

國立成功大學
工業與資訊管理研究所
碩士論文

鋼鐵業直棒製程之平行機台重工排程問題
On Scheduling Reworks for Jobs on Parallel Machines
under Steel Bar Manufacturing Process

研究生：黃麟貴

指導教授：王逸琳

中華民國一百零二年六月

國立成功大學

碩士在職專班論文

鋼鐵業直棒製程之平行機台重工排程問題

On Scheduling Reworks for Jobs on Parallel

Machines under Steel Bar Manufacturing Process

研究生：黃麟貴

本論文業經審查及口試合格特此證明

論文考試委員：王逸琳

李家宏

孔令傑 李宇欣

指導教授：王逸琳

系(所)主管：黃宇和

中華民國 102 年 5 月 29 日

鋼鐵業直棒製程之平行機台重工排程問題

黃麟貴

國立成功大學工業與資訊在職專班

指導教授：王逸琳 博士

摘要

近年由於大陸鋼廠崛起，挾著規模經濟採取多量少樣的生產方式，以價格及產量優勢搶占大量市場；台灣鋼廠之規模雖然無法與其相比，但在技術與品質仍有其優勢，從多量少樣計劃生產方式轉為多樣少量的訂單生產方式。為了符合客製化的訂單生產，必須滿足客戶品質、訂單交期及零庫存的要求，然而製程加工時通常會產生瑕疵品，為要趕上訂單交期必須以重工方式修復瑕疵，因此可能影響原先生產計劃的時間安排，進而造成其它訂單的生產延遲。

本研究擬探討盤元與直棒製程中經常處理之平行機台工件重工排程問題，在考慮交期與整備時間下，依平行機台生產條件限制，決定各工件應在哪一平行機台之排程順序，再假設各平行機台產生瑕疵工件之比例為已知下，決定重工工件應於哪一平行機台之排程順序。利用貪婪演算法 ARI (All Reworks Insertion Rule) 和 SRI (Single Rework Insertion Rule) 二種不同角度的法則，以單一製程為試算單位，計算出原工件的生產排序與重工工件的插入排序，來求解區域的最佳解以達到全域的最佳近似解，並以禁忌搜尋法求解全域的最佳近似解，推算符合現場實際生產狀況的排程完成時間，提昇客戶訂單的達交率，以達到最小化平行機台中之最晚完工時間為目標。

在求解效率與品質的表現方面，SRI 較 ARI 為佳；由於 ARI 在交期、檢驗時間與工件有長短工時幾種因素影響下，其總完工時間數據的起伏變化較大，若將 ARI 經禁忌搜尋法進一步改善後，其表現將大為改善，在鋼鐵業的直棒製程中，運用 SRI 與 ARI 的運算結果可優於現行既有的排程方式。

關鍵字：重工；排程；平行機台；貪婪演算法；禁忌搜尋法

On Scheduling Reworks for Jobs on Parallel Machines under Steel Bar Manufacturing Process

Benny Huang

Institute of Manufacturing Engineering National Cheng Kung University

Advisor : Dr. I-Lin Wang

Abstract

China's steel mills have been a major player in the market because it follows a high-volume-low-variety strategy. This strategy involves selling large quantities but just a few types of steel in order to maximize production efficiency, which keeps prices low. High efficiency and low prices are China's primary advantages in the market. Even though the economic scale of Taiwan's steel mills is smaller than China's, Taiwan's technology and quality of the steel are still competitive in the market. Hence, Taiwan's production model has changed to a make-to-order strategy, meaning Taiwan produces a large number of customized steel products but low volumes of each type. In order to satisfy the customized specifications by different clients, steel mills have to fulfill different requirements on steel quality, deadlines, and zero stock. Because each customized steel product usually requires its own manufacturing process, defective products are common. Steel mills have to fix defects and rework their manufacturing process in order to meet deadlines of orders. Therefore, the reworking process may affect manufacturing schedules, which will in turn delay manufacturing of other orders.

This thesis discusses issues of rescheduling jobs on parallel machines under steel bar manufacturing process. Due-dates, setup times, and production limitations of parallel machines can be used to determine individual work in a certain order for parallel machine scheduling. Furthermore, assuming that the rate of defective products of each parallel machine are known, one can reorganize tasks in a certain order to optimize the output of

using parallel machines. In order to efficiently calculate good manufacturing schedules, we propose greedy algorithms based on two rules: ARI (All Reworks Insertion Rule) and SRI (Single Rework Insertion Rule). Once the original orders and the new order are known, these data can be used to calculate good local optimal solutions, which can then be further improved. We also give a tabu search method to solve our problem. By our proposed solution methods, steel mills could increase more production with shorter completion time of parallel machines.

Our computational experiments indicate SRI performs better than ARI. However, the performance of ARI could be much improved when it integrates with the Tabu search mechanism. Both SRI and ARI are useful for making better schedules for manufacturing steel bar.

Keywords : Rework; Scheduling; Parallel Machine; Greedy Algorithm; Tabu Search

誌謝

在研究所期間，由於在職之故，並未能如一般學生全心投入學業與論文著述，而能完成本論文，首先要特別感謝指導教授 王逸琳老師的悉心指導與鼎力協助，在繁忙的教學與研究外，尚撥冗予以逐字檢查斧正，使得學生在繁忙的工作壓力下，得以清礎掌握論文方向與架構，並於付梓之前，讓本論文在期限前完成，在此致上最誠摯的感謝與敬意。

另外在口試期間，承蒙口試委員成功大學 李宇欣博士、成功大學 李家岩博士與台灣大學 孔令傑 博士之不吝指導，提供本研究許多寶貴的意見，使本研究更臻完善，在此深表謝意。

在我繁重的課業與上課期間中，要感謝公司的同仁與主管，能因應我的狀況給予協助與照顧，讓我能夠適時調整而不影響工作，尚要感謝我的家人與朋友們，有你們在我背後的支持，讓我更有心往自己的目標前進。

最後，要感謝我的妻子詩怡，在求學期間，無數個南來北往的日子，從準備結婚牽手誓約到購置新屋建構家園，一路走來互相扶持，並對我學業、工作上的支持，讓我無後顧之憂順利完成學業，謝謝！

黃麟貴 謹誌

民國 102 年七月于 成大

目錄

摘要	i
Abstract	ii
誌謝	iv
目錄	v
圖目錄	viii
表目錄	x
1. 緒論	1
1.1 研究背景	1
1.2 研究動機	2
1.3 研究目的	2
1.4 研究流程	3
1.5 論文架構	3
2. 文獻探討	5
2.1 平行機台之排程問題	5
2.1.1 機台環境	5
2.1.2 機台特性與限制	6
2.1.3 績效衡量準則	7
2.2 重工	8
2.2.1 重工之瑕疵問題	8
2.2.2 重工之運作流程	9
2.3 排程相關技術	10
2.2.1 派工法則	10
2.2.2 精確最佳解方法	12

2.2.2 近似解方法	12
2.2.3 電腦模擬方法	12
2.4 盤元與直棒製程	13
2.5 小結	15
3. 研究方法	17
3.1 問題描述	17
3.2 基本假設	18
3.3 貪婪演算法模式建立與說明	18
3.3.1 參數說明	19
3.3.2 EDD+SPT with WIDR 演算法	20
3.3.3 ARI 演算法	22
3.3.4 SRI 演算法	29
3.4 禁忌搜尋法說明	33
3.4.1 禁忌搜尋法的基本概念	34
3.4.2 禁忌搜尋法的步驟	35
4. 實驗結果分析與討論	38
4.1 實驗環境	38
4.2 模式資料設定	38
4.3 實驗分析與討論	39
4.3.1 求解效率	39
4.3.2 求解品質	40
4.4 小結	45
5. 結論與建議	47
5.1 結論	47
5.2 未來研究與建議	48



圖目錄

圖 1.1 鋼材的化性與物性	1
圖 2-1 兩階段製造系統的重工結構圖 資料來源：Inderfurth et al. (2005)	9
圖 2-2 平行機台重工生產系統的重工結構圖 資料來源：Kang & Shin (2010)	10
圖 2.3 鋼鐵產業架構	13
圖 2.4 軋鋼流程圖	14
圖 2.5 固溶酸洗流程圖	14
圖 2.6 直棒流程圖	15
圖 2.7 細線盤元與冷精直棒流程圖	15
圖 3.1 單一製程平行機台重工流程圖	17
圖 3.2 ARI 演算流程圖	24
圖 3.3 工件排入平行機台示意圖	26
圖 3.4 工件檢驗完成時間示意圖	27
圖 3.5 平行機台插入重工工件排程示意圖	27
圖 3.6 工件檢驗時間與完工時間比對示意圖	28
圖 3.7 SRI 演算流程圖	31
圖 3.8 平行機台插入重工工件排程示意圖	32
圖 3.9 平行機台插入重工工件排程示意圖	33
圖 4.1 Greedy 與 Tabu_A 三種法則求解時間	39
圖 4.2 Greedy 二種法則求解時間	40
圖 4.3 不異動工件之平行機台最長完工時間	41
圖 4.4 不異動工件之平行機台負荷時間	41
圖 4.5 異動交期之平行機台最長完工時間	42
圖 4.6 異動交期之平行機台負荷時間	42

圖 4.7 異動檢驗時間之平行機台最長完工時間	43
圖 4.8 異動檢驗時間之平行機台負荷時間	43
圖 4.9 不考慮整備時間之平行機台最長完工時	44
圖 4.10 不考慮整備時間之平行機台負荷時間	44
圖 4.11 平行機台最長完工時間之長短工時	45
圖 4.12 平行機台負荷時間之長短工時	45



表目錄

表 2-1 常用的排程績效衡量標準	7
表 3.1 工件資料	21
表 3.2 尺寸級距資料	21
表 3.3 EDD+SPT with WIDR 工件排序	22
表 3.4 平行機台處理時間與重工率資料	24
表 3.5 平行機台資料	25
表 3.6 工件排序及可在平行機台生產的數據	25
表 3.7 ARI 排程運算數據	29
表 3.8 SRI 排程運算數據	33



1. 緒論

近年由於大陸鋼廠崛起，挾著規模經濟採取多量少樣的生產方式，以價格及產量優勢搶占大量市場；台灣鋼廠之規模雖然無法與其相比，但在技術與品質仍有其優勢，從多量少樣計劃生產方式轉為多樣少量的訂單生產方式，從生產附加價值低的鋼材提昇生產附加價值高的鋼材。為了符合客製化的訂單生產，必須滿足客戶品質、訂單交期及零庫存的要求，然而製程加工時通常會產生瑕疵品，為要趕上訂單交期必須以重工方式修復瑕疵，因此可能影響原先生產計劃的時間安排，進而造成其它訂單的生產延遲。若能在生產計劃中將一般排程比較少考慮的重工時間一併估算，應該更能符合現場實際運作狀況，並推算出更合理的訂單交期時間。

1.1 研究背景

在煉鋼廠所煉製的同類型鋼材，每一個鋼材的化性與物性皆會有些許的不同，如圖 1.1 所示，由於中下游鋼廠通常自多方上游煉鋼廠購買同類型鋼材，因此其鋼材的硬度、抗拉強度、屈服強度、斷後伸長率、斷面收縮率等機械性質會有些微的差異。

化學成份 (WT%)											機械性質				
Kind of Test	C	Si	Mn	P	S	Ni	Cr	Mo	Cu	N	硬度 HB	抗拉 強度 N/mm2	屈服 強度 N/mm2	斷後 伸長率	斷面 收縮率
Spec.	MAX	MAX	MAX	MAX	MAX	12.00	20.00					MIN	MIN	MIN	MIN
	0.03	1.00	2.00	0.045	0.030	8.00	18.00					485	170	40	50
01	0.022	0.33	1.79	0.034	0.002	8.18	18.03	0.04	0.15		167	565	235	49	73

圖 1.1 鋼材的化性與物性

所以即使同類型鋼材之不同工件經過同機台並使用同製程參數，其產品之機械性質可能不盡一致，若無法合乎客戶需求時，就需要進行重工處理。此外，製程上的平行機台亦可能有新舊與功能性質上之差異，如此亦將造成產出後有不同之不良品機率，譬如產出之工件尺寸不符、直度不平、圓度不合等瑕疵，也必須決定是否進行重

工處理。

由於是多樣少量、客製化的生產模式，通常「重工」的過程，是指工件經檢驗判斷後，現場人員依該工件的瑕疵程度進行判斷，若能於原製程重新加工後就能修復的瑕疵，即於該製程符合其生產條件限制時投入原機台或另一平行機台進行重工處理，再移往下一製程；而無法於原製程重工處理的瑕疵，將依照其瑕疵程度來判斷是否為廢品或移往其它製程再加工處理。

1.2 研究動機

由於生產流程中可能產生必須重工處理之瑕疵工件，導致該製程之實際產出時間較原先的計劃完成時間有所延誤，影響客戶原預定之交期。本研究之個案公司目前大多先將其瑕疵工件轉做無主庫存，待更適合的客戶出現之後再重工處理並將之掛單給該客戶。然而這種做法可能導致惡性循環，致使庫存一直追補訂單交期較晚之客戶原料，並增加無主之逾齡存貨。因此，若能事先做好重工的計劃生產判斷法則，提交合乎實際現況之預計訂單交期，應有利於避免無法判斷重工所導致之逾齡存貨、頻繁重複的轉單與補料動作。

1.3 研究目的

為了解決重工過程會打亂原始排程，進而導致難以準確推算出最佳的生產完成時間，本研究以某公司位於台南一貫化鋼廠的盤元與直棒製程為研究案例，探討單一製程中平行機台之工件重工排程問題。為達到最小化平行機台之最長完工時間的目標，並儘量均衡平行機台之負荷，本研究擬針對單一製程之平行機台重工排程問題提出具有即時性且能有效改善求解品質的啟發式排程演算法。

1.4 研究流程

本文以研究盤元與直棒廠的重工排程問題為主體，首先探討目前產業背景，針對現況問題進行分析以定義問題。在界定研究主題及範圍後，依據研究主題收集及回顧相關文獻；接著針對問題的特性，提出使用貪婪法則之排程演算法，再結合禁忌搜尋法加以改善求解品質，進行數值測試與分析演算法之排程品質與求解效率，最後進行結論，並建議未來可行之研究方向。研究流程如圖 1.2 所示。

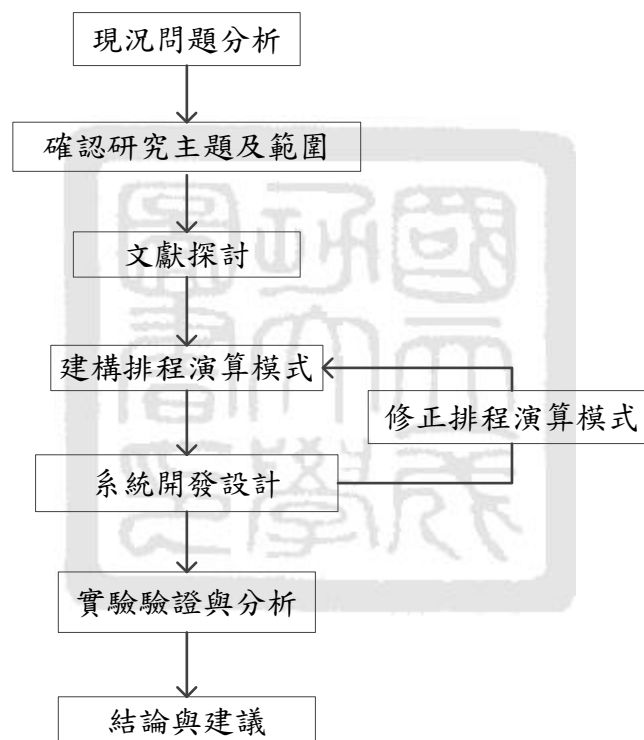


圖 1.2 研究流程圖

1.5 論文架構

本研究共分為五章，各章節的架構簡述如下：

第一章：主要為確定研究主題，描述研究背景、研究目的、研究流程以及論文架構。

第二章：收集並回顧相關文獻，以整理既有的理論與相關文獻研究成果。

第三章：闡述單一製程平行機台工件重工的排程問題，介紹本研究所提出的排程演算法細節。

第四章：撰寫的程式來進行數值測試與分析，整理我們所設計的排程演算法與禁忌搜尋法在求解效率與求解品質上的績效。

第五章：總結本論文並對未來研究發展方向提出建議參考。

本研究依照以上各章節的架構簡述，後續將內容詳細的介紹於後。



2. 文獻探討

本章主要分為四個部份，首先回顧平行機台之排程問題的相關文獻，然後針對重工之瑕疵問題與運作流程進行探討，並彙整文獻所提之排程相關技術方法，接著介紹盤元與直棒製程。

2.1 平行機台之排程問題

不同於單機排程問題，平行機台的排程由於負載平衡與配置方式的不同，總完工時間、最大延遲時間等指標表現便會有所不同。而決定平行機台排程可分兩步驟，首先需決定各工件應於何機台進行加工，再決定各機台配置之工件加工順序（葉子嘉, 2007）。Graham et al. (1979) 提出 $\alpha/\beta/\gamma$ 三種排程分類問題， α 代表機台環境， β 代表機台特性與限制， γ 代表績效衡量準則，以下將分別說明之。

