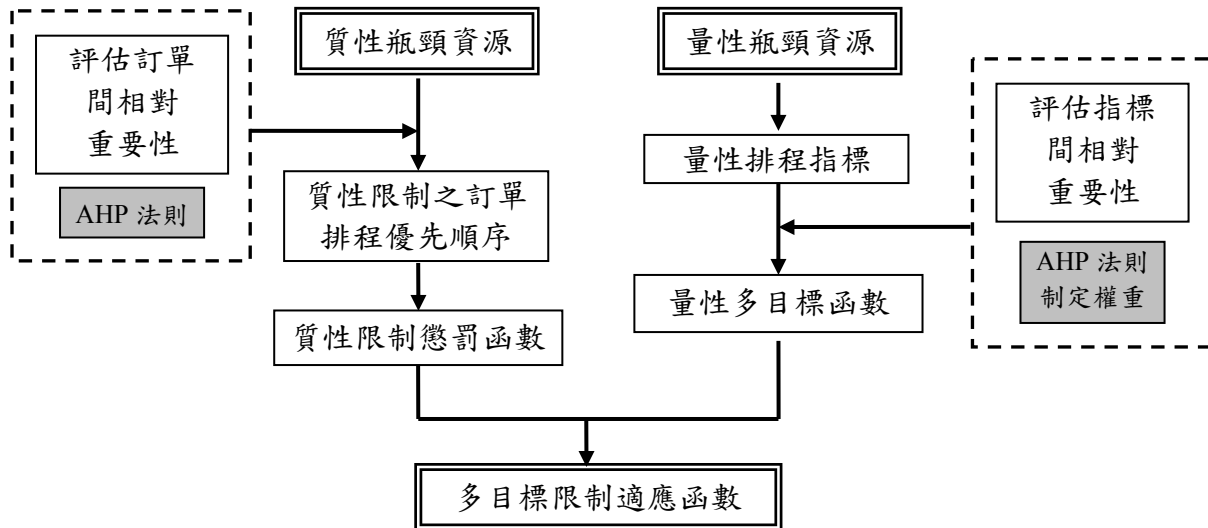


圖 3.2 多目標限制適應函數架構圖

3.1.1 多瓶頸資源與零工式生產架構

零工式生產 (Job shop) 模式是指每張訂單皆有多個加工途程，且每張訂單的製造途程、加工時間、交期時間不同，因此各訂單作業的加工先後順序與加工時程決定一個排程計劃的績效優劣。而零工式排程問題受到現場產能設備限制、物料限制、訂單生產優序限制、人力資源限制等，在進行生產排程規劃時，規劃者得由諸多限制考量下，追求多項目標績效，制定一合適的排程規劃。

本研究探討零工式生產排程問題，假設是有 n 張不同訂單與 m 個不同功能的工作中心所組成，每個工作中心僅有一台機器。本研究零工式排程問題的基本限制如下：

- 1、每張訂單有其不同的作業途程順序及工作中心站，但其操作順序與工作站則需是固定且明確的；
- 2、每台機器在同一個時間內，最多只能執行一個加工作業；
- 3、每個操作進行時必須是連續的；
- 4、同張訂單在同一個時間內，只能有一個操作步驟在機器上被執行；
- 5、當時間開始時，所有訂單皆已準備就緒。

由於排程問題須於有限資源下，進行資源分配最佳化。考量各機器的產能負荷分配、不同訂單之製造優序、顧客訂單交期、企業獲利考量等限

制。本研究針對多種受限資源探討，包含量性限制資源與質性限制資源。其中量性限制資源為實體限制，以產能負荷最重的工作中心站為量性瓶頸 (Quantitative Bottleneck)；質性限制資源為非實體限制，以不同訂單對企業營運績效的影響，利用 AHP 法則評定出對企業營運績效的訂單排序，該訂單生產優先處理政策的質性限制，稱之質性瓶頸 (Qualitative Bottleneck)。

本研究同步考量質性與量性瓶頸資源，將其轉化為不同的績效目標，利用 AHP 法則比較相關績效目標的重要性，並設定權重，將問題轉化為多目標函數，建構多目標排程模式。多目標排程模式建構流程與方法於後續小結說明。3.1.2 質性限制懲罰函數模式架構；3.1.3 量性多目標函數模式架構。

3.1.2 質性瓶頸－懲罰函數模式架構

本研究中所指之質性瓶頸為非實體限制，而這類型的限制，稱之非實體產能限制資源或稱政策限制資源 (Policy constrained resources)，如組織制度文化、市場因素考量等，屬於不能明確加以量化或是不需精確數量化的限制資源。以市場因素而言，該張訂單產品在市場中的定位將是主要關鍵，假若該產品的市場競爭者多，則產品完工的時效性顯的重要，因為其能使產品有更好的競爭優勢。換句話說，此訂單在市場因素之下應該擁有較優先的製造順序，但是此限制並不能直接加以量化；訂單所屬的顧客也是屬於無法明確量化的質性限制，對公司而言，不同顧客與其重要性必然會有程度上差異，可能是因為彼此間有策略聯盟或是其它策略上的因素所造成。同樣地，每個顧客未來潛在的訂單利潤，也應納入此階段中應考量的限制考量之一。

本研究中的考量的質性限制資源包含：訂單利潤、顧客以往的交易記錄、市場因素、潛在訂單利潤。排程規劃人員在評估訂單製造的優先順序時，根據這些資源限制做主觀性的比較判斷。本研究中將利用 AHP 法來評估不同訂單在各個質性限制下對企業營運績效的影響，進而計算出訂單生產優先順序的質性限制，稱之質性瓶頸，依此順序訂出排程時違背此順序的懲罰函數，做為多目標排程模式中的質性限制懲罰函數。以下說明研究中 AHP 法的評估計算。

層級分析法(AHP)的評估計算

關於如何設定各因素權重值的方法有很多，本研究中將利用階層式分析程序法來訂定權重值。AHP 法是由 Saaty (1977) 所提出，是用來評估方案或是因素間相對權重，其原理是採用配對比較法來設定因素間的相對權重，將評估因素的重要性以配對的方式兩兩比較，並給予比較之分數，再加以計算各因素的權重值，最後利用 C.R. 值來衡量成對比較矩陣的一致性，主要是採用一致性指標 (Consistency Index, C.I.) 及一致性比率 (Consistency Ratio, C.R.) 為依歸。C.I. 的計算方式如式 3.1：

$$C.I. = \frac{\lambda_{\max} - n}{n - 1} \begin{cases} = 0, & \text{表示判斷一致} \\ > 0.1, & \text{表示判斷有誤差不連貫} \\ \leq 0.1, & \text{表示判斷雖不一致, 但仍為可接受的偏誤} \end{cases} \quad \text{式 3.1}$$

表 3.1 隨機指標表

階數	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R.I.	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.48	1.56	1.57	1.59

資料來源：Saaty (1977)

另外，Saaty (1977) 提出隨機指標 (Random Index, R.I.) 如表 3.1 所示，指出在不同決策階數下，產生不同的 C.I. 值。在相同階數的矩陣下，C.I. 與 R.I. 的比率，稱為一致性比率 (C.R.)。C.R. = C.I./R.I.，若 C.R. ≤ 0.1，則一致性程度視為滿意。但實務上若 C.R. 小於 0.15 亦為可接受之範圍。

AHP 法最大的優點是在於當人們面臨多個評估因素情形下，提供決策者一客觀的方法來決定各因素的權重。同時，AHP 也可以應用於多個方案的選擇評估。AHP 法中的評估尺度如表 3.2。

表 3.2 AHP 評估尺度

評估尺度	定義
1	同等重要
3	稍重要
5	重要
7	極重要
9	絕對重要
2、4、6、8	相鄰尺度的中間值
以上數值之倒數	如上定義之相對不重要程度

舉例如下：假設有 5 張訂單，而需考量因素為 F1、F2、F3、F4。

步驟一：將所有因素配對以表中的評估尺度來做比較。(若規劃人員評估 F1 較 F2 稍不重要，評估尺度為 1/3，相反的 F2 較 F1 稍重要，評估尺度為 3)。

步驟二：計算此矩陣的特徵值 (eigenvalue)，取其最大特徵值 (λ_{max})。

步驟三：計算 λ_{max} 對應的特徵向量 (eigenvector)。

步驟四：將此特徵向量正規化 (normalize)，此正規化特徵向量即為各個因素相對的權重值 (W_i)，並進行一致性檢定，如表 3.3。

表 3.3 各因素之權重制訂

因素	F ₁	F ₂	F ₃	F ₄	特徵值 $\lambda_{max}=4.11$	權重 W_i
F ₁	1	1/3	2	7	0.3773	0.24
F ₂	3	1	5	9	0.8976	0.58
F ₃	1/2	1/5	1	5	0.2192	0.14
F ₄	1/7	1/9	1/5	1	0.0628	0.04
C.I.=0.03666 ≤ 0.1 C.R.=0.0407 ≤ 0.1						符合一致性檢定

步驟五：規劃人員在因素 F_i 的考量下，將所有訂單配對比較評估，利用步驟一至步驟四的計算方式，求出一正規化特徵向量，此即為每一張訂單在因素 F_i 下的評估值 (E_i)，如表 3.4。其中 E_{ij} 表示訂單 j 在因素 i 的評估值，與一致性檢定。

表 3.4 各方案之評估值

F_i	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	特徵值 λ_{\max} =5.0938	評估值 E_{ij}
訂單 1	1	5	2	3	3	0.7758	0.4055
訂單 2	1/5	1	1/3	1/2	1/3	0.1277	0.0668
訂單 3	1/2	3	1	3	2	0.4912	0.2568
訂單 4	1/3	2	1/3	1	1/2	0.203	0.1061
訂單 5	1/3	3	1/2	2	1	0.3153	0.1648
C.I.= 0.0233 <= 0.1 C.R.=0.0259 <= 0.1							符合一致性檢定

步驟六：計算出所有的評估值(E_i)。

步驟七：計算各訂單的加權值，並加以排序，如表 3.5。

表 3.5 各方案之加權排序

F_i	F_1	F_2	F_3	F_4	加權值	排序
W_i	0.24	0.58	0.14	0.04		
評估值	E_{1j}	E_{2j}	E_{3j}	E_{4j}		
訂單 1	0.4055	0.30	0.10	0.15	0.2913	1
訂單 2	0.0668	0.20	0.25	0.25	0.1770	3
訂單 3	0.2568	0.20	0.30	0.15	0.2256	2
訂單 4	0.1061	0.15	0.15	0.35	0.1745	4
訂單 5	0.1648	0.15	0.20	0.10	0.1586	5

依加總值再加以排序順序就是所有訂單在質性限制資源考量下的優先順序，此規劃優序為質性瓶頸。本階段利用 AHP 法評估每個訂單在質性限制資源下的重要性，並求得的質性訂單順序，配合限制驅導排程技術結合混合式遺傳演算法所求解出的規劃訂單順序，作為計算質性限制懲罰函數值，將此懲罰函數值轉換為多目標適應函數中的質性瓶頸考量。質性限制懲罰函數詳細計算與整體適應函數的設計於 3.2 節中說明。

3.1.3 量性瓶頸—多目標函數模式架構

關於量性限制資源的考量方面，由於排程規劃人員須從多種受限資源下追求多目標與最佳化的資源配置，一般排程規劃受到量性限制資源如：訂單量、現場可用產能供給、人力資源負荷等影響，往往影響到製距、週

期時間、交期達交率等績效。本研究設定為產能負荷最高之工作站為量性瓶頸，受此瓶頸所影響的到排程績效：製距、瓶頸工作站使用率、交期滿足度、非瓶頸資源配合度，等四個目標，轉換為單一多目標函數，各目標間的權重藉由 AHP 法則來制定。在本研究的排程演算法搜尋求解過程中，不單是利用一項指標來評估一個體解的優劣，相對的是強調如何在多個相衝突的目標中折衷求得一最適解，為了能正確利用經由 AHP 法則制定的權重值，在適應函數的設計中每一項目標評估值都應被正規化後再加權計算。以下說明四個衡量指標。

1. 製距 (makespan) 績效評估

製距的評估方式為第一張訂單之起始作業到最後一張訂單的最後一個作業結束所需的時間。首先由個體解排程結果求出此次排程製距，再將製距評估值正規化，製距評估值正規化的計算方式如式 3.2

$$\frac{\text{MIN } MS}{MS_x} \quad \text{式 (3.2)}$$

其中，MIN MS 為至目前為止，搜尋過程中最小製距； MS_x ：個體解 x 製距。

在搜尋求解過程中，當找到一個體解製距比 MIN MS 小，則 MIN MS 就會被取代，以上述方式來正規化製距評估值會造成同一個體在不同族代的適應函數值不相同的現象發生。舉例來說，在第 n 代中，MIN MS 等於 80 且某一個體的製距為 100，製距評估值正規化後為 $80/100=0.8$ ，而如果在第 $(n+1)$ 代，搜尋到一新個體的製距為 70，若前一個體有被保留至 $(n+1)$ 代的族群中，則此個體在 $(n+1)$ 代中的正規化製距評估值等於 $70/100=0.7$ ，會造成其適應函數值降低的現象。

雖然在演算的過程中，會發生同一個體在不同代中有不同的適應函數值。但是以相對的角度而言，在同一代中的個體適應函數值高者恆高，低者恆低，對於演算法中的機制不會造成明顯的影響。

2. 交期滿足度評估

本研究中將交期定義於模糊集合上，如圖 3.3。個體解的交期滿足度評估方式為在該次排程結果中，先將每張訂單完成時間利用模糊隸屬函數求算出交期滿足度，再將該次排程中所有訂單交期滿足度取平均值即為所

求。公式如下所示：

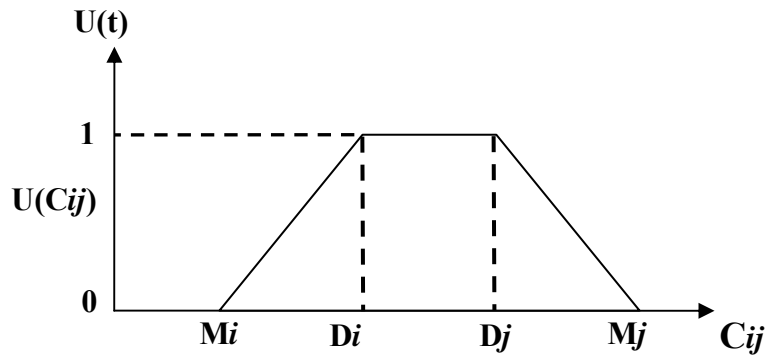


圖 3.3 交期之模糊隸屬函數

$$\left\{ \begin{array}{l} U(C_{ij}) = \frac{C_{ij} - M_i}{D_i - M_i}, \quad M_i \leq C_{ij} \leq D_i \\ U(C_{ij}) = 1, \quad D_i \leq C_{ij} \leq D_j \\ U(C_{ij}) = \frac{M_j - C_{ij}}{M_j - D_i}, \quad D_j \leq C_{ij} \leq M_j \\ U(C_{ij}) = 0, \quad C_{ij} \leq M_i \text{ or } C_{ij} \geq M_j \end{array} \right. \quad \text{式 (3.3)}$$

$$Y_i = \frac{\sum_{j=1}^n U(C_{ij})}{n} \quad \text{式 (3.4)}$$

其中， n ：該次排程之訂單數目

$U(C_{ij})$ ：個體解 i 的第 j 張訂單之交期滿足度

C_{ij} ：個體解 i 的第 j 張訂單完成時間

D_i ：訂單 j 之最早達交時間

D_j ：訂單 j 之最晚達交時間

M_i ：訂單 j 之交期下限

M_j ：訂單 j 之交期上限

Y_i ：個體解 i 之平均交期評估值

3. 瓶頸工作站使用率評估

排程規劃時除了製距、交期的評估之外，機器使用率也是常見的績效衡量指標。在限制理論中，講求的瓶頸資源利用最大化，非瓶頸資源僅需配合瓶頸資源即可，提升非瓶頸資源的利用率對系統產出並無影響，因此瓶頸資源使用率應作為排程績效考量，而不探討非瓶頸資源使用率。本研究定義瓶頸資源為產能負荷最高的工作站，以瓶頸工作站的機器使用率，做為排程績效考量。關於瓶頸機器使用率的評估方式是先個別加總瓶頸工作站上每項作業的加工時間，再除以瓶頸工作站上最後一項作業的完成時間，可得到瓶頸機器使用率，此評估值即為所求。

4. 非瓶頸資源配合度 (Subordinating) 評估

在 Goldratt and Cox (1986) 提出的限制理論中曾提到，執行 TOC 的第三個步驟為系統全力配合 (Subordinate) 步驟二的決定，而本研究所採用的限制驅導排程技術是先將瓶頸作業的排程規劃 (Drum) 制定後，再求解非瓶頸作業的排程，而系統全力配合是指在瓶頸作業開工前，確定非瓶頸作業已如期完工，瓶頸作業可順利開工，如此可讓瓶頸作業持續製造，維持生產系統產出速率。但若是過早完成非瓶頸作業，則會造成製造現場多餘的在製品增加，這對現場營運績效而言，是無益的；另一方面，若是非瓶頸作業完工時間與瓶頸作業開工時間緊密銜接，對排程規劃來說，如此的排程規劃缺乏彈性，一旦非瓶頸作業發生中斷，相對影響到瓶頸作業既定的製造時程，則破壞生產系統產出，瓶頸資源利用率降低，製造營運績效降低。如此可知，過於鬆散與緊密的配合度皆對排程績效有不良影響，因此，本研究將非瓶頸資源配合度定義於模糊集合上，以梯形模糊數表示之，如圖 3.4。個體解的非瓶頸資源配合度評估方式為在該次排程結果中，先算出每張訂單中的第一個瓶頸作業的開工時間與其前一個作業的完工時間的差異值，若該訂單的第一個作業即為瓶頸作業或是該訂單內並沒有瓶頸作業，則不計算該訂單差異值，再利用模糊隸屬函數求算出配合度，再將該次排程中所有訂單中的配合度取平均值即為所求。公式如下所示：

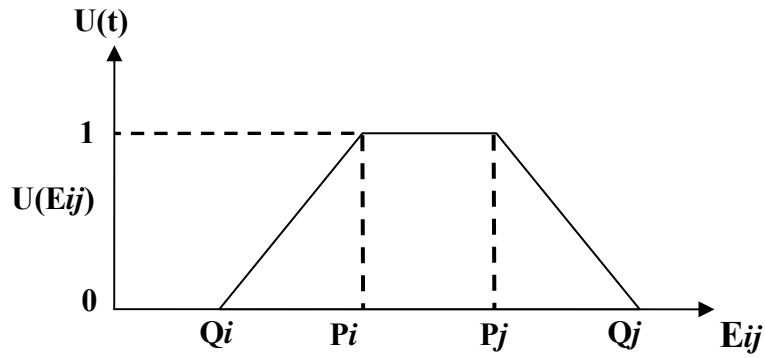


圖 3.4 配合度之模糊隸屬函數

$$\left\{ \begin{array}{l} U(E_{ij}) = \frac{E_{ij} - Q_i}{P_i - Q_i}, \quad Q_i \leq E_{ij} \leq P_i \\ U(E_{ij}) = 1, \quad P_i \leq E_{ij} \leq P_j \\ U(E_{ij}) = \frac{Q_j - E_{ij}}{Q_j - P_j}, \quad P_j \leq E_{ij} \leq Q_j \\ U(E_{ij}) = 0, \quad E_{ij} \leq Q_i \text{ or } E_{ij} \geq Q_j \end{array} \right. \quad \text{式 (3.5)}$$

$$S_i = \frac{\sum_{j=1}^m U(E_{ij})}{m} \quad \text{式 (3.6)}$$

其中， m ：該次排程內，具瓶頸作業且第一個瓶頸作業不為第一個加工途程的訂單數目

$U(E_{ij})$ ：個體解 i 的第 j 張訂單之配合隸屬度

E_{ij} ：個體解 i 的第 j 張訂單的配合時間差異值

P_i ：訂單 j 之理想配合時間差異值下界

P_j ：訂單 j 之理想配合時間差異值上界

Q_i ：訂單 j 之配合時間差異值下限

Q_j ：訂單 j 之配合時間差異值上限

S_i ：個體解 i 之平均配合度評估值

3.2 限制驅導排程演算法

針對零工式 (Job Shop) 排程問題而言，由於各訂單製造途程不完全相同、加上訂單數目、現場現場產能設備限制、物料限制、訂單生產優序限制、人力資源限制等諸多限制下，執行排程規劃更加複雜。規劃者得由諸多限制考量下，確認系統瓶頸，追求多項目標績效，制定一合適的排程規劃。

零工式排程問題亦為一種 NP-hard 問題。傳統的數理規劃，如分支界限法已無法在合理時間內求取出最佳解，而近年來，學者改以用啟發式演算法來求得近似最佳解，雖可在短時間內求得解，但其規劃結果缺乏對排程系統限制資源的考量，排程規劃無法容忍波動發生，一旦發生機器當機或缺料的情事，則生產系統會發生閒置，週期時間內的產出減少，排程績效降低。

本研究的零工式排程問題求解方法是以限制驅導排程技術 (DBR) 為基礎，以系統限制及瓶頸需求導向為先，進行有限資源分配。為補強傳統 DBR 方法不足之處，本研究將遺傳演算法或混合式遺傳演算法機制結合原有的 DBR 方法，作為排程問題求解方法，利用演化求解架構得到排程最適解，而遺傳演算求解過程必須引用到的染色體編碼、解碼方法於 3.2.3 小節說明，適應性函數同上述 3.1.2 與 3.1.3 小節說明。3.2.1 小節敘述限制驅導排程運作方式。

3.2.1 限制驅導排程技術的運作

本小節說明本研究所使用的限制驅導排程技術 (DBR) 運作方法與流程。由於傳統 DBR 架構中，對於緩衝設置與投料繩回推機制具有造成緩衝長度過長或緊密的缺點，對排程績效存有不良影響，本研究採用 Sirikrai 及 Yenradee (2006) 提出的逐層回推機制，從瓶頸作業的開工時間回推前一個作業的開工時程，逐層回推至投料點，如此可避免傳統型 DBR 缺點，但為了更確保瓶頸計劃 Drum 不被中斷打擾，本研究擬增設一小緩衝時間做為吸收統計波動的緩衝時間。本研究採用的修正型 DBR 架構如圖 3.5 所示。

而執行 DBR 流程，則採用 Schragenheim 及 Ronen (1990) 所提出的限制驅導排程技術的執行步驟：步驟一、確認系統限制；步驟二、決定系統緩衝 (Buffer) 大小；步驟三、限制驅導節奏 (Drum) 之設計；步驟四、

