

國立交通大學
工業工程與管理學系

碩士論文

中斷式幾何良率分配及交期限制下二階段生產
系統投料決策之研究

Multiple lot-sizing decisions in a two-stage production with an
interrupted geometric yield and non-rigid demand

研究生：黃亮銓

指導教授：巫木誠教授

中華民國九十八年六月

中文摘要

本研究探討一個新的多次投料問題，此問題為一有交期限制的二階段生產系統，其製程良率服從中斷式幾何分配。我們建立遞迴數學模式代表此問題並以動態規劃求解。本研究也提出界定最佳解範圍的定理，幫助求解此問題。然而利用動態規劃求解大規模問題可能存在求解效率議題。因此本研究提出一個簡化動態規劃網路的演算法，利用將數個分支結合為一個的手法，將原本的網路轉變為較簡單的網路。在大量的實驗中，結果顯示本研究提出之簡化演算法可幫助現場人員有效率地求解大規模及高良率情境下之多次投料問題。

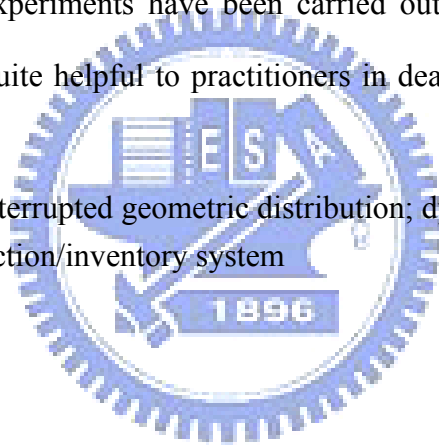
關鍵詞：批量、中斷式幾何分配、動態規劃、兩階段系統、生產/存貨系統



Abstract

This research examines a new multiple lot-sizing problem, which is in the context of a two-stage production system with a due-date-based demand and the process yields are both governed by interrupted geometric (IG) distributions. We model this problem as a recursive formula and solve it by dynamic programming. This research also develops lemmas for solving this problem. However, there may still be many computational efforts in solving this DP problem. An efficient algorithm for resolving the computational issue is proposed. This algorithm is designed to reduce the DP network into a much simpler one—through combining a group of DP branches into a single one. Extensive experiments have been carried out. Results indicate that the reduction algorithm is quite helpful to practitioners in dealing with large-scale cases with high-yield.

Keywords: lot-sizing; interrupted geometric distribution; dynamic programming; two-stage system; production/inventory system



致謝

能完成此碩士論文最要感謝的是指導教授巫木誠老師。老師在學術研究上教導學生系統性的思考、創新及表達能力，使我在有正確的研究方向與方法。在待人處世上，老師不斷地傳達誠信正直的信念與辨清關鍵的管理方法，讓學生受益匪淺。也感謝許錫美老師、彭德保老師、張仲儒老師、陳文智老師在口試中給予的具體建議，大大地提升了本論文的品質。

在碩士生涯的兩年中，感謝許多學長協助我成長。首先感謝蘇泰盛學長提供我良好的研究基礎與研究方法。接著感謝徐揚康學長於合作研究中的耐心討論指導。也感謝施昌甫學長在生活上的照顧以及陳振富學長於出國實習計畫中的陪伴與照料。此外，由衷感謝實驗室同學昭宏、佳玟、文旻與慈盈在生活與學業上的陪伴讓我獲得非常快樂的兩年碩士生涯。

最後要感謝我的家人總是在背後挺著我。特別是我的父母親給予充沛的支持與關懷、姐姐在生活上的分擔解憂都是我順利完成此論文的關鍵。

一路上能受到那麼多人的教導與陪伴實在非常幸運。而感謝的方式僅限於文字是不夠的。我會將這兩年學習到的寶貴知識與處事方法努力地應用於未來的人生中，以不致辜負眾人的幫助與期待，進而希望能在社會上發揚交大校譽，報答母校栽培之恩。



黃亮銓 于 新竹交通大學

2009/6/30

目錄

中文摘要	ii
Abstract.....	iii
致謝.....	iv
目錄.....	v
表目錄	vi
圖目錄	vii
第一章 緒論	1
第二章 文獻回顧	3
2.1 依生產階段的數量分類	3
2.2 依客戶需求的模式分類	4
2.3 依生產的良率分配分類	4
2.4 依求解方法分類	5
第三章 中斷式幾何良率分配及交期限制下多次投料問題	6
3.1 符號定義	6
3.2 研究問題介紹	7
3.3 多次投料問題決策的結果	9
3.4 成本函式與動態規劃求解	10
第四章 最佳投料決策之界限	13
第五章 簡化動態規劃網路之求解演算法	17
第六章 實驗	19
第七章 結論	23
參考文獻	24

表目錄

表 1	探討各種分配的相關文獻.....	5
表 2	利用 comprehensive algorithm 求解的 C_p and T_p	20
表 3	$n = 5$ 時利用 reduction algorithm 求解的 R_C and R_T	21
表 4	$n = 10$ 時利用 reduction algorithm 求解的 R_C and R_T	21
表 5	$n = 15$ 時利用 reduction algorithm 求解的 R_C and R_T	22
表 6	$n = k_i + 1$ 時利用 reduction algorithm 求解的 R_C and R_T	22
表 7	各 (T, D) 組合下利用動態規劃求解的平均時間(單位:秒)	22



圖目錄

圖 1 兩階段生產系統圖示.....	8
圖 2 動態規劃模式的投料決策結構.....	10



第一章 緒論

隨著日益增加的客製化或短生命週期產品，生產者如何在有限的時間滿足顧客的需求是很重要的。在多次的投料生產機會中，如何制定良好的投料策略以滿足顧客需求稱為多次投料問題。多次投料問題本質上源自於製造良率的變化。在一個擁有隨機性良率的生產系統中，利用多次投料去滿足顧客的訂單可能會產生較低的成本。這種多次投料問題被稱為 MLPO (Multiple Lot-Sizing Production to Order) 問題(Grosfeld-Nir & Gerchak, 1996)。多次投料問題是相當重要的。一個好的投料決策模式，也就是決定什麼時候投到哪部機台以及投入多少原料，可以提高訂單的滿足程度及降低生產製造的成本。然而這樣的投料決策是很複雜的，訂單模式、生產模式、機台設置成本、單位製造成本、持貨成本、良率分配等都是會影響投料決策的重要因素。因為多次投料問題的重要性與複雜性，使得該問題非常有研究的價值。

本論文探討一個新的多次投料問題。首先，產品的完成需要經過兩道製程，而這兩道製程具有互斥性，不能同時加工。第二，本研究探討的生產系統中，各機台生產製造的良率服從中斷式幾何(Interrupted Geometric, IG)分配，也就是當第一個不良品出現，即代表其後的所有產品皆是不良品。第三，顧客的訂單中包含交期限制，交期內無法滿足訂單的部份將被罰款。這樣的需求在文獻上也被稱為非嚴格需求(Non-Rigid Demand)，也就是說生產者可以不必全部地滿足顧客的需求。最後，本研究假設的情境為客製化產品的生產，超出客戶需求量的產品不具有任何殘餘價值。

本研究探討的情境可以用下例說明。顧客的訂單中所購買的客制化產品需要經由兩道不同的成型製程來完成。而這兩道製程是由同一台資本密集的機台分別搭配兩個不同的模具生產，因此同一時間只能進行其中一道製程。模具會隨著使用逐漸磨耗，當模具已被磨耗至無法生產良品的程度，製造出來的所有產品將都是不良品，這個特性也導致此二個製造程序符合中斷式幾何的良率分配。

已經有許多探討中斷式幾何分配下多次投料問題的研究被發表。這些研究大部分假設客戶的訂單屬於嚴格需求(Non-Rigid Demand)，也就是生產者必須生產到顧客的訂單完全地被滿足。只有少部分的研究探討非嚴格需求下中斷式幾何良率分配的多次投料問題，然而他們的研究著重在單階段的生產系統。因此本論文所探討的多次投料問題是相當缺乏研究的。