2.1.1 機台環境

Mokotoff (2001) 發表一般排程問題之平行機台環境以 P、Q、R 三種符號來表示，分別代表：完全相同機台 (identical machine, P)、等效機台 (uniform machine, Q)、非相關平行機台 (unrelated parallel machine, R)。完全相同機台之環境可將所有機台視為相同，表示同一件工作不管在哪部機台上進行加工都需要相同的加工時間；等效機台之特性為所有機台具有個別的加工速率，因此若將同一件工作放在不同機台上，可能會因機台加工速率不一而需要不同的加工時間。舉例來說，可做相同加工作業的機台，卻可能因新舊款式之分而造成加工速率的差異；非相關平行機台表示各工作在個別機台上都有其相對應的加工時間，此類型機台環境之問題通常會提供一個對應矩陣以供查詢所有工作在個別機台上所需的加工時間，且各工作在不同機台上的加工時間不一定與機台速率成比例(余侑儒, 2008)。

2.1.2 機台特性與限制

鄧紫文 (2004) 研究整理，依照工件與機台的特性與限制，提出需要考慮的因素如下所示：

■ 釋出時間(Release Time)

工件依到達時間，可以分為靜態模式 (Static) 與動態模式 (Dynamic)，在靜態模式中，工件數目與作業內容均為固定且已知，同一批次的所有工件同時到達等候線且完成所有操作後，才可以再考慮下一批新進的工件；動態模式則是工件可隨著不同時間點被釋放到現行排程中。

■ 順序相依的前置時間 (Sequence Dependent Setup times)

兩個 i, j 工件前後在同一部機台上加工，在 i 完成， j 開始加工前的機台前置時間。

■ 中斷 (Preemptive)

正在加工中的工件是否可被中斷，優先處理其它工件。

■ 先後順序(Precedence Constraint)

明確的先後順序。

■ 機台不連續(Breakdowns)

機器不會永遠持續的運作，可能因為故障或維修而停頓。

■ 候選機器限制(Machine Eligibility Restrictions)

會進行工件加工的候選機台有那些。

■ 排列(permutation)

在流線式(Flow-Shop)生產中，工件加工的順序是先進先出(FIFO)。

■ 阻塞(blocking)

若工件 i 下一部機器的緩衝區已經滿了，則剛進行完工件 i 加工的機器將沒辦法釋放工件 i 。

■ 不等待(no-wait)

工件進入生產系統中，一旦開始加工就必須持續地進行，直到完工，不允許工件

有閒置或等待的狀況發生。

■ 重複循環(recirculation)

工件重複經過同一部機器。

2.1.3 績效衡量準則

早期的排程績效衡量指標皆以單一衡量指標為主，後來則較多使用多重衡量指標，陳建良(1995)彙整與分類排程績效衡量準則如表 2-1。不同的績效衡量指標可能必須設計不同的排程方法，因此訂定排程的衡量指標是研究排程問題的首要工作。

表 2-1 常用的排程績效衡量標準

分類	衡量標準
完工時間相關	平均流程時間(Mean flowtime) 最大流程時間(Maximum flowtime) 流程時間的變異數(Variance of flowtime) 總完工時間(Total completion time) 總加權完工時間(Total weighted completion time) 總時程(Makespan) 生產週期(Cycle time) 單位時間產量(Throughput)
交期相關	平均差異時間(Mean lateness) 平均早交時間(Mean earliness) 平均延遲時間(Mean tardiness) 最大延遲時間(Maximum tardiness) 總延遲時間(Total tardiness) 總加權延遲時間(Total weighted tardiness) 延遲工件數(Number of tardy jobs) 延遲工件比率(Proportion of tardy jobs)
機器及暫存區相關	平均機器使用率(Average utilization) 平均準備時間(Average setup time) 平均在製品數量(Average work in process) 平均等候時間(Average wait time) 最大等候時間(Maximum wait time)

物料搬運相關	平均物料搬運系統使用率 物料搬運系統空車時間
--------	---------------------------

(資料來源：陳建良，1995)

2.2 重工

重工發生在工件經過機台加工後所產生的瑕疵處理流程，在單一機台下，其瑕疵的部份只能在原製程同機台修復；在平行機台下，瑕疵部份的排程處理需考量是否符合機台的生產條件限制及其負荷量。重工會耗費更多的時間及成本來修復工件，以符合品質要求 (Inderfurth et al. ,2005)。

2.2.1 重工之瑕疵問題

由於工件發生的瑕疵狀況不確定，因此大多數相關文獻探討重工的最佳生產批量、重工時最低的生產成本，甚至是重工的最佳訂購決策。Chiu et al. (2004) 假設瑕疵為隨機產生，所有生產的工件都必須檢查，不良品分為可修復與廢棄二種狀況；可修復的不良品經過重工，有可能再產生瑕疵。在此不完美的品質經濟批量模式下，不允許短缺，以最小的總成本建構數學模式，以獲得最佳的生產批量。Chen et al. (2001) 研究具有多個不可靠供應來源的生產庫存系統，透過檢驗與重工，可以改善工件從製程得到的品質改善，該問題有二個交互的決策：(1)從原料的訂購量，(2)工件返回原製程的重工量，該研究設計貪婪演算法確定訂購決策的供應來源與最佳訂購數量。Chiu et al. (2011) 在經濟批量下考慮重工與不同的出貨量，建立數學模型以決定最佳的存貨訂購決策，解決現實生產過程中隨機產生的缺陷不良品問題。

Kang & Shin (2010) 整理分析工件經機台加工產生的瑕疵而導致重工之歷史數據，定期更新並以其推測理論重工率，考慮整備時間與到期日期，提出二種啟發式演算 MRPD (Minimum Rework Probabilities with Due-dates)與 PSBS (Problem-Space-Based Search)以解決平行機台重工排程問題。Shin & Kang (2010) 推導理論重工率，應用貪

筭演算法考慮整備時間與到期日期，以解決 TFT-LCD 廠之平行機台重工排程問題。

2.2.2 重工之運作流程

以單一製程來看，工件經製程加工後，若有產生瑕疵，則返回原製程再加工修復，因此該工件在製程中經歷了二階段的處理。 Inderfurth et al. (2005) 探討兩階段製造系統的單一產品靜態批量生產問題（如圖 2-1 所示），由於新的來料有部份不符合品質規定或經過製程而產瑕疵品，導致必須重工；又因為經濟與環保考量，不考慮出售可重工處理的次級品，該研究並假設經過重工的瑕疵品其品質一定會滿足預先規定的標準。

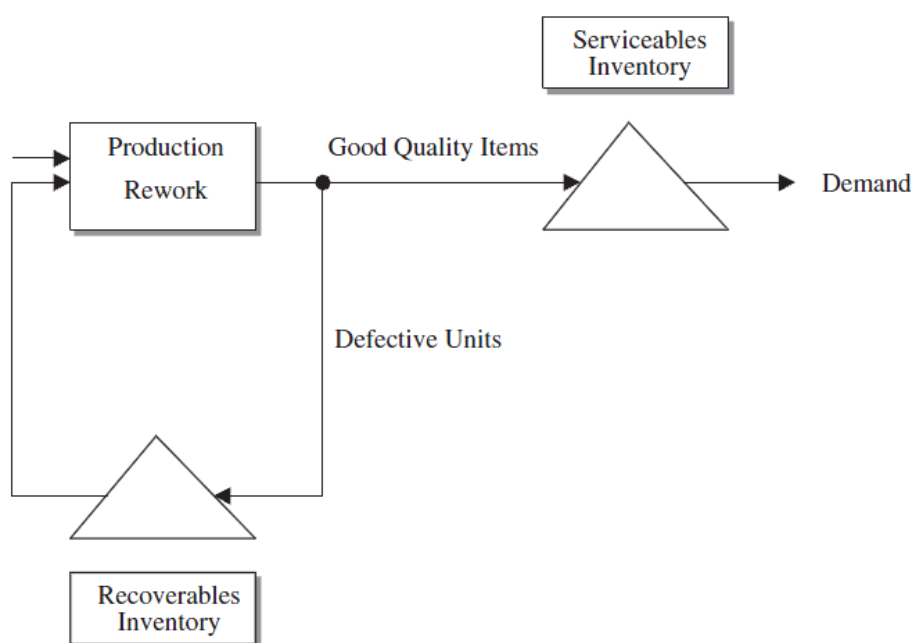


圖 2-1 兩階段製造系統的重工結構圖 資料來源：Inderfurth et al. (2005)

Kang & Shin (2010) 指出平行機台中每個機台的生產條件與重工率不太一致，但每一機台的加工時間卻大致相同。為了減少問題的複雜性，重工的時間等於工件在機台的加工時間，等待的工件在排入符合條件之機台加工後，必須檢驗是否產生瑕疵工件。若有，則將之送回某機台以重工處理之，如圖 2-2 所示。

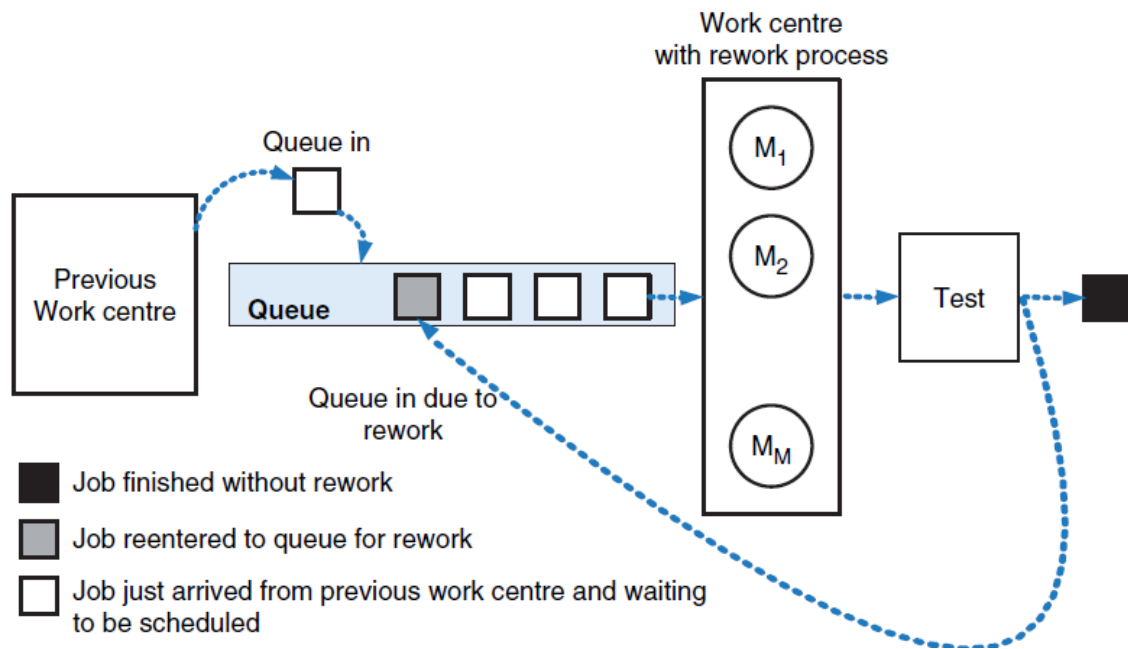


圖 2-2 平行機台重工生產系統的重工結構圖 資料來源：Kang & Shin (2010)

2.3 排程相關技術

排程技術經過數十年的發展，已出現不少常用的法則與解法可供使用，主要分成派工法則(dispatching Rule)、精確最佳解方法(Exact Optimization Methods)、近似解方法(Approximation Methods)、電腦模擬方法(Computer Simulation Approaches)等(Jain & Meeran, 1998; Kelton *et al.*, 2009)，以下將個別說明之。

2.2.1 派工法則

派工法則是一種決定工件加工順序的依據，針對工件或資源計算各作業的優先順序值，然後再根據此優先順序值大小，指派優先值順序最高的工件。派工法則大多使用數種貪婪(greedy)準則來進行工件排序，其優點在於容易建構且求解速度快，但卻無法同時考慮到整個排程計畫(Herrmann, 2001)，因此對較複雜的排程問題無法保證可求得最佳解。陳春龍 (2006) 依 Rajendran *et al.*(1999) 彙整派工法則歸類成下列五種類型：

■ 與處理時間(Processing Time)相關的法則

此種派工法只考慮到工件在機器上的處理時間。例如：SPT (Shortest Processing Time)，即是考量最小處理時間優先。

■ 與交期相關的法則

此種派工法考量與交期相關的參數，且在與交期相關的指標上表現較好。例如：EDD (Earliest Due Date)為交期最早的工件優先的法則。

■ 與處理時間或是交期皆無關的法則

FIFO (First In First Out)是此類派工法則中的代表。此法則只考慮工件到達機器的時間，越早到達者優先權越高。

■ 與現場條件(Shop Floor Condition)有關的法則

此類派工法則考慮到現場機器的條件，而非只是單純的考量工件的特性。最著名的派工法則為 WINQ (Work In Next Queue)，此項法則考慮到下一個操作機器的壅塞程度，被廣泛使用在零工式生產中(Conway, 1965; Haupt, 1989)。

■ 包含上述四種類型兩種以上的混合法則

有很多派工法則同時考慮處理時間及交期，例如：CR (Critical Ratio) (Haupt, 1989) 同時考慮到工件寬鬆時間(Slack Time)與剩餘操作步驟數的關係。而同時考慮處理時間、交期和現場條件的則有 PT+WINQ+SL (Processing Time Plus Work In Next Queue Plus Negative Slack)(Oliver & Chandrasekharan, 1997)。

Subramaniam et al. (2000) 指出近年來派工法則的研究分成兩大主流：第一種為修改已經存在的派工法則，此種方法將幾個績效指標較好的派工法則合併成一個派工法則(Combination of Dispatching Rules)或是促進及縮短派工法則(Expediting and Truncating Dispatching Rules)；第二種主流重心為在派工法則中加入反覆的方法(Iterative Method)，屬於這個主流的方法主要有三種，分別為執行多次派工法則的多次啟發式方法(Multiple-Pass Heuristic Rule)、考慮未來會發生情形的前瞻式啟發性方法(Look-Ahead Heuristic Rules)及考慮重排的啟發式法則(Rescheduling Heuristic Rules)。

2.2.2 精確最佳解方法

根據系統的限制條件和目標，利用作業研究的數學規劃模式求得最佳解。此方法雖能保證得到理論最佳解，但實作上經常需耗費許多時間。舉例來說，許多排程問題即使以目前最先進的數學規劃軟體求解，仍僅能處理小規模的問題，根本無法滿足短時間內處理現實製程之需求。

2.2.2 近似解方法

此方式主要是希望在有限的計算時間內，求得一逼近最佳解的可行解，尤其適合求解大型的排程問題。近似解方法可以分成兩大部分：一是人工智慧(Artificial Intelligence)，另一是區域搜尋法(Local Search)。前者主要的方法包括：限制條件滿足法(Constraint Satisfaction)、類神經網路法(Neural Network)或是不同方法合併使用；後者常見的是禁忌搜尋法(Tabu Search)、基因演算法(Genetic Algorithm)、模擬退火法(Simulated Annealing)，以及一併使用多種方法等方式，其求解績效亦相當理想(王治平，2003)。

2.2.3 電腦模擬方法

由於真實的製造環境過於複雜，不容易透過理論模式以詳盡敘述系統特徵；反之，電腦模擬卻可在適當的成本之下，利用高階的模擬語言來呈現幾近真實的系統，提供較為真實的介面給予專家參考。然而模擬的結果並無法保證求得最佳的排程結果，亦不能提供排程的即時決策，這些產生的結果僅能讓決策者知道所輸入的排程策略究竟有多好，或者僅能輸入不同的策略來比較彼此之間的優劣。因此，許多的電腦模擬研究依舊使用常見的簡單派工法，然而在一些先進的系統中已可內建原有的模擬核心技術，藉以提出一些簡單的決策(郭宜雍，2004)。

2.4 盤元與直棒製程

線材及棒鋼產品是以小鋼胚加熱後軋延而成，依外形可區分為盤元及直棒兩類，如圖 2-3 所示。盤元再加工後可生產鋼線、螺絲螺帽、鐵絲等下游產品；直棒可提供下游加工應用於電腦周邊之設備零件、工業用途之軸件、船舶用件、醫療、化學工業設備等零組件，其生產流程分別說明如下：