規劃投料時程 (Rope)。細部說明如下：

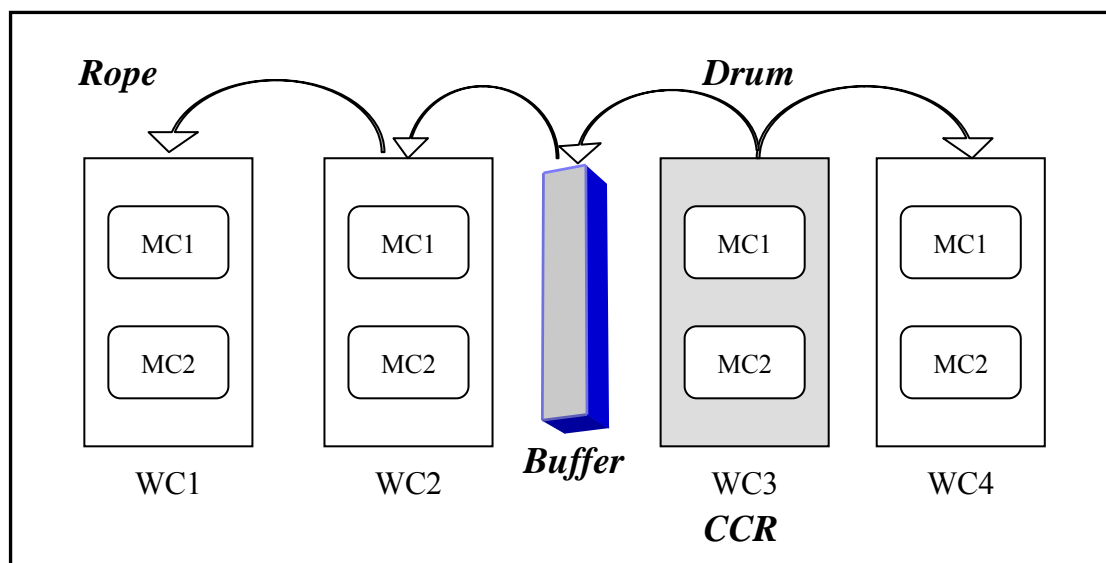


圖 3.5 修正型 DBR 架構示意圖

步驟 1. 確認系統限制：

本研究中，質性瓶頸為非實體限制，以不同訂單對企業營運績效的影響，利用 AHP 法則評定出理想訂單規劃優序為質性瓶頸，將非實體限制納入排程系統考量，質性限制架構如 3.1.2 小節；關於量性限制資源方面，則依據訂單資訊，計算所有工作中心的產能負荷，以產能負荷最重的工作中心為量性瓶頸，所有在瓶頸工作站進行加工的作業稱之限制作業 (Constraint operations)，若該訂單有一作業在瓶頸工作站進行加工，則其它作業稱之非限制作業 (Non-constraint operations)，若該訂單無任何作業在瓶頸工作站進行加工，則其全部作業稱之未佔用作業 (Free operations)，以上述分類方法，將該次排程內的所有訂單作業進行分群，以利後續作業排程規畫順序考量。

步驟 2. 決定系統緩衝(Buffer)大小：

本研究於瓶頸作業之同訂單中的前一個非瓶頸作業後設置緩衝，為產能受限資源緩衝 (Capacity Constrained Resource Buffer)。架構如圖 3.6。圖中 1-2 為瓶頸作業，1-1 是同訂單中的非瓶頸作業，緩衝則設於非瓶頸作業之後。

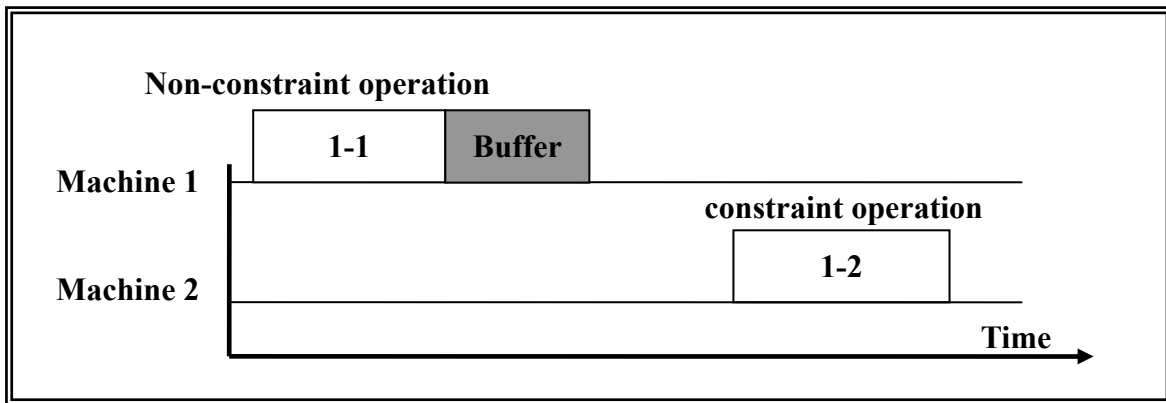


圖 3.6 緩衝設置架構示意圖

由於緩衝時間長度設定影響排程績效甚鉅，過去多數研究對緩衝長度多以經驗值訂定，面對於新訂單或是新的加工機器則無法有效明定之，實屬缺乏一個有效的緩衝長度建立機制。本研究擬將緩衝長度設定為一決策變數，以編碼方式表示並透過遺傳演算法的演化求解機制，求算適合的緩衝長度。另一方面，緩衝長度影響系統的穩健度，設定過長的緩衝時間，雖可以更有效的吸收系統波動，排程系統彈性變高，穩健程度也提高，但生產系統的製距時間會拉長，交期達交率亦會受到影響；反之，緩衝時間過短，排程系統缺乏彈性，吸收波動能力、穩健程度降低。

瓶頸作業開工時間與前一個作業完工時間的時間差異其作用與緩衝時間類似，由於排程是一種有限資源分配，所以並非每個作業銜接緊密，故本研究將上述的時間差異，做為總緩衝時間，如圖 3.7 所示。並以非瓶頸資源配合度這項指標來評估此總緩衝時間的績效。

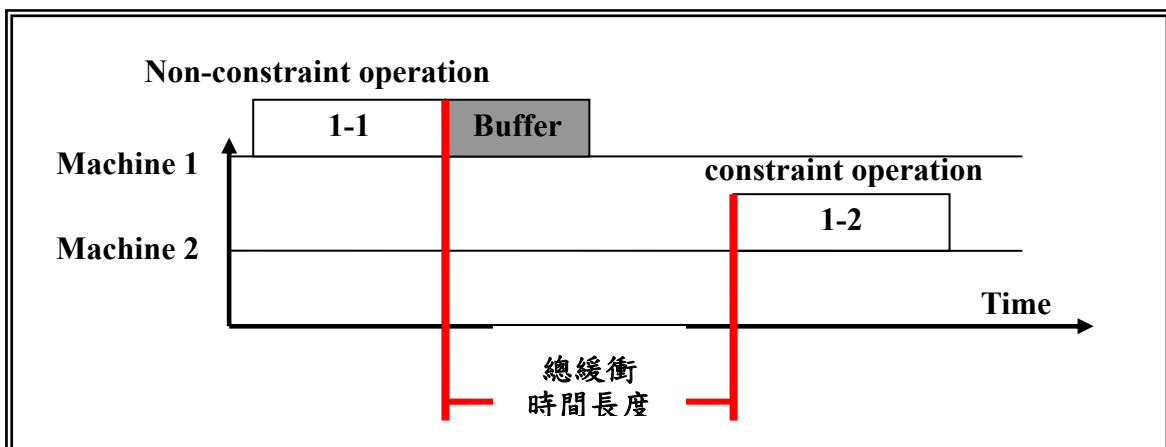


圖 3.7 總緩衝時間長度示意圖

步驟 3. 限制驅導節奏 (Drum) 之設計：

在步驟一確定瓶頸所在，並將作業分群，選出限制作業 (Constraint operations)。本研究以 Goldratt (1986)，Wu 及 Yeh (2006) 提出制定 Drum 的程序，先將所有限制作業以產能無限假設下，以規劃時程起點加上非瓶頸作業時間總和，得到瓶頸作業開工時間，以此類推，可制定其理想生產時程，即為廢墟圖，接著將作業加工重疊的時程部份推平，便可以得到合理的瓶頸生產排程規劃，此瓶頸作業排程計畫稱之鼓 (Drum)。舉一例：訂單一、二皆有兩個作業，表示為 (作業代號/加工機器/加工時間)：(1-1/M1/2)、(1-2/M2/2)、(2-1/M3/1)、(2-2/M2/4)，假設兩訂單的緩衝長度皆為 1，其中機器 2 產能負荷最高，故為瓶頸機台，則 1-2 及 2-2 作業為瓶頸作業，1-1 與 2-1 為非瓶頸作業，廢墟圖表示為圖 3.8，假設 1-2 生產規劃優序高於 2-2，所以先確定 1-2 開工時程，並將 2-2 開工時程延至 1-2 後，即為推平廢墟過程，最後可得到一合理 Drum 計畫，其甘特圖如圖 3.9。

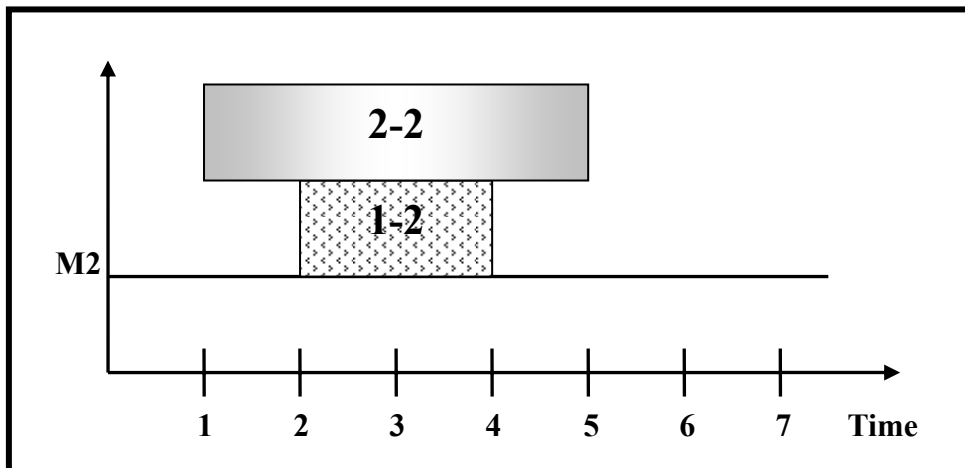


圖 3.8 廢墟圖

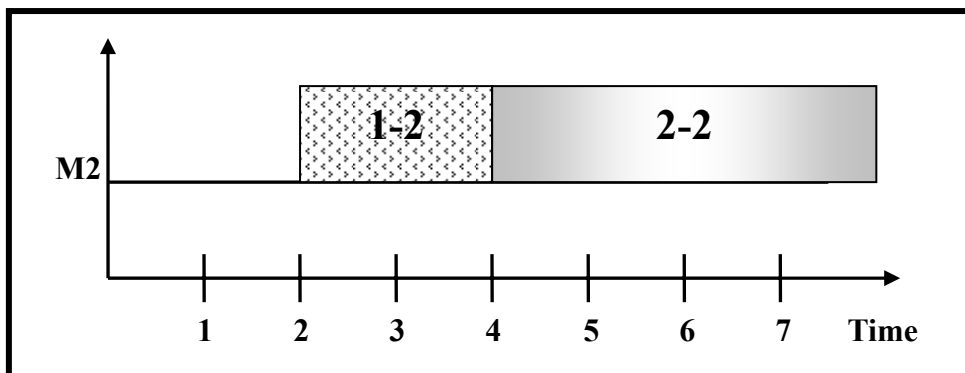


圖 3.9 廢墟推平後之甘特圖

步驟 4. 規劃投料時程(Rope)

本研究使用的投料時程是採用 Sirikrai 及 Yenradee (2006) 提出的逐層逆推方式，再加以修正。由瓶頸作業開工時間，增設一緩衝，再逐層逆推先前一層作業的開工時程，依序往前回推得到投料時間。而瓶頸作業之後的非瓶頸作業，亦循此則，往後回推各作業開工時程。最後可得到完整的預先排程結果。依循前述小範例，必須逐層回推 1-1 作業與 2-1 作業的開工時程，假設 1-1 作業生產規劃優序高於 2-1 作業，故先制定 1-1 作業開工時程，其中必須包含時間長度為 1 的緩衝，循同理回推 2-1 作業的開工時程，如圖 3.10。

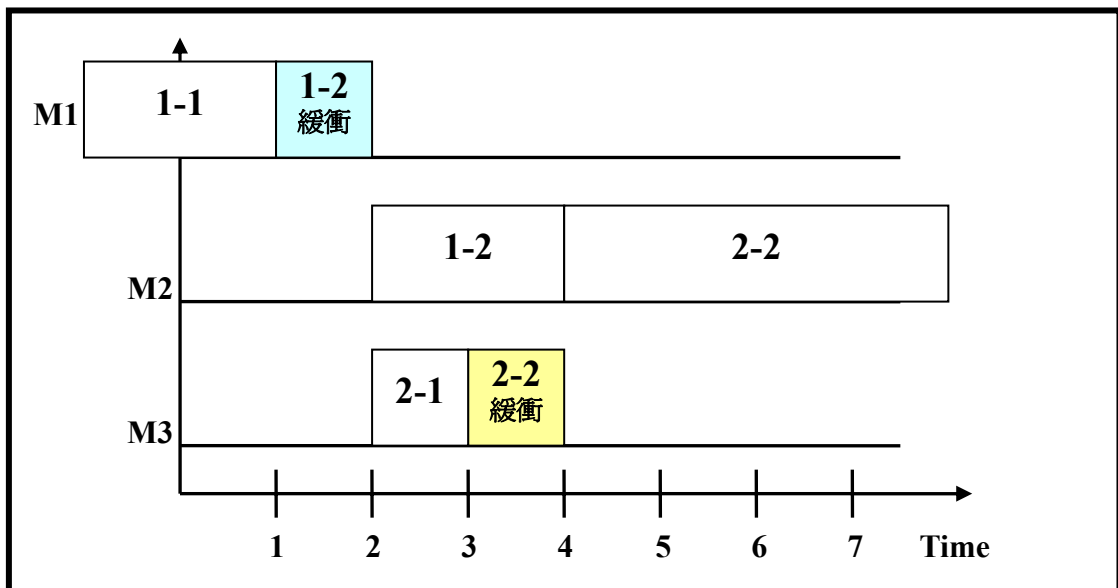


圖 3.10 投料 Rope 制定之甘特圖

其中由於 1-1 作業的開始加工時間已經超過規劃時程起點，為不可行的規劃結果，必須進行投料時程修正，其修正方式為將全部作業右移，右移時間間隔長度為 1-1 作業的開工時間與規劃時程起點時間間隔差異值，經過投料時程修正可得一最後的排程結果，如圖 3.11。

限制驅導排程技術雖是有效的排程方法，但其中尚有許多問題有待解決，如：合宜的緩衝時間長度設定、廢墟推平的作業順序制定、細部瓶頸作業排程方法。本研究擬用遺傳演算法與混合式遺傳演算法將排程問題編碼，再以限制驅導排程技術進行解碼，並演化求解，如此便可解決上述問題。關於限制驅導排程技術結合遺傳演算法與混合式遺傳演算法於 3.2.2 小

節說明。細部解碼排程法則於 3.2.3 小節說明。

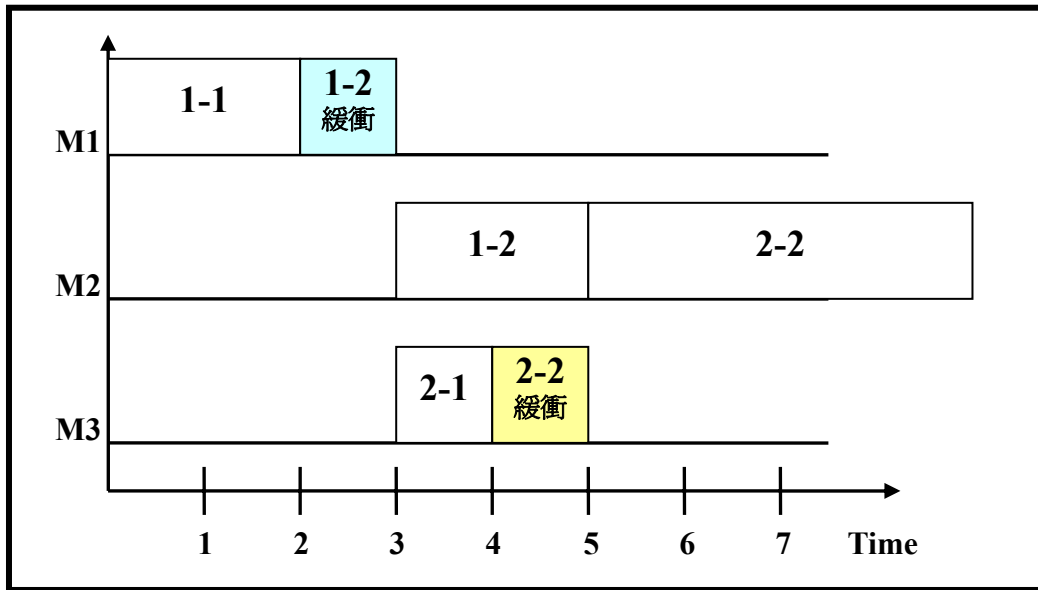


圖 3.11 投料時程修正後之排程甘特圖

3.2.2 限制驅導排程技術結合遺傳演算法機制的運作架構

本小節說明研究中關於限制驅導排程技術結合遺傳演算法機制的設計及其運算過程。限制驅導排程技術為本研究主要的排程求解方法，但因傳統 DBR 機制有其不足之處，如：緩衝時間長度設定、廢墟推平的作業順序等問題皆無明確法則，本研究擬將過去 DBR 缺點以遺傳演算法機制作為修正方法，其中緩衝時間長度以編碼方式表示之，作業廢墟推平順序則以隨機編碼次序為依據，透過解碼流程將各染色體專換為明確排程結果，計算其多目標適應函數值，評估此個體之排程績效優劣。加上遺傳演算法的演化求解機制，可在多目標下，求取一折衷的排程最佳解（含各訂單的緩衝時間長度設定、完整的作業派工時程）。限制驅導排程技術 DBR 結合遺傳演算法 GA 機制之架構如圖 3.12。

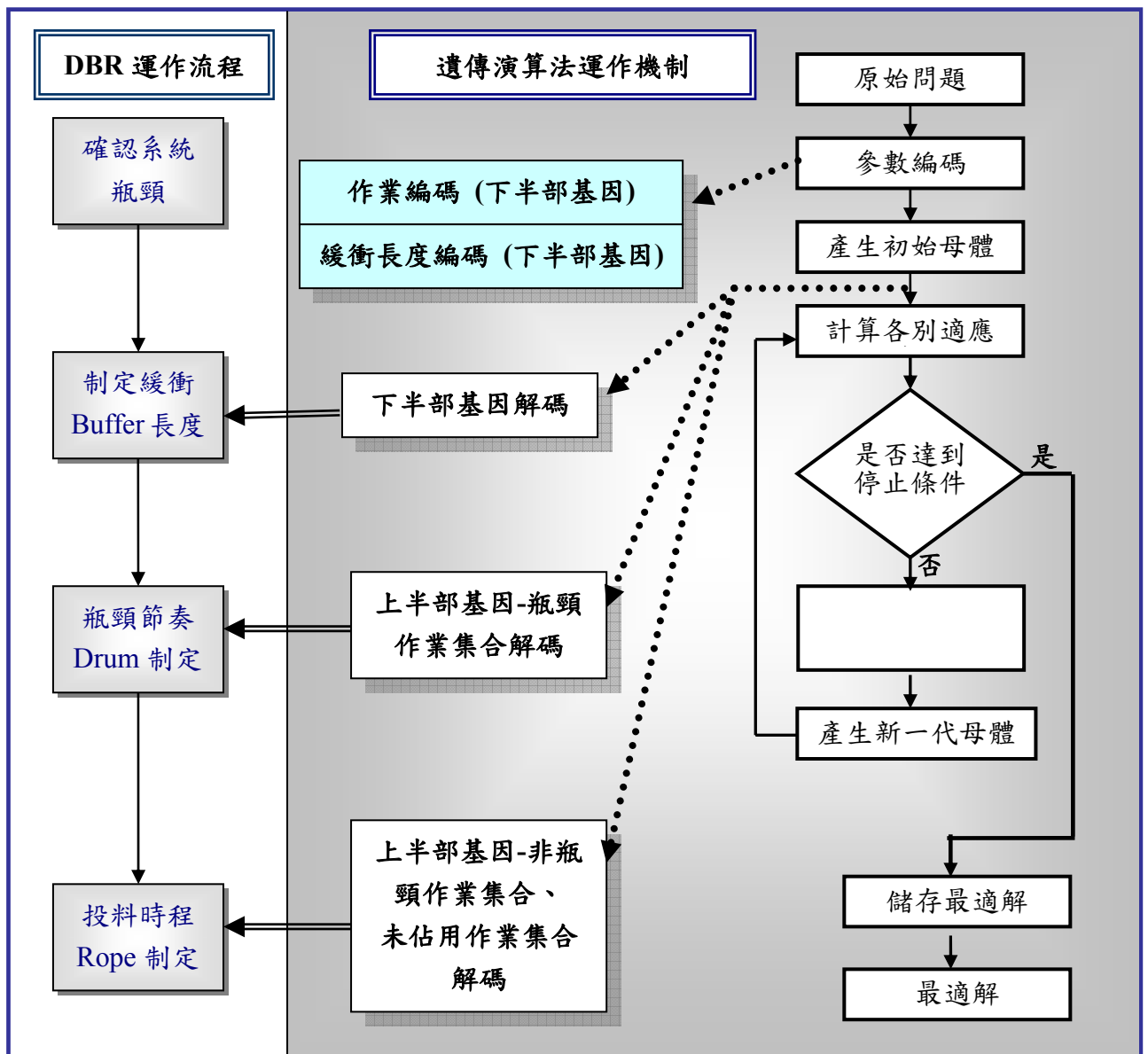


圖 3.12 限制驅導排程技術 DBR 結合遺傳演算法 GA 之架構

其中遺傳演算法將單一染色體個體分為上半部與下半部基因，本小節先就編碼方式說明如下：

1. 上半部基因編碼

染色體個體上半部的基因，每一基因代表一作業的排程的優先順序，而基因的排列是先依照訂單再依照訂單本身的加工作業順序。也就是說第一個基因為訂單一的第一個加工作業，第二個基因為訂單一的第二個加工作業，直至訂單一的最後一個加工作業；而其下一個基因即為訂單二的第一個加工作業，依此類推。上半部基因數目等於所有訂單作業數總和。上半部基因的編碼示意圖如圖 3.13。