本研究將這個多次投料問題以數學模式表示並利用動態規劃求解。同時並提出一些定理，以提供最佳投料量的界限而幫助求解。在小規模的情境下，單純使用動態規劃方法可以有效率地求得最佳投料決策。然而在大規模且高良率的情境中，求解此動態規劃的網路可能相當耗時。為了解決這個效率問題，本研究提出了一個簡化原本動態規劃網路的演算法。此演算法的基本概念是利用將多個網路的分支群聚成一個的方法，使得原先的求解網路轉變的較簡單。本研究也利用大量的電腦實驗檢驗兩個演算法在不同情境下的績效。結果顯示本研究提出的簡化演算法可以幫助現場人員有效率地求解大規模且高良率情境下之多次投料問題。

本論文的編排如下。第二章回顧多次投料問題的相關文獻。在第三章中，我們將詳細說明此多次投料問題，並用數學模式表示，以動態規劃求解。在第四章中，本研究提出三個定理及定理成立需要的性質，給予最佳投料決策的搜尋界限。在第五章中，本研究提出簡化動態規劃網路之演算法。第六章進行各種測試情境下的電腦實驗，以檢驗兩種演算法之求解績效。最後，結論與未來研究方向則於第七章中說明。

第二章 文獻回顧

研究多次投料問題的文獻非常豐富，已經有兩篇重要的文獻整理之研究發表 (Yano & Lee, 1995; Grosfeld-Nir & Gerchak, 2004)。多次投料問題基本上可以分別用生產階段的數量、客戶需求的模式、生產的良率分配及求解方法等角度來分類。

2.1 依生產階段的數量分類

依據生產階段的數量，多次投料問題的相關文獻可分為單階段及多階段兩種。目前大多數的文獻屬於單階段生產系統(Beja, 1977; Sepheri *et al.*, 1986; Pentico, 1988; Grosfeld-Nir & Gerchak, 1990; Anily, 1995; Grosfeld-Nir & Gerchak, 1996; Zhang & Guu, 1997; Zhang & Guu, 1998; Guu & Liou, 1999; Guu, 1999; Anily *et al.*, 2002; Guu & Zhang, 2003)。單階段生產系統指的是產品從投入到產出只需經過一道成功的製造程序。

而探討多階段生產系統的文獻相對較少(Lee & Yano, 1988; Wein, 1992; Grosfeld-Nir & Ronen, 1993; Pentico, 1994; Grosfeld-Nir & Robinson, 1995; Grosfeld-Nir, 1995; Barad & Braha, 1996; Barad, 1999; Grosfeld-Nir & Gerchak, 2002; Grosfeld-Nir, 2005; Grosfeld-Nir *et al.*, 2006; Ben-Zvi & Grosfeld-Nir, 2007)。多階段生產系統指的是產品從投入到產出只需經過兩道或兩道以上成功的製造程序。其中 Grosfeld-Nir *et al.* (2006)探討多階層的生產系統。所謂多階層 (Multiple Echelon)系統指的是一個階層包含多道製程，這些製程彼此間並無順序關係，而該階層的各製程須產出一給定的數量才能算完成該階段作業，進入下一階段。實務上組裝系統即屬於多階層生產系統，各零件的生產製造皆須完成規格規定的數量才能進入組裝階層進行組裝。

2.2 依客戶需求的模式分類

依據客戶的需求模式，多次投料問題的相關文獻可分為嚴格需求(Rigid Demand)及非嚴格需求(Non-rigid Demand)兩種。目前大部分的多次投料文獻皆屬於嚴格需求(Wein, 1992; Grosfeld-Nir & Ronen, 1993; Pentico, 1994; Grosfeld-Nir & Robinson, 1995; Grosfeld-Nir, 1995; Grosfeld-Nir & Gerchak, 2002; Grosfeld-Nir, 2005; Grosfeld-Nir *et al.*, 2006; Ben-Zvi & Grosfeld-Nir, 2007)。嚴格需求指的是客戶的訂單需求量必須要全部地滿足。因為必須全部滿足，因此生產者必須直到客戶的需求完全滿足後才能停止生產。其中 Gerchak & Grosfeld-Nir(1999)特別探討不同等級產品的組合訂單，即高級品、一般品。高級品與一般品的價格不同，其中高級品可作為一般品交給顧客。而生產者分別產出高級品、一般品及不良品符合三項式分配。

非嚴格需求指的是客戶的訂單需求量並不一定要完全被滿足。這類的問題通常都有給定交期限限制，也就是期限內若無法滿足顧客，將必須支付缺交的罰金，事實上罰金也可視為無法賺到客戶的支出之機會成本。考慮交期限限制的文獻較少(Lee & Yano, 1988; Barad & Braha, 1996; Braha, 1999)。另外 Gerchak & Grosfeld-Nir(1998) 探討客戶需求存在變化的問題，也就是客戶最後的需求可能大於或小於產的量，因此決策者應將需求的變化考慮進去，才能決定應該生產多少的量。

2.3 依生產的良率分配分類

不同的良率分配會影響投料的決策。依生產系統中製程的良率分配分類，相關文獻大致上可分為一般型均勻分配(General Discrete Distribution)、二項式機率分配(Binomial Distribution)、中斷式幾何機率分配(IG Distribution)、隨機比例機率分配(Stochastically Proportional Distribution)、全良品或全不良品分配(All-or-nothing Distribution)。表 1 顯示探討各種分配的一些相關文獻。

表 1 探討各種分配的相關文獻

良率分配	相關文獻
一般離散型分配	Grosfeld-Nir & Robinson(1995); Grosfeld-Nir(1995); Grosfeld-Nir & Gerchak(2002); Grosfeld-Nir <i>et al.</i> (2006)
二項式機率分配	Grosfeld-Nir & Ronen(1993); Pentico(1994); Grosfeld-Nir & Robinson(1995); Grosfeld-Nir(1995); Barad & Braha(1996); Braha(1999); Grosfeld-Nir & Gerchak(2002); Grosfeld-Nir(2005); Grosfeld-Nir <i>et al.</i> (2006); Ben-Zvi & Grosfeld-Nir(2007)
全良品或全不良品分配	Grosfeld-Nir & Robinson(1995); Grosfeld-Nir(1995); Grosfeld-Nir & Gerchak(2002); Grosfeld-Nir <i>et al.</i> (2006); Ben-Zvi & Grosfeld-Nir(2007)
隨機比例機率分配	Lee & Yano(1988); Wein(1992); Grosfeld-Nir(1995)
中斷式幾何機率分配	Grosfeld-Nir & Robinson(1995); Grosfeld-Nir & Gerchak(2002); Grosfeld-Nir <i>et al.</i> (2006); Ben-Zvi & Grosfeld-Nir(2007)

2.4 依求解方法分類

多次投料問題擁有遞迴形式的成本函式，因此本質上常用動態規劃求解。然而在一些多次投料問題中，使用動態規劃求解可能是極沒效率的，因此有許多文獻針對問題發展更有效率的求解方法。其中有一些文獻提出能縮小求解空間的定理(Beja, 1977; Anily, 1995; Zhang & Guu, 1998)。有些文獻則提出啟發式的投料法則以求解近似最佳解(Silver, & New, 1986; Pentico, 1988)，也就是在什麼樣的條件下應該如何決定投料決策。另外少數文獻針對一些特殊情境提出非動態規劃求解的公式求解方法，如 Anily *et al.* (2002)提出需求極高或極低時的求解公式，可以快速求得最佳解。

第三章 中斷式幾何良率分配及交期限制下多次投料問題

本章首先定義本研究使用的符號。接著在 3.2 節中，我們介紹本研究探討的多次投料問題和中斷式幾何分配。投料決策制定後所產生的結果在 3.3 節說明。最後在 3.4 節中，本研究將此多次投料問題以數學模式表示並用動態規劃求解。