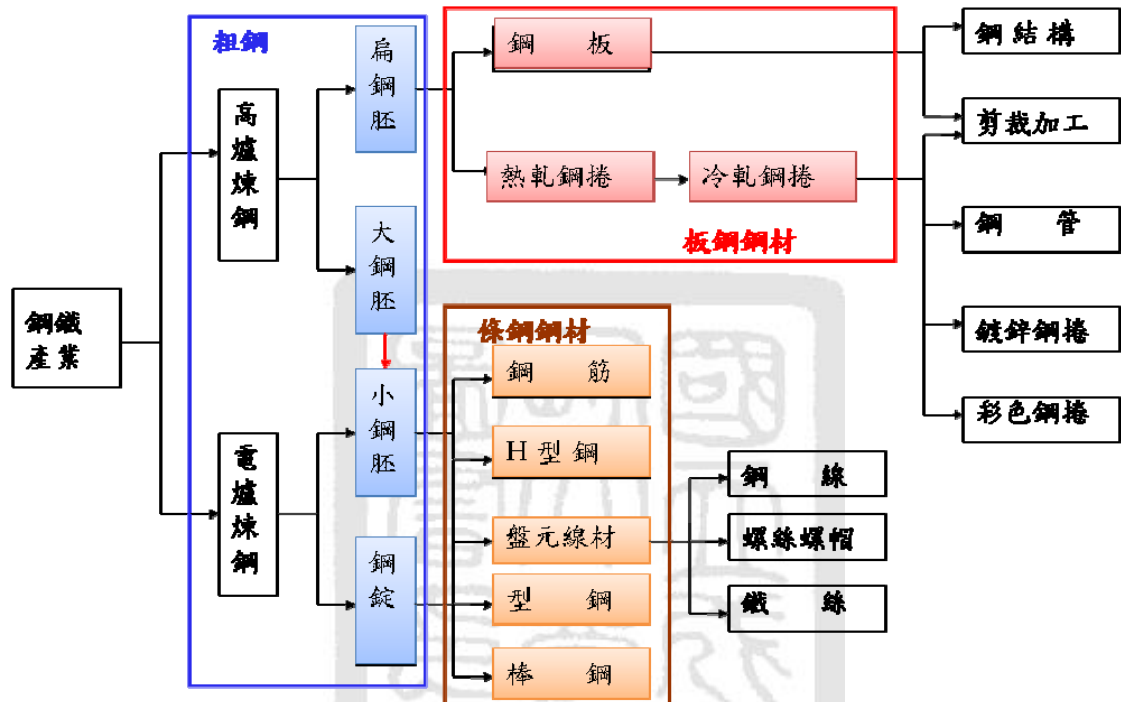


圖 2.3 鋼鐵產業架構

資料來源：萬用手冊，台灣經濟研究院產經資料庫整理（2007 年 6 月）。

■ 軋鋼流程

將小鋼胚加熱至可供軋延之溫度，經由不同功能的軋機，其旋轉軋輥間的壓力改變小鋼坯形狀，產出線材盤元、鋼棒盤元與熱軋直棒，如圖 2-4 所示，其目的一方面是為了得到需要的形狀，另一方面是為了改善鋼的內部質量。

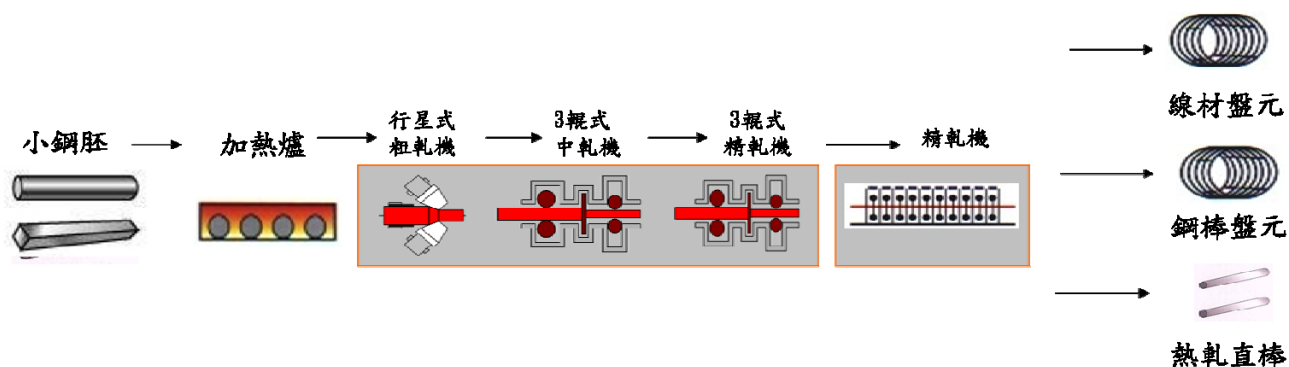


圖 2.4 軋鋼流程圖

■ 固溶與酸洗製程

固溶製程為將線材盤元、鋼棒盤元加溫到適當溫度並維持此溫度一段時間，再將其緩慢冷卻，其功用在於恢復因冷加工而降低的性質，增加柔軟性、延性和韌性，並釋放內部殘留應力、以及產生特定的顯微結構；酸洗製程為利用鹽浴、硫酸、混酸等酸洗製程，除去表面之氧化皮層，並增加潤滑及防銹作用，使表面呈現灰白金屬色澤，如圖 2-5 所示。

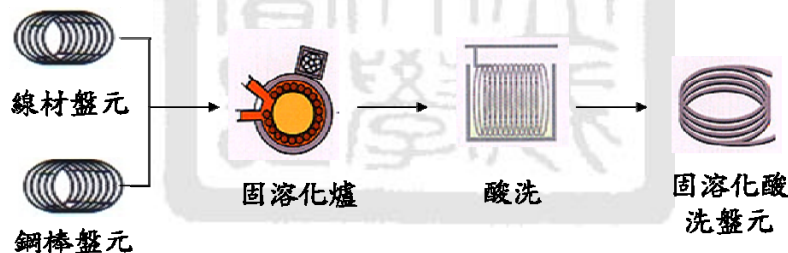


圖 2.5 固溶酸洗流程圖

■ 細線盤元與冷精直棒流程

熱軋直棒可分為經由固溶、矯直、削皮、軋光等製成削皮棒或軋光棒，如圖 2-6 所示；矯直製程為熱軋直棒從上下兩排相互交錯排列的輥子之間通過時，因受到多次反覆彎曲而得到的對形狀缺陷的矯正；削皮製程為以車削原理去除表面銹皮及缺陷，獲得較佳表面及尺寸精度；軋光製程為使表面粗糙度降低，以獲得光亮平整表面。

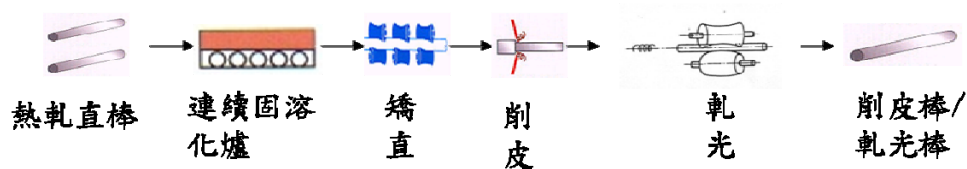


圖 2.6 直棒流程圖

固溶化酸洗盤元、削皮棒或軋光棒，經由披覆與冷抽加工製成細線盤元與冷精直棒，冷精直棒包含圓棒及各式異形棒，如圖 2-7 所示。鋼材之冷精加工其變形磨擦壓力極大；披覆製程為經過酸洗除銹進行表面皮模的披覆，以保護加工表面確保產品之品質；冷抽製程為經過眼模模具以拉長成形，同時改善屈服強度和抗拉強度的過程。



圖 2.7 細線盤元與冷精直棒流程圖

生產為盤元或直棒的加工製程中，在細線盤元與冷精直棒流程中，其製程大部份都為平行機台，相關的生產條件限制、製程參數與整備時間皆類似，所以本研究以單一製程之平行機台為研究對象，在有重工的問題下，推算工件的最早完工時間，以期將此排程方法延伸應用到各製程之平行機台。

2.5 小結

本研究以盤元與直棒製程面臨非相關平行機台之工件重工排程問題，假設工件經

機台重工之歷史數據，透過整理分析並定期更新，推測理論重工率。考慮整備時間與訂單交期，並評估處理本問題績效指標較好的兩種以上之派工法則，依平行機台生產條件限制，投入到符合條件的機台加工。第三章將發展貪婪演算法與禁忌搜尋法，對工件與重工工件做最適合的生產順序安排，並推算符合現場實際生產狀況的排程完工時間。



3. 研究方法

本章將說明本研究提出的方法。第一節描述問題；第二節說明基本假設；第三節詳述我們提出之貪婪演算法原理與步驟；第四節則說明如何使用禁忌搜尋法進一步改善排程結果。

3.1 問題描述

盤元與直棒廠的製程加工可歸類於單一製程平行機台的排程問題，在單一製程排隊等候的工件，會依平行機台生產條件限制，投入到符合條件的平行機台加工；而完工後的工件立刻經檢驗判定，若為正常工件則會移往下一製程；若為瑕疵工件則會依平行機台生產條件限制，馬上投入到符合條件的平行機台再重工處理以修復瑕疵；而針對再次重工處理仍無法修復的瑕疵工件，則將之判為廢品或移往其它製程加工處理，如圖 3.1 所示。

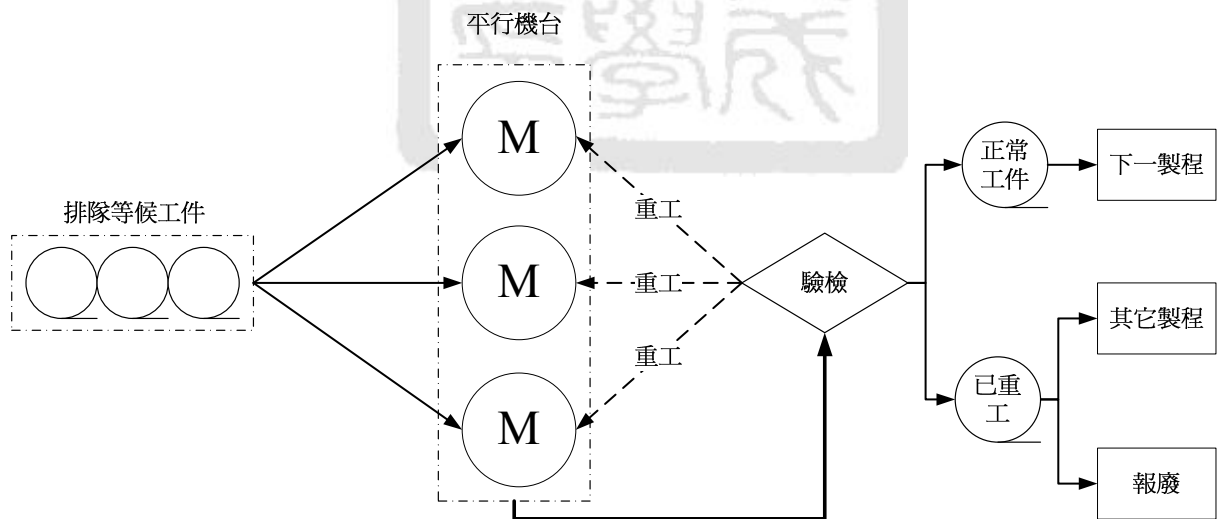


圖 3.1 單一製程平行機台重工流程圖

平行機台由於購置時間或生產條件限制，或多或少會產生瑕疵工件，而每一工完全產出後才開始檢驗。製程中如須重工，將增加重工與等待重工的時間，導致該製程

之實際產出時間與其原先規劃的完成時間有所延遲；若無法預估製程中之瑕疵工件發生機率，將更難預測可能需增加之排程時間。在假設各平行機台產生瑕疵工件的機率為已知的情況下，本研究擬設計排程演算法，儘量均衡平行機台之負荷來決定各工件及其重工工件應在哪一平行機台之排程順序，達到最小化平行機台之最晚完工時間的目標。

3.2 基本假設

本研究的基本假設如下：

- 平行機台會因為新舊與功能之差異，有不同的瑕疵工件產生機率。
- 平行機台因功能之差異，所以工件的加工時間可能不一致。
- 工件會因平行機台生產條件的限制而只能在某些特定的機台上加工。
- 當工件進入機台之後，該機台必須在結束其加工之後才能進行下個工件加工。
- 由於是多樣少量、客製化的生產模式，故產生的瑕疵工件一律需經重工處理。
- 瑕疵工件在單一製程不論經那一台平行機台重工，因為物性或成本關係只會重工一次。
- 工件若必須重工處理，需在工件完工且檢驗完成後才能開始重工。
- 由於平行機台相鄰近，所以不考慮重工之工件搬運時間。
- 不考慮經過製程產生損耗的問題。
- 不考慮平行機台故障的情形發生。
- 不考慮重工的加工成本

3.3 貪婪演算法模式建立與說明

本節中將說明本研究提出的排程方法：(1) EDD+SPT with WIDR (Work In Dia Range)演算法；(2)貪婪演算法 ARI (All Reworks Insertion Rule)；(3) 貪婪演算法 SRI

(Single Rework Insertion Rule)；以及(4)禁忌搜尋法，並以簡單實例說明。

3.3.1 參數說明

M :所有機台集合

J :所有工件之集合

J^D :所有已排完程（原工件及其重工工件）之工件集合， D 表示 Done

J^G :所有當下之原工件已排完程但其重工工件尚未被排程之工件集合， G 表示 On-Going

J^W :所有當下之原工件未排程之工件集合， W 表示 Will be done

i :第 i 個原工件(即原工件 i)

M_i :可處理原工件 i 之機台集合

m_i :實際處理原工件 i 之機台

\bar{m}_i :實際處理原工件 i 重工工件之機台

u :第 u 個機台(即機台 u)

$\Delta t_{i,j,u}^r$:原工件 i 於機台 u 轉換成處理原工件 j （或其重工工件）之整備時間

$top(u)$:當下排入機台 u 的最後一件原工件編號

$top^r(u)$:當下排入機台 u 的最後一件重工工件編號

$\Delta t_{i,u}^p$:原工件 i 於機台 u 之處理時間

$\Delta \bar{t}_{i,u}^p$:原工件 i 之重工工件於機台 u 之處理時間

Δt_i^c :原工件 i 的檢驗時間(假設檢驗時間與機台無關)

t_i^s :原工件 i 的實際開始處理時間

t_i^f :原工件 i 的實際完工時間

$\tilde{t}_{i,u}^f$:原工件*i*於機台*u*之的暫存完工時間

$\bar{t}_{i,u}^s$:原工件*i*之重工工件在機台*u*的實際開始處理時間

\bar{t}_i^s :原工件*i*之重工工件的實際開始處理時間

\bar{t}_i^f :原工件*i*之重工工件的實際完工時間

$\tilde{t}_{i,u}^s$:原工件*i*之重工工件在機台*u*的暫存開始處理時間

\tilde{T}_u^a :當下機台*u*的暫存可開始處理時間

3.3.2 EDD+SPT with WIDR 演算法

以工件資料如表 3.1 與尺寸級距資料如表 3.2 為例，整理出工件的交期與工件的尺寸級距。若在同一機台中相鄰兩排序工件的尺寸級距不同，則必須在工件轉換時增加機台的整備時間。所以我們先運用 EDD 的派工法則，以工件的交期時間先後做第一排序條件，再以 SPT with WIDR 的派工法則做第二排序條件。WIDR 的命名規則參考 WINQ，兩者都是考慮到現場機器的條件，其中 WINQ 預先考慮到下一個操作機器的壅塞程度(Conway, 1965; Haupt, 1989)，而 WIDR 則預先考慮不同尺寸工件造成機台更換設備的整備時間。SPT with WIDR 為在同一交期時間中，依各尺寸級距其加總之處理時間由小到大排序，為了避免增加整備時間，所以加入工件的尺寸級距判斷，也就是先針對處理時間最短的一組尺寸級距，將該尺寸級距的所有工件依 SPT 做排序，再處理第二短處理時間的尺寸級距工件，展開如表 3.3 所示之工件排序資料。

表 3.1 工件資料

工件	尺寸	重量(公斤)	交期
ID1	8	500	2013/6/20
ID2	15	1000	2013/6/23
ID3	4	1300	2013/6/25
ID4	19	1000	2013/6/27
ID5	10	550	2013/6/21
ID6	7	1500	2013/6/22
ID7	11	1000	2013/6/25
ID8	10	900	2013/6/28
ID9	5	500	2013/6/25
ID10	5	1200	2013/6/25

表 3.2 尺寸級距資料

尺寸 級距	尺寸min	尺寸max	整備時 間
01	0.1	1	10
02	1.1	2	10
03	2.1	3	10
04	3.1	4	10
05	4.1	5	10
06	5.1	6	10
07	6.1	7	10
08	7.1	8	10
09	8.1	9	10
10	9.1	10	10
11	10.1	11	10
12	11.1	12	10
13	12.1	13	10
14	13.1	14	10
15	14.1	15	10
16	15.1	16	10
17	16.1	17	10
18	17.1	18	10
19	18.1	19	10
20	19.1	20	10