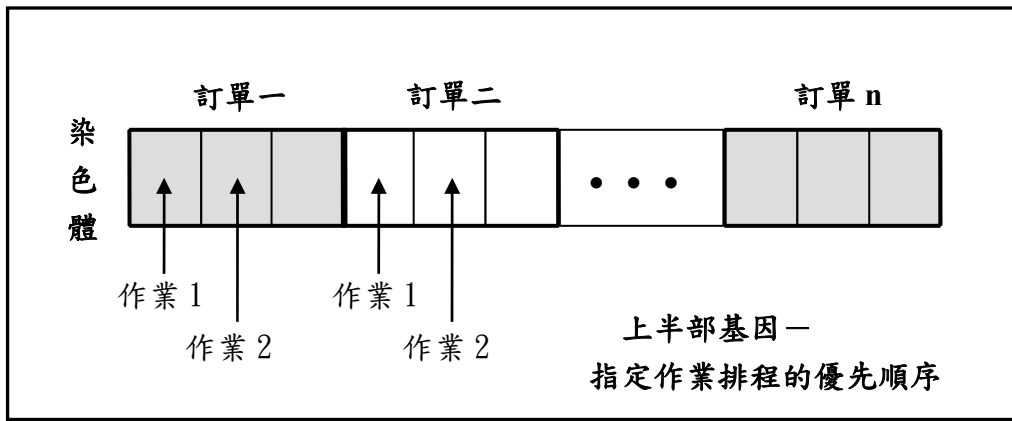


圖 3.13 上半部基因編碼表示

2. 下半部基因編碼

染色體個體中下半部的基因，採用 2 元編碼制，每 3 碼基因代表一個訂單的緩衝時間長度。舉例來說，某條染色體的前三碼為 011，表示為訂單 1 緩衝時間長度為 $0 * 20 + 1 * 21 + 1 * 22 = 6$ 。則訂單 1 首個瓶頸作業的前一個作業其加工時間變為原加工時間加上下半部基因碼設定的緩衝時間，成為調整後加工時間，在求解過程，以調整後加工時間做為排程資訊。下半部基因的編碼示意圖如圖 3.14。

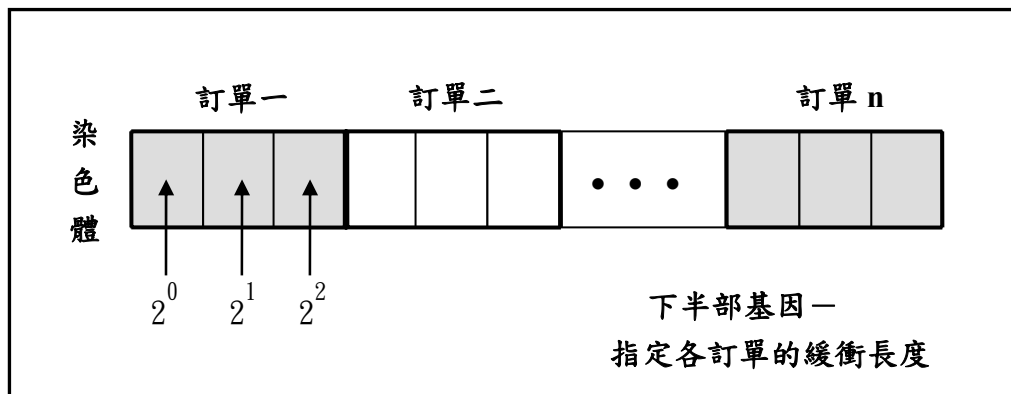


圖 3.14 下半部基因編碼表示

3. 初始族群的產生

為了能讓系統在起始搜尋時，對於每一狀態空間(state space)都有同等機會，本研究將採用隨機的方式產生初始族群。假設所有訂單的總作業數共為 N 個作業，對於上半部基因來說，初始族群產生方式是隨機產生一個 $1 \sim N$ 不重複的數字串列個體的起始基因，如圖 3.15。

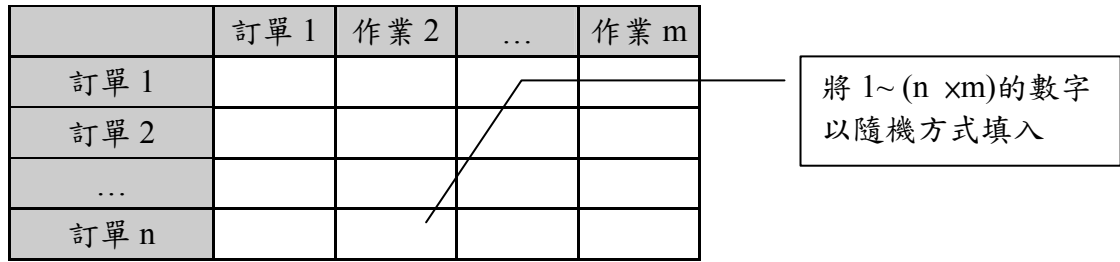


圖 3.15 上半部基因初始族群示意圖

但是，以這種方式產生初始族群會有不理解的現象發生，例如，訂單 1 有 5 個作業，而其基因分別為[5, 8, 6, 10, 12]，我們發現訂單 1 中的作業 3 因為加工途程的限制必須在作業 2 之後方可開始加工，但作業基因優先於作業 2，此個體為一不理解的，關於這一點，我們將藉由一啟發式修正的方式將其合理化。修正方式是將不理解的兩個基因做交換。在上例中，我們將 8 與 6 的互換之後的個體即為一理解。

染色體個體中下半部的基因，代表各訂單的緩衝長度時間，隨機填入 0 或 1 的值，再依 2 元解碼方式，計算出緩衝長度時間。圖 3.16 為下半部基因初始族群示意圖。

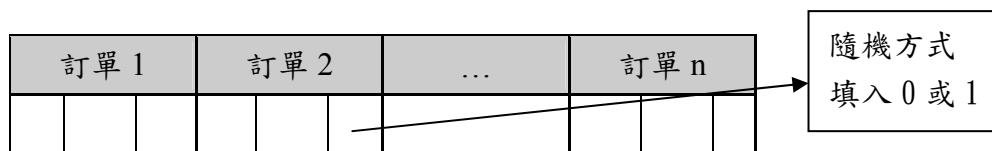


圖 3.16 下半部基因初始族群示意圖

4. 多目標限制適應性函數之設計

本研究所設計的多目標限制適應性函數由兩個部分構成：量性多目標函數與質性限制懲罰函數。量性多目標函數是針對製造現場中量性瓶頸資源設定的績效衡量指標，而質性懲罰函數則是利用 AHP 法所求得質性瓶頸的訂單順序計算轉換而來。由於是利用遺傳演算法來搜尋出較佳的作業順序，其中每一個體皆是一組作業順序，將其轉換為訂單順序後，即可算出其與質性訂單順序的差異，並以其作為懲罰函數值。本研究以 3.1.2 節中所

計算出之質性限制資源考量下的訂單順序作為質性瓶頸，演算所得的個體基因再與此訂單順序比較，以懲罰函數來處理違反質性限制的情形。首先，求出個體解中訂單的順序，計算方式是由個體的基因解碼順序算出每張訂單平均的解碼順序，將其排序。懲罰函數的計算方式則是先計算排序後的訂單順序與質性限制下之訂單順序之間的加權後差異平方總和，再將其除以可能的加權後最大差異平方總和，以正規化之。而最大差異平方總和則是以最大違反質性限制的訂單順序所計算出的。計算取加權之原因，則是為了讓重要的訂單更顯示其重要程度，不重要的訂單在排程時若違反質性優序也無妨。詳細的懲罰函數計算步驟舉例如下：假設目前有 5 張訂單，這些訂單經 AHP 求得之權重如表 3.6。

表 3.6 各訂單之加權值

F_i	F_1	F_2	F_3	F_4	加權值	排序
訂單 1	0.1061	0.15	0.15	0.35	0.1745	4
訂單 2	0.0668	0.20	0.25	0.25	0.1770	3
訂單 3	0.4055	0.30	0.10	0.15	0.2913	1
訂單 4	0.1648	0.15	0.20	0.10	0.1586	5
訂單 5	0.2568	0.20	0.30	0.15	0.2256	2

透過 AHP 得到在質性限制的訂單順序為：訂單 3→訂單 5→訂單 2→訂單 1→訂單 4。而由遺傳演算法所得的個體解中的基因解碼順序的計算如表 3.7，加權差異平方和之計算如表 3.8。

表 3.7 訂單平均基因解碼順序

	作業 1	作業 2	作業 3	作業 4	作業 5	加總	平均基因順序	排序
訂單 1	14	11	6	1	15	55	9.4	1
訂單 2	12	7	2	16	20	60	11.4	2
訂單 3	13	8	3	17	21	65	12.4	3
訂單 4	9	4	18	22	24	70	15.4	4
訂單 5	10	5	19	23	25	75	16.4	5

表 3.8 加權後差異平方總和之計算

訂單編號	1	2	3	4	5	加權差異平方總和
訂單加權值	0.1745	0.1770	0.2913	0.1586	0.2256	
質性訂單順序	4	3	1	5	2	
平均訂單解碼順序	1	2	3	4	5	
差異平方	3^2	1^2	$(-2)^2$	1^2	$(-3)^2$	
加權差異平方	1.5705	0.1770	1.1652	0.1586	2.0304	5.1017

加權差異平方的總和為 $1.5705+0.1770+1.1652+0.1586+2.0304=5.1017$

依照質性限制考量之訂單順序，找出最大違反質性限制的訂單順序，以求出最大差異平方和。本例依表的質性限制順序，找出的最大違反訂單順序為：訂單 4→訂單 1→訂單 2→訂單 5→訂單 3。以此最大違反的訂單順序與質性訂單順序，求出加權後最大差異平方的總和為 8.7988，計算如表 3.9。

表 3.9 加權後最大差異平方總和之計算

訂單編號	1	2	3	4	5	加權最大差異平方總和
訂單加權值	0.1745	0.1770	0.2913	0.1586	0.2256	
質性訂單順序	4	3	1	5	2	
最大違反訂單順序	2	3	5	1	4	
差異平方	2^2	0^2	$(-4)^2$	4^2	$(-2)^2$	
加權差異平方	0.6980	0	4.6608	2.5376	0.9024	8.7988

將加權後差異平方總和除以加權後最大差異平方總和，以正規化後，所得懲罰函數值如下： $5.1017 / 8.7988 = 0.5798$

本研究利用 AHP 法計算出質性限制資源與量性限制資源的權重，整體多目標限制適應適應函數值計算如下式

$$f(x) = W_1 \cdot (\sum \alpha_i \cdot f_i) + W_2 \cdot (1 - p(x)) \quad \text{式 (3.7)}$$

其中， $f(x)$ ：個體解 x 的量性多目標適應函數值。

$p(x)$ ：個體解 x 的質性懲罰函數值。

W_1 ：量性限制資源的權重。 W_2 ：質性限制限制的權重。

α_i ：量性目標 i 的權重。 f_i ：量性目標 i 的評估值。 $W_1 + W_2 = 1$ ， $\sum \alpha_i = 1$ 。

5. 育種選擇

在遺傳演算法中，選取個體來產生下一子代，通常是藉由輪盤法（roulette wheel）來進行此一機制。而關於輪盤法中每個槽（slot）大小的設計方式一般常用的是直接以個體的適應函數值來設計，如下

$$P(x_i) = \frac{f(x_i)}{\sum_i f(x_i)} \quad \text{式 (3.8)}$$

其中， $P(x_i)$ 為個體 x_i 被選中的機率； $f(x_i)$ 為個體 x_i 的適應函數值。

但是，以這種方式來執行選擇機制，隨著族群演化，族群中個體的適應函數值逐漸提高，而且彼此之間的差距愈趨減小，如此一來，將會造成最佳解與最差解被選中的機率接近 1：1，使得整個演算過程中族群的適應函數值發生停滯。基於上述，本研究中在育種選擇機制上採用另一種方式，以競賽法（Tournament selection）當作育種的機制，在每一代的演化過程中，首先隨機選取兩個或更多個物種，具有最大適應函數值的物種即被選中送至交配池中。由於競賽法所需的計算量較少，且可以藉由一次選取物種個數的多寡來控制競爭的速度，因此採用競爭式選擇。

6. 交配運算子設計

由於染色體中包含了訂單之作業排程的優先順序與其加工途程所指定的工作中心中佔有的平行機台資訊，因此一般常用的交配運算子已無法使用。本研究的交配運算分為上半部基因的交配與下半部基因的交配。上半部基因屬於排序數值，需採用排序問題的交配運算子，本研究引用 Bean（1994）所提出的 Random Key Representation 修正重新編碼與交配方法並加以修改；而下半部基因的交配則以一般的兩點交配處理。上半部基因之交配步驟如下：

步驟一：在育種選擇策略選出的母體中，隨機選擇二條染色體作為父代。

步驟二：在選取到的染色體上的每個基因，以 0 至 1 的均勻分配隨機產生一數值，依照原基因由小大到重新代表之，即重新以一個 0 至 1 的隨機數編碼為此基因值。

步驟三：以兩點交配方式，隨機選取兩點作為交配點，互換兩交配點之間的基因值。

步驟四：將染色體各基因值由小到大重新排序編號，即步驟二重新編碼後

的解碼。完成此兩條染色體的交配運算。

步驟五：重複步驟二～四，直到所有子代染色體產生完畢。本研究上半部基因的交配舉例如圖 3.17：

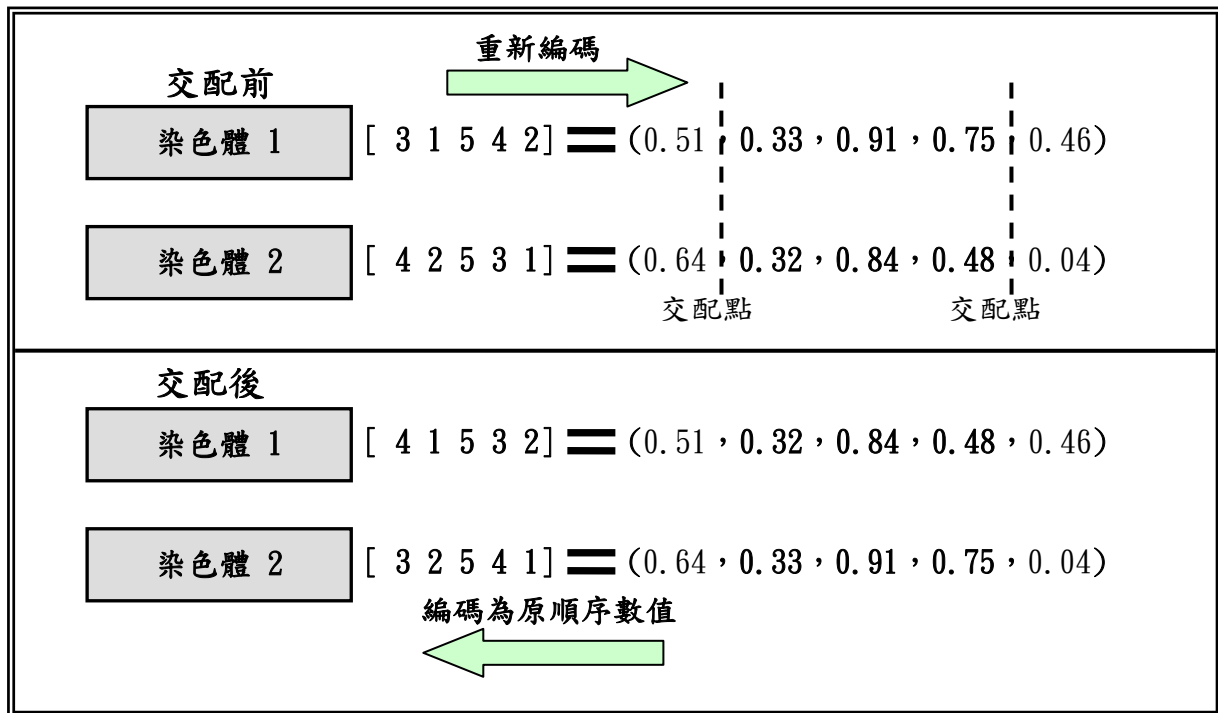


圖 3.17 上半部基因交配示意圖

7. 突變運算子設定

在突變運算子設計上以考量每一基因為主，不同於以往大部分研究著重於考慮每一染色體突變機率，本研究對每一基因均給予突變機率來決定是否需要進行突變，希望藉此能拓展遺傳演算法的搜尋空間，使其搜尋更有效率；另外由於排程問題的特殊型態，需考量作業排程的優先順序與其佔有的平行機器資訊，因此本研究採取以下突變方法，分為上半部基因的突變與下半部基因的突變。

新產生的子代染色體中的上半部基因，依突變機率決定要突變，則隨機選擇同一染色體中另一上半部基因，二基因數字互換。圖 3.18 所示即為此染色體上半部基因中的第二位置依突變機率決定要突變，再隨機選擇第四位置，兩位置的上半部基因數字交換，代表第二位置的作業與第四位置的作業排程順序互換。下半部基因的突變仍是對每一基因均給予突變機率來決定是否需要進行突變，若該基因決定突變，則基因碼由原本的 0 變為 1，或是由原本的 1 變為 0。

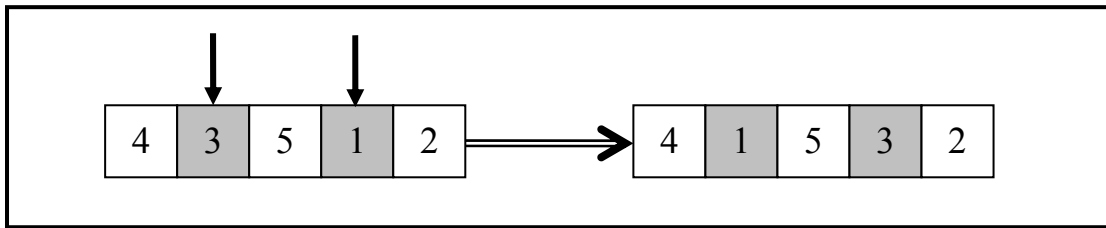


圖 3.18 上半部突變示意圖

3.2.3 解碼法則

本小節介紹的遺傳演算法的解碼流程，即為進行作業排程，將單一個體中的染色體資訊轉換為明確排程結果（一組染色體代表一個排程解），以供計算適應函數值，判斷個體之排程績效優劣。

在本研究所提出之排程演算法中，以限制驅導排程技術為基礎，於解碼過程，仍以限制驅導技術之概念為基礎，依照 3.2.1 小節介紹之 DBR 執行架構進行解碼（作業排程）。解碼流程如圖 3.19：

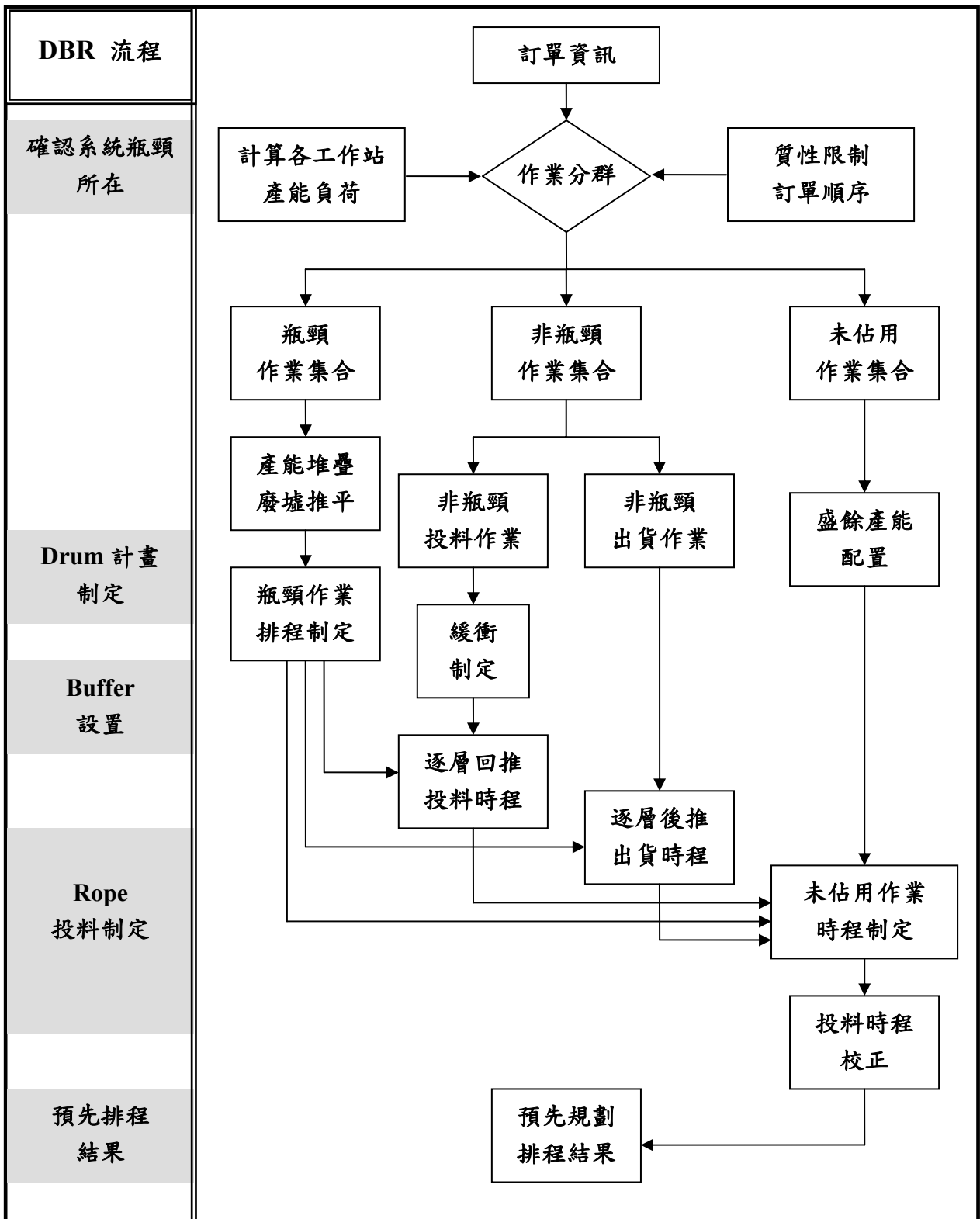


圖 3.19 解碼流程圖

以限制驅導排程法則為基礎之解碼步驟（作業排程）如下：

步驟一：先將基因碼分成三種集合，如下：瓶頸作業基因碼集合，非瓶頸作業基因碼集合，未佔用作業基因碼集合。

- 步驟二：將瓶頸作業基因碼組合進行排序，按照排序的優序，作為瓶頸作業解碼的順序。
- 步驟三：將欲排作業所屬訂單中的先前作業加工時間加總，作為欲排作業的理想開工時間。判斷在理想開工時間點上，欲排作業使用的加工機台，是否閒置，且閒置時間區段足以排入該作業，若是則至步驟五，若否則至步驟四。
- 步驟四：尋找大於理想開工時間之後的機器空閒區段，判斷足以排入該作業的區段並選擇最早區段作為欲排作業的開工時間，並推算欲排作業的完工時間。至步驟六。
- 步驟五：將理想開工時間作為欲排作業的開工時間，並推算欲排作業的完工時間。至步驟六。
- 步驟六：依照步驟二~步驟五的方式，依序將所有瓶頸作業排定。至步驟七。
- 步驟七：將非瓶頸作業基因集合分為：非瓶頸投料作業集合（瓶頸作業之前的非瓶頸作業）、非瓶頸出貨作業集合（瓶頸作業之後的非瓶頸作業）。至步驟八。
- 步驟八：判斷非瓶頸投料作業集合中，為各訂單中的瓶頸作業之前第一個作業基因代碼並排序，作為第一層非瓶頸作業之作業代碼及順序。至步驟九。
- 步驟九：判斷非瓶頸投料作業集合中，為各訂單中的瓶頸作業之前第二個作業基因代碼並排序，作為第二層非瓶頸作業之作業代碼及順序。依步驟八~步驟九的方式，逐層回推。依每一層中的順序及各層順序得到非瓶頸投料作業的解碼優序，至步驟十。
- 步驟十：依步驟九所得順序，進行非瓶頸投料作業解碼，至步驟十一。
- 步驟十一：尋找欲排作業所屬訂單中，欲排作業之下一個途程作業的開工時間，作為欲排作業的理想完工時間，並尋找欲排作業的加工機器上，小於等於理想完工時間之後的機器閒置區段。至步驟十二。
- 步驟十二：判斷足以排入該作業的閒置區段，若有，則選擇最晚區段的閒置結束時間作為欲排作業的完工時間，並推算欲排作業的開工時間，至步驟十四。若否，則至步驟十三。
- 步驟十三：判斷在該機器已經排定之作業的最早開工時間，作為欲排作業