3.1 符號定義

本節定義本論文使用之所有符號如下

T ：可生產的總期數，即從接單至交期可用於生產的期數。

t ：剩餘的可生產期數。 $t = T, T-1, \dots, 2, 1, 0$ ，其中 $t=T$ 代表開始生產的時間點，而 $t=0$ 代表交期。

i ：生產階段 M_i 使用的編號， $i = 1, 2$ 。

$\alpha^{(i)}$ ：投料至 M_i 階段生產所需要的設置成本， $i = 1, 2$ 。

$\beta^{(i)}$ ：投料至 M_i 階段生產所需要的單位製造成本， $i = 1, 2$ 。

D ：起始的客戶需求量，也就是 $t=T$ 時的剩餘需求量。

D_t ：第 t 期時，還剩餘未滿足的顧客需求量。

h ：一單位成品持有一期的存貨持有成本。

m ：交期到時的單位缺貨成本。

$\theta^{(i)}$ ：各階段 M_i 的中斷式幾何良率參數，即製程失去控制的機率， $i = 1, 2$ 。

$k_t^{(i)}$ ：第 t 期投至 M_i 的批量， $i = 1, 2$ 。

$W_t^{(i)}$ ：代表階段 M_i 是否有投料的二元變數。

$$W_t^{(i)} = \begin{cases} 0 & \text{if } k_t^{(i)} = 0 \\ 1 & \text{if } k_t^{(i)} > 0. \end{cases}$$

$Y_{k_t^{(i)}}^{(i)}$ ：代表第 t 期投料 $k_t^{(i)}$ 單位至階段 M_i 而產出的良品個數之隨機變數，而 $y_{k_t^{(i)}}^{(i)}$

表示隨機變數 $Y_{k_t^{(i)}}$ 的結果， $y_{k_t^{(i)}} = 0, 1, \dots, k_t^{(i)}$ ， $i=1, 2$ 。

$p(y_{k_t^{(i)}})$ ：投入 $k_t^{(i)}$ 單位原料至階段 M_i 而產出良品個數剛好為 $y_{k_t^{(i)}}$ 的機率，而

$$\sum_{y_{k_t^{(i)}}=0}^{k_t^{(i)}} p(y_{k_t^{(i)}}) = 1, i = 1, 2。$$

B_t ：第 t 期時在製品的存貨水準，而 $B_t = B_{t+1} - k_{t+1}^{(2)} + Y_{k_{t+1}^{(1)}}$ 。

$s_t = (D_t, B_t)$ ：生產系統在第 t 期時的狀態，由剩餘需求量 D_t 與在製品存貨水準 B_t 組成，也稱為狀態 s_t 。

$N_t(s_t) = (k_t^{(1)}, k_t^{(2)})$ ：在狀態 s_t 的情況下的決策投料量組合，也就是投 $k_t^{(1)}$ 單位至階段 M_1 及 $k_t^{(2)}$ 單位至階段 M_2 。必須注意的是，由於本問題中兩階段的製程限制無法

同時加工，因此我們必須限制 $k_t^{(1)} \cdot k_t^{(2)} = 0$ 。

$C_t(N_t(s_t))$ ：在投料決策組合 $N_t(s_t)$ 的情況下，從第 t 期至第 0 期的期望總成本。

而 $C_t^*(s_t) = \underset{(k_t^{(1)}, k_t^{(2)})}{\text{Min}} \{C_t(N_t(s_t))\}$ ，也就是代表狀態 s_t 下，採用所有可行決策中的最佳決策，所產生的期望總成本。

$N_t^*(s_t) = (k_t^{(1)*}, k_t^{(2)*})$ ，表示狀態 s_t 下的最佳投料決策組合，也就是

$$C_t(N_t^*(s_t)) = \underset{(k_t^{(1)}, k_t^{(2)})}{\text{Min}} \{C_t(N_t(s_t))\} = C_t^*(s_t)。$$

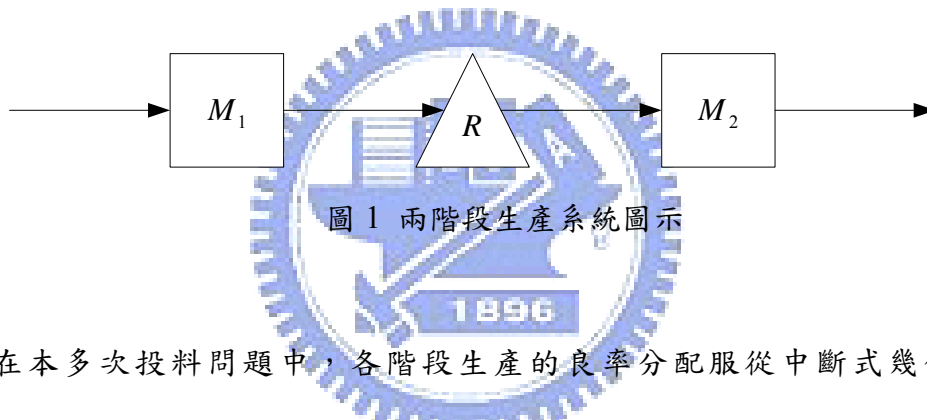
3.2 研究問題介紹

本論文探討的多次投料問題之生產系統包含兩階段的製程： M_1 及 M_2 ，中間有在製品暫存的緩衝區 R ，如圖 1 所示。 M_1 及 M_2 製程加工的生產週期時間都為一期。每道製程結束後進行整批加工品的品質檢驗。品質檢驗結果有缺陷的不良品將直接丟棄報廢。

為了在交期內滿足非嚴格的需求量，生產者會不斷的投入原料至 M_1 生產及

投入在製品進 M_2 生產。當上一期的產出在品質檢驗後得知有多少可用的良品及在製品時，我們首先必須決定本期應該進行第一階段或是第二階段的加工。接下來我們必須決定應該投入多少的量至被選定加工的階段。這兩個決策我們可以用投料決策組合 $N_t(s_t) = (k_t^{(1)}, k_t^{(2)})$ 表示，其中 $k_t^{(1)} \cdot k_t^{(2)} = 0$ 必須要成立，代表最多只進行某一階段的加工。

當然在第 t 期投至 M_2 生產的數量一定要小於可用的在製品存量，也就是 B_t 。這樣的生產系統必須不斷的進行投料直到交期或顧客的需求量已經完全被滿足了。在本多次投料問題中，只有考慮成品的持有成本，忽略在製品的持有成本。



在本多次投料問題中，各階段生產的良率分配服從中斷式幾何分配 (Interrupted Geometric)。此分配的機率密度函數 (Probability Density Function, PDF) 定義如下

$$P(Y_{k_t^{(i)}} = y_{k_t^{(i)}}) = \begin{cases} (1 - \theta^{(i)}) \cdot (\theta^{(i)})^{y_{k_t^{(i)}}} & y_{k_t^{(i)}} = 0, 1, 2, \dots, k_t^{(i)} - 1 \\ (\theta^{(i)})^{k_t^{(i)}} & y_{k_t^{(i)}} = k_t^{(i)} \end{cases}$$

這代表每道製程程序的加工狀態可能有兩種：正常運作中及失去控制的狀態。當製程處於正常運作的時候，加工製造出來的產品均是沒有缺陷的；然而若製程處於失去控制的狀態時，所有在此狀態下生產製造的產品皆屬於不良品。正常加工狀態下，有一固定的機率，系統會轉變為失去控制的狀態，然而當製程失去控制的時候，無法自動地恢復正常之運作。在中斷式幾何分配良率下，欲產出剛好 $y_{k_t^{(i)}}$ 單位的良品，系統在製造前 $y_{k_t^{(i)}}$ 單位時是屬於正常運作的狀態，而第

$(y_{k_t^{(i)}} + 1)$ 單位時系統故障轉變為失去控制的狀態。成型加工製程即為符合中斷式幾何分配良率的實務範例。許多成型加工製程需要搭配模具，當模具經由使用磨耗至無法正常生產的程度，生產系統就進入無法控制的狀態，用壞的模具生產出來的產品也將都是不良品。