表 3.3 EDD+SPT with WIDR 工件排序

工件	交期	重量(kg)	尺寸 級距
ID1	2013/6/20	500	08
ID5	2013/6/21	550	10
ID6	2013/6/22	1500	07
ID2	2013/6/23	1000	15
ID7	2013/6/25	1000	11
ID3	2013/6/25	1300	04
ID9	2013/6/25	500	05
ID10	2013/6/25	1200	05
ID4	2013/6/27	1000	19
ID8	2013/6/28	900	10

3.3.3 ARI 演算法

為了均衡平行機台的工作負荷並儘量滿足訂單的交期，ARI 演算法先將所有原工件排入平行機台後（步驟一至二），再一一安插其重工工件的排程（步驟三至六），其流程圖如圖 3.2 所示，而詳細步驟如下：

步驟一：原工件先依 EDD+SPT with WIDR 演算法排序，亦即先依照派工法則 EDD

為第一排序，再以 SPT with WIDR 為第二排序，將原工件排列等候順序。

針對每原工件 i 及可被處理的機台 $u \in M_i$ 而言，令

$$\tilde{T}_u^a = \text{top}(u) = \text{top}^r(u) = 0, \quad m_i = \bar{m}_i = 0.$$

步驟二：依步驟一之原工件順序，逐一將各原工件排程。其中，原工件 i 將被排

在機台 $u^* = \arg \min_{u \in M_i} \{\tilde{T}_u^a + \Delta t_{i, \text{top}(u), u}^r\}$ 的最後面，亦即 $m_i = u^*$ 。令

$$t_i^s = \tilde{T}_{u^*}^a + \Delta t_{i, \text{top}(u^*), u^*}^r, \quad t_i^f = t_i^s + \Delta t_{i, u}^p, \quad \tilde{T}_{u^*}^a = t_i^f, \quad \text{top}(u) = i. \text{ 重複以上程序直}$$

至全部的原工件皆完成排程。

步驟三：針對目前尚未完成其重工排程的原工件 i （即 $\bar{m}_i = 0$ ），選擇其重工工件

可被最早處理的機台 $v_i^* = \arg \min_{v \in M_i} \{\tilde{T}_v^a + \Delta t_{i, \text{top}(v), v}^r + \Delta t_i^c\}$ ，並在所有的這

些工件中選擇最早能被處理其重工工件的工件

$i^* = \arg \min_i \{ \tilde{T}_{v_i^*}^a + \Delta t_{i, \text{top}(v_i^*), v_i^*}^r + \Delta t_i^c \}$ ，以開始其重工排程。

步驟四：檢查該原工件 i^* 之處理機台 m_{i^*} 在其完工時間 ($t_{i^*}^f$) 後是否已被安插別

的重工工件，若有 (即 $\text{top}^r(m_{i^*}) \neq 0$)，且該機台的最後一個被安插重工

的工件之原工件 $\text{top}^r(m_{i^*})$ 之完工時間 $\bar{t}_{\text{top}^r(m_{i^*})}^f$ 晚於 $t_{i^*}^f + \Delta t_{i^*}^c$ ，則且 i^* 的重工

工件暫存開始處理時間 $\tilde{t}_{i^*, m_{i^*}}^s = \bar{t}_{\text{top}^r(m_{i^*})}^f$ ；反之 (即 $\text{top}^r(m_{i^*}) = 0$ ，或

$\text{top}^r(m_{i^*}) \neq 0$ 但 $\bar{t}_{\text{top}^r(m_{i^*})}^f < t_{i^*}^f + \Delta t_{i^*}^c$)，則 i^* 的重工工件的開始處理時間

$$\tilde{t}_{i^*, \bar{m}_{i^*}}^s = t_{i^*}^f + \Delta t_{i^*}^c。$$

步驟五：以步驟四一一檢驗所有其它可處理原工件 i^* 之重工工件機台

$v \in M_{i^*} - \{m_{i^*}\}$ ，並計算其在機台 v 的可能開始重工處理時間 $\tilde{t}_{i^*, v}^s$ ，再取其

最小者所對應的機台 $\bar{m}_{i^*} = \arg \min_{v \in M_{i^*}} \tilde{t}_{i^*, v}^s$ 來處理原工件 i^* 之重工工件。

設定 $\bar{t}_{i^*}^s = \tilde{t}_{i^*, \bar{m}_{i^*}}^s$ ， $\bar{t}_{i^*}^f = \bar{t}_{i^*}^s + \Delta \bar{t}_{i^*, u}^p$ ， $\text{top}^r(\bar{m}_{i^*}) = i^*$ 。

步驟六：返回步驟三直至所有工件 i 與重工工件 i^* 皆完成排程才結束。

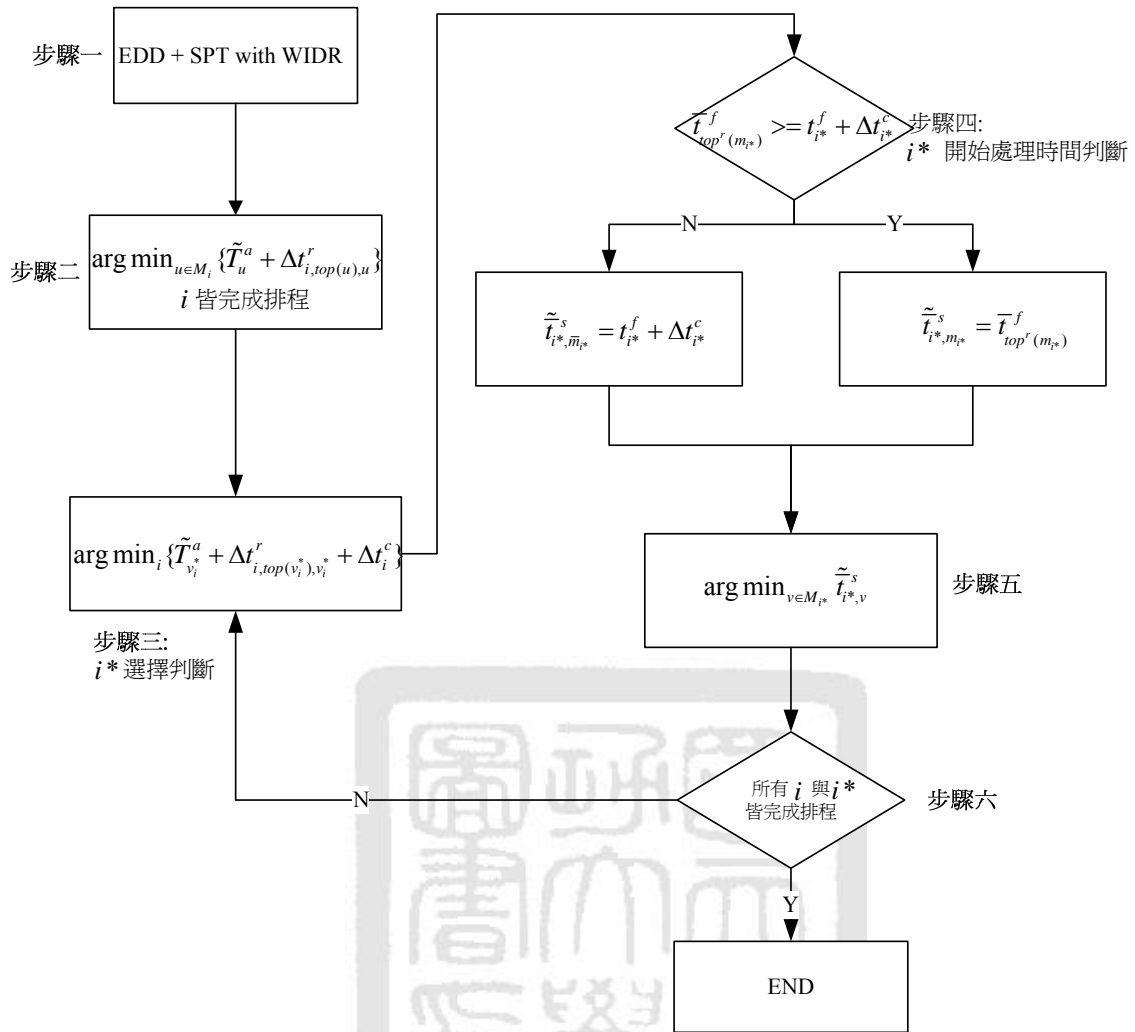


圖 3.2 ARI 演算流程圖

根據上述的步驟提出簡例說明，工件資料如表 3.1，並加入平行機台處理時間與重工率資料如表 3.4 與平行機台資料如表 3.5，做生產條件限制與機台整備時間判斷，依步驟一產生如表 3.6 工件的排列等候順序及可在平行機台生產的數據。

表 3.4 平行機台處理時間與重工率資料

機台	尺寸min	尺寸max	處理時間(分/公斤)	重工率
A	0.1	11	0.30	0.15
B	5.1	16	0.28	0.12
C	9.1	20	0.25	0.1

表 3.5 平行機台資料

機台	尺寸 min	尺寸 max	尺寸 級距	整備 時間	機台	尺寸 min	尺寸 max	尺寸 級距	整備 時間	機台	尺寸 min	尺寸 max	尺寸 級距	整備 時間
A	0.1	1	01	10	B	5.1	6	06	10	C	9.1	10	10	10
A	1.1	2	02	10	B	6.1	7	07	10	C	10.1	11	11	10
A	2.1	3	03	10	B	7.1	8	08	10	C	11.1	12	12	10
A	3.1	4	04	10	B	8.1	9	09	10	C	12.1	13	13	10
A	4.1	5	05	10	B	9.1	10	10	10	C	13.1	14	14	10
A	5.1	6	06	10	B	10.1	11	11	10	C	14.1	15	15	10
A	6.1	7	07	10	B	11.1	12	12	10	C	15.1	16	16	10
A	7.1	8	08	10	B	12.1	13	13	10	C	16.1	17	17	10
A	8.1	9	09	10	B	13.1	14	14	10	C	17.1	18	18	10
A	9.1	10	10	10	B	14.1	15	15	10	C	18.1	19	19	10
A	10.1	11	11	10	B	15.1	16	16	10	C	19.1	20	20	10

表 3.6 工件排序及可在平行機台生產的數據

工件	交期	重量 (kg)	尺寸 級距	機台	處理 時間	機台	處理 時間	機台	處理 時間
ID1	2013/6/20	500	08	A	151	B	141		
ID5	2013/6/21	550	10	A	166	B	155	C	138
ID6	2013/6/22	1500	07	A	451	B	421		
ID2	2013/6/23	1000	15			B	281	C	250
ID7	2013/6/25	1000	11	A	301	B	281	C	250
ID3	2013/6/25	1300	04	A	391				
ID10	2013/6/25	1200	05	A	361				
ID9	2013/6/25	500	05	A	151				
ID4	2013/6/27	1000	19					C	250
ID8	2013/6/28	900	10	A	271	B	253	C	225

依步驟二，第一順位是 ID1，ID1 符合平行機台生產條件限制有 A 機台與 B 機台，由於未有工件投入，所以平行機台的可開工時間皆為 0 分鐘且未有尺寸級距。ID1 在 A 機台的處理時間 151 分鐘，加上尺寸級距不一致整備時間 10 分鐘，其加總的完工時間為 161 分鐘；在 B 機台的處理時間 141 分鐘，加上尺寸級距不一致的整備時間 10 分鐘，其加總的完工時間為 151 分鐘，可知 ID1 在 B 機台的完工時間最早，所以將 ID1

排入到 B 機台生產。由於尚有工件未排程，因此返回步驟二處理第二順位工件 ID5。

由於 A、B、C 三機台皆可處理 ID5，其中 ID5 在 A 機台（可開工時間為 0 分）的處理時間 161 分鐘加上尺寸級距不一致的整備時間 10 分鐘，其加總完工時間為 176 分鐘；在 B 機台（可開工時間為 151 分，因先前已投入 ID1 的完工時間為 151 分）的處理時間 161 分鐘以及尺寸級距不一致的整備時間 10 分鐘，其加總完工時間為 $151+161+10=316$ 分；在 C 機台（可開工時間為 0 分）的處理時間 138 分鐘加上尺寸級距不一致的整備時間 10 分鐘，其加總完工時間為 148 分鐘為最早，所以將 ID5 排入到 C 機台生產。第三順位的 ID6 可在 A 或 B 機台處理，ID6 在 A 機台（可開工時間為 0 分）的處理時間 451 分鐘加上尺寸級距不一致的整備時間 10 分鐘，其加總完工時間為 461 分；在 B 機台（可開工時間為 151 分）的處理時間 421 分鐘加上尺寸級距不一致的整備時間 10 分鐘，其加總完工時間為 $151+421+10=582$ 分，所以 ID6 在 A 機台的完工時間為最早，所以將 ID6 排入到 A 機台生產；依此類推，如圖 3.3 所示將所有排列等候的工件排入平行機台。

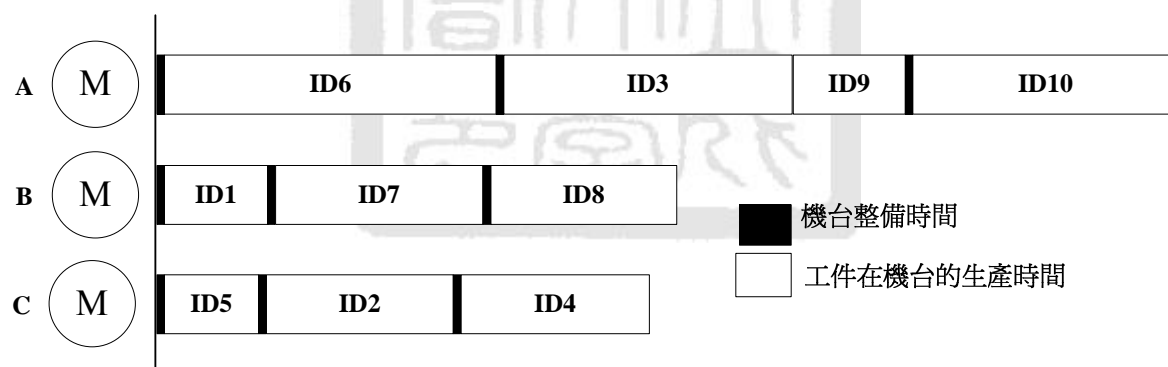


圖 3.3 工件排入平行機台示意圖

若已完成所有工件的排程，則進行步驟三，假設檢驗時間為每公斤 0.05 分鐘，ID6 重量為 1500 公斤共需花費 75 分鐘檢驗，因此其可開始重工時間為 $461+75=536$ 分；ID1 重量為 500 公斤共需花費 25 分鐘檢驗，因此其可開始重工時間為 $151+25=176$ 分；ID5 重量為 550 公斤共需花費 27.5 分鐘檢驗，因此其可開始重工時間為 $148+27.5=176$ 分；所以最早的可開始重工時間為 ID1 與 ID5，假設先選擇 ID1 做重工插入排序如圖

3.4 所示。

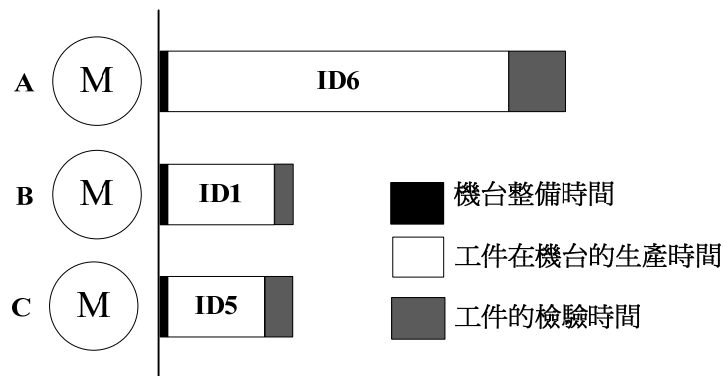


圖 3.4 工件檢驗完成時間示意圖

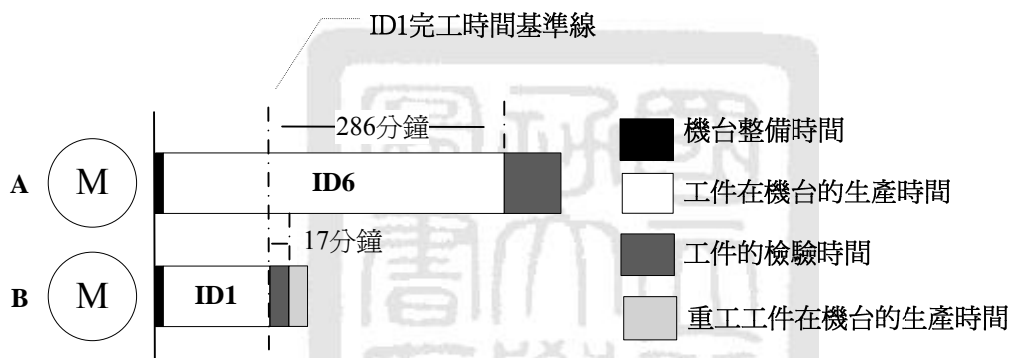


圖 3.5 平行機台插入重工工件排程示意圖

依步驟四，之後被安插重工的工件不存在，工件可重工處理時間為原工件驗檢完工時間，以 ID1 符合機台 A 與機台 B 的生產條件限制，依步驟五，比對在符合平行機台生產限制條件最小閒置等待的重工處理時間。ID6 完工時間 461 分鐘大於 ID1 檢驗的完成時間 176 分鐘，所以若要在 ID6 後重工需等待 285 分鐘，再加上尺寸級距不一致的整備時間 10 分鐘*B 機台的重工率 0.12，共需等待 286 分鐘；ID1 檢驗的完成時間 176 分鐘大於 ID1 完工時間 151 分鐘，所以若要在 ID1 後重工需等待 25 分鐘，因此在 ID1 後重工等待時間是最小，所以將 ID1 的重工安排在 ID1 後如圖 3.5 示，ID1 的重工的完成時間為 193 分鐘，ID1 完工時間 151 分鐘加上等待時間 25 分鐘，再加上重工工件處理時間，ID1 重量 500 公斤*B 機台的重工率 0.12*B 機台的處理時間 0.28