的完工時間，並推算欲排作業的開工時間，至步驟十四。

步驟十四：按照步驟十一~步驟十三的方法，依序且逐層的將所有非瓶頸之前作業排定，至步驟十五。

步驟十五：將瓶頸作業後的非瓶頸組合中，為各訂單中的瓶頸作業之後的第一個作業基因代碼並排序，作為第一層非瓶頸作業之後作業代碼及順序。

步驟十六：將瓶頸作業後的非瓶頸組合中，為各訂單中的瓶頸作業之後的第二個作業基因代碼並排序，作為第二層非瓶頸作業之後作業代碼及順序。

步驟十七：依此步驟十五~步驟十六的原則，逐層後溯。依每一層中的順序及各層順序得到非瓶頸之後作業基因的解碼優序，至步驟十八。

步驟十八：依步驟十七所得順序，進行非瓶頸之後作業解碼，至步驟十九。

步驟十九：尋找該訂單中，欲排作業之前一個途程作業的完工時間，作為欲排作業的理想開工時間，並尋找欲排作業其加工機器中，大於等於理想完工時間之後的機器閒置區段。至步驟二十。

步驟二十：判斷足以排入該欲排作業的閒置區段，若有，則選擇最早區段的閒置開始時間作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟二二。若否則至步驟二一。

步驟二一：判斷在該機器已經排定之作業的最晚完工時間，作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟二二。

步驟二二：按照步驟十八~步驟二一的方法，依序且逐層的將所有非瓶頸之後作業排定，至步驟二三。

步驟二三：判斷訂單中未含瓶頸作業之基因組合是否為空集合，若是，則至步驟三四，若否則至步驟二四。

步驟二四：將訂單中未含瓶頸作業之基因組合排序，得到未含瓶頸作業解碼優序。至步驟二五。

步驟二五：依照步驟二四得到的順序，進行作業解碼，至步驟二六。

步驟二六：判斷欲排作業是否為該訂單的第一個作業途程，若是，則至步驟二七，若否則至步驟三十。

步驟二七：將欲排作業的理想開工時間設定為 0，並尋找欲排作業其加工

機器中，大於等於理想完工時間之後的機器閒置區段。至步驟二八。

步驟二八：判斷足以排入該欲排作業的閒置區段，若有，則選擇最早區段的閒置開始時間作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟三三。若否則至步驟二九。

步驟二九：判斷在該機器已經排定之作業的最晚完工時間，作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟三三。

步驟三十：將欲排作業的理想開工時間設定為欲排作業於該訂單中之前一個途程作業完工時間，並尋找欲排作業其加工機器中，大於等於理想完工時間之後的機器閒置區段。至步驟三一。

步驟三一：判斷足以排入該欲排作業的閒置區段，若有，則選擇最早區段的閒置開始時間作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟三三。若否則至步驟三二。

步驟三二：判斷在該機器已經排定之作業的最晚完工時間，作為欲排作業的開工時間，並推算欲排作業的完工時間，至步驟三三。

步驟三三：按照步驟二三~步驟三二的方法，依序且逐層的將所有訂單中未含瓶頸作業排定，至步驟三四。

步驟三四：判斷全部作業的開工時間是否小於 0，若有，則至步驟三五，若否則至步驟三六。

步驟三五：尋找最早開工時間，計算其最早開工時間與時間點 0 的差值，將全部作業的開工時間及完工時間加上此差值，作為校正後的開工時間及完工時間。至步驟三六。

步驟三六：完成解碼。

透過上述的步驟，可將一組染色體解碼為轉為一排程解。以利計算計算個體適應函數值。

3.2.4 混合式遺傳演算法(HGA)

傳統遺傳演算法是透過編碼將問題的解答以基因形式來表達，再依據生物演化「適者生存」的原理進行交配與突變等運算；而禁忌搜尋法 (Tabu Search) 主要利用彈性記憶 (禁忌串列) 讓搜尋過程有跳離區域最佳解的機

會。在遺傳演算法與禁忌搜尋法各有其求解的獨特優點之下，本研究將以混合式遺傳演算法（HGA）來求解排程解，其中是以傳統 GA 先求出一代族群解，然後將所有個體進行禁忌搜尋，除了讓最佳解有搜尋到更佳解的機會，最差解也有跳脫區域的機會。

本研究將採用短期記憶結構的禁忌搜尋法對之前遺傳演算法所得的排程解進行鄰域搜尋，以提升系統搜尋的效率，執行禁忌搜尋時所使用的目標函數與遺傳演算法中的適應性函數相同，關於禁忌搜尋機制的步驟如下。

步驟一：由個體 I 中每一訂單以隨機的方式抽選一作業來進行禁忌搜尋，被抽選到的作業順序元素為 S。

步驟二：設定禁忌串列的大小為 M 及搜尋次數 Z。設定最佳排程順序 S^* 為目前個體的作業順序。設定最佳目標函數 $G(S^*)=G(I)$ 、設定 $S_1=S$ 、設定 $z=1$ 。

步驟三：搜尋 S_z 的鄰域， S^1, S^2, \dots, S^{N-1} ，將 S^1, S^2, \dots, S^{N-1} ，回填至個體 I 中，檢驗其是否有違反順序限制，若有則進行調整，否則並計算其目標函數值 $G(S^1), G(S^2), \dots, G(S^{N-1})$ ；N 為排列順序中的元素個數。

步驟四：從 $G(S^1), G(S^2), \dots, G(S^{N-1})$ 選取不屬於禁忌串列中之元素對的最佳目標函數值 $G(S_z^*)$ 。如果 $G(S^*) < G(S_z^*)$ ，則令 $S^*=S_z^*$ 。

步驟五：以 FIFO 的方式更新禁忌串列； $z=z+1$ ，如果 $z=Z$ 則停止。

步驟六：將最初個體 S 以 S^* 替代，進行下一族群的遺傳演算法運作。

舉例說明：假設問題為 5×5 的 Job shop 排程問題，而目前染色體基因 I，如表 3.10，其適應性函數 $G(I)=10$ 。

表 3.10 各訂單作業之基因

	作業 1	作業 2	作業 3	作業 4	作業 5
訂單 1	1	6	11	16	21
訂單 2	2	7	12	17	22
訂單 3	3	8	13	18	23
訂單 4	4	9	14	19	24
訂單 5	5	10	15	20	25

*灰色表示被選中者

1. 訂單 1~5 被選中的作業之順序元素

$$S = [6, 12, 3, 14, 20] = S_1$$

2. 搜尋 S_1 的鄰域，並將其回填至原個體 I 中，計算其目標函數值。

設定搜尋次數 $Z = 7$ ，禁忌串列 $M = 3$ ， $G(S^*) = 10$ 。

當 $z = 1$

表 3.11 S_1 的鄰域

鄰域解	12, 6, 3, 14, 20	6, 3, 12, 14, 20	6, 12, 14, 3, 20	6, 12, 3, 20, 14
目標函數	9.5*	8	8	6

此時，禁忌串列 = $\{(6, 12)\}$ 。

當將鄰域解(12, 6, 3, 14, 20)回傳至原個體時，訂單 1、2 的作業中的順序分別為： $(1, 12, 11, 16, 21)$ 與 $(2, 7, 6, 17, 22)$ 皆違反訂單加工途程的限制必需將其調整為： $(1, 11, 12, 16, 21)$ 與 $(2, 6, 7, 17, 22)$ 。

當 $z = 2$

表 3.12 S_2 的鄰域

鄰域解	6, 12, 3, 14, 20	12, 3, 6, 14, 20	12, 6, 14, 3, 20	12, 6, 3, 20, 14
目標函數	10	9.8*	9	8

雖然鄰域解 1 的目標函數最高，但其是由 $(6, 12)$ 互換所得，而 $(6, 12)$ 是目前禁忌串列中的元素對，不能選取，所以選取第 2 個鄰域解，此時，禁忌串列為 $\{(6, 12), (3, 6)\}$ 。

當 $z = 3$

表 3.13 S_3 的鄰域

鄰域解	3, 12, 6, 14, 20	12, 6, 3, 14, 20	12, 3, 14, 6, 20	12, 3, 6, 20, 14
目標函數	9.8*	9.5	7	9

此時禁忌串列為 $\{(6, 12), (3, 6), (12, 3)\}$ ，而因為此鄰域解的目標函數 $12 > 10$ (原個體的適應性函數)，所以原個體將會被替換。

當 $z = 4$

表 3.14 S4 的鄰域

鄰域解	12, 3, 6, 14, 20	3, 6, 12, 14, 20	3, 12, 14, 6, 20	3, 12, 6, 20, 14
目標函數	9.8	9	8	11*

此時禁忌串列為 $\{(3, 6), (12, 3), (14, 20)\}$ ，持續進行至 $z = 7$ 。

本節中混合式遺傳演算法(HGA)的設計理念，主要是藉由傳統 GA 交配、突變及育種選擇等運算子，使得求解過程可以搜尋出與原族群差異較大的個體，讓系統保有開發 (explore) 的功能。另一方面，當每獲得一排程解時，利用禁忌搜尋在此基因型態下的鄰域解，提供系統有精鍊 (refine) 的功能。結合禁忌搜尋後的混合式遺傳演算法 (HGA) 運作流程如圖 3.20。

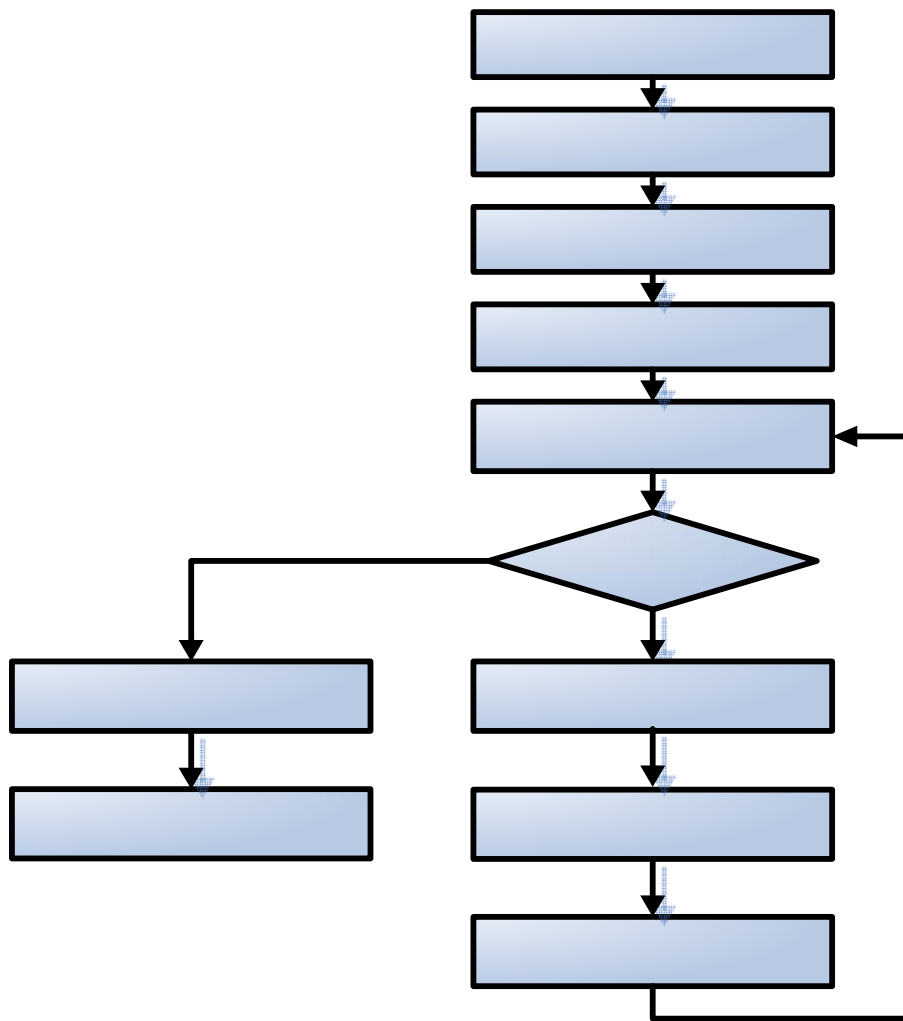


圖 3.20 混合式遺傳演算法(HGA)運作流程圖

3.3 中斷測試與穩健性

當排程解的目標函數值若不會因為中斷事件的干擾而下降太多，則稱此排程為穩健排程。在實際的生產環境中，中斷事件的發生可能原因眾多，如：緊急插單、機台故障、物料短缺、抽單、變更交期、作業進度落後或超前、訂貨數量變更等等。其中緊急插單、機台故障與物料短缺三項對於生產系統的負面影響極大。

在正常週期排程規劃中，若可事先規劃出一個具有穩健性(Robustness)的排程，則生產作業若遭遇中斷事件，整體生產仍可維持高的績效指標。故本研究在預先規劃排程中，事先考量事件中斷發生，冀望以排程規劃的彈性來吸收中斷干擾所造成的影響，使規劃結果具有容忍波動現象的能力，以利日後工廠，讓生產系統在實際運作時，更具有穩健性。本研究以機台故障、物料短缺與緊急插單三種中斷情節做為干擾測試，於 3.3.1 小節簡述此三項中斷測試架構。

3.3.1 中斷測試架構：

在本研究中，對於機台故障與物料短缺兩種中斷事件，以右移法進行調整。只變動受到干擾影響作業的開工時程，使大部分的作業仍然能維持於原本的排程相對位置，且由於此法不需經過遺傳演算法重新演化求解，因此調整過程快速許多。

而緊急插單的中斷事件由於涉及到較多的質性限制資源改變，且訂單作業數也增加改變，須重新評定加入緊急插單後的質性限制，調整方法以剩餘產能作為緊急訂單的可用產能，將各作業安插至各機器的閒置時間，如此原先作業仍然能維持於原本的排程相對位置。各中斷情節測試架構與說明如下：

中斷測試：機台故障

雖然工廠會進行例行保養與預防保養，但機器仍會發生偶發性故障，一旦機器發生故障，其對現場的影響是立即的。機器發生故障時，需用到該機器的生產作業都需要延後到機器修復為止，才能再進行。生產排程人員需要預估機器恢復正常運轉所需的時間，再依此資訊作重新規劃排程的動作。

機器故障中斷測試步驟如下：

- 步驟 1：紀錄機器故障機台與發生時間點，故障時間點上正在處理的作業為中斷作業。
- 步驟 2：預估機器修護時間，即為機台當機狀態持續時間。機台修護時間點則為機台生故障點加上機台維修時間。
- 步驟 3：將中斷作業所剩餘加工時間加上機台修護時間作為中斷作業完工時間。
- 步驟 4：同樣於故障機器上加工的其他作業依照中斷作業完工時間評估作業開工時間，若中斷作業完工時間已超過下一個作業的開工時間，則下一個作業開工時間則右移至中斷作業完工時間，以此方式，逐層推估其他作業開工時間。舉一例子：若中斷作業完工時間為 15，而下一個作業的原先開工時間為 12，則開工時間必須延至 15，推算此作業的完工時間，再評估下一個作業開工時間，以此類推至在該機器上的最後一個加工作業。
- 步驟 5：紀錄受到變動的作業，並依訂單別去判斷其受到波動作業的該筆訂單下一個後續作業的開工時間是否受到影響，若受到影響則進行調整，調整方式如步驟 4；若否，則維持原先後續作業開工時程。
- 步驟 6：以步驟 5 的方式，判斷全部訂單受到波動的加工作業，並進行時程右移調整。
- 步驟 7：完成中斷測試，並紀錄新的排程規劃結果。
- 步驟 8：計算新排程結果的多目標適應函數值。

中斷測試：物料短缺

在一般的工廠中，缺料也是常見造成現場中斷的因素，雖然缺料與當機對於現場的影響都是立即性，但是以缺料而言，其直接影響的是該缺料訂單的作業以及其後續途程的作業，而其他作業仍可以進行不會受到直接影響。缺料發生時的調整方式是先取消缺料的訂單，等到該缺料的物料或零件到達時再將其重新插入排程。

物料短缺中斷測試步驟如下：

- 步驟 1：紀錄發現物料短缺的時間點與缺料作業代號。

- 步驟 2：預估物料補貨時間，即為缺料狀態持續時間。物料到達時間點則為發現物料短缺時間點加上物料補貨時間。
- 步驟 3：缺料作業所加工的途程工作站上，於物料到達時間點之後，找尋第一段機器閒置時間，並將此閒置時間的開始點作為缺料作業開工時間，並推估缺料作業完工時間。
- 步驟 4：判斷與缺料作業加工途程站相同的後一個作業開工時程是否受到缺料作業完工時間影響，若缺料作業完工時間已超過下一個作業的開工時間，則下一個作業開工時間則右移至缺料作業完工時間，以此方式，逐層推估其他後續作業開工時間。舉一例子：若缺料作業完工時間為 15，而下一個作業的原先開工時間為 12，則開工時間必須延至 15，推算此作業的完工時間，再評估下一個作業開工時間，以此類推至在該機器上的最後一個加工作業。
- 步驟 5：紀錄受到變動的作業，並依訂單別去判斷其受到波動作業的該筆訂單下一個後續作業的開工時間是否受到影響，若受到影響則進行調整，調整方式如步驟 4；若否，則維持原先後續作業開工時程。
- 步驟 6：以步驟 5 的方式，判斷全部訂單受到波動的加工作業，並進行時程右移調整。
- 步驟 7：完成中斷測試，並紀錄新的排程規劃結果。
- 步驟 8：計算新排程結果的多目標適應函數值。

中斷測試：緊急插單

所謂緊急插單指的是該張訂單原先並未安排在本次的排程週期內生產，而因為某些原因必須要優先處理該張訂單，盡快安排生產其所指定之產品的相關作業。這些原因可能是訂單來自於交易頻繁的顧客，基於顧客關係給予優先處理、或是承接該張訂單將有額外的收益、訂單的數量或金額達到公司所認定的標準，給予優先生產的權利。緊急插單發生時，排程規劃人員必須重新評定該單的質性因素，決定此插單的重要程度，再進行排程規劃。

- 步驟 1：紀錄得知緊急訂單時間與訂單加工資訊、交期。
- 步驟 2：以 AHP 法，重新評估質性限制的訂單優先順序與各訂單權重值。以預先規劃排程結果的剩餘產能作為緊急訂單的可用產能。

- 步驟 3：以預先規劃排程結果的剩餘產能作為緊急訂單的可用產能。
- 步驟 4：依得知緊急訂單時間點之後的各機器閒置時間來安插緊急訂單作業。
- 步驟 5：依訂單作業順序將所有作業插入機器閒置時間。
- 步驟 6：完成中斷測試，並紀錄含緊急訂單的新排程規劃結果。
- 步驟 7：計算新排程結果的多目標適應函數值。

實際生產環境的中斷事件非常多，排程規劃人員可依照其需求，將日常生產活動中最嚴重或最常發生的中斷事件，納入測試性干擾機制中，在排程求解過程中來測試各個體是否對於此中斷事件能有好的穩健與穩定性表現，最後求得一個具有穩健性的排程規劃，可更實用於真實環境中。

第四章 系統實證

本章節以零工式工廠的排程問題為例，依據第三章所述的系統架構來進行驗證。首先於 4.1 節中說明問題實證一之排程環境，並說明質性限制資源評估與各績效指標權重值訂定過程，以三種方法：傳統遺傳演算法 (GA)、具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 進行求解，最後針對結果探討與分析。4.2 節說明問題實證二之排程環境，亦同以上述三方法求解，再進行果探討與分析。4.3 節說明上述兩種實證問題中加入測試性干擾機制，求得穩健排程之步驟結果，並驗證穩健排程的效用。4.4 小節進行不同質性與量性瓶頸權重組合的實驗結果並分析。4.5 小節說明本章節的結論。

4.1 Job shop 排程實證一

於實證一中，Job shop 排程環境假設為：現場有 4 個不同工作中心站，各工作中心站僅有一台機器，本次排程週期中有 5 張訂單，每一訂單均代表一種產品，每張訂單的加工作業數並不相同；而訂單交期為一線性梯形模糊數；在此實證情境中，非瓶頸資源配合度的歸屬函數亦是以一梯形模糊數表示，假設模糊數為 $(0, 2, 7, 13)$ ，表示理想的非瓶頸作業與瓶頸作業銜接時間差異為 2 至 7 個時間單位，在此範圍內，表示兩作業銜接程度極佳，有一定的緩衝空間，且不致將製距拉長，排程具有彈性，具一定吸收波動能力，維持瓶頸作業不受中斷干擾，0 為此時間差異之上界，當該值為 0 時，表示兩作業緊密銜接，無任何延遲，13 為此時間差異之下界，當該值超越 13 時，則兩作業銜接程度過於鬆散，配合度極差。實證一的訂單資料如表 4.1 所示：

表 4.1 情境一的訂單資料

	加工途程工作站-(加工時間)				交期模糊數			
訂單 1	3-(12)	4-(6)	2-(25)		43	56	61	76
訂單 2	1-(14)	3-(24)			36	45	53	67
訂單 3	2-(13)	4-(19)	3-(7)		44	63	71	88
訂單 4	1-(18)	3-(14)	4-(5)	3-(20)	141	162	176	198
訂單 5	4-(9)	1-(12)	2-(17)		44	63	79	91

由表 4.1 中，計算各工作站產能負荷，如表 4.2，其中產能負荷最高的工作中心站 3，設定為瓶頸工作站，所有在瓶頸工作站進行的作業，即為瓶頸作業，但由於訂單 4 中，發生迴流 (re-entrant) 情況，將迴流中的所有作業 (即作業 4-3) 亦視為瓶頸作業，而訂單 5 沒有任何作業在工作中心站 3 進行加工，故訂單 5 的全部加工作業則設定為未佔用作業，其他作業則為非瓶頸作業。在投料至瓶頸作業間的非瓶頸作業稱之非瓶頸投料作業，而瓶頸作業至完工出貨的作業稱之非瓶頸交貨作業。再由瓶頸作業分別往前與向後推估各個非瓶頸作業隸屬於第幾層，作為逐層規劃排程的依據，作業分群資訊如表 4.3。