3.3 多次投料問題決策的結果

如前所述，本系統之兩道製程具有互斥性，同時只能進行其中一道製程，因此 $k_t^{(1)} \cdot k_t^{(2)} = 0$ 。對第 t 期的投料決策 $(k_t^{(1)}, k_t^{(2)})$ 而言，其在第 $t-1$ 期可能產生的結果可以用圖 2 來表示。在第 t 期時生產系統的狀態為 $s_t = (D_t, B_t)$ ，表示剩餘未滿足的需求量為 D_t 而系統的在製品存貨水準為 B_t 。投入到 M_1 及 M_2 生產的量為 $k_t^{(1)}$ 及 $k_t^{(2)}$ ，而其相對應的產出則分別為 $y_{k_t^{(1)}}$ 及 $y_{k_t^{(2)}}$ 。根據投入跟產出的資訊，我們可以推得下一期的狀態 $s_{t-1} = (D_t - y_{k_t^{(2)}}, B_t - k_t^{(2)} + y_{k_t^{(1)}})$ 。

而 M_1 及 M_2 階段分別產出 $y_{k_t^{(1)}}$ 及 $y_{k_t^{(2)}}$ 單位良品的機率分別為 $p(y_{k_t^{(1)}})$ 及 $p(y_{k_t^{(2)}})$ 。因為產出的良品數目不會超過投入生產的量，所以我們可以得到 $0 \leq y_{k_t^{(1)}} \leq k_t^{(1)}$ 及 $0 \leq y_{k_t^{(2)}} \leq k_t^{(2)}$ 。根據產出量的可能範圍，一組投料決策 $(k_t^{(1)}, k_t^{(2)})$ 可能產生的結果數目有 $(k_t^{(1)} + 1) \cdot (k_t^{(2)} + 1)$ 個，其中 $k_t^{(1)} \cdot k_t^{(2)} = 0$ 。而每個結果的發生機率為 $p(y_{k_t^{(1)}}) \cdot p(y_{k_t^{(2)}})$ ，而所有可能結果的發生機率總和為一，即

$$\sum_{y_{k_t^{(1)}}=0}^{k_t^{(1)}} \sum_{y_{k_t^{(2)}}=0}^{k_t^{(2)}} p(y_{k_t^{(1)}}) \cdot p(y_{k_t^{(2)}}) = 1。$$

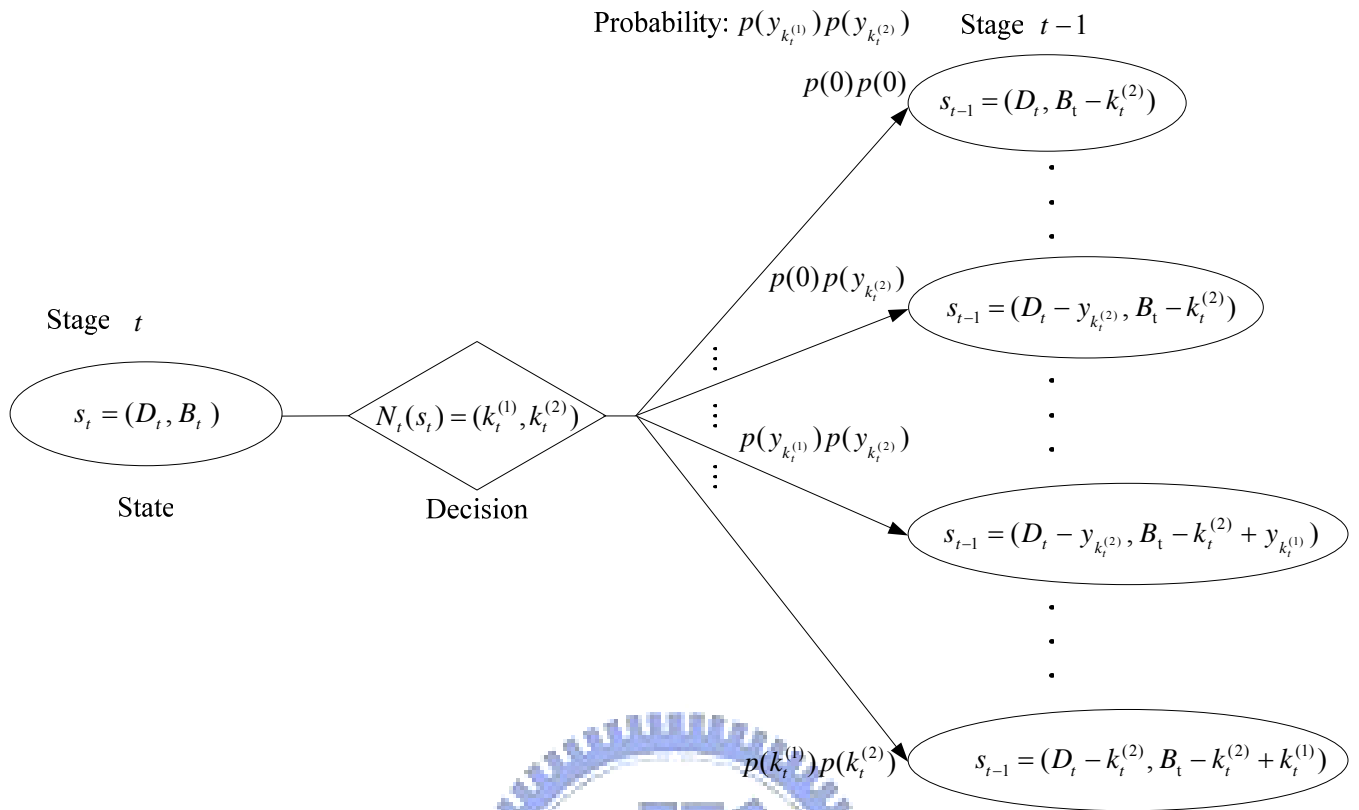


圖 2 動態規劃模式的投料決策結構

3.4 成本函式與動態規劃求解

假設我們在狀態 $s_t = (D_t, B_t)$ 時決定了一組投料決策 $N_t(s_t) = (k_t^{(1)}, k_t^{(2)})$ ，相對應之從第 t 期至第 0 其所產生的期望總成本可以用下式(1)表示

$$C_t(N_t(s_t)) = H_1 + H_2 + \sum_{y_{k_t^{(1)}}=0}^{k_t^{(1)}} \sum_{y_{k_t^{(2)}}=0}^{k_t^{(2)}} p(y_{k_t^{(1)}}) \cdot p(y_{k_t^{(2)}}) \cdot H_3 + \sum_{y_{k_t^{(2)}}=0}^{k_t^{(2)}} p(y_{k_t^{(2)}}) \cdot H_4 \quad (1)$$

其中

$$k_t^{(1)} \cdot k_t^{(2)} = 0$$

$$H_1 = \alpha^{(1)} W_t^{(1)} + \beta^{(1)} k_t^{(1)}$$

$$H_2 = \alpha^{(2)} W_t^{(2)} + \beta^{(2)} k_t^{(2)}$$

$$H_3 = C_{t-1}^*(s_{t-1}) \text{ , 而 } s_{t-1} = (D_t - y_{k_t^{(2)}} \text{ , } B_t - k_t^{(2)} + y_{k_t^{(1)}})$$

$$H_4 = h \cdot (t-1) \cdot y_{k_t^{(2)}}$$

式(1)中， H_1 項代表投批量 $k_t^{(1)}$ 至 M_1 生產的製造成本。製造成本包含只有投料量大於零才會產生的設置成本 $\alpha^{(1)}W_t^{(1)}$ 及變動製造成本 $\beta^{(1)}k_t^{(1)}$ 。同理 H_2 項代表投批量 $k_t^{(2)}$ 至 M_2 生產的製造成本。在式(1)的第三項中， $p(y_{k_t^{(1)}}) \cdot p(y_{k_t^{(2)}})$ 表示 M_1 階段生產出恰好 $y_{k_t^{(1)}}$ 單位的合格在製品且 M_2 階段生產出恰好 $y_{k_t^{(2)}}$ 單位的合格良品的聯合機率。而因為 $H_3 = C_{t-1}^*(s_{t-1})$ 代表 s_{t-1} 一組特定狀態下的最低期望總成本，