四捨五入為 17 分鐘。

進行步驟六是否原工件皆完成重工工件排程，還有原工件未完成，所以返回步驟三，其檢驗的完成時間最早的為 ID5，在 ID5 後重工等待時間是最小，所以將 ID5 的重工安排在 ID5 後。執行步驟六卻返回步驟三，其檢驗的完成時間最早的為 ID2，ID2 符合機台 B 與機台 C 的生產條件限制，ID2 檢驗的完成時間 500 分鐘大於 ID7 完工時間 484 分鐘，所以若要在 ID6 後重工需等待 16 分鐘，再加上尺寸級距不一致的整備時間 10 分鐘*C 機台的重工率 0.1，共需等待 17 分鐘；ID2 檢驗的完成時間 500 分鐘大於 ID2 完工時間 450 分鐘，所以若要在 ID2 後重工需等待 50 分鐘，所以將 ID2 的重工安排在 ID7 後。再執行步驟六卻返回步驟三，其檢驗的完成時間最小的為 ID7，在 ID6 後重工等待時間是最小，所以將 ID7 的重工安排在 ID6 後，ID7 的重工的完成時間為 509 分鐘，ID6 完工時間 461 分鐘加上等待時間 11 分鐘，再加上尺寸級距不一致的整備時間 10 分鐘*B 機台的重工率 0.12，再加上重工工件處理時間，ID7 重量 1000 公斤*B 機台的重工率 0.12*A 機台的處理時間 0.3 四捨五入為 36 分鐘。

再執行步驟六卻返回步驟三，其檢驗的完成時間最小的為 ID6，由於 ID6 之後有 ID7 的重工工件簡稱為 ID7R。ID6 重量為 1500 公斤，其檢驗完成時間為完工時間 461 分鐘加上檢驗時間 30 分鐘為 491 分鐘，所以 ID7R 的完工時間 509 分鐘大於 ID6 檢驗完成時間，如圖 3.6 所示，由於 ID6 的重工需等 ID7R 完工才能開始重工處理，ID6 重工工件可重工開始處理時間為 ID7R 的完工時間為 509 分鐘。

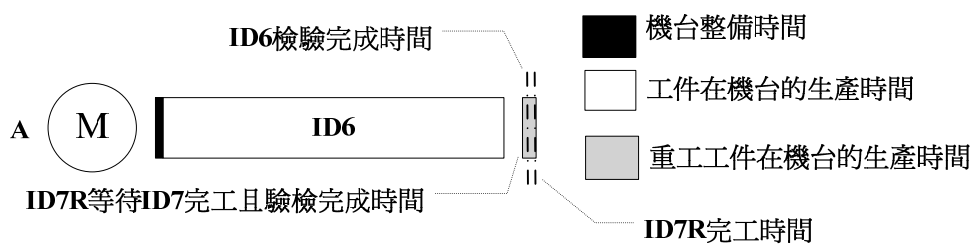


圖 3.6 工件檢驗時間與完工時間比對示意圖

其餘類推，依步驟六直到將原工件皆完成重工工件排程為止，假設檢驗時間為每公斤 0.05 分鐘運算完的數據如表 3.7 所示。

表 3.7 ARI 排程運算數據

機台A					機台B					機台C				
工件	生產 時間	完成 時間	等待 時間	整備 時間	工件	生產 時間	完成 時間	等待 時間	整備 時間	工件	生產 時間	完成 時間	等待 時間	整備 時間
ID6	451	461	0	10	ID1	141	151	0	10	ID5	138	148	0	10
ID7R	36	509	11	1	ID1R	17	193	25	0	ID5R	14	190	28	0
ID6R	67	652	75	1	ID7	281	484	0	10	ID2	250	450	0	10
ID3	391	1053	0	10	ID2R	28	529	16	1	ID4	250	710	0	10
ID3R	58	1176	65	0	ID8	253	792	0	10					
ID9	151	1327	0	0	ID4R	28	853	32	1					
ID9R	22	1374	25	0	ID8R	30	884	0	1					
ID10	361	1745	0	10										
ID10R	54	1859	60	0										

3.3.4 SRI 演算法

為了從另一角度滿足上述需求，SRI 演算法規劃將原工件排入平行機台後，再判斷是否要插入重工工件的排程，還是依序等候之原工件的排程，如圖 3.7 所示，其步驟如下：

步驟一：同 ARI 演算法之步驟一，可得原工件之所有排序。

步驟二：依步驟一之原工件順序，逐一將各原工件或其重工工件排程。原工件 i 及

可被處理的機台 $u \in M_i$ 而言，其完工時間 $\tilde{t}_{i,u}^f$ 加總完工時間最小的機台

$$\arg \min_{v \in M_i} \{\tilde{t}_{i,u}^f + \Delta t_{i,top(u),u}^r\}, \text{ 且 } \arg \min_{v \in M_i} \{\tilde{t}_{i,u}^f + \Delta t_{i,top(u),u}^r\} \geq \tilde{t}_{i,u}^f + \Delta t_{i,top(u),u}^r,$$

則選擇原工件 i 的排序，反之 ($\arg \min_{v \in M_i} \{\tilde{t}_{i,u}^f + \Delta t_{i,top(u),u}^r\} < \tilde{t}_{i,u}^f + \Delta t_{i,top(u),u}^r$ 或

i 不存在)，則選擇重工工件 i^* 的排序。

步驟三：針對目前尚未完成其重工排程的原工件 i (即 $\bar{m}_i = 0$)，選擇其重工工件

可被最早處理的機台 $v_i^* = \arg \min_{v \in M_i} \{\tilde{T}_v^a + \Delta t_{i,top(v),v}^r + \Delta t_i^c\}$ ，並在所有的這些

工件中選擇最早能被處理其重工工件的工件

$i^* = \arg \min_i \{ \tilde{T}_{v_i^*}^a + \Delta t_{i, \text{top}(v_i^*), v_i^*}^r + \Delta t_i^c \}$ ，以開始其重工排程。

步驟四：檢查該原工件 i^* 之處理機台 m_{i^*} 在其完工時間 ($t_{i^*}^f$) 後是否已被安插別的原工件或別的重工工件，若有 (即 $\text{top}^r(m_{i^*}) \neq 0$)，且該機台的最後一個被安插原工件或重工的工件之原工件 $\text{top}^r(m_{i^*})$ 之完工時間 $\bar{t}_{\text{top}^r(m_{i^*})}^f$ 晚於 $t_{i^*}^f + \Delta t_{i^*}^c$ ，則且 i^* 的重工工件暫存開始處理時間 $\tilde{t}_{i^*, m_{i^*}}^s = \bar{t}_{\text{top}^r(m_{i^*})}^f$ ；反之 (即 $\text{top}^r(m_{i^*}) = 0$ ，或 $\text{top}^r(m_{i^*}) \neq 0$ 但 $\bar{t}_{\text{top}^r(m_{i^*})}^f < t_{i^*}^f + \Delta t_{i^*}^c$)，則 i^* 的重工工件的開始處理時間 $\tilde{t}_{i^*, \bar{m}_{i^*}}^s = t_{i^*}^f + \Delta t_{i^*}^c$ 。

步驟五：以步驟四一一檢驗所有其它可處理原工件 i^* 之重工工件機台 $v \in M_{i^*} - \{m_{i^*}\}$ ，並計算其在機台 v 的可能開始重工處理時間 $\tilde{t}_{i^*, v}^s$ ，再取其最小者所對應的機台 $\bar{m}_{i^*} = \arg \min_{v \in M_{i^*}} \tilde{t}_{i^*, v}^s$ 來處理原工件 i^* 之重工工件。設定 $\bar{t}_{i^*}^s = \tilde{t}_{i^*, \bar{m}_{i^*}}^s$ ， $\bar{t}_{i^*}^f = \bar{t}_{i^*}^s + \Delta \bar{t}_{i, u}^p$ ， $\text{top}^r(\bar{m}_{i^*}) = i^*$ 。

步驟六：原工件 i 與被選擇重工的工件 i^* 皆完成排程，若不是則返回步驟二，若是則結束。

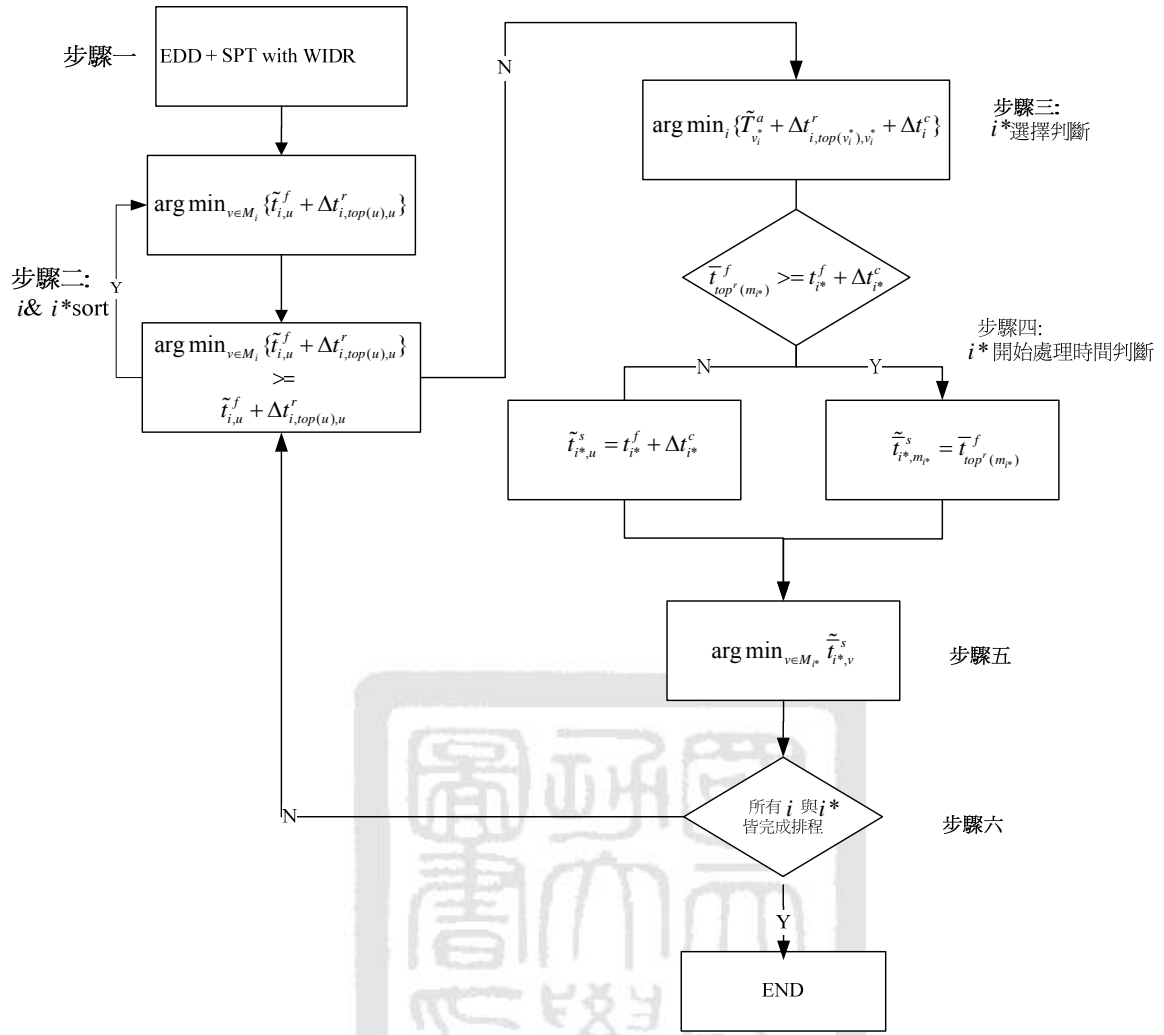


圖 3.7 SRI 演算流程圖

根據上述的步驟提出簡例說明，其步驟一同 ARI 的說明，依步驟二，ID1 在 B 機台的完工時間是最小，所以將 ID1 排入到 B 機台生產，目前最小完工時間 ID1 符合機台 A 與機台 B 的生產條件限制，ID1 的完工時間大於機台 A 目前完成時間為 0，為了先讓未重工的原工件，在平行機台盡量平均完工時間後，再安排重工工件的排程，再繼續原工件 i 的排序；第二順位是 ID5 在 C 機台的完工時間是最小，所以將 ID5 排入到 C 機台生產，目前最小完工時間 ID5 符合機台 A、機台 B 與機台 C 的生產條件限制，ID5 的完工時間大於機台 A 目前完成時間為 0，再繼續原工件 i 的排序；第三順位是 ID6 在 A 機台的完工時間是最小，所以將 ID6 排入到 A 機台生產。

由於 ID6、ID1 與 ID5 都沒有小於 ID5 在 C 機台的最小完工時間，則進行步驟三，最小檢驗的完成時間為 ID1 與 ID5，先選擇 ID1 做重工插入排序，依步驟四，之後被安插別的原工件或別的重工工件不存在，工件可重工處理時間為原工件驗檢完工時間，再進行步驟五，ID1 在機台 B 中 ID1 後重工等待時間是最小的，所以將 ID1 的重工插入在 ID1 之後，如圖 3.8 所示，其重工的完成時間為 193 分鐘。

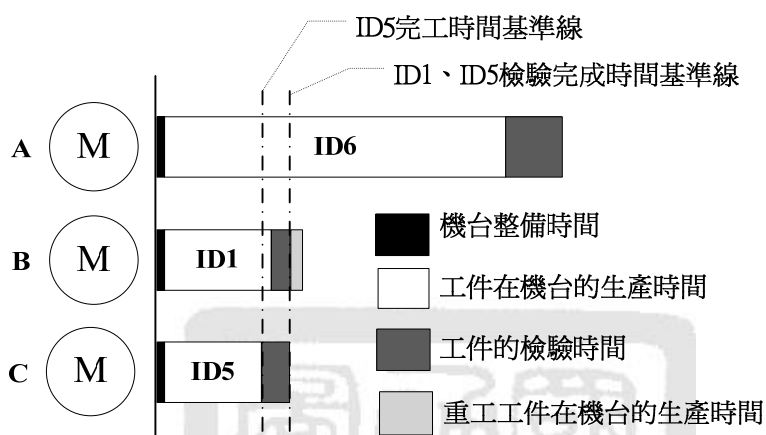


圖 3.8 平行機台插入重工工件排程示意圖

再進行步驟六的判斷，還有原工件與被選擇重工的工件皆未完成排程，所以返回步驟二，目前最小完工時間 ID5 符合機台 A、機台 B 與機台 C 的生產條件限制，由於機台 B 的原工件 i 已排入被選擇重工的工件 i^* ，機台 B 的完工時間為 0，ID5 的完工時間大於機台 B 目前完成時間為 0，再繼續原工件 i 的排序，第四順位 ID2 在 C 機台的完工時間是最小，所以將 ID2 排入到 C 機台生產，第五順位 ID7 在 B 機台的完工時間是最小，所以將 ID7 排入到 B 機台生產，由於 ID6、ID7 與 ID5 都沒有小於 ID5 在 C 機台的最小完工時間，則進行步驟三，最小檢驗的完成時間為 ID5，先選擇 ID5 做重工插入排序，依步驟四，之後有被安插別的原工件 ID2 之完工時間晚於 ID5 的檢驗完成時間，則 ID5 的重工工件開始處理時間為安插別的原工件 ID2 之完工時間，如圖 3.9 所示，其重工的完成時間為 423 分鐘，其餘以此類推，直到將原工件 i 與被選擇重工的工件 i^* 皆完成排程為止，假設檢驗時間為每公斤 0.05 分鐘運算完的數據如表 3.8 所示。