表 4.2 各工作站產能負荷表

工作中心站編號	1	2	3	4
產能負荷總量	45	55	57	39

表 4.3 作業分群表

訂單別	作業代碼 (作業分群種類)			
訂單 1	1-1 (瓶頸)	1-2 (非)	1-3 (非)	
訂單 2	2-1 (非)	2-2 (瓶頸)		
訂單 3	3-1 (非)	3-2 (非)	3-3 (瓶頸)	
訂單 4	4-1 (非)	4-2 (瓶頸)	4-3 (瓶頸)	4-4 (瓶頸)
訂單 5	5-1 (未)	5-2 (未)	5-3 (未)	
註：(瓶頸)，表示該作業設定為瓶頸作業 (非)，表示該作業設定為非瓶頸作業 (未)，表示該作業設定為未佔用作業				
瓶頸作業群組細分				
作業群組	作業代碼			
瓶頸作業	1-1、2-2、3-3、4-2、4-3、4-4			
非瓶頸投料作業 (第一層)	2-1、3-2、4-1			
非瓶頸投料作業 (第二層)	3-1			
非瓶頸交貨作業 (第一層)	1-2			
非瓶頸交貨作業 (第二層)	1-3			
未佔用作業	5-1、5-2、5-3			

實證問題於求解過程中，必須確定所使用演算法進行時的多目標限制適應性函數，此多目標限制適應性函數由質性限制資源與量性限制資源評估求得，於下做一簡短介紹與結果說明。

1. 質性限制資源評估－質性限制懲罰函數

在質性限制懲罰函數，依照本章 3.1.2 小節所述之架構，在實證一中，考量訂單本身的利潤、該訂單顧客以往的歷史交易、該訂單產品在市場上的因素以及該訂單顧客未來的潛在訂單四項因素，利用 AHP 法求得各質性限制資源的權重，如表 4.4。

再由規劃人員依據各質性限制來對訂單作兩兩配對比較。如表 4.5 至表 4.8。最後計算各訂單在質性限制下的加權值，求出質性限制之訂單順序，如表 4.9。所求得質性限制的訂單順序與經過遺傳演算法的訂單作業解碼順序轉換後，可算出其與質性訂單順序的最大差異，並以其作為懲罰函數值。

表 4.4 質性限制之權重值

質性限制	訂單利潤	歷史交易	市場因素	潛在訂單	$\lambda = 4.1315$	評估值
訂單利潤	1	3	2	5	0.8028	0.47
歷史交易	1/3	1	1/3	3	0.2694	0.16
市場因素	1/2	3	1	3	0.5153	0.30
潛在訂單	1/5	1/3	1/3	1	0.1348	0.07
CI=0.0438 ; CR=0.487 符合一致性檢定						

表 4.5 實證一的質性限制(一)－訂單利潤

利潤	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	$\lambda = 0.0782$	評估值
訂單 1	1	2	1	1/2	3	0.4475	0.2195
訂單 2	1/2	1	1/2	1/3	2	0.2495	0.1227
訂單 3	1	2	1	1	4	0.5416	0.2665
訂單 4	2	3	1	1	3	0.6486	0.3160
訂單 5	1/3	1/2	1/4	1/3	1	0.1534	0.0753
C.I.=0.0196 ; C.R.=0.0175 符合一致性檢定							

表 4.6 實證一的質性限制(二)－歷史交易

歷史交易	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	$\lambda=5.2097$	評估值
訂單 1	1	1	1/2	1/2	1	0.3072	0.1451
訂單 2	1	1	1	1	3	0.5080	0.2378
訂單 3	2	1	1	2	3	0.6447	0.3027
訂單 4	2	1	1/2	1	1	0.4051	0.1919
訂單 5	1	1/3	1/3	1	1	0.2605	0.1225
C.I.=0.0524 ; C.R.=0.0468 符合一致性檢定							

表 4.7 實證一的質性限制(三)－市場因素

市場因素	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	$\lambda=5.1064$	評估值
訂單 1	1	3	1/2	1/4	4	0.2898	0.1633
訂單 2	1/3	1	1/3	1/5	2	0.1413	0.0806
訂單 3	2	3	1	1/2	6	0.4645	0.2585
訂單 4	4	5	2	1	7	0.8207	0.4511
訂單 5	1/4	1/2	1/6	1/7	1	0.0822	0.0466
C.I.=0.0266 ; C.R.=0.0238 符合一致性檢定							

表 4.8 實證一的質性限制(四)－潛在訂單

潛在訂單	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	$\lambda=5.1006$	評估值
訂單 1	1	1/4	1	1/2	1/2	0.1990	0.1000
訂單 2	4	1	3	2	2	0.7519	0.3779
訂單 3	1	1/3	1	1	1/3	0.2247	0.1142
訂單 4	2	1/2	1	1	1/2	0.3047	0.1548
訂單 5	2	1/2	3	2	1	0.5018	0.2531
C.I.=0.0252 ; C.R.=0.0225 符合一致性檢定							

表 4.9 實證一的質性限制之訂單順序

	利潤	歷史	市場	潛在	總加權	排序
權重	0.47	0.16	0.3	0.07		
訂單 1	0.2195	0.1451	0.1633	0.1000	0.1824	3
訂單 2	0.1227	0.2378	0.0806	0.3779	0.1464	4
訂單 3	0.2665	0.3027	0.2585	0.1142	0.2592	2
訂單 4	0.3160	0.1919	0.4511	0.1548	0.3254	1
訂單 5	0.0753	0.1225	0.0466	0.2531	0.0867	5

2. 量性限制資源評估—多目標適應函數

關於量性資源評估，本研究將量性限制資源設定為產能負荷最高之工作站，由於量性限制資源影響到的排程績效種類諸多，本研究依 3.1.3 小節所說明之架構，以製距、瓶頸工作站使用率、交期滿足度、非瓶頸資源配合度等四個目標，由規劃人員評判各目標之重要性做配對比較，將此四項衡量指標轉換為單一多目標函數，各目標間的權重藉由 AHP 法則來制定，其中經由 AHP 法則制定的各目標之權重值如下表 4.10。

表 4.10 量性限制之權重值

量性限制	製距	達交率	使用率	配合度	$\lambda=4.0458$	評估值
製距	1	1/3	1/2	1/2	0.2129	0.1202
達交率	3	1	3	2	0.8201	0.4566
使用率	2	1/3	1	1	0.3583	0.2020
配合度	2	1/2	1	1	0.3921	0.2212
C.I.=0.0153 ; C.R.=0.0170 符合一致性檢定						

3. 多目標適應性函數

多目標適應性函數中包含質性與量性限制資源考量。質性限制資源考量是由規劃人員在訂單利潤、顧客以往的交易記錄、市場因素、潛在訂單利潤等四項質性限制下，評判各訂單對企業營運績效影響，在實證問題一中，求出質性限制之訂單順序為訂單 4=>訂單 3=>訂單 1=>訂單 2=>訂單 5。

量性限制資源考量現場產能負荷，以製距、瓶頸工作站使用率、交期滿足度、非瓶頸資源配合度等四個目標，作為瓶頸資源規劃依據指標，由規劃人員評判各量性目標之重要性做配對比較，藉由 AHP 法則制定各目標間的權重，將此四項衡量指標轉換為單一多目標函數（表 4.10 所得）。

最後再評估量性限制資源、質性限制資源，並制定權重，如表 4.11。

表 4.11 質性與量性限制之權重值

	質性限制	量性限制	$\lambda=2$	評估值
質性限制	1	1/5	0.1961	0.1667
量性限制	5	1	0.9806	0.8333
C.I.=0 符合一致性檢定				

由表 4.9、表 4.10、表 4.11 與式 3.7 可得，本研究排程系統進行遺傳演算法或混合式遺傳演算法所使用的多目標限制適應性函數如下：

$$\text{Fitness} = \{0.8333 \times [0.1202 \times \text{製距} + 0.4566 \times \text{交期滿足度} + 0.2020 \times \text{機器使用率} + 0.2212 \times \text{非瓶頸資源配合度}] + \{0.1667 \times (1 - \text{質性懲罰函數})\}$$

式 4.1

4. 排程演算法參數設定

本研究所使用的排程演算法，包含傳統遺傳演算法 (GA)、具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 三種，其中關於這三種方法的基本參數設定方面，本研究將遺傳演算法之母體大小設定為 20 條個體，個體中的基因編碼同 3.2.2 節中所提及之架構。其它相關設定如表 4.12、表 4.13 所示。

表 4.12 遺傳演算法(GA)中參數設定值

參數	參數值
群體大小	20
演算代數	300
交配方式	Random key representation
上半部基因突變機率	0.01
下半部基因突變機率	0.01

表 4.13 DBR_GA 與 DBR_HGA 參數設定值

參數	參數值
群體大小	20
演算代數	300
交配方式	Random key representation
上半部基因突變機率	0.01
下半部基因突變機率	0.01
禁忌搜尋次數	4
禁忌串列大小	3

4.1.1 Job shop 排程實證一之求解結果

以表 4.12、表 4.13 進行參數設定，將傳統遺傳演算法 (GA)、具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 三種方法代入 4.1 小節假設的 Job shop 問題實證一求解。

利用三種方法所求解出的最高適應函數值為 0.936797，其中 DBR_GA 與 DBR_HGA 兩種方法曾求得此解。此最適排程結果如下表 4.14，製距為 83，搜尋過程中最短之製距為 79，將個體的製距正規化為 0.9518，各訂單的交期滿足度如表 4.15，除訂單 2 為 0.6667 之外，其他訂單皆在理想交期時間內完工，得一平均訂單達交率為 0.9333。

而在實證一的訂單資料中，由於工作站 3 為瓶頸工作站，其中訂單 1 第一個途程作業即為瓶頸作業，而訂單 5 沒有任何作業於瓶頸工作站加工，故於實證一中，僅考慮訂單 2、3、4 之第一個瓶頸作業與先前一個的非瓶頸作業加工時程之銜接時間差距，作為評估非瓶頸資源配合度之依據，非瓶頸資源配合度如表 4.16 所示，得一非瓶頸資源平均配合度為 1。而工作站 3 之排程甘特圖如圖 4.1 所示，可得瓶頸工作站使用率為： $(83 - (18 - 12)) \div 83 = 0.9277$ 。

表 4.14 實證一的最適排程結果

作業加工時程表					
訂單別		途程 1	途程 2	途程 3	途程 4
訂單 1	開工時間	0	12	31	
	完工時間	12	18	56	
訂單 2	開工時間	0	18		
	完工時間	14	42		
訂單 3	開工時間	18	31	56	
	完工時間	31	50	63	
訂單 4	開工時間	20	42	56	63
	完工時間	38	56	61	83
訂單 5	開工時間	0	42	56	
	完工時間	9	54	73	

表 4.15 實證一的訂單達交結果

訂單達交率						
訂單別	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	平均達交率
訂單達交率	1	0.667	1	1	1	0.9333

表 4.16 實證一的非瓶頸資源配合度結果

訂單別	第一層非瓶頸 作業之代碼	第一個瓶頸 作業之代碼	時間差異	非瓶頸資源 配合度	非瓶頸資源 平均配合度
	完工時間	開工時間			
訂單 2	2-1	2-2	4 (18-14)	1	1
	14	18			
訂單 3	3-2	3-3	6 (56-50)	1	
	50	56			
訂單 4	4-3	4-4	2 (63-61)	1	
	61	63			

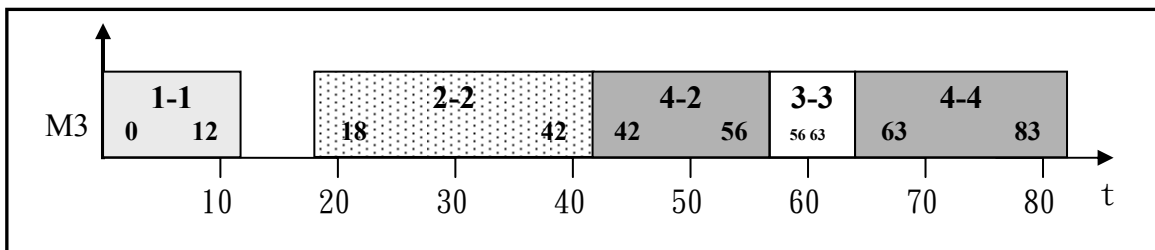


圖 4.1 瓶頸工作站之排程甘特圖

根據本次個體的染色體資訊，解碼順序按瓶頸作業、非瓶頸投料作業各層、非瓶頸投交貨作業各層，最後為未佔用作業。首先，瓶頸作業集合有{1-1、2-2、3-3、4-2、4-3、4-4}，而各自基因碼為{6、8、10、9、13、15}，依基因碼大小排序作為解碼順序，可得{6、8、9、10、13、15}，故瓶頸作業解碼順序為{1-1、2-2、4-2、3-3、4-3、4-4}；另外非瓶頸投料作業第一層的作業集合有{2-1、3-2、4-1}，基因碼為{7、5、3}，依基因碼大小排序，可得{3、5、7}，故非瓶頸投料作業第一層解碼順序為{4-1、3-2、2-1}，以此逐層類推非瓶頸投料作業與非瓶頸投料交貨作業之解碼順序，

最後再制定未佔用作業解碼順序。

全部作業解碼順序、各訂單的平均解碼順序與訂單排序結果如表 4.17 與表 4.18。

表 4.17 全部作業解碼順序

作業分群級別	瓶頸作業						非瓶頸投料作業 (第一層)		
	1	2	3	4	5	6	7	8	9
解碼順序	1	2	3	4	5	6	7	8	9
作業代碼	1-1	2-2	4-2	3-3	4-3	4-4	4-1	3-2	2-1
基因代碼	6	8	9	10	13	15	3	5	7
作業分群級別	非瓶頸投料作業 (第二層)		非瓶頸投料作業 (第一層)		非瓶頸投料作業 (第二層)		未佔用作業		
	10	11	12	13	14	15			
解碼順序	10	11	12	13	14	15			
作業代碼	3-1	1-2	1-3	5-1	5-2	5-3			
基因代碼	4	11	14	1	2	12			

表 4.18 各訂單的作業解碼順序平均與排序

訂單別	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5
作業解碼順序總和	24	11	22	21	42
作業解碼順序平均值	8	5.5	7.333	5.25	14
排序	4	2	3	1	5

依照質性限制訂單順序，找出最大違反質性限制的訂單順序，以求出最大差異平方和。找出的最大違反訂單順序為：訂單 5→訂單 2→訂單 1→訂單 3→訂單 4，以此最大違反的訂單順序與質性限制訂單順序，求出加權後最大差異平方的總和為 8.216；而遺傳演算法求解之解碼順序為：訂單 4→訂單 2→訂單 3→訂單 1→訂單 5，與質性限制訂單順序求出加權後差異平方的總和為 1.0272，計算方式如表 4.19。將加權後差異平方總和除以加權後最大差異平方總和，以正規化後，所得懲罰函數值如下： $1.0272 / 8.216 = 0.125$ 。

根據式 4.1，可計算出總目標函數： $0.8333 \times \{(0.1202 \times (79/83)) + (0.4566 \times 0.93333) + (0.2020 \times 0.9277) + (0.2212 \times 1)\} + \{0.1667 \times (1 - 0.125)\} = 0.936797$ 。

表 4.19 加權後差異平方總和之計算

訂單編號	1	2	3	4	5	最大加權 差異平方 總和
訂單加權值	0.1824	0.1464	0.2592	0.3254	0.0867	
質性限制訂單順序	3	4	2	1	5	
最大違反訂單順序	3	2	4	5	1	
最大違反差異平方	0	4	4	16	16	
加權差異平方	0	0.5856	1.0368	5.2064	1.3872	8.216
訂單編號	1	2	3	4	5	加權差異 平方總和
質性限制訂單順序	3	4	2	1	5	
解碼之訂單順序	4	2	3	1	5	
排序差異平方	1	4	1	0	0	
加權差異平方	0.1824	0.5856	0.2592	0	0	

DBR_GA 與 DBR_HGA 兩種方法求得最高適應函數值為 0.9368，兩種演算法之求解過程其適應函數值收斂過程如圖 4.2 與圖 4.3：

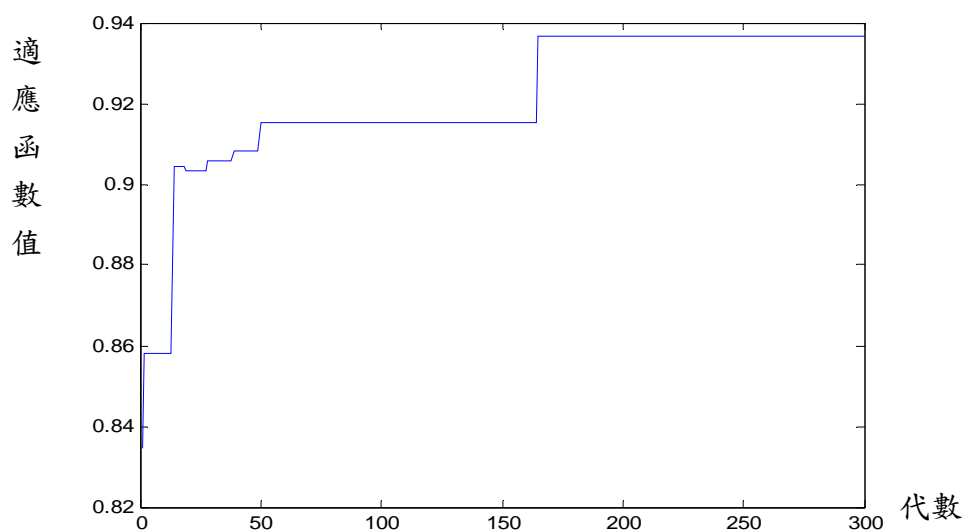


圖 4.2 實證一之 DBR_GA 適應函數值趨勢圖

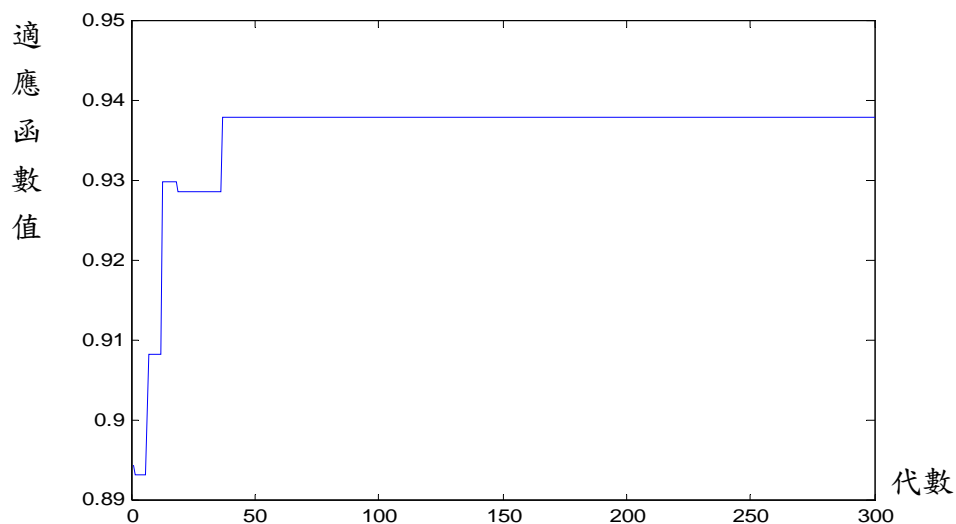


圖 4.3 實證一之 DBR_HGA 適應函數值趨勢圖

4.1.2 實證一結果探討與分析

以表 4.12、表 4.13 參數設定求解，將上述的 GA、DBR_GA 與 DBR_HGA 三種方法分別代入 4.1 小節假設的實證問題求解，每種方法各實驗 30 次，其結果顯示如表 4.20。

表 4.20 實證一以三種方法求解之數據表

實證一：小型問題					
傳統式 GA		DBR_GA		DBR_HGA	
實驗回數	最佳解	實驗回數	最佳解	實驗回數	最佳解
1	0.818777	1	0.927404	1	0.936797
2	0.772747	2	0.918087	2	0.936797
3	0.814584	3	0.927404	3	0.936797
4	0.784162	4	0.922425	4	0.927404
5	0.793129	5	0.936797	5	0.936797
6	0.770223	6	0.90949	6	0.927404
7	0.787175	7	0.929576	7	0.936797
8	0.800037	8	0.936797	8	0.936797
9	0.802783	9	0.929576	9	0.936797
10	0.792815	10	0.927404	10	0.936797
11	0.822866	11	0.936797	11	0.927404
12	0.800037	12	0.915343	12	0.936797
13	0.815644	13	0.921551	13	0.936797
14	0.804155	14	0.936797	14	0.936797
15	0.79859	15	0.936797	15	0.936797
16	0.802508	16	0.927404	16	0.936797
17	0.789181	17	0.936797	17	0.936797

實證一：小型問題					
傳統式 GA		DBR GA		DBR HGA	
實驗回數	最佳解	實驗回數	最佳解	實驗回數	最佳解
18	0.800611	18	0.930944	18	0.936797
19	0.793986	19	0.936797	19	0.936797
20	0.793555	20	0.936797	20	0.936797
21	0.781401	21	0.915343	21	0.936797
22	0.788797	22	0.936797	22	0.936797
23	0.791293	23	0.922425	23	0.936797
24	0.772011	24	0.915343	24	0.936797
25	0.811451	25	0.922425	25	0.936797
26	0.811626	26	0.915343	26	0.936797
27	0.811451	27	0.929576	27	0.936797
28	0.804155	28	0.915343	28	0.936797
29	0.80141	29	0.936797	29	0.936797
30	0.815644	30	0.918087	30	0.936797
最大值	0.822866	最大值	0.936797	最大值	0.936797
平均值	0.798227	平均值	0.926949	平均值	0.935858
標準差	0.013899	標準差	0.008776	標準差	0.002866

於實證一中，以三種方法求解出的最適解為 0.936797，其中 DBR_GA 與 DBR_HGA 皆曾求解出此最適解，並由表 4.20 中可以看出以具遺傳演算法機制之限制驅導技術(DBR_GA)或具混合式遺傳演算法機制之限制驅導技術(DBR_HGA)皆可求出比傳統型遺傳演算法較高的排程結果，以下針對三種方法，根據表 4.20 進行統計檢定：

1. 檢定 DBR_GA 法與 GA 方法求解品質：

$$H_0: \mu_{DBR_GA} \leq \mu_{GA}$$

$$H_1: \mu_{DBR_GA} > \mu_{GA}$$

以 $\alpha = 0.05$ 為顯著水準，

$$t = \frac{\mu_{DBR_GA} - \mu_{GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_GA}^2 + (30-1) \cdot s_{GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30}\right)}} = \frac{0.926949 - 0.798227}{\sqrt{((0.008776)^2 + (0.013899)^2) \cdot \frac{29}{58} \cdot \frac{2}{30}}} = 42.8914$$

因為 $|t| \geq t_{\alpha} = 1.69$ ，所以拒絕虛無假設。

得證：以 DBR_GA 求解問題，所求得解的平均值高於傳統 GA，由統計檢定結果得知 DBR_GA 比傳統 GA 於求解小型零工式排程問題較績效優異。

2. 檢定 DBR_HGA 法與 GA 方法求解品質：

$$H_0: \mu_{DBR_HGA} \leq \mu_{GA}$$

$$H_1: \mu_{DBR_HGA} > \mu_{GA} \quad \text{以 } \alpha = 0.05 \text{ 為顯著水準，}$$

$$t = \frac{\mu_{DBR_HGA} - \mu_{GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_HGA}^2 + (30-1) \cdot s_{GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30}\right)}} = \frac{0.935858 - 0.798227}{\sqrt{((0.002866)^2 + (0.013899)^2) \cdot \frac{29}{58} \cdot \frac{2}{30}}} = 53.1192$$