本， $\sum_{y_{k_t^{(1)}}=0}^{k_t^{(1)}} \sum_{y_{k_t^{(2)}}=0}^{k_t^{(2)}} p(y_{k_t^{(1)}}) \cdot p(y_{k_t^{(2)}}) \cdot H_3$ 即代表所有 s_{t-1} 可能狀態下的最低期望總成本。而 H_4 項表示 $y_{k_t^{(2)}}$ 單位良品，即投 $k_t^{(2)}$ 單位生產的一個可能結果，的存貨持有成本，因此

成本，因此 $\sum_{y_{k_t^{(2)}}=0}^{k_t^{(2)}} p(y_{k_t^{(2)}}) \cdot H_4$ 代表 $k_t^{(2)}$ 的所有可能產出結果之期望存貨持有成本。

式(1)包含了 $H_3 = C_{t-1}^*(s_{t-1})$ 項，這表示式(1)是一個遞迴函式。隱含著為了決定狀態 s_t 下的投料決策，我們必須知道在所有可能的 s_{t-1} 狀態下的最佳決策。依此類推，如果我們想要獲得起始狀態 s_T 下的最佳決策，我們必須先獲得所有可能的 s_{T-1}, \dots, s_1, s_0 狀態下的最佳決策。這也意味到遞迴函式(1)本質上是一個動態規劃問題。

因為狀態 s_{t-1} 是由狀態 s_t and $s_T = (D, 0)$ 而來，我們可以獲得整個動態規劃求解的網路—各個 t 下的所有可能狀態。不過必須注意的是，整個動態規劃求解的網路基本上是由投料決策 $k_t^{(1)}$ 及 $k_t^{(2)}$ 的搜尋上界決定，因此如何找到好的投料量搜尋上限是非常重要的。好的投料決策搜尋上限可以幫助降低求解方法的計算

複雜度與求解時間效率。

遞迴總成本函式(1)有兩個邊界條件定義了計算結束條件。第一個邊界條件是當狀態 $s_t = (0, B_t)$ 時，此時已經沒有剩餘未滿足的需求量了，也就是我們已經在交期前完全地滿足客戶的需求。所以在這個狀態下我們已經不再需要製造任何的產品，因此相對應的成本是零。依此我們可以獲得下述的式(2)及式(3)

$$N_t^*(s_t = (0, B_t)) = (0, 0) \quad (2)$$

$$C_t^*(s_t = (0, B_t)) = 0 \quad (3)$$

第二個邊界條件被定義在狀態 $s_0 = (D_0, B_0)$ 下。在 $t = 0$ 時表示交期已經到了，此時生產任何產品都無法賣給顧客，因此是不能滿足剩餘的需求量的。因為無法繼續生產，此時會產生缺貨的處罰成本，如下列式(4)。

$$C_0^*(s_0 = (D_0, B_0)) = mD_0 \quad (4)$$



第四章 最佳投料決策之界限

為了利用動態規劃求解此多次投料問題，本研究提出三個提供最佳投料界限的定理。Lemma 1 說明第二階段的最佳投料量上限為 $\text{Min}\{D_t, B_t\}$ 。Lemma 2 以及 Lemma 3 則共同提供了第一階段的最佳投料量上限，也就是 $\text{Min}\left\{(t-1) \cdot D_t - B_t, \left\lceil \frac{\ln \beta^{(1)} - \ln m}{\ln \theta^{(1)}} \right\rceil - 1 \right\}$ 。以下為這三個定理及所需的定理成立所需要的性質。

Proposition 1: $C_{t-1}^*(s_{t-1} = (D_t, B_t - n)) \geq C_{t-1}^*(s_{t-1} = (D_t, B_t))$ 在 $B_t \geq n \geq 1$ 且 n 是整數的情況下恆成立

Proof

在本問題設定的情境中，若我們刻意保留 n 單位在製品不生產直到交期日，該 n 單位在製品會被丟棄，而不會產生多餘的成本。也就是說，多餘的在製品並不會造成任何的 cost。且狀態 $s_{t-1} = (D_t, B_t - n)$ 下的所有可行解在狀態 $s_{t-1} = (D_t, B_t)$ 下也都是可行解，即狀態 $s_{t-1} = (D_t, B_t - n)$ 下之解空間為狀態 $s_{t-1} = (D_t, B_t)$ 下之解空間的子集合。因此我們可以推論在 $s_{t-1} = (D_t, B_t - n)$ 情況下的最佳成本 $C_{t-1}^*(s_{t-1} = (D_t, B_t - n))$ 一定可以在 $s_{t-1} = (D_t, B_t)$ 的狀態下達成。綜合上述，我們可以得知 $C_{t-1}^*(s_{t-1} = (D_t, B_t - n)) \geq C_{t-1}^*(s_{t-1} = (D_t, B_t))$ 。□

Lemma 1: 給定 $s_t = (D_t, B_t)$ ，其中 $t \geq 1$ 且 $D_t \geq 1$ ，我們可以知道 $k_t^{(2)*} \leq D_t$ 且

$$k_t^{(2)*} \leq B_t \text{ 恆成立}$$

Proof

B_t 是可用於第二階段生產的在製品數，因此 $k_t^{(2)*} \leq B_t$ 是一定成立的。

為了證明 $k_t^{(2)*} \leq D_t$ ，我們可以考慮兩個情況

情況 1： $k_t^{(1)*} > 0$

因為 $k_t^{(1)*} \cdot k_t^{(2)*} = 0$ 的限制，使得 $k_t^{(2)*} = 0 \leq D_t$ 。

情況 2： $k_t^{(1)*} = 0$

令 $N_t'(s_t) = (0, D_t)$ 及 $N_t''(s_t) = (0, D_t + n)$

其相對應的成本則分別為

$$C_t(N_t'(s_t)) = \alpha^{(2)} + \beta^{(2)} \cdot (D_t) + h(t-1) \cdot E[Y_{D_t}]$$

$$+ \sum_{y_{D_t}=0}^{D_t} p(y_{D_t}) \cdot C_{t-1}^*(s_{t-1} = (D_t - y_{D_t}, B_t - D_t))$$

及

$$C_t(N_t''(s_t)) = \alpha^{(2)} + \beta^{(2)} \cdot (D_t + n) + h(t-1) \cdot E[Y_{D_t+n}]$$

$$+ \sum_{y_{D_t+n}=0}^{D_t+n} p(y_{D_t+n}) \cdot C_{t-1}^*(s_{t-1} = (D_t - y_{D_t+n}, B_t - (D_t + n)))$$

$$C_t(N_t''(s_t)) - C_t(N_t'(s_t))$$

$$= n \cdot \beta^{(2)} + h \cdot (t-1) \cdot (E[Y_{D_t+n}] - E[Y_{D_t}]) + A - B, \text{ 其中}$$

$$A = \sum_{y_{D_t+n}=0}^{D_t-1} p(y_{D_t+n}) \cdot C_{t-1}^*(s_{t-1} = (D_t - y_{D_t+n}, B_t - (D_t + n)))$$

$$B = \sum_{y_{D_t}=0}^{D_t-1} p(y_{D_t}) \cdot C_{t-1}^*(s_{t-1} = (D_t - y_{D_t}, B_t - (D_t)))$$

根據性質 1，我們可以獲得 $A \geq B$ 。因此

$$C_t(N_t''(s_t)) - C_t(N_t'(s_t)) \geq n \cdot \beta^{(2)} + h \cdot (t-1) \cdot (E[Y_{D_t+n}] - E[Y_{D_t}]) > 0$$