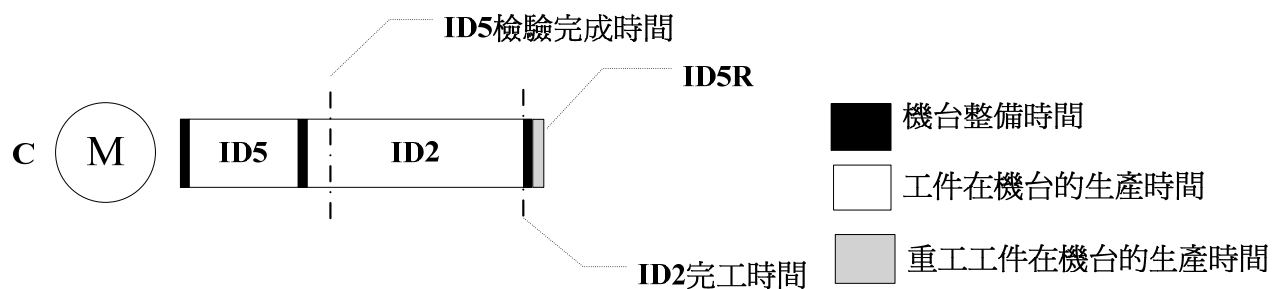


圖 3.9 平行機台插入重工工件排程示意圖

表 3.8 SRI 排程運算數據

機台A					機台B					機台C				
工件	生產時間	完成時間	等待時間	整備時間	工件	生產時間	完成時間	等待時間	整備時間	工件	生產時間	完成時間	等待時間	整備時間
ID6	451	461	0	10	ID1	141	151	0	10	ID5	138	148	0	10
ID3	391	862	0	10	ID1R	17	193	25	0	ID2	250	408	0	10
ID9	151	1023	0	10	ID7	281	484	0	10	ID5R	14	423	0	1
ID6R	67	1091	0	1	ID2R	28	527	14	1	ID4	250	683	0	10
ID3R	58	1150	0	1	ID7R	34	612	50	1	ID4R	25	758	50	0
ID9R	22	1173	0	1	ID8	253	875	0	10					
ID10	361	1534	0	0	ID8R	30	950	45	0					
ID10R	54	1648	60	0										

3.4 禁忌搜尋法說明

禁忌搜尋法是一種常用的啟發式搜尋法，由 Glover 於 1989 年所提出具有記憶的啟發式演算法，利用記錄先前之搜尋結果，以避免演算過程陷於局部最佳解，其演算流程利用禁忌串列（Tabu List）記錄最近幾個移動（Move），避免搜尋過程中，重複相同的路徑或形成迴圈（Cycle），其步驟由一個起始解（Initial Solution）開始，在每

次的鄰近解搜尋中移動到未被禁忌限制住的最佳鄰近解，直到滿足終止條件而停止移動。

3.4.1 禁忌搜尋法的基本概念

■ 移步 (Move)

從一個可行解移至另一個新的可行解。

■ 鄰近搜尋 (Neighborhood Search)

所有鄰近之可行解稱為鄰域，也就是透過移步法則將現形解轉移至另一解之所有可能解；常見的方法有鄰近交換法 (Adjacent Exchange)、兩兩交換法 (Pairwise Exchange)、插入法 (Insertion) 以及群插入法 (Block Insertion)。

■ 禁忌名單 (Tabu List)

記錄前幾次的移動，並且存在一個列表(List)中而成為短期記憶，是為了避免重複的搜尋而落入區域的最佳解，這個短暫記憶就稱為禁忌名單。除非獲得免禁準則，否則禁忌名單中的所有記錄，不可以再被選用。

■ 禁忌期限

每當有新的移步發生時，則所有被記錄在禁忌名單中的移步將會被更新。而更新手法則是採先進先出 (FIFO) 的方式，亦即下一個新的移步進來，且經過一段禁忌期限之後，就把禁忌名單中最舊的移步從名單中移除。由於名單只固定存放最近幾次的記錄，因此禁忌名單中的移步會一直變動。Yang *et al.* (1999) 提出假設問題的大小為 q ，則禁忌期限應該介於 $1/3q$ 到 $3/2q$ 之間的整數。

■ 免禁準則 (Aspiration criterion)

當所有的移步都無法再找到更佳의 鄰域解時，則會開始搜尋禁忌名單中的移步；若從禁忌名單中的移步發現更佳의 鄰域解時，將以此鄰域解來取代現有的解而成為新的解；由於此種作法忽略了原本禁忌名單的機制，因此稱為免禁準則。

■ 記憶方式

- ◆ 短期記憶將最近的移步記錄於禁忌名單中，避免重覆搜尋而落入區域最佳解，並加入免禁準則，讓較佳移步不受禁忌名單限制。
- ◆ 長期記憶一方面使用加強性（intensification）策略，將搜尋的焦點集中在特定的區域最佳解中，對這附近的解一再去作重新組合，且仔細針對這個區域再作搜尋；另一方面使用多樣性（diversification）策略，指在搜尋過程中，盡量選取未曾探索過區域中的解，使跳脫舊的搜尋區域，移往探索新的求解區域。

■ 搜尋停止準則

禁忌搜尋法利用簡單的法則先找出一個起始解。由這一個起始解開始，移動到未被禁忌住的鄰近解中目標函數最佳者，反覆這些步驟，一直到搜尋過程滿足終止條件，則停止搜尋。

用來終止搜尋進行之條件，常用方法如下：

- ◆ 到達預設最大迭代次數後即終止。
- ◆ 目標函數在一定迭次內未改善則終止。
- ◆ 達到預設 CPU 最常執行時間則終止。
- ◆ 達到預設之可接受目標值及停止搜尋。

3.4.2 禁忌搜尋法的步驟

為了達到平行機台中之最長的完工時間為最小化的目標，以貪婪演算法之 ARI 模式加上禁忌搜尋法的搭配組合，規劃如圖 3.9 所示之 Tabu 演算流程，其步驟如下所示：

步驟一：依 ARI 分配工件到平行機台的方式，將所有工件先排入平行機台的排序，做為起始解。

步驟二：依 ARI 分配重工工件插入排程到平行機台的方式；其運算完成的數據分為未排重工的平行機台排序做為暫時起始解，與已排重工的平行機台排序做為比較的數據，將其存成暫時最佳解。

步驟三：依起始解的排序，使用插入法將已排重工運算完成之最大完工時間的機台中，排在最前面的工件抽出，插入到符合平行機台條件限制，且已排重工運算完成之最小完工時間的機台中。

步驟四：在同一機台之中的工件，滿足加強性策略如工件的完工時間合乎交期，則使用相鄰兩點交換法，將工件重新排序。

步驟五：判斷新的排序是否有在禁忌名單，若沒有即進行下一步驟；若有則返回步驟四。

步驟六：依 ARI 分配重工工件插入排程到平行機台的方式，其運算完成的數據若小於暫時最佳解中已排重工運算完成平行機台中之最晚的完工時間，將其數據更新暫時起始解與暫時最佳解；若大於則返回步驟四。

步驟七：判斷搜尋停止準則，若滿足則結束；若未滿足則進行下一步驟。

步驟八：判斷目前排序的排列組合是否已排完，若已排完，則依照免禁準則將暫時起始解更新為起始解，再從步驟二開始執行；若未排完則返回步驟四。

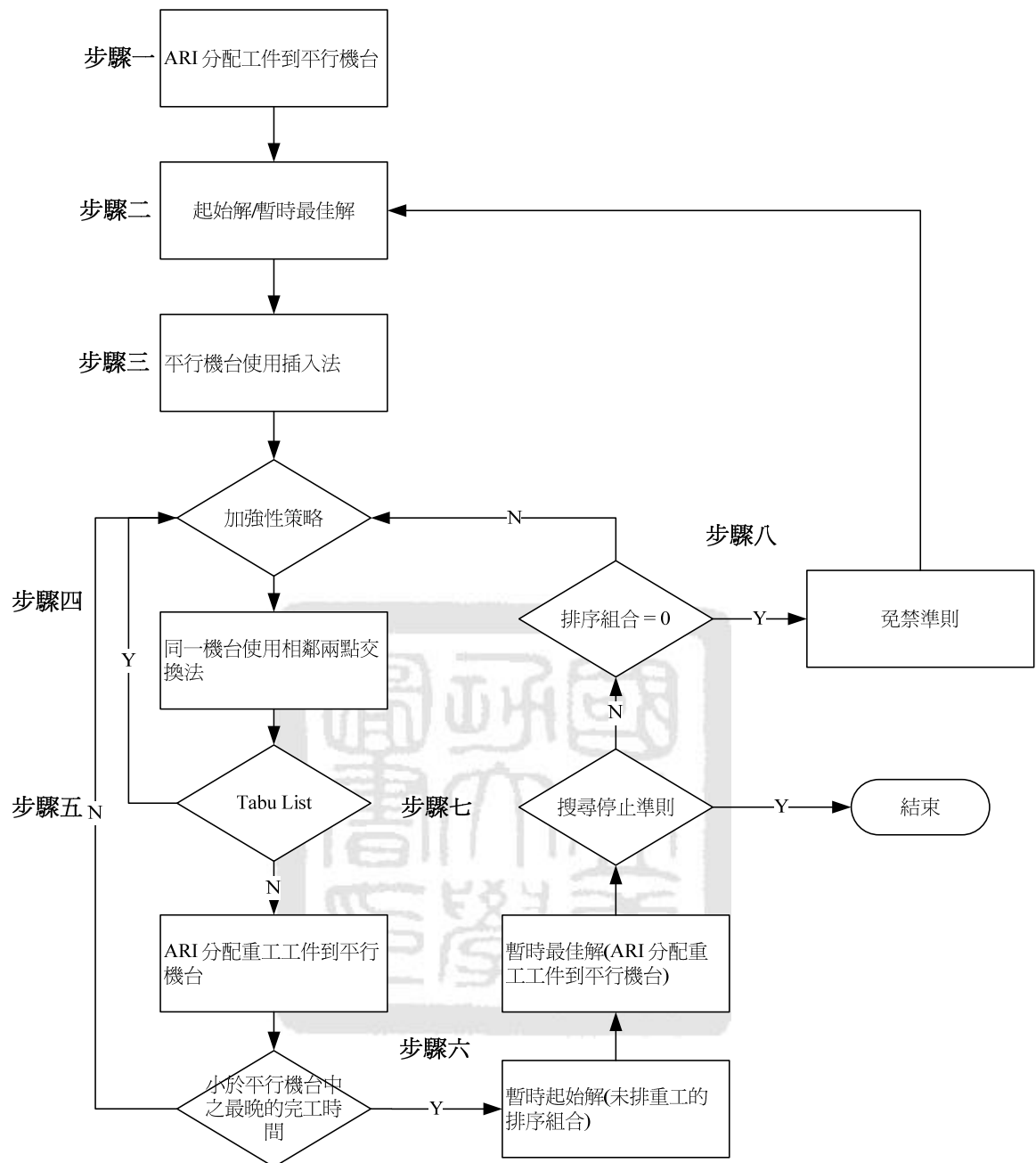


圖 3.9 Tabu 演算流程圖圖

4. 實驗結果分析與討論

本研究目的為建立單一製程平行機台重工排程模式，以達到平行機台中之最長的完工時間為最小化的目標，將對於所採用之貪婪演算法 ARI、SRI 和禁忌搜尋法之三種方法，利用現場相關的真實資料評比其求解品質與效率之差異。

4.1 實驗環境

本研究使用 Delphi 開發工具與 MySQL 資料庫來設計程式與求解；作業系統為 Windows XP，而硬體設備為 Intel Core i3-2310M CPU 以及 2GB 的記憶體。

4.2 模式資料設定

本研究以案例公司提供的機台資訊、機台重工率、機台加工時間、工件檢驗時間等基本資料，隨機收集數組工件與訂單資訊的測試資料，藉此來觀察各種排程機制的績效。

本研究的參數設定如下所示：

- 機台數為 3 台

各製程平行機台數介於 2~10 台，平均機台數為 3 台。

- 工件檢驗時間 0.05 分/公斤

- 尺寸級距不同的整備時間為 10 分鐘

- 工件數為 50 件

製程歷史中，每一個製程最多等待工件數為 300 件，平均等待排程工件數為 50 件。

4.3 實驗分析與討論

本研究擬測試 ARI、SRI 與 Tabu+ARI 等排程機制，而 Tabu+ARI 的起始解部份，分別為隨機的等待順序與 EDD+SPT with WIDR 的等待順序，將所有工件的等待順序資料先後排入平行機台的排序數據，為了後續說明，將名稱簡化為 Tabu_R 與 Tabu_W。

4.3.1 求解效率

從製程歷史中，每一個製程最多等待工件數為 300 件，平均等待排程工件數為 50 件，需在 10 分鐘內完成運算，所以 Tabu 設定搜尋停止準則為等待工件數 q ，若禁忌名單中有記錄各平行機台排序組合數超過 $8q/9$ 時就停止，為了不造成記憶體不足，禁忌期限設定為禁忌名單中有各平行機台之工件排序組合數時，且其中一機台之工件排序組合數為 $q/2$ 時，就把禁忌名單中最舊的機台之工件排序組合從名單中移除。由於只是比較程式運算時間，Tabu_R 與 Tabu_W 只是起始解不同，所以只比較 ARI、SRI 及 Tabu_R，如圖 4.1 所示，Tabu_R 的求解時間最慢，在超過 50 個工件以後，即無法於 10 分鐘內求得可行的解。在將時間間距縮小下，如圖 4.2 所示，ARI 的求解時間最快，SRI 次之。

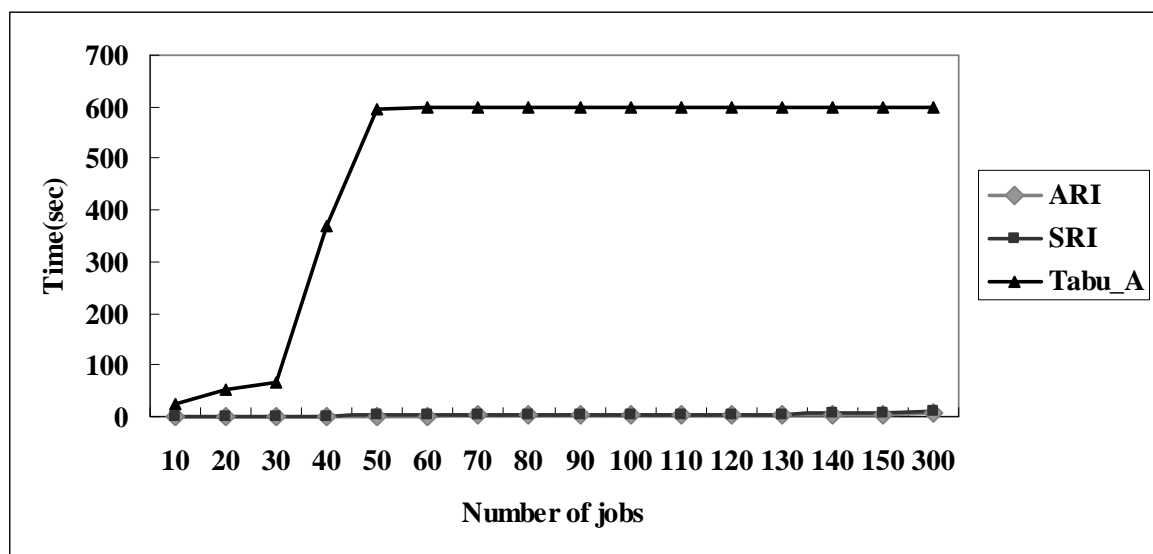


圖 4.1 Greedy 與 Tabu_A 三種法則求解時間

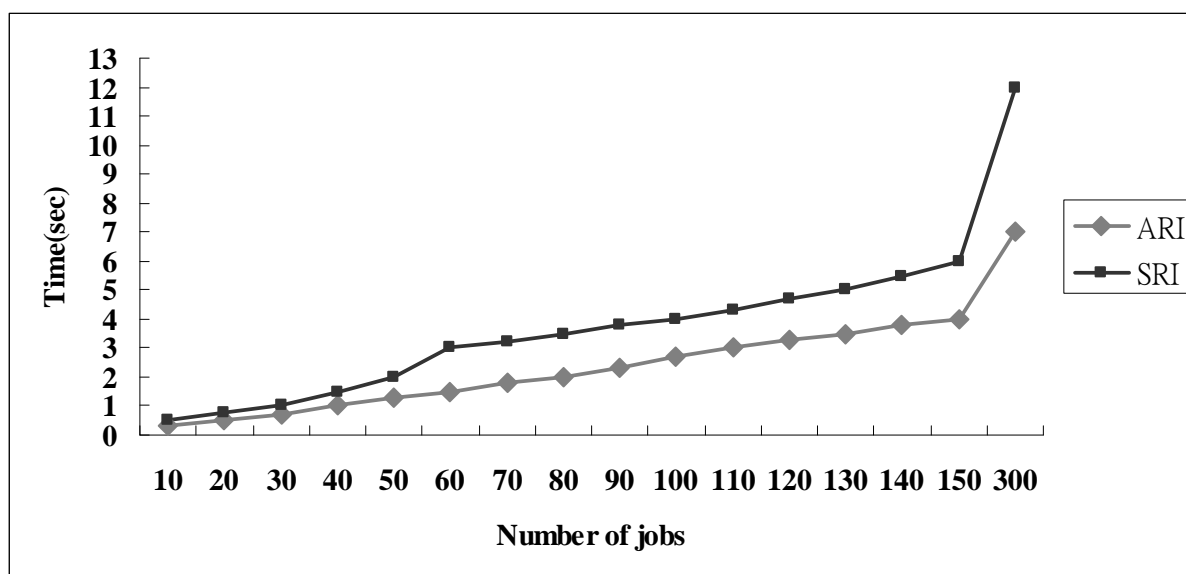


圖 4.2 Greedy 二種法則求解時間

4.3.2 求解品質

在求解品質方面，以平均等待工件數 50 件為測試依據，分別測試實際工件數據、工件平均工時數據與工件長短工時數據。工件平均工時的數據產生為依實際工件數據的加總重量除以 50 後的平均重量，而工件長短工時的數據產生為隨機配給 25 個工件為短工時與 25 個工件為長工時。假設其短工時為 100 公斤，而長工時為平均重量*2-100 的重量，再測試不同的狀況為不異動工件、異動工件交期、異動檢驗時間、不考慮整備時間，在比較平行機台最長完工時間與平行機台負荷時間的數據，來分析那種方法的求解品質最佳，以下為測試說明：