因為 $|t| \geq t\alpha = 1.69$ ，所以拒絕虛無假設。

得證：以 DBR_HGA 求解問題，所求得解的平均值高於傳統 GA，由統計檢定結果得知 DBR_HGA 比傳統 GA 於求解小型零工式排程問題績效較優異。

3. 檢定 DBR_HGA 法與 DBR_GA 方法求解品質：

$$H_0: \mu_{DBR_HGA} \leq \mu_{DBR_GA}$$

$$H_1: \mu_{DBR_HGA+TS} > \mu_{DBR_GA} \quad \text{以 } \alpha = 0.05 \text{ 為顯著水準，}$$

$$t = \frac{\mu_{DBR_HGA} - \mu_{DBR_GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_HGA}^2 + (30-1) \cdot s_{DBR_GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30}\right)}} = \frac{0.935858 - 0.926949}{\sqrt{((0.002866)^2 + (0.008776)^2) \cdot \frac{29}{58} \cdot \frac{2}{30}}} = 5.2855$$

因為 $|t| > t\alpha = 1.69$ ，拒絕虛無假設，

得證：以 DBR_HGA 求解問題，所求得解的平均值高於 DBR_GA，由統計檢定結果得知 DBR_HGA 比 DBR_GA 於求解小型零工式排程問題績效較優異。

從三次統計檢定中可發現，具遺傳演算法機制之限制驅導技術 (DBR_GA) 或具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 所求解的品質較傳統型遺傳演算法要來的高，不僅樣本平均數較高，且樣本

標準差亦小於傳統型遺傳演算法，其中又以限制驅導技術結合混合式遺傳演算法 (DBR_HGA) 的平均績效最好，經由統計檢定，可證明在實證一中 (小型零工式排程問題)，以 DBR_HGA 所求的解為最好。

4.2 Job shop 排程實證二

問題實證二中，Job shop 排程環境假設為：現場有 8 個不同的工作中心站，每個工作中心站僅有一台機器，本次排程週期中有 10 張訂單，每一訂單均代表一種產品；而訂單之交期為一線性梯形模糊數；在此情境中，非瓶頸資源配合度的歸屬函數以一梯形模糊數表示，設定模糊數為 (0, 2, 7, 13)。實證二的訂單資料如表 4.21。其中產能負荷最高的工作中心站 4，並設定工作站 4 為瓶頸工作站，將作業依照瓶頸作業、非瓶頸投料作業、非瓶頸交貨作業、未佔用作業四種類別進行分群，作業分群資訊如表 4.22。

表 4.21 實證二的訂單資料

訂單別	加工途程工作站-(加工時間)								交期模糊數			
訂單 1	5-(3)	6-(9)	1-(14)	4-(20)	2-(17)	8-(8)			163	181	206	215
訂單 2	7-(16)	6-(12)	3-(6)	4-(22)	1-(14)	7-(17)			154	166	189	211
訂單 3	3-(16)	7-(12)	4-(23)	5-(15)	6-(8)	2-(11)	8-(16)		165	176	192	220
訂單 4	6-(5)	7-(8)	5-(14)	3-(10)	2-(12)	1-(7)	4-(13)		141	162	176	198
訂單 5	4-(14)	1-(16)	2-(7)	5-(18)	7-(9)	6-(20)			157	169	181	196
訂單 6	3-(14)	5-(10)	8-(16)	5-(18)	7-(22)	2-(26)	4-(19)	3-(11)	163	172	183	190
訂單 7	5-(12)	4-(7)	3-(8)	1-(16)	2-(15)	5-(23)			140	153	161	174
訂單 8	1-(18)	8-(21)	6-(7)	4-(26)					167	179	191	206
訂單 9	8-(14)	2-(16)	4-(17)	3-(8)	6-(22)				148	163	176	190
訂單 10	4-(15)	8-(9)	2-(6)	1-(20)					99	113	126	145

表 4.22 實證二之作業分群資訊

訂單別	作業代碼-(分群群組代碼)						
訂單 1	1-1 (非)	1-2 (非)	1-3 (非)	1-4 (瓶頸)	1-5 (非)	1-6 (非)	
訂單 2	2-1 (非)	2-2 (非)	2-3 (非)	2-4 (瓶頸)	2-5 (非)	2-6 (非)	

訂單別	作業代碼-(分群群組代碼)							
訂單 3	3-1 (非)	3-2 (非)	3-3 (瓶頸)	3-4 (非)	3-5 (非)	3-6 (非)	3-7 (非)	
訂單 4	4-1 (非)	4-2 (非)	4-3 (非)	4-4 (非)	4-5 (非)	4-6 (非)	4-7 (瓶頸)	
訂單 5	5-1 (瓶頸)	5-2 (非)	5-3 (非)	5-4 (非)	5-5 (非)	5-6 (非)		
訂單 6	6-1 (非)	6-2 (非)	6-3 (非)	6-4 (非)	6-5 (非)	6-6 (非)	6-7 (瓶頸)	6-8 (非)
訂單 7	7-1 (非)	7-2 (瓶頸)	7-3 (非)	7-4 (非)	7-5 (非)	7-6 (非)		
訂單 8	8-1 (非)	8-2 (非)	8-3 (非)	8-4 (瓶頸)				
訂單 9	9-1 (非)	9-2 (非)	9-3 (瓶頸)	9-4 (非)	9-5 (非)			
訂單 10	10-1 (瓶頸)	10-2 (非)	10-3 (非)	10-4 (非)				
瓶頸作業群組細分								
作業群組		作業代碼						
瓶頸作業		1-4、2-4、3-3、4-7、5-1、6-7、7-2、8-4、9-3、10-1						
非瓶頸投料作業 (第一層)		1-3、2-3、3-2、4-6、6-6、7-1、8-3、9-2						
非瓶頸投料作業 (第二層)		1-2、2-2、3-1、4-5、6-5、8-2、9-1、						
非瓶頸投料作業 (第三層)		1-1、2-1、4-4、6-4、8-1						
非瓶頸投料作業 (第四層)		4-3、6-3						
非瓶頸投料作業 (第五層)		4-2、6-2						
非瓶頸投料作業 (第六層)		4-1、6-1						
非瓶頸交貨作業 (第一層)		1-5、2-5、3-4、5-2、6-8、7-3、9-4、10-2						
非瓶頸交貨作業 (第二層)		1-6、2-6、3-5、5-3、7-4、9-5、10-3						
非瓶頸交貨作業 (第三層)		3-6、5-4、7-5、10-4						
非瓶頸交貨作業 (第四層)		3-7、5-5、7-6						
非瓶頸交貨作業 (第五層)		5-6						

實證二之求解步驟同 4.1 小節所述之流程，以下說明多目標限制適應函數過程。

1. 質性限制資源評估－質性限制懲罰函數

本章實證二中，各質性限制對訂單作兩兩配對比較。如表 4.23 至表 4.26。最後計算各訂單在質性限制下的加權值，求出質性限制之訂單順序，如表 4.27。所求得質性限制的訂單順序與經過遺傳演算法的訂單作業編碼轉換後，可算出其與質性訂單順序的最大差異，並以其作為懲罰函數值。

在實證問題二中，質性限制下之訂單順序為 訂單 4=> 訂單 1=> 訂單 3=> 訂單 9=> 訂單 8=> 訂單 7=> 訂單 6=> 訂單 2=> 訂單 5=> 訂單 10 (表 4.27 所得)。

表 4.23 實證二的質性限制(一)－訂單利潤之權重值

利潤	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	$\lambda = 10.3197$	評估值
訂單1	1	5	1	1/2	5	4	4	3	2	7	0.4295	0.1727
訂單2	1/5	1	1/4	1/7	2	1	1/2	1/3	1/4	3	0.0936	0.0393
訂單3	1	4	1	1/2	7	4	3	3	2	7	0.4199	0.1697
訂單4	2	7	2	1	7	7	5	5	3	8	0.6882	0.2767
訂單5	1/5	1/2	1/7	1/7	1	1/2	1/3	1/4	1/4	2	0.0650	0.0272
訂單6	1/4	1	1/4	1/7	2	1	1	1/2	1/4	4	0.1005	0.0414
訂單7	1/4	2	1/3	1/5	3	1	1	1/2	1/3	4	0.1350	0.0564
訂單8	1/3	3	1/3	1/5	4	2	2	1	1/2	4	0.1907	0.0781
訂單9	1/2	4	1/2	1/3	4	4	3	2	1	5	0.2923	0.1181
訂單10	1/7	1/3	1/7	1/8	1/2	1/4	14	1/4	1/5	1	0.0495	0.0204
C.I.= 0.0355 ; C.R.= 0.0238 符合一致性檢定												

表 4.24 實證二的質性限制(二)－歷史交易

歷史交易	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	$\lambda = 10.3237$	評估值
訂單1	1	2	1/3	1/4	2	3	1/3	3	1/2	4	0.2215	0.0852
訂單2	1/2	1	1/3	1/4	1	2	1/3	2	1/2	3	0.1554	0.0598
訂單3	3	3	1	1/2	3	4	1	5	2	5	0.4423	0.17
訂單4	4	4	2	1	4	4	2	5	3	5	0.638	0.2454
訂單5	1/2	1	1/3	1/4	1	2	1/3	2	1/2	3	0.1554	0.0598
訂單6	1/3	1/2	1/4	1/4	1/2	1	1/4	2	1/3	2	0.1082	0.0416
訂單7	3	3	1	1/2	3	4	1	5	2	5	0.4423	0.17
訂單8	1/3	1/2	1/5	1/5	1/2	1/2	1/5	1	1/4	2	0.0851	0.0328
訂單9	2	2	1/2	1/3	2	3	1/2	4	1	4	0.2845	0.1094
訂單10	1/4	1/3	1/5	1/5	1/3	1/2	1/5	1/2	1/4	1	0.0676	0.026
C.I.=0.0371 ; C.R.=0.0249 符合一致性檢定												

表 4.25 實證二的質性限制(三)－市場因素

市場考量	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	$\lambda = 10.5937$	評估值
訂單1	1	4	3	2	5	3	4	1/2	2	5	0.4753	0.1839
訂單2	1/4	1	1/2	1/3	3	1/2	2	1/5	1/3	2	0.1284	0.0497
訂單3	1/3	2	1	1/3	4	1/2	3	1/4	1/2	3	0.1841	0.0712
訂單4	1/2	3	3	1	5	2	4	1/2	2	4	0.3751	0.1452
訂單5	1/5	1/3	1/4	1/5	1	1/4	1/2	1/7	1/4	1/2	0.0599	0.0232
訂單6	1/3	2	2	1/2	4	1	3	1/4	1/2	4	0.2269	0.0878
訂單7	1/4	1/2	1/3	1/4	2	1/3	1	1/5	1/4	2	0.096	0.0372
訂單8	2	5	4	2	7	4	5	1	3	5	0.65	0.2515
訂單9	1/2	3	2	1/2	4	2	4	1/3	1	4	0.296	0.1145
訂單10	1/5	1/2	1/3	1/4	2	1/4	2	1/5	1/4	1	0.0924	0.0358
C.I.=0.0486 ; C.R.=0.0326 符合一致性檢定												

表 4.26 實證二的質性限制(四)－潛在訂單

潛在訂單	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	$\lambda = 10.5526$	評估值
訂單1	1	1/2	1/4	1/5	1/2	1/3	1/4	2	1/4	3	0.1076	0.0414
訂單2	2	1	1/3	1/5	1	1/5	1/3	2	1/4	3	0.1456	0.056
訂單3	4	3	1	1/2	3	2	1	4	1/2	5	0.3508	0.1349
訂單4	5	5	2	1	5	3	2	5	2	7	0.625	0.2403
訂單5	2	1	1/3	1/5	1	1/2	1/3	2	1/4	3	0.1403	0.054
訂單6	3	2	1/2	1/3	2	1	1/5	3	1/3	4	0.2259	0.0869
訂單7	4	3	1	1/2	3	2	1	4	1/5	5	0.3508	0.1349
訂單8	1/2	1/2	1/4	1/5	1/2	1/3	1/4	1	1/5	2	0.0853	0.0328
訂單9	4	4	2	1/2	4	3	2	5	1	5	0.4974	0.1913
訂單10	1/3	1/3	1/5	1/7	1/3	1/4	1/5	2	1/5	1	0.0716	0.0275
C.I.=0.0421 ; C.R.=0.0282 符合一致性檢定												

表 4.27 實證二的質性限制之訂單順序

	利潤	歷史交易	市場考量	潛在訂單	加權值	排序
權重	0.47	0.16	0.30	0.07		
訂單1	0.1727	0.0852	0.1839	0.0414	0.1529	2
訂單2	0.0393	0.0598	0.0497	0.056	0.0469	8
訂單3	0.1697	0.17	0.0712	0.1349	0.1378	3
訂單4	0.2767	0.2454	0.1452	0.2403	0.2297	1
訂單5	0.0272	0.0598	0.0232	0.054	0.0331	9
訂單6	0.0414	0.0416	0.0878	0.0869	0.0585	7
訂單7	0.0564	0.17	0.0372	0.1349	0.0743	6
訂單8	0.0781	0.0328	0.2515	0.0328	0.1197	5
訂單9	0.1181	0.1094	0.1145	0.1913	0.1208	4
訂單10	0.0204	0.026	0.0358	0.0275	0.0264	10

2. 量性限制資源評估—多目標適應函數

關於量性資源評估，同實證一之架構。各目標之權重如表 4.10。

3. 多目標適應性函數

在實證二中，所使用的多目標適應性函數為式 4.1。

4. 排程演算法參數設定

相關設定如表 4.12、表 4.13 所示。

4.2.1 Job shop 排程實證二之求解結果

以表 4.12、表 4.13 參數設定求解，將 4.1 小節所述之 GA、DBR_GA 與 DBR_HGA 三種方法代入 4.2 小節假設的 Job shop 問題實證二中進行求解。

在實證二中，以三種方法求解，所求解出的最高適應函數值為 0.8638，其中僅 DBR_HGA 方法曾求得此解，此解為問題實證二之最適排程。而排程結果、平均交期滿足度、非瓶頸資源配合率、質性懲罰值等資訊如表 4.28：

表 4.28 實證二之最適排程結果

各作業開工時程										
		途程	途程	途程	途程	途程	途程	途程	途程	途程
		1	2	3	4	5	6	7	8	
訂單 1	開工時間	72	75	84	102	157	174			
	完工時間	75	84	98	122	174	182			
訂單 2	開工時間	83	99	111	122	144	158			
	完工時間	99	111	117	144	158	175			
訂單 3	開工時間	1	17	36	105	120	174	185		
	完工時間	17	29	59	120	128	185	201		
訂單 4	開工時間	35	40	58	101	115	135	144		
	完工時間	40	58	72	111	127	142	157		
訂單 5	開工時間	15	29	45	120	138	152			
	完工時間	29	45	52	138	147	172			
訂單 6	開工時間	34	48	71	87	105	127	157	176	
	完工時間	48	58	87	105	127	153	176	187	
訂單 7	開工時間	14	29	48	56	85	138			
	完工時間	26	36	56	72	100	161			
訂單 8	開工時間	10	28	49	59					
	完工時間	28	49	56	85					
訂單 9	開工時間	51	65	85	122	130				
	完工時間	65	81	102	130	152				
訂單 10	開工時間	0	15	24	102					
	完工時間	15	24	30	122					
各訂單達交率										
達 交 率	訂單	訂單	訂單	訂單	訂單	訂單	訂單	訂單	訂單	訂單
	1	2	3	4	5	6	7	8	9	10
	1	1	0.678	0.762	1	0.428	1	0	0.266	1
製距：201 曾經搜尋最短製距：195					製距績效：0.9701					
瓶頸利用率：1					非瓶頸資源配合率：1					
平均訂單達交率：0.7136					質性懲罰值：0.1455					
總目標函數： $0.8333*((0.1202*(195/201))+(0.4566*0.7136)+(0.2020*1)+(0.2212*1))$ $+0.1667*(1-0.1455)=0.8638$										

在實證二中，僅 DBR_HGA 求解出的最高適應函數值 0.8638，其法求解過程之適應函數值趨勢如圖 4.4：

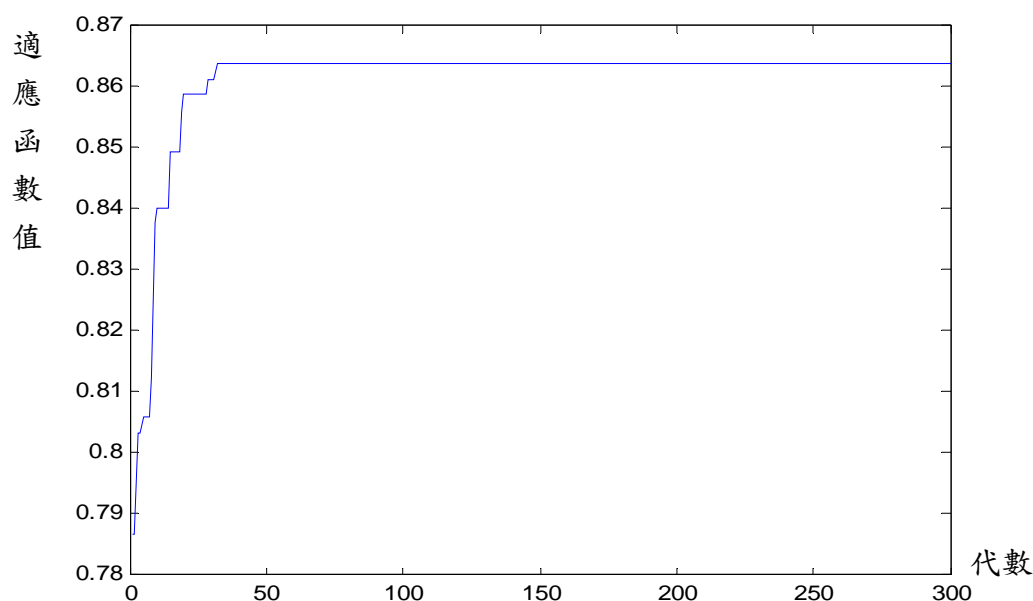


圖 4.4 實證二之 DBR_HGA 適應函數值趨勢圖

4.2.2 實證二結果探討與分析

以表 4.12、表 4.13 參數設定求解，將前述的 GA、DBR_GA 與 DBR_HGA 三種方法分別代入 4.2 小節假設的實證問題二求解，每種方法各實驗 30 次，其結果顯示如表 4.29。

表 4.29 實證二以三種方法求解之數據表

實證二：中型問題					
傳統式 GA		DBR_GA		混合式 DBR_HGA	
實驗回數	最佳解	實驗回數	最佳解	實驗回數	最佳解
1	0.668379	1	0.834054	1	0.863766
2	0.729585	2	0.832187	2	0.863766
3	0.697845	3	0.826088	3	0.863766
4	0.734776	4	0.826917	4	0.863436
5	0.670278	5	0.830182	5	0.863072
6	0.695667	6	0.820880	6	0.863766
7	0.729879	7	0.839514	7	0.863766
8	0.684258	8	0.843446	8	0.863766

實證二：中型問題					
傳統式 GA		DBR GA		混合式 DBR HGA	
實驗回數	最佳解	實驗回數	最佳解	實驗回數	最佳解
9	0.720410	9	0.833704	9	0.857133
10	0.682854	10	0.837495	10	0.863072
11	0.682852	11	0.821925	11	0.863766
12	0.700196	12	0.831108	12	0.863071
13	0.700120	13	0.821553	13	0.855082
14	0.705204	14	0.807757	14	0.863766
15	0.717708	15	0.845282	15	0.857133
16	0.751307	16	0.833997	16	0.859197
17	0.708261	17	0.813834	17	0.856862
18	0.676107	18	0.831964	18	0.857878
19	0.697856	19	0.826410	19	0.857133
20	0.717672	20	0.815216	20	0.863766
21	0.724589	21	0.842878	21	0.855082
22	0.667514	22	0.836023	22	0.863766
23	0.688490	23	0.841778	23	0.863766
24	0.702233	24	0.829625	24	0.859197
25	0.672675	25	0.827292	25	0.860839
26	0.736066	26	0.825426	26	0.863766
27	0.692530	27	0.831856	27	0.863766
28	0.689010	28	0.810149	28	0.863766
29	0.716719	29	0.811760	29	0.863766
30	0.677632	30	0.832356	30	0.862827
最大值	0.751307	最大值	0.845282	最大值	0.863766
平均值	0.701289	平均值	0.828755	平均值	0.861583
標準差	0.022604	標準差	0.010013	標準差	0.003079

於實證問題二中，以三種方法求解出的最適解為 0.863766，其中僅 DBR_HGA 方法曾求解出此最適解，並由表 4.29 中可看出具遺傳演算法機制之限制驅導技術(DBR_GA)或具混合式遺傳演算法機制之限制驅導技術(DBR_HGA)法皆可求出比傳統型遺傳演算法績效較高的排程結果，以下針對三種方法，根據表 4.29 進行統計檢定：

1. 檢定 DBR_GA 法與 GA 方法求解品質：

$$H_0 : \mu_{DBR_GA} \leq \mu_{GA}$$

$$H_1 : \mu_{DBR_GA} > \mu_{GA}$$

以 $\alpha = 0.05$ 為顯著水準，

$$t = \frac{\mu_{DBR_GA} - \mu_{GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_GA}^2 + (30-1) \cdot s_{GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30} \right)}} = 28.2399$$

$\therefore |t| \geq t\alpha = 1.69$ ，所以拒絕虛無假設。

得證：以 DBR_GA 求解問題，所求得解的平均值高於傳統 GA，界統計檢定結果得知 DBR_GA 比傳統 GA 於求解中型零工式排程問題較績效優異。

2. 檢定 DBR_HGA 法與 GA 方法求解品質：

$$H_0: \mu_{DBR_HGA} \leq \mu_{DBR_GA}$$

$$H_1: \mu_{DBR_HGA} > \mu_{DBR_GA}$$

$$t = \frac{\mu_{DBR_HGA} - \mu_{GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_HGA}^2 + (30-1) \cdot s_{GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30}\right)}} = \frac{0.861583 - 0.701289}{\sqrt{((0.003079)^2 + (0.022604)^2) \cdot \frac{29}{58} \cdot \frac{2}{30}}} = 38.4858$$