上述結果表示 $k_t^{(2)*} \leq D_t$ 。 □

Lemma 2: $k_t^{(1)*} \leq (t-1) \cdot D_t - B_t$

在任何的期數 t 下 $D_t \geq D_{t-1}$ 皆成立。然後根據定理 1，我們可以推論得到 $\sum_{\tau=1}^{t-1} k_{\tau}^{(2)*} \leq (t-1) \cdot D_t$ 。也就是第二階段從第 $t-1$ 期至第一期的總投料量不會超過 $(t-1) \cdot D_t$ 。這表示階段二所需要的在製品個數最多是 $(t-1) \cdot D_t$ 。而第 t 期時生產系統中尚有 B_t 的在製品存貨量水準，因此第 $t-1$ 期時第一階段所需要產出之良品最多是 $(t-1) \cdot D_t - B_t$ 。我們可以把第二階段是為第一階段的顧客，其需求量最多是 $(t-1) \cdot D_t - B_t$ 。根據已發表的文獻(Guu & Zhang, 2003)，擁有中斷式幾何良率分配的單階段生產系統中，最佳投料量不會超過需求量。綜合上述，我們可以得知第 t 期時第一階段的最佳投料量不會超過 $(t-1) \cdot D_t - B_t$ 。也就是 $k_t^{(1)*}$ 的上界為 $(t-1) \cdot D_t - B_t$ 。



Lemma 3: 如果 $t \geq 1$ ， $D_t \geq 1$ 且 $s_t = (D_t, B_t)$ ，其中 $B_t \geq 0$ ，

則 $k_t^{(1)*} \leq \left\lceil \frac{\ln \beta^{(1)} - \ln m}{\ln \theta^{(1)}} \right\rceil - 1$ 恆成立

Proof

令 $N_t'(s_t) = (k, 0)$ for $k \geq 1$

$$\text{則 } C_t(N_t'(s_t)) = \alpha^{(1)} + \beta^{(1)} \cdot (k) + \sum_{y_k=0}^k p(y_k) \cdot C_{t-1}^*(s_{t-1} = (D_t, B_t + y_k))$$

令 $N_t''(s_t) = (k+1, 0)$ for $k \geq 1$

$$\text{則 } C_t(N_t''(s_t)) = \alpha^{(1)} + \beta^{(1)} \cdot (k+1) + \sum_{y_{k+1}=0}^{k+1} p(y_{k+1}) \cdot C_{t-1}^*(s_{t-1} = (D_t, B_t + y_{k+1}))$$

$$C_t(N_t''(s_t)) - C_t(N_t'(s_t))$$

$$= \beta^{(1)} - (\theta^{(1)})^{k+1} \cdot [C_{t-1}^*(s_{t-1} = (D_t, B_t + k)) - C_{t-1}^*(s_{t-1} = (D_t, B_t + k + 1))]$$

考慮一個特定狀態 $(D_t, B_t + k)$ 和比它多一單位在製品的狀態 $(D_t, B_t + k + 1)$ 。假設該多的在製品不需要任何成本即可轉換為一單位的良品。在這個極端的情況

下，多一單位在製品的狀態最多也只能比原本的狀態減少一單位的缺貨或節省一單位的缺貨處罰成本。因此我們可以推論得到

$$C_{t-1}^*(s_{t-1} = (D_t, B_t + k)) - C_{t-1}^*(s_{t-1} = (D_t, B_t + k + 1)) \leq m$$

根據上述，我們可以得到 $C_t(N_t''(s_t)) - C_t(N_t'(s_t)) \geq \beta^{(1)} - (\theta^{(1)})^{k+1} \cdot m$

所以如果 $\beta^{(1)} \geq (\theta^{(1)})^{k+1} \cdot m$ ，我們可以推得 $C_t(N_t''(s_t)) - C_t(N_t'(s_t)) \geq 0$ 。

這表示如果投料量 k 在 $\beta^{(1)} \geq (\theta^{(1)})^{k+1} \cdot m$ 的條件下，多一單位的投料量無法得到更低的期望總成本。

因此， $\beta^{(1)} = (\theta^{(1)})^{k+1} \cdot m$ 定義了最佳投料量的上限

也就是 $k = \frac{\ln \beta^{(1)} - \ln m}{\ln \theta^{(1)}} - 1$ 最佳投料量 $k_t^{(1)*}$ 的上限

$$\text{故 } k_t^{(1)*} \leq \left\lceil \frac{\ln \beta^{(1)} - \ln m}{\ln \theta^{(1)}} \right\rceil - 1 \quad \square$$



本論文提出三個最佳投料決策界限的定理，使得利用動態規劃求得最佳投料決策的方法是可行的。我們利用動態規劃手法求解時，只要知道哪些狀態是必須計算的，就能由後向前逐一計算各狀態的最佳投料決策與相對應成本。

第五章 簡化動態規劃網路之求解演算法

本研究提出一個可以簡化原始動態規劃網路的演算法。此演算法基本的概念是將數個投料決策的結果結合成一個，因此可以大幅縮減需要計算的狀態。演算法可以分為三個主要的步驟：首先是將一個投料決策可能產生的所有結果群聚成幾組。第二步驟是計算每一組的代表值。第三步驟則利用這些代表值取代原本的所有可能結果，簡化原本動態規劃網路。

第一個主要步驟—投料決策結果的群聚說明如下。如前所述，一個投料決策 $(k_t^{(1)}, k_t^{(2)})$ 所可能產生的結果有 $(k_t^{(1)} + 1) \cdot (k_t^{(2)} + 1)$ 個。假使我們制定了一個投料決策 $k_t^{(1)}$ ，也就是在第一部機台投入 $k_t^{(1)}$ 數量的原料，所有可能的結果有 $k_t^{(1)} + 1$ 個，即 $0, 1, \dots, k_t^{(1)}$ 。在這些可能的結果中，我們可以透過把每 n 個群聚成一組的方法，將這些結果變成 $\left\lceil \frac{k_t^{(1)} + 1}{n} \right\rceil$ 組。例如當我們決定 $n = 2$ 和 $k_t^{(1)} = 6$ ，我們可以得

到四組： $\{0, 1\}, \{2, 3\}, \{4, 5\}, \{6\}$ 。同理我們也可以將第二部機台的投料決策 $k_t^{(2)}$ 分為 $\left\lceil \frac{k_t^{(2)} + 1}{n} \right\rceil$ 組。

第二個主要步驟是決定各組的代表值與各組發生的機率。令 $Q_j = \{y \mid n(j-1) \leq y \leq nj-1, y \text{ denotes the possible outcome in group } j\}$ 代表第 j 組結

果，包含投料決策 $k_t^{(1)}$ 或 $k_t^{(2)}$ 下 n 個連續的可能結果， $j = 1, \dots, \left\lceil \frac{k_t^{(i)} + 1}{n} \right\rceil$ ；若每

種可能結果的發生機率用 $p(y)$ 表示，則第 j 組結果 Q_j 可以用

$Round\left(\frac{\sum_{y=n(j-1)}^{nj-1} y \cdot p(y)}{\sum_{y=n(j-1)}^{nj-1} p(y)}\right)$ 來代表，其對應的機率為 $\sum_{y=n(j-1)}^{nj-1} p(y)$ 。換句話

說，我們用所有可能結果的期望值做為該組的代表值，而期望值發生的機率為原本該組內所有可能結果的機率總和。接續上例，第二組 ($j = 2$) 的結果 $Q_2 = \{2, 3\}$ ，

各結果發生的機率分別為 $p(2) = 0.1$ 及 $p(3) = 0.2$ 。所以第二組的代表值為 $Round\left(\frac{(2 \cdot 0.1) + (3 \cdot 0.2)}{(0.1 + 0.2)}\right) = Round(2.7) = 3$ 。而該組的發生機率為 $0.1 + 0.2 = 0.3$ 。

第三個步驟則是簡化原本的動態規劃網路。簡化的基本邏輯是利用一個單一結果來代表每一組，也就是用每一組的代表值及機率取代該組的各種可能結果。假設一組投料決策 $(k_t^{(1)}, k_t^{(2)})$ 被制定，且每 n 個連續的可能結果被歸為一組。特過簡