■ 不異動工件

在假設檢驗時間為每公斤 0.05 分鐘，與尺寸級距不同的整備時間為 10 分鐘，比較平行機台最長完工時間，如圖 4.3 所示，時間愈短愈好的情形下，以實際工件數據而言，Tabu_A 最好、SRI 次之、Tabu_R 最差；以工件平均工時數據而言，Tabu_A 最好、SRI 次之、Tabu_R 最差；但在工件長短工時數據來看，SRI 最好、Tabu_A 次之、Tabu_R 最差。

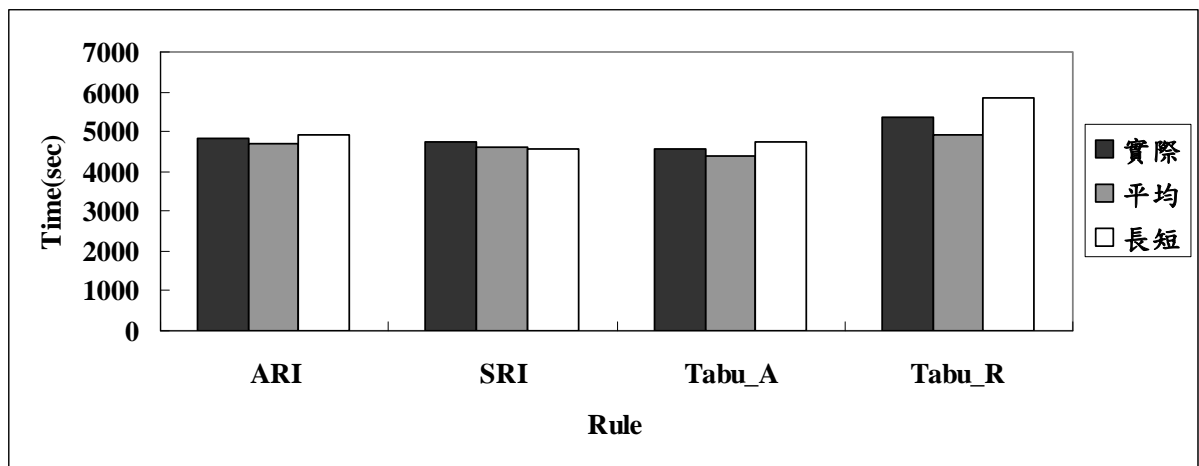


圖 4.3 不異動工件之平行機台最長完工時間

比較平行機台負荷時間，其值為平行機台加總的完工時間除於機台數 3，再相減平行機台的完工時間最大取絕對值的數據，如圖 4.4 所示，其結果與比較平行機台最長完工時間之分析一致。

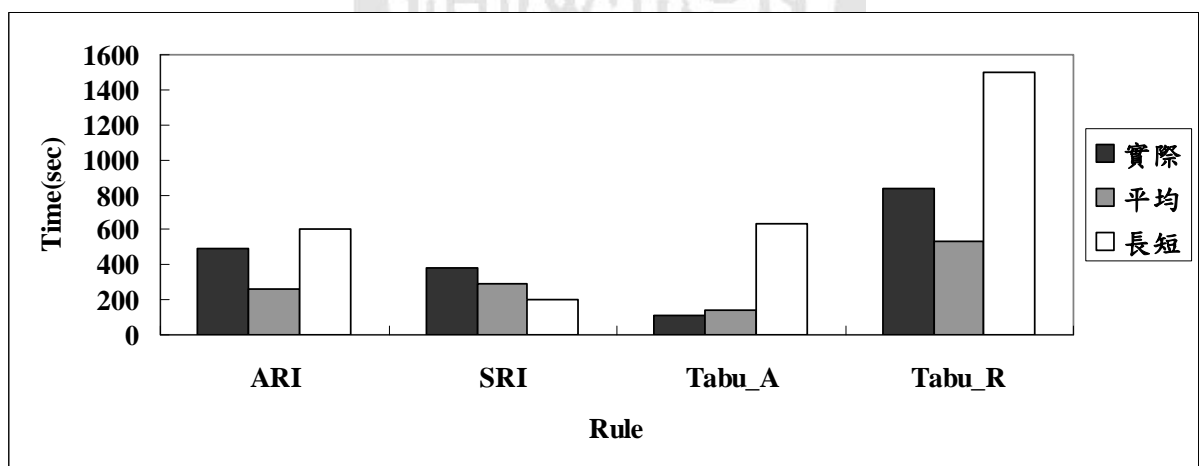


圖 4.4 不異動工件之平行機台負荷時間

■ 異動工件交期

在假設檢驗時間為每公斤 0.05 分鐘，與尺寸級距不同的整備時間為 10 分鐘，隨機異動工件的交期，藉此影響工件的等待排序，分析是否會造成不同的結果。比較平行機台最長完工時間，如圖 4.5 所示，以實際工件數據而言，Tabu_A 最好、SRI 次之、ARI 最差；以工件平均工時數據而言，Tabu_A 最好、ARI 次之、Tabu_R 最差；但在工件長短工時數據來看，SRI 最好、Tabu_A 次之、Tabu_R。

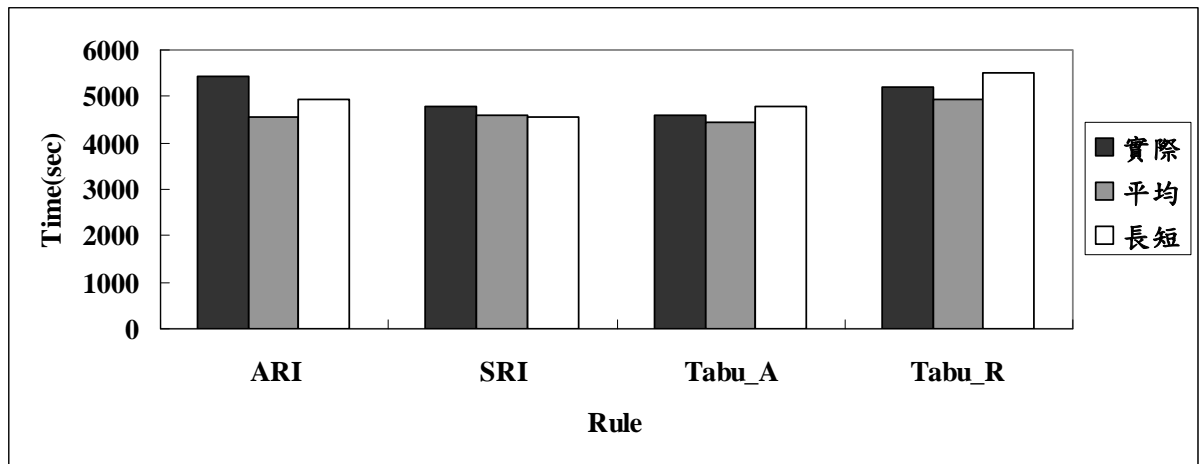


圖 4.5 異動交期之平行機台最長完工時間

比較平行機台負荷時間，如圖 4.6 所示，其結果與比較平行機台最長完工時間之分析一致。

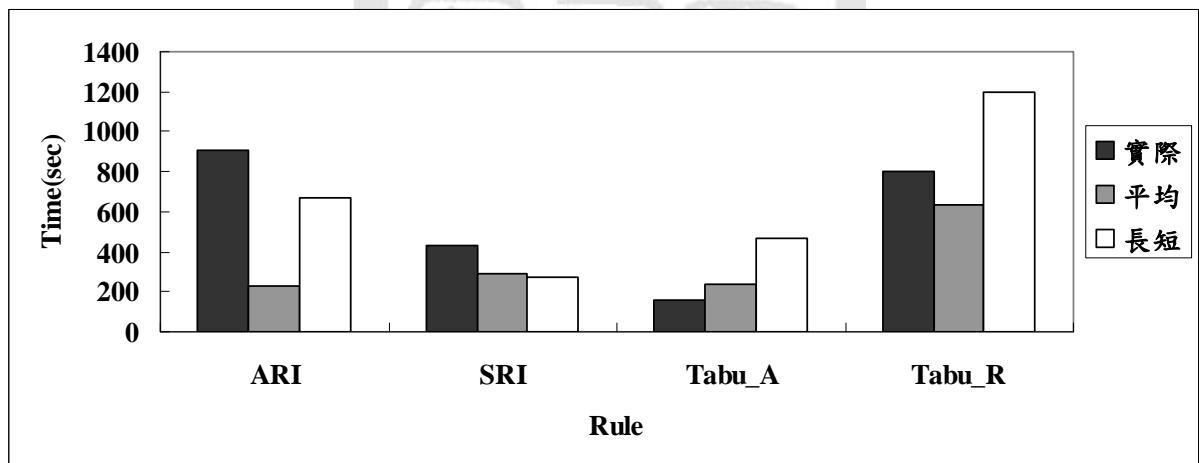


圖 4.6 異動交期之平行機台負荷時間

■ 異動檢驗時間

在假設檢驗時間為每公斤 0.1 分鐘，與尺寸級距不同的整備時間為 10 分鐘，藉此影響工件檢驗完工時間，分析是否會造成不同的結果。比較平行機台最長完工時間，如圖 4.7 所示，以實際工件數據而言，Tabu_A 最好、SRI 次之、Tabu_R 最差；以工件平均工時數據而言，SRI 最好、Tabu_A 次之、Tabu_R 最差；但在工件長短工時數據來看，SRI 最好、Tabu_A 次之、Tabu_R 最差。

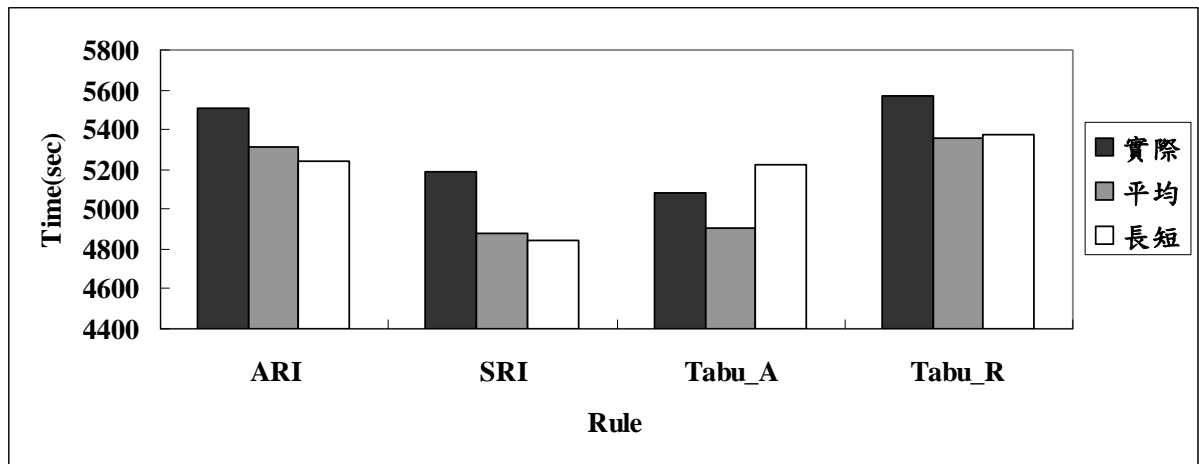


圖 4.7 異動檢驗時間之平行機台最長完工時間

比較平行機台負荷時間，如圖 4.8 所示，以實際工件數據而言，SRI 最好、Tabu_A 次之、ARI 最差；以工件平均工時數據而言，SRI 最好、Tabu_A 次之、Tabu_R 最差；但在工件長短工時數據來看，SRI 最好、Tabu_A 次之、Tabu_R 最差。

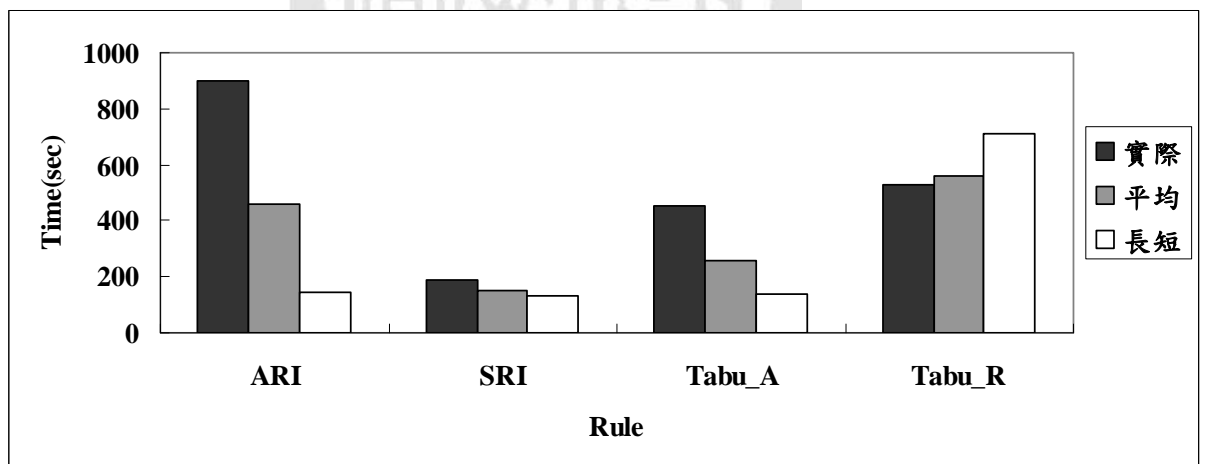


圖 4.8 異動檢驗時間之平行機台負荷時間

■ 不考慮整備時間

在假設檢驗時間為每公斤 0.05 分鐘，不考慮整備時間，藉此影響工件的等待排序，分析是否會造成不同的結果。比較平行機台最長完工時間，如圖 4.9 所示，以實際工件數據而言，Tabu_A 最好、ARI 次之、Tabu_R 最差；以工件平均工時數據而言，Tabu_A 最好、SRI 次之、Tabu_R 最差；但在工件長短工時數據來看，SRI 最好、Tabu_A 次之、Tabu_R 最差。