$\therefore |t| \geq t\alpha = 1.69$ ，所以拒絕虛無假設。

得證：以 DBR_HGA 求解問題，所求得解的平均值高於傳統 GA，界統計檢定結果得知 DBR_HGA 比傳統 GA 於求解中型零工式排程問題績效較優異。

3. 檢定 DBR_HGA 法與 DBR_GA 方法求解品質：

$$H_0: \mu_{DBR_HGA} \leq \mu_{DBR_GA}$$

$$H_1: \mu_{DBR_HGA} > \mu_{DBR_GA} \quad \text{以 } \alpha = 0.05 \text{ 為顯著水準}$$

$$t = \frac{\mu_{DBR_HGA} - \mu_{DBR_GA}}{\sqrt{\frac{(30-1) \cdot s_{DBR_HGA}^2 + (30-1) \cdot s_{DBR_GA}^2}{30+30-2} \left(\frac{1}{30} + \frac{1}{30}\right)}} = \frac{0.861583 - 0.828755}{\sqrt{((0.003079)^2 + (0.010013)^2) \cdot \frac{29}{58} \cdot \frac{2}{30}}} = 17.1641$$

$\therefore |t| \geq t\alpha = 1.69$ ，所以拒絕虛無假設。

得證：以 DBR_HGA 求解問題，所求得解的平均值高於 DBR_GA，界統計檢定結果得知 DBR_HGA 比 DBR_GA 於求解中型零工式排程問題績效較優異。

從三次統計檢定中可發現，具遺傳演算法機制之限制驅導技術 (DBR_GA) 或具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 所求解的品質較傳統型遺傳演算法要來的高，不僅樣本平均數較高，且樣本標準差亦小於傳統型遺傳演算法，其中又以具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 的平均績效最好，經由統計檢定，可證明在實證二 (中型零工式排程問題) 中，以 DBR_HGA 所求的解為最好。

4.3 中斷測試與穩健排程

在 4.1 與 4.2 兩小節中，建構兩種實證問題情境，並以傳統遺傳演算法 (GA)、具遺傳演算法機制之限制驅導技術 (DBR_GA) 或具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 等三種方法進行求解，得最適預先規劃排程結果。在本節中，本研究將分別於兩種實證情境中，以三方法所求解出的各前二名個體(共 6 個個體)，進行中斷測試，測試方式依照本章 3.3.1 小節中介紹的三種中斷情節，以下分別探討三種中斷測試結果與分析。

4.3.1 中斷測試：機台故障

本小節說明機台故障中斷測試，依照本章 3.3.1 小節方法，進行中斷後排程規劃結果調整。在兩種實證問題情境中，分別加入機台故障的情節，本例以隨機方式從所有工作中選取一站作為故障機台，以均勻分配隨機從此機台的工作時程中選取一時間點作為機台當機的時間點，當機情節將發生兩次。關於機台故障資訊如表 4.30 所示：

表 4.30 實證一與實證二之故障資訊

實證一：	故障情節 1	故障情節 2
故障機台	1	2
機台故障時間點	10	69
機台故障狀態持續時間 (機台維修時間)	7	10
實證二：	故障情節 1	故障情節 2
故障機台	3	2
機台故障時間點	15	103
機台故障狀態持續時間	5	116

於兩種實證情境問題中，挑出原先預排結果中 GA、DBR_GA 與 DBR_HGA 三方法中，各法績效最佳之前兩組個體進行故障測試干擾，干擾結果如下表 4.31、表 4.32：

表 4.31 實證一的機器故障中斷測試干擾結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9518	0.8778	0.9518	0.9518	0.9518	0.9186
交期	0.89	0.69	0.9333	0.7333	0.9333	0.8333
配合度	0.6667	0.3333	1	0.6667	1	0.5556
使用率	0.9277	0.8653	0.9277	0.9277	0.9277	0.8953
懲罰值	0.341	0.341	0.125	0.125	0.125	0.125
總函數	0.8299	0.6725	0.9368	0.7993	0.9368	0.8081
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.940	0.8778	0.940	0.9186	0.9634	0.9186
交期	0.8789	0.69	0.9222	0.8333	0.9006	0.8167
配合度	0.6667	0.3333	1	0.5556	1	0.6111
使用率	0.9167	0.8953	0.9167	0.8653	0.939	0.8953
懲罰值	0.341	0.341	0.125	0.125	0.125	0.125
總函數	0.8156	0.6725	0.9296	0.8081	0.9274	0.8120

於實證一的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 兩方法之個體（共四個個體）績效表現，皆遠高於傳統 GA 的績效。其四個個體經中斷後目標函數值平均約為 0.80，而傳統 GA 則下降至 0.6725，可看出傳統 GA 在求解小型零工式排程問題時，缺乏考量機器中斷造成的中斷影響，而 DBR_HGA 與 DBR_GA 由於在方法本身加入緩衝機制，做為

吸收波動的能力，故其受中斷後，仍可維持較高的績效表現，證明其具有穩健性。

表 4.32 實證二的機器故障中斷測試干擾結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9701	0.9701	0.9750	0.9750	0.9701	0.9701
交期	0.7302	0.7317	0.7124	0.7124	0.7136	0.7136
配合度	0.5	0.3750	1	0.8750	1	1
使用率	0.8756	0.8756	0.9362	0.9362	1	1
懲罰值	0.1796	0.1796	0.1921	0.1921	0.1455	0.1455
總函數	0.7513	0.7287	0.8453	0.8222	0.8638	0.8638
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9898	0.9606	0.9701	0.9701	0.9750	0.9559
交期	0.6848	0.6286	0.6992	0.6992	0.7397	0.6501
配合度	0.5	0.3750	1	1	1	0.8333
使用率	0.8934	0.8670	1	1	0.9362	0.9167
懲罰值	0.1971	0.1971	0.2347	0.2347	0.1455	0.1455
總函數	0.7361	0.6843	0.8434	0.8434	0.8634	0.7934

於實證二的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 個體績效表現高於傳統 GA 的績效。經中斷後目標函數值平均約為 0.84，而傳統 GA 則下降至 0.70，可看出傳統 GA 在求解中型零工式排程問題時，缺乏考量機器中斷造成的中斷影響，而 DBR_HGA 與 DBR_GA，仍可維持較高的績效表現，證明其具有穩健性。

4.3.2 中斷測試：物料短缺

本小節說明物料短缺中斷測試，並依照 3.3.1 小節方法進行排程規劃調整。在兩種實證情境中，加入物料短缺的情節。假設生產現場在生產週期的期初巡檢物料時，發現有一物料不足或是該物料尚未入廠，而無法有效備料，進而發出採購單，經過採購、運送過程後(即經過一段前置期)，物料才可順利到廠。

本例以以均勻分配隨機從工作時程期初(時間點 0 至時間點 20)選取一時間點作為發現物料短缺的時間點，並隨機方式選取一訂單的作業當作發生缺料的作業，並指定物料到廠時間。物料短缺資訊如表 4.33 所示，干擾結果如表 4.34、表 4.35：

表 4.33 兩實證問題之物料短缺資訊

實證一：物料短缺情境假設	
發現缺料時間點	6
缺料作業	3-2 (第 3 張訂單的第 2 個作業)
物料到廠時間	33
實證二：物料短缺情境假設	
發現缺料時間點	2
缺料作業	6-1 (第 6 張訂單的第 1 個作業)
物料到廠時間	36

表 4.34 實證一的物料短缺中斷干擾測試結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
各項指標						
製距	0.9518	0.940	0.9518	0.9518	0.9518	0.9518
交期	0.89	0.8576	0.9333	0.9333	0.9333	0.9333
配合度	0.6667	0.6667	1	1	1	1
使用率	0.9277	0.9167	0.9277	0.9277	0.9277	0.9277
懲罰值	0.341	0.341	0.125	0.125	0.125	0.125
總函數	0.8299	0.8072	0.9368	0.9368	0.9368	0.9368
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
各項指標						
製距	0.94	0.94	0.94	0.94	0.9634	0.9634
交期	0.8789	0.8789	0.9222	0.9222	0.9006	0.9006
配合度	0.667	0.667	1	1	1	1
使用率	0.9167	0.9167	0.9167	0.9167	0.939	0.939
懲罰值	0.341	0.341	0.125	0.125	0.125	0.125
總函數	0.8156	0.8156	0.9296	0.9296	0.9274	0.9274

於實證一的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 個體績效表現高於傳統 GA 的績效。經中斷後目標函數值平均約為 0.93，而傳統 GA 則下降至 0.80，可看出傳統 GA 在求解小型零工式排程問題時，缺乏考量物料短缺造成的中斷影響，而 DBR_HGA 與 DBR_GA 由於仍可維持較高的績效表現，證明其具有穩健性。

表 4.35 實證二的物料短缺中斷干擾測試結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9701	0.9701	0.9750	0.9750	0.9701	0.9701
交期	0.7302	0.7060	0.7124	0.7124	0.7136	0.7136
配合度	0.5	0.5	1	1	1	1
使用率	0.8756	0.8756	0.9362	0.9362	1	1
懲罰值	0.1796	0.1796	0.1921	0.1921	0.1455	0.1455
總函數	0.7513	0.7421	0.8453	0.8453	0.8638	0.8638
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9898	0.9606	0.9701	0.9701	0.9750	0.9750
交期	0.6848	0.6291	0.6992	0.6992	0.7397	0.7397
配合度	0.5	0.25	1	1	1	1
使用率	0.8934	0.8670	1	1	0.9362	0.9362
懲罰值	0.1971	0.1971	0.2347	0.2347	0.1455	0.1455
總函數	0.7361	0.6587	0.8434	0.8434	0.8634	0.8634

於實證二的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 的兩個體績效表現，皆遠高於傳統 GA 的績效。經中斷後目標函數值平均約為 0.93，而傳統 GA 則下降至 0.80，可看出傳統 GA 在求解中型零工式排程問題時，缺乏考量物料短缺造成的中斷影響，而 DBR_HGA 與 DBR_GA 仍可維持較高的績效表現，證明其具有穩健性。

4.3.3 中斷測試：緊急插單

本小節說明緊急插單中斷測試，依照 3.3.1 小節方法進行調整。在兩種實證問題情境中，分別加入緊急插單的情節，將預先規劃排程結果的剩餘

閒置產能作為緊急插單的可用產能。

本例在兩種實證問題情境中，各加入一筆緊急訂單，排程規劃人員重新評定該單的質性因素，決定此插單的重要程度，再進行排程規劃調整。實證一與實證二的緊急訂單資訊與整體質性限制資訊如表 4.36 至表 4.47，而干擾結果如表 4.48 與表 4.49 所示：

表 4.36 實證一之緊急訂單資訊

緊急訂單資料：訂單 6	
得知緊急訂單時間	13
交期(梯形模糊數)	68, 75, 89, 101
總作業途程數	4
加工途程站-(加工時間)	M4-(10), M1-(8), M2-(16), M4-(7)

表 4.37 實證一的質性限制(一)-訂單利潤

利潤	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	訂單 6	$\lambda=6.3420$	評估值
訂單 1	1	2	1	1/2	3	1/2	0.3351	0.1571
訂單 2	1/2	1	1/2	1/3	2	1/4	0.1860	0.087725
訂單 3	1	2	1	1	4	2	0.5054	0.22662
訂單 4	2	3	1	1	3	3	0.6341	0.27745
訂單 5	1/3	1/2	1/4	1/3	1	1/3	0.1280	0.058733
訂單 6	2	4	1/2	1/3	3	1	0.4234	0.19237
C.I.= 0.0684 ; C.R.= 0.0552 ; 符合一致性檢定								

表 4.38 實證一的質性限制(二)-歷史交易

歷史交易	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	訂單 6	$\lambda=6.5104$	評估值
訂單 1	1	1	1/2	1/2	1	1	0.29529	0.12674
訂單 2	1	1	1	1	3	1/2	0.38338	0.16379
訂單 3	2	1	1	2	3	1/4	0.47011	0.20386
訂單 4	2	1	1/2	1	1	1	0.36899	0.16039
訂單 5	1	1/3	1/3	1	1	1/2	0.22643	0.098401
訂單 6	1	2	3	1	2	1	0.59782	0.24682
C.I.= 0.1021 ; C.R.= 0.0823 ; 符合一致性檢定								

表 4.39 實證一的質性限制(三)－市場因素

市場因素	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	訂單 6	$\lambda=6.1012$	評估值
訂單 1	1	3	1/2	1/4	4	2	0.29754	0.15396
訂單 2	1/3	1	1/3	1/5	2	1	0.145	0.075911
訂單 3	2	3	1	1/2	6	3	0.47068	0.24124
訂單 4	4	5	2	1	7	4	0.79791	0.40327
訂單 5	1/4	1/2	1/6	1/7	1	1/2	0.082167	0.042722
訂單 6	1/2	1	1/3	1/4	2	1	0.15966	0.082904
C.I.= 0.0202 ; C.R.= 0.0163 ; 符合一致性檢定								

表 4.40 實證一的質性限制(四)－潛在訂單

潛在訂單	訂單 1	訂單 2	訂單 3	訂單 4	訂單 5	訂單 6	$\lambda=6.1305$	評估值
訂單 1	1	1/4	1	1/2	1/2	1/4	0.15855	0.072933
訂單 2	4	1	3	2	2	2	0.68241	0.31063
訂單 3	1	1/3	1	1	1/3	1/3	0.1833	0.084765
訂單 4	2	1/2	1	1	1/2	1/2	0.25235	0.11635
訂單 5	2	1/2	3	2	1	1	0.42722	0.19576
訂單 6	4	1/2	3	2	1	1	0.47894	0.21957
C.I.=0.0261 ; C.R.=0.0210 ; 符合一致性檢定								

表 4.41 實證一的質性限制之訂單順序

	利潤	歷史	市場	潛在	總加權	排序
權重	0.47	0.16	0.3	0.07		
訂單 1	0.1571	0.12674	0.15396	0.072933	0.1454	4
訂單 2	0.087725	0.16379	0.075911	0.31063	0.1120	5
訂單 3	0.22662	0.20386	0.24124	0.084765	0.2174	2
訂單 4	0.27745	0.16039	0.40327	0.11635	0.2852	1
訂單 5	0.058733	0.098401	0.042722	0.19576	0.0699	6
訂單 6	0.19237	0.24682	0.082904	0.21957	0.1701	3

表 4.42 實證二之緊急訂單資訊

緊急訂單資料：訂單 11	
得知緊急訂單時間	13
交期(梯形模糊數)	178, 188, 195, 209
總作業途程數	6
加工途程站-(加工時間)	M6-(15), M1-(8), M3-(23), M8-(20), M7-(9), M5-(24)

表 4.43 實證二的質性限制(一)- 訂單利潤之權重值

利潤	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	訂單11	$\lambda = 11.364$	評估值
訂單1	1	5	1	1/2	5	4	4	3	2	7	4	0.42796	0.16539
訂單2	1/5	1	1/4	1/7	2	1	1/2	1/3	1/4	3	1/2	0.088815	0.035688
訂單3	1	4	1	1/2	7	4	3	3	2	7	4	0.41946	0.16275
訂單4	2	7	2	1	7	7	5	5	3	8	6	0.68362	0.26389
訂單5	1/5	1/2	1/7	1/7	1	1/2	1/3	1/4	1/4	2	1/3	0.06152	0.024689
訂單6	1/4	1	1/4	1/7	2	1	1	1/2	1/4	4	1	0.10901	0.043528
訂單7	1/4	2	1/3	1/5	3	1	1	1/2	1/3	4	1	0.13083	0.052399
訂單8	1/3	3	1/3	1/5	4	2	2	1	1/2	4	1	0.18039	0.07106
訂單9	1/2	4	1/2	1/3	4	4	3	2	1	5	2	0.2835	0.10992
訂單10	1/7	1/3	1/7	1/8	1/2	1/4	1/4	1/4	1/5	1	1/3	0.045662	0.018105
訂單11	1/4	2	1/4	1/6	3	1	1	1	1/2	3	1	0.13382	0.052579

C.I.= 0.0364 ; C.R.= 0.0241 ; 符合一致性檢定

表 4.44 實證二的質性限制(二)－歷史交易

歷史交易	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	訂單11	$\lambda = 11.357$	評估值
訂單1	1	2	1/3	1/4	2	3	1/3	3	1/2	4	1	0.21058	0.078446
訂單2	1/2	1	1/3	1/4	1	2	1/3	2	1/2	3	1	0.15311	0.057179
訂單3	3	3	1	1/2	3	4	1	5	2	5	3	0.43747	0.15898
訂單4	4	4	2	1	4	4	2	5	3	5	4	0.62783	0.22765
訂單5	1/2	1	1/3	1/4	1	2	1/3	2	1/2	3	1/2	0.14509	0.054452
訂單6	1/3	1/2	1/4	1/4	1/2	1	1/4	2	1/3	2	1/2	0.10346	0.03898
訂單7	3	3	1	1/2	3	4	1	5	2	5	3	0.43747	0.15898
訂單8	1/3	1/2	1/5	1/5	1/2	1/2	1/5	1	1/4	2	1/3	0.080459	0.030283
訂單9	2	2	1/2	1/3	2	3	1/2	4	1	4	2	0.28212	0.10318
訂單10	1/4	1/3	1/5	1/5	1/3	1/2	1/5	1/2	1/4	1	1/3	0.065251	0.024341
訂單11	1	1	1/3	1/4	2	2	1/3	3	1/2	3	1	0.18224	0.067534
C.I.=0.0357 ; C.R.=0.0237 ; 符合一致性檢定													

表 4.45 實證二的質性限制(三)－市場因素

市場考量	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	訂單11	$\lambda = 11.422$	評估值
訂單1	1	4	3	2	5	3	4	1/2	2	5	4	0.4696	0.17361
訂單2	1/4	1	1/2	1/3	3	1/2	2	1/5	1/3	2	1	0.1252	0.048266
訂單3	1/3	2	1	1/3	4	1/2	3	1/4	1/2	3	2	0.1839	0.070869
訂單4	1/2	3	3	1	5	2	4	1/2	2	4	4	0.3768	0.14034
訂單5	1/5	1/3	1/4	1/5	1	1/4	1/2	1/7	1/4	1/2	1/2	0.0593	0.022571
訂單6	1/3	2	2	1/2	4	1	3	1/4	1/2	4	3	0.2321	0.087901
訂單7	1/4	1/2	1/3	1/4	2	1/3	1	1/5	1/4	2	1	0.0842	0.032586
訂單8	2	5	4	2	7	4	5	1	3	5	6	0.6470	0.24058
訂單9	1/2	3	2	1/2	4	2	4	1/3	1	4	3	0.2953	0.11046
訂單10	1/5	1/2	1/3	1/4	2	1/4	2	1/5	1/4	1	1	0.0918	0.035866
訂單11	1/4	1	1/2	1/4	2	1/3	1	1/6	1/3	1	1	0.0966	0.036949
C.I.=0.0421 ; C.R.=0.0279 ; 符合一致性檢定													

表 4.46 實證二的質性限制(四)－潛在訂單

潛在訂單	訂單1	訂單2	訂單3	訂單4	訂單5	訂單6	訂單7	訂單8	訂單9	訂單10	訂單11	$\lambda = 11.373$	評估值
訂單1	1	1/2	1/4	1/5	1/2	1/3	1/4	2	1/4	3	1/4	0.098799	0.036634
訂單2	2	1	1/3	1/5	1	1/5	1/3	2	1/4	3	1/3	0.12905	0.047156
訂單3	4	3	1	1/2	3	2	1	4	1/2	5	1	0.32707	0.11839
訂單4	5	5	2	1	5	3	2	5	2	7	2	0.59317	0.21435
訂單5	2	1	1/3	1/5	1	1/2	1/3	2	1/4	3	1/3	0.12905	0.047156
訂單6	3	2	1/2	1/3	2	1	1/5	3	1/3	4	1/2	0.20651	0.075135
訂單7	4	3	1	1/2	3	2	1	4	1/5	5	1	0.32707	0.11839
訂單8	1/2	1/2	1/4	1/5	1/2	1/3	1/4	1	1/5	2	1/4	0.07142	0.026201
訂單9	4	4	2	1/2	4	3	2	5	1	5	2	0.48165	0.1733
訂單10	1/3	1/3	1/5	1/7	1/3	1/4	1/5	2	1/5	1	1/5	0.06659	0.024902
訂單11	4	3	1	1/2	3	2	1	4	1/2	5	1	0.32707	0.11839
C.I.=0.0373 ; C.R.=0.0247 ; 符合一致性檢定													

表 4.47 實證二的質性限制之訂單順序

	利潤	歷史交易	市場考量	潛在訂單	加權值	排序
權重	0.47	0.16	0.30	0.07		
訂單1	0.16539	0.078446	0.17361	0.036634	0.1449	2
訂單2	0.035688	0.057179	0.048266	0.047156	0.0437	9
訂單3	0.16275	0.15898	0.070869	0.11839	0.1315	3
訂單4	0.26389	0.22765	0.14034	0.21435	0.2176	1
訂單5	0.024689	0.054452	0.022571	0.047156	0.0304	10
訂單6	0.043528	0.03898	0.087901	0.075135	0.0583	7
訂單7	0.052399	0.15898	0.032586	0.11839	0.0681	6
訂單8	0.07106	0.030283	0.24058	0.026201	0.1123	5
訂單9	0.10992	0.10318	0.11046	0.1733	0.1134	4
訂單10	0.018105	0.024341	0.035866	0.024902	0.0249	11
訂單11	0.052579	0.067534	0.036949	0.11839	0.0549	8

表 4.48 實證一緊急插單中斷干擾測試結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9518	0.7670	0.9518	0.8229	0.9518	0.8229
交期	0.89	0.7417	0.9333	0.8472	0.9333	0.8472
配合度	0.6667	0.6667	1	1	1	1
使用率	0.9277	0.9277	0.9277	0.9277	0.9277	0.9277
懲罰值	0.341	0.3729	0.125	0.2325	0.125	0.2325
總函數	0.8299	0.7426	0.9368	0.8732	0.9368	0.8732
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.94	0.7670	0.94	0.8229	0.9634	0.8144
交期	0.8789	0.7324	0.922	0.8380	0.9006	0.8060
配合度	0.6667	0.6667	1	1	1	1
使用率	0.9167	0.9167	0.9167	0.9167	0.939	0.939
懲罰值	0.341	0.3729	0.125	0.2325	0.125	0.2325
總函數	0.8156	0.7372	0.9296	0.8678	0.9274	0.8586

於實證一的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 的兩個體績效表現，皆遠高於傳統 GA 的績效。經中斷後目標函數值平均約為 0.86，而傳統 GA 則下降至 0.73，可看出傳統 GA 在求解小型零工式排程問題時，缺乏考量物料短缺造成的中斷影響，而 DBR_HGA 與 DBR_GA 由於在方法本身加入緩衝機制，做為吸收波動的能力，故其受中斷後，仍可維持較高的績效表現，證明其具有穩健性。

表 4.49 實證二緊急插單中斷干擾測試結果

方法別 (個體別)	傳統 GA (個體 1)		DBR_GA (個體 1)		DBR_HGA (個體 1)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9701	0.8125	0.9750	0.9750	0.9701	0.9701
交期	0.7302	0.6225	0.7124	0.7385	0.7136	0.7123
配合度	0.5	0.5	1	1	1	1
使用率	0.8756	0.8756	0.9362	0.9362	1	1
懲罰值	0.1796	0.1492	0.1921	0.1645	0.1455	0.1275
總函數	0.7513	0.6996	0.8453	0.8598	0.8638	0.8663
方法別 (個體別)	傳統 GA (個體 2)		DBR_GA (個體 2)		DBR_HGA (個體 2)	
	中斷前	中斷後	中斷前	中斷後	中斷前	中斷後
製距	0.9898	0.8553	0.9701	0.9375	0.9750	0.9750
交期	0.6848	0.6225	0.6992	0.6421	0.7397	0.7309
配合度	0.5	0.5	1	1	1	1
使用率	0.8934	0.8934	1	1	0.9362	0.9362
懲罰值	0.1971	0.1674	0.2347	0.1996	0.1455	0.1275
總函數	0.7361	0.7039	0.8634	0.8243	0.8634	0.8631