化的方法，投料決策 $k_t^{(1)}$ 現在只有 $\left\lceil \frac{k_t^{(1)} + 1}{n} \right\rceil$ 種結果而 $k_t^{(2)}$ 只有 $\left\lceil \frac{k_t^{(2)} + 1}{n} \right\rceil$ 種結果。換

句話說，投料決策 $(k_t^{(1)}, k_t^{(2)})$ 產生的結果現在變成 $\left\lceil \frac{k_t^{(1)} + 1}{n} \right\rceil \cdot \left\lceil \frac{k_t^{(2)} + 1}{n} \right\rceil$ 個。簡化的

結果會使得需要計算的狀態數比起原先的動態規劃網路大幅減少。

當設定 $n = 1$ 時，簡化的方法就等同於原本的方法。為了後續章節演算法比較上的方便，用動態規劃求解原本遞迴網路的方法我們稱之為 *comprehensive algorithm*，而利用簡化網路求解的方法則稱之為 *reduction algorithm*。



第六章 實驗

本論文提出兩種求解此多次投料問題的演算法，分別是 *comprehensive algorithm* 及 *reduction algorithm*。為了比較兩種演算法的績效以及適用情境，我們執行了許多不同情境下的電腦實驗。執行實驗所使用的電腦規格為：3.4 GHZ 的 CPU 及 504MB 的記憶體。

演算法的求解品質與求解效率分別用兩個指標來衡量： $R_C = \frac{C_r - C_p}{C_p}$ 及 $R_T = \frac{T_r}{T_p}$ ，其中 C_r 是 *reduction algorithm* 制定之投料決策產生的期望總成本， T_r 是利用 *reduction algorithm* 求解所需的時間；而 C_p 是由 *comprehensive algorithm* 制定之投料決策產生的期望總成本， T_p 則是 *comprehensive algorithm* 的平均計算時間。

根據實驗結果，我們建議在大規模問題(也就是 TD 大)且高良率的情境下使用本論文提出的 *reduction algorithm*。而在其他情境下，可以使用完整的動態規劃方法，可以在實務允許的計算時間下求得最佳決策。

表 2 顯示在某些情境下運用 *comprehensive algorithm* 是沒有效率的。表 2 的實驗情境是固定 $h=1, m=100, \alpha^{(1)}=50, \alpha^{(2)}=50$ ，下透過變動 $(T, D) \in \{(10,100), (10,200), (20,100)\}$ 及 $\theta^{(1)}, \theta^{(2)} \in \{0.9, 0.99, 0.999\}$ 產生 27 個情境範例。實驗結果顯示當 $\theta^{(1)} \geq 0.99$ 及 $(T, D) \in \{(10,200), (20,100)\}$ 時，使用 *comprehensive algorithm* 計算耗費約 5.4-16.0 小時。會需要那麼長的計算時間主要是因為 $k_i^{(1)*}$ 的搜尋界限，也就是

$$k_i^{(1)*} \leq \min \left\{ \left(\left\lceil \frac{\ln \beta^{(1)} - \ln m}{\ln \theta^{(1)}} \right\rceil - 1 \right), ((t-1)D_i - B_i) \right\},$$

此界限顯示當 $\theta^{(1)}, D, T$ 值越高時，最佳投料決策 $k_i^{(1)*}$ 也越高，因此也需要更多的計算時間。

如表 3 所示，這些大規模的高良率問題可以利用 reduction algorithm 求解，其求解效率非常好且求解誤差很小。當 $n = 5$ 時， R_T 的範圍從 2.94% 到 3.50% 而 R_c 則從 0.06% 到 4.30%。使用 reduction algorithm 可以大幅縮減計算所需的時間。以 $(T, D) = (20, 100)$ 及 $(\theta^{(1)}, \theta^{(2)}) = (0.999, 0.999)$ 的情境為例，利用 reduction algorithm 可以將計算時間由 16 小時縮短至 30 分鐘。此外，表 4-6 顯示當 n 的值增加， R_T 會趨向減少而 R_c 會漸增。

相反的，comprehensive algorithm 則最適合用於良率相對較低的情境(也就是 $\theta^{(1)} \leq 0.9$)。表 7 是以 comprehensive algorithm 在良率相對較低之情境的平均求解時間。該表中的值是在各個 (T, D) 情境下 16 個測試範例的平均求解時間。此 16 個測試範例設定如下： $\theta^{(1)}, \theta^{(2)} \in \{0.6, 0.8\}$ ， $\beta^{(1)} = \beta^{(2)} \in \{1, 2\}$ ， $m \in \{100, 200\}$ ，以及 $\alpha^{(1)} = \alpha^{(2)} = 50$ 。

實務上，量率高於 0.99 (也就是 $\theta^{(1)} \geq 0.99$) 的製程並不少見。成型製程中，模具的良率往往高於 0.99；否則就會因為時常需要更換模具而造成很高的成本。因此，本研究提出之 reduction algorithm 可以幫助現場人員快速的求解多次投料問題。

表 2 利用 comprehensive algorithm 求解的 C_p and T_p

(T, D) $(\theta^{(1)}, \theta^{(2)})$		(10,100)		(10,200)		(20,100)	
		C_p (\$)	T_p (sec.)	C_p (\$)	T_p (sec.)	C_p (\$)	T_p (sec.)
0.9	0.9	8070.76	220	18070.80	1019	6228.48	1875
0.9	0.99	5058.79	169	15026.30	707	2151.69	821
0.9	0.999	3578.58	158	13304.30	710	1437.32	789
0.99	0.9	5016.30	2394	14972.70	19489	2629.97	23309
0.99	0.99	1018.96	2412	2561.08	22080	1017.97	23057
0.99	0.999	689.91	2324	1363.89	20789	689.90	22979
0.999	0.9	3984.39	2754	13742.00	39414	2343.91	57688
0.999	0.99	847.77	2738	1846.54	43191	847.77	57377

0.999	0.999	541.07	2703	1057.13	41408	541.07	57056
-------	-------	--------	------	---------	-------	--------	-------

表 3 $n = 5$ 時利用 reduction algorithm 求解的 R_C and R_T

(T, D) $(\theta^{(1)}, \theta^{(2)})$		(10,100)		(10,200)		(20,100)	
		R_C (%)	R_T (%)	R_C (%)	R_T (%)	R_C (%)	R_T (%)
0.9	0.9	0.28	3.29	0.13	2.05	0.72	2.36
0.9	0.99	0.05	5.19	0.02	4.48	0.33	4.70
0.9	0.999	0.07	4.91	0.00	4.36	0.29	4.45
0.99	0.9	0.58	3.68	0.06	3.34	1.18	3.16
0.99	0.99	0.31	3.75	0.14	3.50	0.17	3.32
0.99	0.999	0.06	3.64	0.03	3.38	0.06	3.15
0.999	0.9	0.33	3.67	0.64	3.24	4.30	2.94
0.999	0.99	0.15	3.75	0.21	3.42	0.15	3.03
0.999	0.999	0.01	3.57	0.03	3.31	0.01	3.03

表 4 $n = 10$ 時利用 reduction algorithm 求解的 R_C and R_T

(T, D) $(\theta^{(1)}, \theta^{(2)})$		(10,100)		(10,200)		(20,100)	
		R_C (%)	R_T (%)	R_C (%)	R_T (%)	R_C (%)	R_T (%)
0.9	0.9	0.27	2.18	0.12	1.25	0.56	1.39
0.9	0.99	0.14	4.45	0.04	2.85	1.70	3.22
0.9	0.999	0.19	3.35	0.03	2.77	2.61	2.92
0.99	0.9	0.92	2.15	0.08	1.92	11.51	1.76
0.99	0.99	1.01	2.23	0.22	2.01	1.11	1.76
0.99	0.999	0.09	2.13	0.07	1.91	0.09	1.73
0.999	0.9	6.10	2.14	0.12	1.77	6.71	1.59
0.999	0.99	0.77	2.23	0.16	1.91	0.77	1.61
0.999	0.999	0.03	2.10	0.17	1.82	0.03	1.60