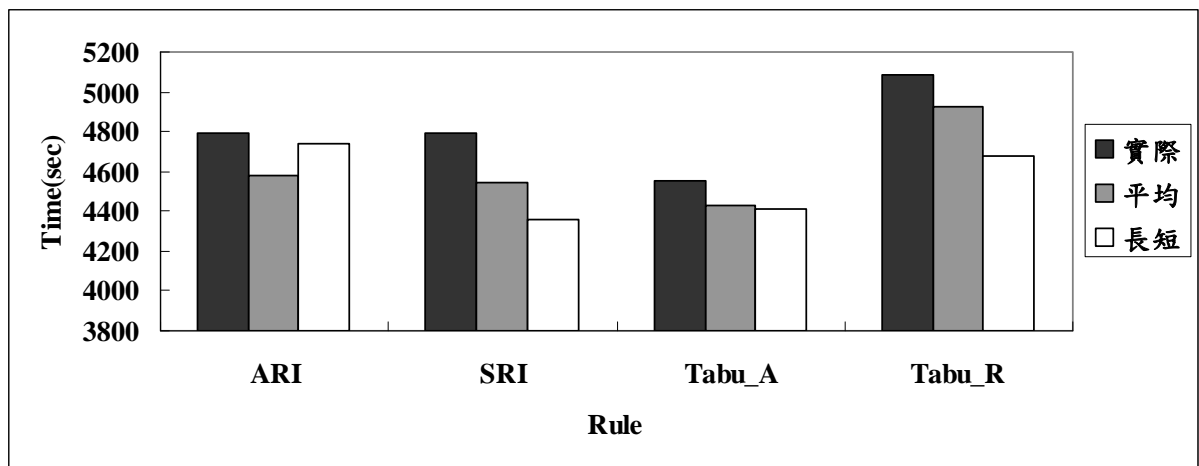


圖 4.9 不考慮整備時間之平行機台最長完工時

比較平行機台負荷時間，如圖 4.10 所示，其結果與比較平行機台最長完工時間之分析一致。

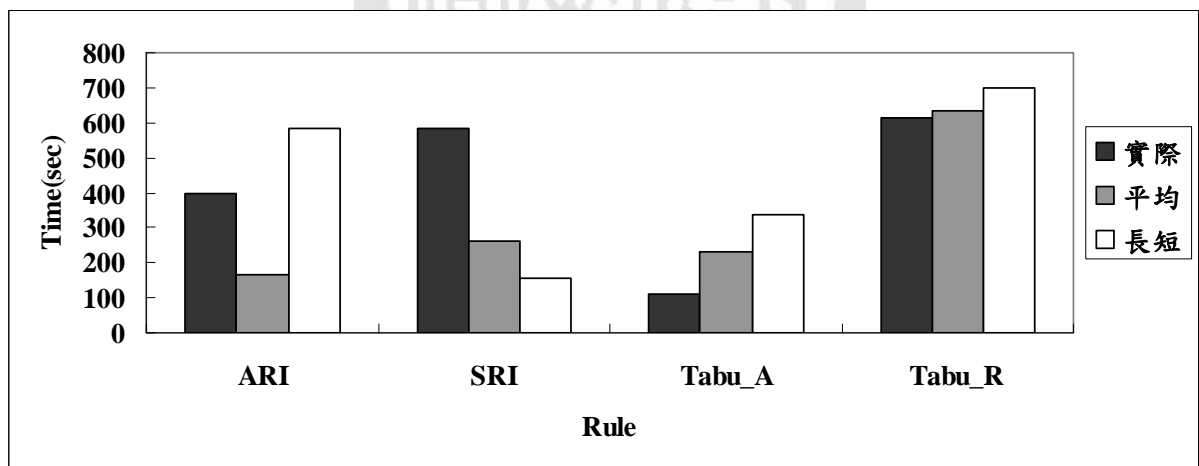


圖 4.10 不考慮整備時間之平行機台負荷時間

以求解品質來看，SRI 在工件的重量有明顯差異，造成工件有明顯長短生產工時狀況下，如圖 4.11 與圖 4.12 所示，其求解品質最好。Tabu_A 次之、Tabu_R 最差。Tabu_A 在不不論幾種因素影響下，如上面分析大部份的求解品質是最好的，SRI 次之、Tabu_R 最差。

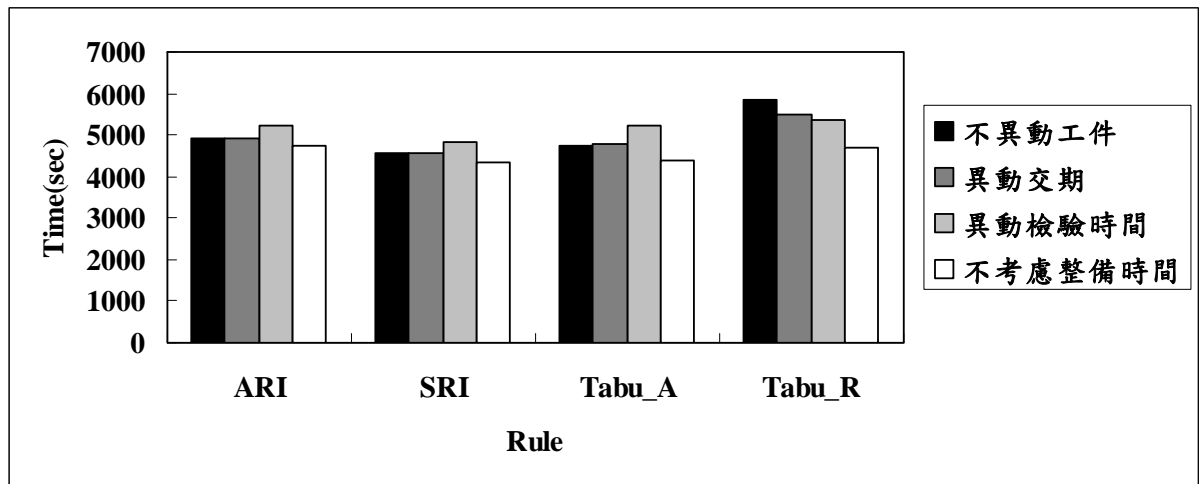


圖 4.11 平行機台最長完工時間之長短工時

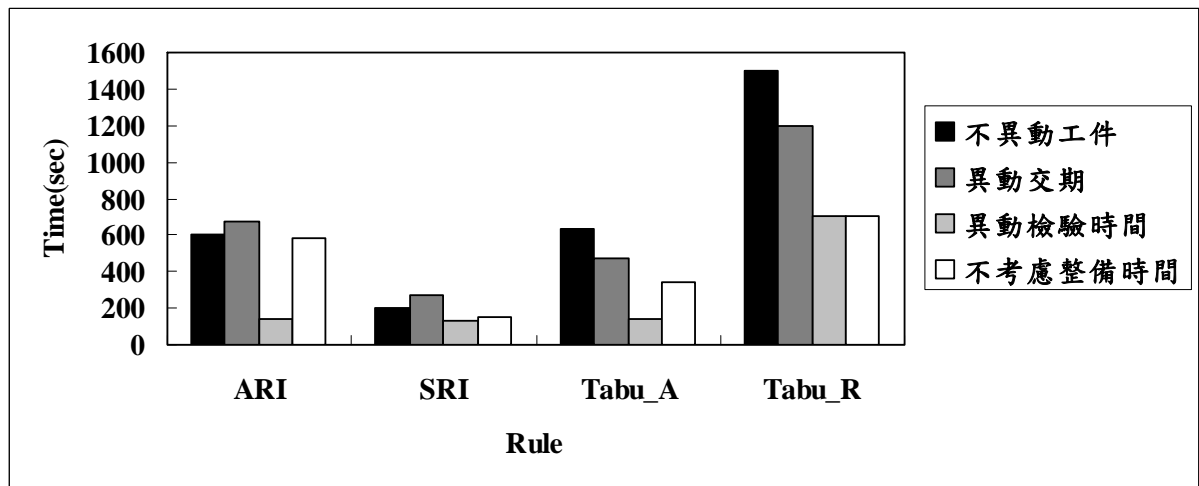


圖 4.12 平行機台負荷時間之長短工時

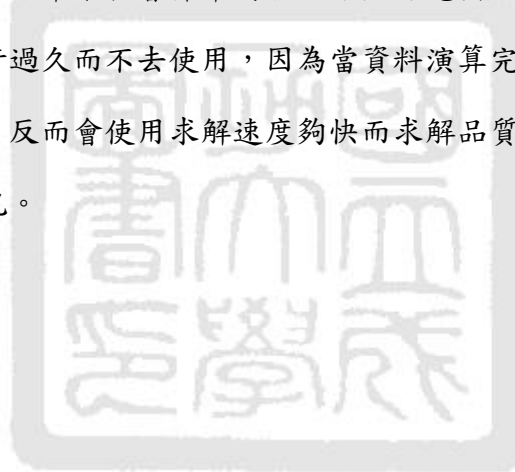
4.4 小結

工件分配到平行機台的排程過程中，會因機台的生產限制而影響投入機台的選擇，而不同的派工法則也會影響投入機台的分配，再加上機台的整備時間考量，也會影響投入機台的判斷，工件重工這因素，會讓所有問題更加複雜，所以經過各項因素測試後，在考慮求解效率與求解品質下，首選為 SRI，ARI 次之。不過 ARI 在上述幾種因素影響下數據的起伏變化大，經過 Tabu_A 的優化修正後，數據改善良多，所以

若以各製程平均等待排程工件數 50 件，10 分鐘內完成程式運算，Tabu_A 最佳、SRI 次之。

ARI 在求解品質下大部份數據不如 SRI，原因為 ARI 是將所有工件排入平行機台後，再插入重工的工件，這會讓重工的工件只能依已排入的工件狀況判斷如何插入重工的工件，而 SRI 是一件工件排入平行機台後，就馬上判斷重工的工件插入或下一等待順序的工件排入，比較能夠避免被限制在所有工件已排入平行機台，而無法異動工件再重新排入平行機台，以求得工件與重工工件在平行機台中更好的完工時間。

Tabu_A 在工件有明顯長短生產工時狀況下，求解品質不如 SRI，是因為為了符合 10 分鐘內完成程式運算才中斷演算，若持續讓 Tabu_A 跑到比 SRI 的數據好時才停止演算是可行的，但在求解效率下就會非常的差，雖然可跑出比較正確的數據，在實務上使用者會因為程式執行過久而不去使用，因為當資料演算完成產生數據時，現場狀況已不同演算時的狀況，反而會使用求解速度夠快而求解品質差不多效果的資料，以面對現場不斷變化的狀況。



5. 結論與建議

5.1 結論

為了符合客製化的訂單生產，必須滿足客戶品質、訂單交期及零庫存，可是製程加工時所產生的瑕疵可能必須馬上以重工方式修復以趕上訂單交期，因此可能影響原先生產計劃的時間安排，進而造成其它訂單的生產延遲，若能在生產計劃中將一般排程比較少考慮的重工時間估算進去，應該更能符合現場實際運作狀況，並推算出更合理的訂單交期時間。

本研究以盤元與直棒製程中經常處理之平行機台工件重工排程問題，在考慮交期與整備時間下，利用貪婪演算法 ARI 與 SRI 二種不同角度的法則，計算出原工件的生產排序與重工工件的插入排序，來求解區域的最佳解以達到全域的最佳近似解，並以禁忌搜尋法求解全域的最佳近似解，推算符合現場實際生產狀況的排程完成時間，並提昇客戶訂單的達交率，以達到最小化平行機台中之最晚的完工時間為目標。

我們以數十組不同大小範圍之個案公司資料，經過各項因素測試後，在考慮求解效率與求解品質下發現 SRI 的表現最好，ARI 次之。不過 ARI 可能在交期、檢驗時間與工件有長短工時幾種因素影響下造成其完工時間數據的起伏變化過大，將 ARI 經禁忌搜尋法進一步改善後的 Tabu_A 則有不錯的表現。

ARI 在求解品質下大部份數據不如 SRI，原因在於 ARI 受限於所有工件已排入平行機台，這會讓重工的工件只能依已排入的工件狀況判斷如何插入重工的工件，而 SRI 則是一件工件排入平行機台後，就馬上判斷重工的工件插入或下一等待順序的工件排入，比較能夠因應目前工件的狀況適當調整在平行機台的排序，以求得工件與重工工件在平行機台中更好的完工時間。Tabu_A 在工件有明顯長短生產工時狀況下，其求解品質不如 SRI，此乃肇因於本研究實際上設定的禁忌期限終止運算機制，實務上我們亦可持續讓 Tabu_A 跑到比 SRI 的數據好時才停止，但如此一來其求解效率將會非常

差，因此實務上使用者大都選擇使用求解速度夠快而求解品質差不多效果的資料，以面對現場不斷變化的狀況。

5.2 未來研究與建議

本研究仍有未盡完善之處，最後提出以下建議，以供日後相關研究發展之參考：

- (1) 從不同鋼廠的來料，或從本廠其它製程的來料，都可能在加工的過程中，工件已產生瑕疵，而必須經過下一製程的重複加工也就是重工才可修復，後續可進一步分析來料的瑕疵在不同狀況下，所提出排程方法其優劣表現情形。
- (2) 未考慮下製程的生產規格循環，由於下製程為了成本與整備時間考量，會排定可生產規格的計劃時間，所以若未考慮下製程的生產規格循環，當製程產出的工件，卻無法馬上投入下製程生產，將影響下製程產生一堆等待生產的工件。
- (3) 本研究只以數十組不同大小範圍之個案公司資料測試，若能經過實際操作者以不同的模式資料運行，再與程式開發人員討論與修正，應該更能符合現場實際生產狀況。

參考文獻

- 陳建良，1995，排程概述，機械工業雜誌，12 月，頁 122-137。
- 陳春龍，2006，以派工法則求解印刷電路板排程問題，行政院國家科學委員會專題計劃
- 王治平，2003，實際零工式生產排程問題的派工法則，國立政治大學資訊管理研究所，
碩士論文。
- 鄧紫文，2004，實際零工式生產派工法則之選擇：靜態系統，國立政治大學資訊管理
研究所，碩士論文。
- 郭宜雍，2004，結合模擬與智慧搜尋法最佳化多機台流線式製程之排程研究，國立成
功大學製造工程研究所，博士論文。
- 葉子嘉，2007，整備資源有限下之平行機台排程方法研究，國立成功大學工業與資訊
管理學系，碩士論文。
- 余侑儒，2008，考量多產品環境下之多目標等效平行機台排程研究，國立成功大學工
業與資訊管理學系，碩士論文。
- Chen, J., Yao, DD., and Zheng, S., 2001, Optimal replenishment and rework with multiple
unreliable supply sources. *Operations Research* , **49**(3),430-443.
- Chiu, S. W., Gong, D.-C., and Wee, H.-M., 2004, Effects of random defective rate and
imperfect Rework Process on economic production quantity model, *Janpan Journal of
Industrial & Applied Mathematics*, **21**(3), 375-389.
- Chiu, Y. P., Liu, S-C., Chiu, C-L., and Chang, H-H., 2011, Mathematical modeling for
determining the replenishment policy for EMQ model with rework and multiple
shipments. *Mathematical and Computer Modelling*, **54**, 2165-2174.
- Conway, R.W., 1965, Priority dispatching and work-in-process inventory in a job shop,
Journal of Industrial Engineering, **16**, 123-130.

- Graham, R. L., Lawler, E. L., Lenstra, J. K., and Rinnooy Kan, A.H.G., 1979, Optimization and approximation in deterministic sequencing and scheduling : a survey, *Annals of Discrete Mathematics*, **5**, 287-326.
- Haupt, R., 1989, A survey of priority rule-based scheduling, *OR Spektrum*, **11**, 3-16.
- Herrmann, J. W., 2001, Improving Manufacturing System Performance through Rescheduling, *Computer Integrated Manufacturing Laboratory*, University of Maryland, June 21
- Inderfurth, K., Lindner, G., and Rachaniotis, N.P., 2005, Lot sizing in a production system with rework and product deterioration, *International Journal of Production Research*, **43**(7), 1355-1374.
- Jain, A. S., and Meeran, S., 1998, A State-of-the-Art Review of Job-Shop Scheduling Techniques, Technical Report, Department of Applied Physics, *Electronic and Mechanical Engineering University of Dundee*, Dundee, Scotland, UK.
- Kang, Y. H., and Shin, H. J., 2010, An adaptive scheduling algorithm for a parallel machine problem with rework processes, *International Journal of Production Research*, **48**(1), 95-115.
- Kelton, W.D., Sadowski, R.P., and Sturrock, D.T., 2009, *Simulation with Arena*, 5th edition, McGraw-Hill, New York
- Oliver, H. and Chandrasekharan, R., 1997, Efficient dispatching rules for scheduling in a job shop, *International Journal of Production Economics*, **48**, 87-105.
- Mokotoff, E., 2001, Parallel machine scheduling problems: A survey. Asia-Pacific, *Journal of Operational Research*, **18**(2), 193-242.
- Shin, H. J., and Kang, Y. H., 2010, A rework-based dispatching algorithm for module process in TFT-LCD manufacture, *International Journal of Production Research* , **48**(3), 915–931
- Subramaniam, V., Lee, G.K., Hong, G.S., Wong, Y.S., and Ramesh, T., 2000, Dynamic selection of dispatching rules for job shop scheduling, *Production Planning &*

Control,**11**(1), pp.73-81.

Yang, T., Rajasekharan, M., and Peters, B.A., 1999, Semiconductor fabrication facility design using a hybrid search methodology, *Computers and Industrial Engineering*, **36**, 565-583.