於實證二的中斷測試中，發現經由中斷測試後，DBR_HGA 與 DBR_GA 的兩個體績效表現，皆遠高於傳統 GA 的績效。經中斷後目標函數值平均約為 0.86，而傳統 GA 則下降至 0.69，可看出傳統 GA 在求解中型零工式排程問題時，缺乏考量物料短缺造成的中斷影響，而 DBR_HGA 與 DBR_GA 由於在方法本身加入緩衝機制，做為吸收波動的能力，故其受中斷後，仍可維持較高的績效表現，證明其具有穩健性。

4.4 質性與量性瓶頸資源之敏度度分析

量性瓶頸與質性瓶頸對於本排程規劃研究中，根據表 4.11 由 AHP 法所制定的權重來說，量性與質性的考量比率為 0.8333：0.1667，尤以量性瓶頸考量過於偏重。為考量規劃者對於兩種瓶頸資源規劃時，所設定的考量設定比重不完全相同，本研究在此小節內，以三組不同權重比例，以 4.1 節所述的三種求解方法：傳統遺傳演算法（GA）、具遺傳演算法機制之限制驅導技術（DBR_GA）及具混合式遺傳演算法機制之限制驅導技術（DBR_HGA），於兩種實證問題情境下求解，來驗證本研究所提之排程方法是否適用於各類質性與量性權重比例，而不因權重設定問題權重，造成求解品質（排程績效）偏低。三組不同權重比例為：第一組：量性瓶頸：質性瓶頸=0.75：0.25；第二組：量性瓶頸：質性瓶頸=0.5：0.5；第三組：量性瓶頸：質性瓶頸=0.25：0.75。以下說明三組權重的實驗結果與分析。

第一組、量性瓶頸：質性瓶頸=0.75：0.25

在此組權重數據上，式 4.1 的多目標限制適應函數則改為：

$$\text{Fitness} = \{0.75 \times [0.1202 \times \text{製距} + 0.4566 \times \text{交期滿足度} + 0.2020 \times \text{機器使用率} + 0.2212 \times \text{非瓶頸資源配合度}]\} + \{0.25 \times (1 - \text{質性懲罰函數})\}。$$

在此多目標限制適應函數下，利用三種方法於兩種實證問題情境下各求解 30 次，其結果如表 4.50、表 4.51：

表 4.50：實證一以權重組合 1 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.809575	最大值	0.930618	最大值	0.931704
平均值	0.786032	平均值	0.924413	平均值	0.930344
標準差	0.016528	標準差	0.007275	標準差	0.001974

表 4.51：實證二以權重組合 1 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.755059	最大值	0.844645	最大值	0.862835
平均值	0.704284	平均值	0.818452	平均值	0.861911
標準差	0.020738	標準差	0.010813	標準差	0.001731

由表 4.50、表 4.51 的實驗結果中，指出規劃者冀望在量性瓶頸考量偏重時，本研究所提出的具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA)，其求解表現比傳統遺傳演算法 (GA) 來的好。因本研究所採用的限制驅導技術，針對量性瓶頸—產能負荷最高之工作站，將現場加工作業分群，依分群結果，再進行排程規畫，使規劃以其工作站需求為主，提高該工作站利用率，進而提升系統整體績效，而傳統遺傳演算法則以隨機方式指定工作排程規劃順序，非以系統瓶頸需求為主要考量，則系統績效偏低。

第二組、量性瓶頸：質性瓶頸=0.5：0.5

在此組權重數據上，式 4.1 的多目標限制適應函數則改為：

$$\text{Fitness} = \{0.5 \times [0.1202 \times \text{製距} + 0.4566 \times \text{交期滿足度} + 0.2020 \times \text{機器使用率} + 0.2212 \times \text{非瓶頸資源配合度}]\} + \{0.5 \times (1 - \text{質性懲罰函數})\}。$$

在此多目標限制適應函數下，利用三種方法於兩種實證問題情境下各求解 30 次，其結果如表 4.52、表 4.53：

表 4.52：實證一以權重組合 2 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.863405	最大值	0.912794	最大值	0.912794
平均值	0.83069	平均值	0.905879	平均值	0.912119
標準差	0.014628	標準差	0.011154	標準差	0.000184

表 4.53：實證二以權重組合 2 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.790572	最大值	0.842049	最大值	0.860041
平均值	0.752941	平均值	0.823656	平均值	0.85775
標準差	0.020951	標準差	0.008451	標準差	0.005242

在此權重設定下，表示規劃者冀望排程規劃考量質性與量性瓶頸時為等重。由表 4.52、表 4.53 的實驗結果中指出，本研究所提的所提出的具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之

限制驅導技術 (DBR_HGA) 可以同步考量兩種瓶頸資源，其實驗結果比傳統遺傳演算法 (GA) 較佳。表示本研究所提之限制驅導技術更可針對兩種瓶頸資源進行同步考量規劃。

第三組、量性瓶頸：質性瓶頸=0.25：0.75

在此組權重數據上，式 4.1 的多目標限制適應函數則改為：

$$\text{Fitness} = \{0.25 \times [0.1202 \times \text{製距} + 0.4566 \times \text{交期滿足度} + 0.2020 \times \text{機器使用率} + 0.2212 \times \text{非瓶頸資源配合度}]\} + \{0.75 \times (1 - \text{質性懲罰函數})\}。$$

在此多目標限制適應函數下，利用三種方法於兩種實證問題情境下各求解 30 次，其結果如表 4.54、表 4.55：

表 4.54：實證一以權重組合 3 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.923329	最大值	0.913358	最大值	0.914694
平均值	0.912498	平均值	0.905875	平均值	0.913251
標準差	0.008284	標準差	0.007851	標準差	0.000776

表 4.55：實證二以權重組合 3 之實驗測試結果

傳統式 GA		DBR_GA		DBR_HGA	
最大值	0.853137	最大值	0.846801	最大值	0.857247
平均值	0.834985	平均值	0.83443	平均值	0.854768
標準差	0.018555	標準差	0.006422	標準差	0.005258

由表 4.54、表 4.55 的實驗結果中指出，本研究所提的所提出的具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 與傳統遺傳演算法 (GA) 求解表現差異不大。

由於此權重設定下，表示規劃者冀望排程規劃考量質性瓶頸偏重，而本研究所提之質性限制為重要定擔憂現考量的政策，為非實體限制，其中值性瓶頸為訂單規劃優先順序。

然而不管是本研究所提的 DBR_GA、DBR_HGA 或是傳統的 GA 對於訂單規劃優先順序都可以提供良好的規劃能力，而在此權重設定下，量性

考量偏低，而本研究所提之 DBR_GA、DBR_HGA 則因多納入量性瓶頸考量，仍以瓶頸作業優先進行排程規劃，則增加違反訂單規劃優序的懲罰值，所以排程績效降低。

本小節以 3 組權重比例，利用傳統遺傳演算法 (GA)、具遺傳演算法機制之限制驅導技術 (DBR_GA) 及具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA)，於兩實證問題求解。結果指出，規劃者考量質性與量性瓶頸資源偏好比重不同時，當考量量性瓶頸較重於質性或是相等時，以本研究所提之 DBR_GA、DBR_HGA 皆可有較高的求解品質，但當規劃者考量質性瓶頸偏重時，則三種方法表現在伯仲之間，表示本研究所提的排程方法仍可考量量性與質性瓶頸資源在不同程度考量下，皆具有良好的排程規劃能力，不受到規劃者考量著重某一瓶頸資源，而使得排程績效低落。

4.5 本章小結

經由本章所述兩組 Job shop 實證問題測試，以三種方法進行求解，每種方法各實驗 30 次，以統計檢定的方法，證實在相同的求解代數下，發現在兩種實證問題測試下，以 DBR_GA、DBR_HGA 求解結果皆明顯優於傳統 GA 方法。同時由統計檢定可得 DBR_HGA 法優於 DBR_GA 法。另外，當排程問題的規模變大，加入禁忌搜尋機制，可使遺傳演算法求解較大規模的排程問題上，可以有更高的品質。而當規劃者考量不同的質性與量性瓶頸的權重組合下，本研究所提之 DBR_GA、DBR_HGA 仍有不錯的績效。

在兩種實證問題中，各以三種中斷情節去測試排程結果的穩健性，從中斷測試中可以發現具混合式遺傳演算法機制之限制驅導技術 (DBR_HGA) 所求解出的排程結果具有高度穩健性，在實證二 (中型排程問題) 中，不管是機台故障、物料短缺、緊急插單的中斷干擾下，原先績效最高的染色體受中斷後仍是績效最高，甚至於緊急插單下，績效往上增加，其原因為新插入訂單在排程規劃後，平均交期達交率提高，進而提升整體訂單達交率，另一方面，在懲罰函數表現上，由於新訂單的增加，降低訂單懲罰函數，故該個體的多目標適應函數向上提升。這樣情況亦反映著增加未佔用作業 (Free operations) 可提升系統額外產出，同時提升系統績效表現，但因剩餘產能挪為新加入訂單所用，所以機器閒置時間減少，會造成系統彈性減低，吸收波動現象的能力亦降低。

第五章 結論與未來研究方向

5.1 結論

本研究於零工式生產環境下，探討排程規劃中會使生產系統績效受限的量的限制資源（機器產能負荷、訂單交期）與質性限制資源（政策文化、市場需求等）。以限制資源的需求為優先導向，將多項限制轉換為量的多目標函數與質性限制懲罰函數，再結合為多目標架構，並發展以限制驅導排程技術結合混合式遺傳演算法進行排程問題求解，得到一個可容忍系統波動、具穩健性的排程規劃。

根據實驗結果，本研究歸納下列幾項結論：

1. 生產系統因為追求多種績效，會受限於不同的資源，以限制資源為優先考量，使系統在資源有限下進行最佳的資源分配，可使系統整體績效更高。
2. 傳統的限制驅導排程技術僅探討實體限制資源，多以產能負荷最重的工作站為瓶頸，本研究將非實體限制，如：市場需求、潛在顧客、歷史交易、訂單利潤等限制作為質性限制並納入考量。排程規劃同步考量多重瓶頸，包含質性與量的架構，可更貼近現實環境，提高實務的應用性。
3. 將限制驅導排程技術結合遺傳演算法，並加上禁忌搜尋機制所得到的排程解之品質可比傳統的遺傳演算法來得更好，雖無法確保其解為最佳，但利用本研究的設計的生產排程系統可以在短時間內求取出一個近似最佳解或可接受解。
4. 利用限制驅導排程技術的緩衝機制，可以讓排程規劃中保有彈性來吸收一些異常情況發生，如：當機或缺料等影響，對生產系統造成的斷開置的不良效應。可使生產系統產出平順，則此排程規劃具有穩健性，可適用於現實環境中複雜的生產環境。

5.2 未來研究方向

雖然本研究經由實驗驗證，確實可達預期之效果，但在發展本研究架構與問題實作的過程中，也發現一些可進一步探討的問題，歸納出幾項未來研究方向，如下：

1. 本研究探討各工作站中僅有一台加工機器，而現實中的生產系統多為更複雜的平行機台模式，其中機器群的新舊、可靠度亦影響機器加工的效度，本研究建議可將平行等效與平行非等效模式納入探討。可更貼近實務。
2. 質性限制資源與公司策略息息相關，本研究建議由企業營運策略出發，探討更深入的質性限制，將質性限制資源進行更多維度探討。
3. 不同的產品組合變更，易造成系統的瓶頸漂移，則排程規劃勢必面臨重新規劃的考量，本研究建議後續研究可探討瓶頸漂移與重排程相互的影響。
4. 排程規劃多屬企業短期規劃，而如何將短期資訊回饋中長期的規劃是困難的。本研究建議將所提之系統架構結合系統模擬的方法進行中長期規劃預測，將中長期模擬與短期排程資訊同步回饋公司營運管理者，則可獲得更多的資訊，以供決策。

經由實際驗證，證實本研究所提出的多目標與限制驅導零工式排程系統之可行性，未來後續研究可以針對上述幾點，作更深入的探討與研究。

參考文獻

中文部份

- [1] 吳鴻輝、李榮貴，限制驅導式現場排程與管理技術，全華科技圖書股份有限公司，民 89 年。
- [2] 潘國丞，「多目標平行機台零工式生產重排程之探討」，東海大學工業工程與經營資訊所，碩士論文，民 95 年。
- [3] 吳信儀，「以改良之進化策略演算法解決排序問題之研究—SRS 演算法與多重工作者系統之發展」，東海大學工業工程研究所碩士論文，民 85 年。

英文部分

- [4] Adenso, D., “An SA/TA mixture algorithm for the scheduling tardiness problem”, *European Journal of Operation Research*, Vol. 88, pp. 6-524, 1996.
- [5] Artigues, C., Billaut, J.-C. ,and Esswein, C., “Maximization of solution flexibility for robust shop scheduling”, *European Journal of Operational Research*, Vol. 165, pp. 14-328, 2005.
- [6] Boaz Ronen and Martin K. Starr, “Synchronized manufacturing as in OPT: from practice to theory”, *Computers ind. Engng*, Vol. 18, No. 4, pp.585-600, 1990.
- [7] Brown, J.R. and Ozgur, C.O., “Priority class scheduling: product scheduling for multi-objective environment”, *Production Planning and Control*, Vol. 8, pp. 2-770, 1997.
- [8] Chang, P.T., Lo, Y. T., “Modelling of job-shop scheduling with multiple quantitative and qualitative objectives and a GA/TS mixture approach”, *International Journal of Computer Integrated Manufacturing*, Vol. 14, No. 4, pp. 67-384, 2001.
- [9] Cheng R., and Gen M., “Parallel machine scheduling problems using genetic algorithms”, *Computer and Industrial Engineering*, Vol. 33, pp. 1-764, 1997.
- [10] Dooley, J.K. and Mahmoodi, F., “Identification of robust scheduling heuristics: application of Taguchi methods in simulation studies”, *Computers ind. Engng*, Vol. 22, No. 4, pp. 59-368, 1992.
- [11] Goldratt, .E.and Cox, J., *The Goal*, Cronton-on-Hudson, NY, 1986.
- [12] Gupta, Amit Kumar, “Single machine scheduling with multiple objectives in semiconductor manufacturing”, *International Journal of Advanced Manufacturing Technology*, Vol. 26, No. 9-10, pp. 50-958, 2005.
- [13] Fang, H. L., “Genetic Algorithm in timetabling and scheduling”, Ph. D. dissertation, Department of Artificial Intelligent, University of Edinburgh, 1994.
- [14] Forgy, T. C., “Varying the probability of mutation in the genetic algorithm”, *Proceedings of the 3rd International Conference on Genetic algorithms*, pp. 04-109,

1989.

- [15] Garey MR, Johnson DS, Sethi R, “The complexity of flow-hop and job-shop scheduling”, *Math Oper Res*, 1976, Vol.1, pp. 117-129.
- [16] Geiger, Martin Josef, “On operators and search space topology in multi-objective flow shop scheduling”, *European Journal of Operational Research*, Vol. 181, No.1, pp. 195-206 2007.
- [17] Glover, F., “Tabu search-part I”, *ORSA Journal on Computing*, Vol. 1, No. 3, pp. 190-206, 1989.
- [18] Glover, F., “Tabu search-part II”, *ORSA Journal on Computing*, Vol. 2, No. 1, pp. 4-32, 1990.
- [19] Glover, F., “Tabu search : a tutorial ”, *Interface*, Vol. 20, No. 4, pp. 74-94, 1990.
- [20] Jensen, T.M., “Generating robust and flexible job shop schedules using genetic algorithms”, *IEEE Transactions on Evolutionary Computation*, Vol. 7, No. 3, June 2003.
- [21] Joseph Y-T. Leung, *Handbook of Scheduling : Algorithms, Models, and Performance Analysis*, Boca Raton :Chapman & Hall/CRC, 2004.
- [22] J. Schaefer, R. Aggoune, F. Becker and R. Fabbri, “TOC-based planning and scheduling model”, *International Journal of Production Research*, Vol. 42, No. 13, pp. 2639 – 2649, 2004.
- [23] J. V. Simons, M.D. Stephens ,and W.P. Simpson, ,”A simultaneous versus sequential scheduling of multiple resources which constrain system throughput”, *International Journal of Production Research*, Vol. 37, No. 1, pp. 21-33, 1999.
- [24] Kim, G. H. and C. S. G.Lee, “An evolutionary approach to the job-shop scheduling problem”, *Proceedings IEEE International Conference on Robotics and Automation*, Vol.1, pp. 501-506, 1994.
- [25] Kumar, N.S.H. and Srinivasan, G, “A genetic algorithm for job shop scheduling- A case study”, *Computers in Industry*, Vol. 31 , pp. 155-160, 1996.
- [26] Leyuan Shi; Yunpeng Pan, “An efficient search method for job-shop scheduling problems”, *Automation Science and Engineering, IEEE Transactions on [see also Robotics and Automation*, Volume 2, Issue 1, pp. 73-77, 2005.
- [27] Liaw, C.-F., “A hybrid genetic algorithm for the open shop scheduling problem”, *European Journal of Operational Research*, Vol. 124, pp. 28-42, 2000.
- [28] Loukil, T., Teghem, J., and Tuyttens, D., “Solving multi-objective production scheduling problems using metaheuristics”, *European Journal of Operational Research*, Vol. 161, pp. 42-61, 2005.
- [29] Low, Chinyao, Wu, Tai-Hsi, and Hsu, Chih-Ming, “Mathematical modelling of multi-objective job shop scheduling with dependent setups and re-entrant operations”, *International Journal of Advanced Manufacturing Technology*, Vol. No. 27, pp. 181-189, 2005.

- [30] Mellor, P., "A review of job shop scheduling", *Operational Research Quarterly*, Vol.17, No.2, pp. 161-170, 1966.
- [31] Michael Pinedo, Xiuli Chao, *Operations Scheduling with Applications in Manufacturing and Services*, rwin/McGraw-Hill, Boston, 1999.
- [32] Michalewicz, Z., "Genetic Algorithm + Data Structures = Evolution Programs", *Springer-Verg Berlin Heidelberg*, 1994
- [33] Mikkel T. Jensen, "Generating robust and flexible job shop schedules using genetic algorithms", *IEEE Transaction on evolutionary computation*, Vol. 7, No. 3, 2003.
- [34] Murata, T. and Ishibuchi, H., "Performance evaluation of genetic algorithms for flowshop scheduling problems," *Proceedings the First IEEE International Conference on Evolutionary Computation*, Vol. 2, pp. 812-817, 1994.
- [35] Norman, B.A., Bean, J.C., "A genetic algorithm methodology for complex scheduling problems", *Naval Research Logistics* , Vol. 46, pp. 199-211, 1999.
- [36] Park, B.J., Choi, H.R. and Kim, H.S., "A hybrid genetic algorithm for the job shop scheduling problems", *Computers & Industrial Engineering*, 45, pp. 597–613, 2003.
- [37] Petty, C. B., M. R. Leuze and J. J. Grefenstette, "A parallel genetic algorithm", *Proceedings of the Second International Conference on Genetic Algorithm*, pp. 155-161, 1987.
- [38] Ravindran, D., Selvakumar, S.J., and Haq, A. Noorul, "Flow shop scheduling with multiple objective of minimizing makespan and total flow time", *International Journal of Advanced Manufacturing Technology*, Vol. 25, No. 9-10, pp. 1007-1012, 2005.
- [39] Saaty T. L., *The Analysis Hierarchy Process*, McGraw-Hill Inc. NewYork, 1980.
- [40] Saidi-Mehrabad, Mohammad and Fattahi, Parviz, "Flexible job shop scheduling with tabu search algorithms", *International Journal of Advanced Manufacturing Technology*, Vol. 32, No. 5-6, pp. 563-570, 2007.
- [41] Satya S. Chakravorty, "An evaluation of the DBR control mechanics min a job shop environment", *Omega*, Vol. 29, pp. 335-342, 2001.
- [42] Satya S. Chakravorty and J. Brian Atwater, "The impact of free goods on the performance of drum-buffer-rope scheduling systems", *International Journal of Production Research*, Vol. 95, pp. 347-357, 2005.
- [43] Schragenheim, E. and Ronen, B. "Drum-buffer-rope shop floor control", *Prod. Invent. Manage. J.*, 3rd ,pp. 18-23, 1990.
- [44] Scrich, Cintia Rigao,. Armentano, Vinicius Amaral, and Laguna Manuel, "Tardiness minimization in a flexible job shop: A tabu search approach", *Journal of Intelligent Manufacturing*, Vol.15, No. 1, pp. 103-115, February, 2004.
- [45] Sevaux, M., Sörensen, K., "A genetic algorithm for robust schedules", *Paper presented at the 8th International Workshop on Project Management and Scheduling*, Valencia, April 3–5, 2002.

- [46] Sirikrai, V., and Yenradee, P., "Modified drum-buffer-rope scheduling mechanism for a non-identical parallel machine flow shop with processing-time variation" , *International Journal of Production Research*, Vol. 44, No. 17, pp. 3509-3531, 2006.
- [47] Sörensen, K., "Tabu searching for robust solutions", *research paper, Faculty of Applied Economic Sciences UFSIA-RUCA*, University of Antwerp, UA, 2002.
- [48] S.-Y. Wu, John S. Morris, and Thomas M. Gordan, "A simulation analysis of effectiveness of drum-buffer-rope scheduling in furniture manufacturing", *Computers ind. Engng*, Vol. 26, No. 4, pp. 757-764, 1994.
- [49] Varadharajan, T.K., Rajendran, and Chandrasekharan, "A multi-objective simulated-annealing algorithm for scheduling in flowshops to minimize the makespan and total flowtime of jobs", *European Journal of Operational Research*, Vol. 167, No. 3, pp. 772-795, 2005.
- [50] V. Daniel, R. Guide Jr., "Scheduling with priority dispatching rules and drum-buffer-rope in a recoverable manufacturing system", *International Journal of Production Economics*, Vol. 53, pp. 101-116, 1997.
- [51] Velagapudi, K.N., "Robust Schedule for Manufacturing Systems", *Computers and Industrial Engineering*, Vol. 23, pp. 133-136, 1992.
- [52] Wellman, M. A. and D. D. Gemmill, "A genetic algorithm approach to optimization of asynchronous automatic assembly systems", *International Journal of Flexible Manufacturing Systems*, Vol. 7, pp. 27-46, 1995.
- [53] Wesley, B. J., and J. B. Chambers, "Solving the job shop scheduling problem with tabu search", *IEEE Transactions*, Vol. 27, No. 2, pp. 257-263, 1995.
- [54] Wu, H.-H. and Yeh, M.-L., "A DBR scheduling method for manufacturing environments with bottleneck re-entrant flows" , *International Journal of Production Research*, Vol. 44, No. 5, pp. 883-902, 2006.