表 5 $n = 15$ 時利用 reduction algorithm 求解的 R_C and R_T

(T, D) $(\theta^{(1)}, \theta^{(2)})$		(10,100)		(10,200)		(20,100)	
		R_C (%)	R_T (%)	R_C (%)	R_T (%)	R_C (%)	R_T (%)
0.9	0.9	0.59	1.49	0.26	0.75	1.06	0.99
0.9	0.99	0.29	2.59	0.07	1.91	3.83	2.23
0.9	0.999	0.54	2.48	0.03	1.89	18.44	2.10
0.99	0.9	0.87	1.65	0.22	1.44	14.67	1.23
0.99	0.99	1.28	1.66	0.58	1.48	1.27	1.27
0.99	0.999	0.85	1.61	0.26	1.41	0.85	1.25
0.999	0.9	0.87	1.61	10.65	1.28	12.36	1.05
0.999	0.99	3.10	1.68	0.92	1.36	3.10	1.12
0.999	0.999	0.05	1.61	0.24	1.32	0.05	1.12

表 6 $n = k_i + 1$ 時利用 reduction algorithm 求解的 R_C and R_T

(T, D) $(\theta^{(1)}, \theta^{(2)})$		(10,100)		(10,200)		(20,100)	
		R_C (%)	R_T (%)	R_C (%)	R_T (%)	R_C (%)	R_T (%)
0.9	0.9	1.31	0.18	0.58	0.04	2.96	0.21
0.9	0.99	4.67	0.73	1.73	0.24	30.71	0.26
0.9	0.999	3.88	0.62	1.61	0.42	53.22	0.26
0.99	0.9	3.30	0.38	1.00	0.12	18.22	0.18
0.99	0.99	103.94	0.35	32.27	0.19	104.14	0.20
0.99	0.999	54.80	0.33	90.92	0.17	54.80	0.19
0.999	0.9	2.36	0.56	0.76	0.18	20.47	0.15
0.999	0.99	87.78	0.59	60.76	0.28	87.78	0.18
0.999	0.999	25.02	0.56	91.44	0.26	25.02	0.17

表 7 各 (T, D) 組合下利用 comprehensive algorithm 求解的平均時間(單位:秒)

$T \backslash D$	10	50	100	200
10	0.13	4.17	13.13	21.20
15	0.34	12.74	43.03	167.95
20	0.33	26.32	123.21	496.78

第七章 結論

本論文探討一個新的兩階段多次投料問題，擁有中斷式幾何良率分配及具有交期限制的訂單需求。兩階段的加工時間都是一期，而每一期只進行其中一階段加工。因此當某一階段的加工完成時，我們必須制定的投料決策包含接下來應該進行哪一階段加工以及應該投入的批量大小。

本研究將這個多次投料問題用數學模式表示，以動態規劃求解。我們也提出最佳投料量上限的定理，有效的減小最佳解搜尋空間。然而在大規模且良率很高的情境下，單純使用動態規劃求解仍然需要很長的時間。為了解決求解效率上的問題，本研究提出了一個簡化動態規劃網路的演算法，將數個動態規劃網路的分支結合為一個，因此得到一個簡單許多的求解網路。

實驗結果顯示簡化的演算法可以有效率且準確地求解大規模且高良率情境下之多次投料問題。一個單純使用動態規劃需要 16 小時的多次投料問題，在使用簡化的演算法後，可以在 30 分鐘內獲得滿意解。

本研究仍然存有許多延伸的空間。其中一個可能的延伸是應用此簡化的演算法於其他的多次投料問題。文獻上有許多多次投料問題以動態規劃求解，有些文獻針對特殊的問題在大規模情境下提出啟發式的解法。因此若應用本研究提出之簡化式演算法於其他的多次投料問題中，可以與其他研究提出之啟發式解法作計算效率及品質上的績效比較。

參考文獻

- Anily S (1995). Single machine lot sizing with uniform yields and rigid demands: robustness of the optimal solution. *IIE Transactions* **27**: 625-633.
- Anily S, Beja A and Mendel A (2002). Optimal lot sizes with geometric production yield and rigid demand. *Operations Research* **5**: 424-432.
- Barad M and Braha D (1996). Control limits for multi-stage manufacturing processes with binomial yield (single and multiple production runs). *Journal of the Operational Research Society* **47**: 98-112.
- Beja A (1977). Optimal reject allowance with constant marginal production efficiency. *Naval Research Logistics Quarterly* **24**: 21-33.
- Ben-Zvi T and Grosfeld-Nir A (2007). Serial production systems with random yield and rigid demand: a heuristic. *Operations Research Letters* **35**: 235-244.
- Braha D (1999). Manufacturing control of a serial system with binomial yields, multiple production runs, and non-rigid demand: a decomposition approach. *IIE Transactions* **31**: 1-9.
- Grosfeld-Nir A (1995). Single bottleneck systems with proportional expected yields and rigid demand. *European Journal of Operational Research* **80**: 297-307.
- Grosfeld-Nir A (2005). A two-bottleneck system with binomial yields and rigid demand. *European Journal of Operational Research* **165**: 231-250.
- Grosfeld-Nir A and Gerchak Y (1990). Multiple lotsizing with random common-cause yield and rigid demand. *Operations Research Letters* **9**: 383-388.

Grosfeld-Nir A and Gerchak Y (1996). Production to order with random yields: single-stage multiple lot-sizing. *IIE Transactions* **28**: 669-676.

Grosfeld-Nir A and Gerchak Y (2002). Multistage production to order with rework capability. *Management Science* **48**: 652–664.

Grosfeld-Nir A and Gerchak Y (2004). Multiple lot sizing in production to order with random yields: review of recent advances. *Annals of Operations Research* **126**: 43-69.

Grosfeld-Nir A and Robinson LW (1995). Production to order on a two machine line with random yields and rigid demand. *European Journal of Operational Research* **80**: 264-276.

Grosfeld-Nir A and Ronen B (1993). A single bottleneck system with binomial yields and rigid demand. *Management Science* **39**: 650-653.

Grosfeld-Nir A, Anily S and Ben-Zvi T (2006). Lot-sizing two-echelon assembly systems with random yields and rigid demand. *European Journal of Operational Research* **173**: 600-616.

Guu SM (1999). Properties of the multiple lot-sizing problem with rigid demand, general cost structures, and interrupted geometric yield. *Operations Research Letters* **25**: 59-65.

Guu SM and Liou FR (1999). An algorithm for the multiple lot sizing problem with rigid demand and interrupted geometric yield. *Journal of Mathematical Analysis and Applications* **234**: 567-579.

Guu SM and Zhang AX (2003). The finite multiple lot sizing problem with

- interrupted geometric yield and holding costs. *European Journal of Operational Research* **145**: 635-644.
- Lee HL and Yano CA (1988). Production control in multistage systems with variable yield losses. *Operations Research* **36**: 269-278.
- Pentico DW (1988) An evaluation and proposed modification of the Sepheri-Silver-New heuristic for multiple lot sizing under variable yield. *IIE Transactions* **20**: 360-363.
- Pentico DW (1994). Multistage production systems with random yield: heuristics and optimality. *International Journal of Production Research* **32**: 2455-2462.
- Sepheri M, Silver EA and New C (1986). A heuristic for multiple lot sizing for an order under variable yield. *IIE Transactions* **18**: 63-69.
- Wein AS (1992). Random yield, rework and scrap in multistage batch manufacturing environment. *Operations Research* **40**: 551-563.
- Yano CA and Lee HL (1995). Lot sizing with random yields: a review. *Operations Research* **43**: 311-334.
- Zhang AX and Guu SM (1997). Properties of the multiple lot-sizing problem with rigid demands and general yield distributions. *Computers and Mathematics with Applications* **33**: 55-65.
- Zhang AX and Guu SM (1998). The multiple lot sizing problem with rigid demand and interrupted geometric yield. *IIE Transactions* **30**: 427-431.