

國立交通大學
工業工程與管理學系

碩士論文

TFT-LCD 排序機派工之研究
A Sorter Dispatching Algorithm for TFT-LCD



研究生：王君豪
指導教授：巫木誠 博士

中華民國九十四年六月

TFT-LCD 排序機派工之研究

A Sorter Dispatching Algorithm for TFT-LCD Manufacturing

研究生：王君豪

Student：Chun-Hao Wang

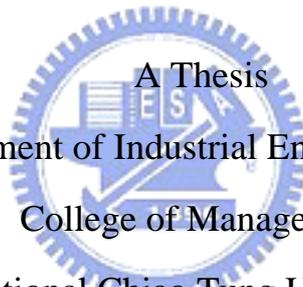
指導教授：巫木誠 博士

Advisor：Dr. Muh-Cherng Wu

國立交通大學

工業工程與管理學系

碩士論文



Submitted to Department of Industrial Engineering and Management

College of Management

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree of Master of Science

In

Industrial Engineering

June 2005

Hsin-Chu, Taiwan, Republic of China

中華民國九十四年六月

TFT-LCD 排序機派工之研究

研究生：王君豪

指導教授：巫木誠 博士

國立交通大學工業工程與管理研究所

中文摘要

薄膜液晶顯示器(thin film transistor-liquid crystal display, TFT-LCD)主要的製程可以分成三部份：陣列製程、組立製程以及模組製程。在組立製程中，為了使薄膜電晶體基板(TFT-plate)及彩色濾光片基板(CF-plate)壓合後的良率提高，必須先將卡匣內的基板進行良率配對、送上排序機抽取及重新安置的動作。本研究提出了一個排序機派工的法則，該法則利用基因演算法，求解卡匣在排序機上重置的完工順序，期望在屬於連續性的製造流程中，達到下游組立作業機台利用率的最大化。經過模擬實例驗證後，本研究發展的派工演算法，其下游機台利用率均顯著優於業界目前使用的方法。

關鍵字：派工、薄膜液晶顯示器、組立作業、排序機

A Sorter Dispatching Algorithm for TFT-LCD manufacturing

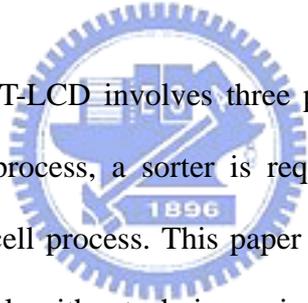
Student : Chun-Hao Wang

Advisor : Dr. Muh-Cherng Wu

Department of Industrial Engineering and Management

National Chiao Tung University

Abstract



The manufacturing of TFT-LCD involves three processes: array process, cell process, and module process. In cell process, a sorter is required to rearrange the TFT plates to facilitate the operation of the cell process. This paper presents a method for dispatching the sorters. The method, a genetic algorithm technique, aims to make the downstream cell process not idle due to lack of WIPs. In terms of the utilization of the cell process, the algorithm is significantly higher than the business-as-usually practice by extensive simulation.

Keywords: TFT-LCD, sorter, dispatching, cell process

誌謝

本論文得以順利完成，首先要感謝巫木誠教授在這兩年來的細心指導，除了學術領域的指引之外，於做事態度及人際相處方面，亦使學生獲益良多，在此致上最誠摯的謝意。同時，十分感謝口試老師許錫美博士及彭德保博士，針對論文提供了許多寶貴的意見，使學生的論文更臻完善，為我兩年的研究生涯劃上了完美的休止符。

研究所兩年中，感謝同門的陳尚宏、謝岳霖、林雅娟、蔡正航、洪挺耀及林聲宇的陪伴、互相激勵與安慰。此外，感謝同研究室的蘇泰盛學長、施昌甫學長、吳宜穆、蘇瑋婷、廖建閔、蔡宜娟及張貽朝，你們的出現，豐富了我的人生，也讓我在緊張、忙碌的生活中，留下了許多美好的回憶。

最後要深深感謝我的父母親王振職先生與連淑媛女士、姊姊香云和弟弟劭瑜，由於你們不斷的鼓勵與支持，讓我能夠埋首於論文學業上，衷心感謝你們所給予我的精神支持。僅以此論文獻給最敬愛的家人以及所有關心我的師長、朋友及學弟妹。

君豪

于 風城交大

2005-06-13

目錄

中文摘要	i
Abstract.....	ii
誌謝	iii
目錄	iv
表目錄	v
圖目錄	vi
第一章 緒論	1
1.1 研究背景	1
1.2 研究問題	5
1.3 論文組織	5
第二章 基板配對決策之整數規劃模型	6
2.1 模型符號	6
2.2 整數規劃模式	7
2.3 實例說明	8
第三章 研究方法	9
3.1 問題分析與解構	9
3.2 第二子題的解法	11
3.3 第一子題的解法	12
3.3.1 符號說明	13
3.3.2 染色體編碼方式及初始母體的產生	13
3.3.3 適應函數(fitness function).....	13
3.3.4 子代產生方式	14
3.3.5 終止條件	16
第四章 案例驗證	17
4.1 輸入資料與參數設定	17
4.2 結果與分析	17
第五章 結論與未來研究方向	22
5.1 結論	22
5.2 未來研究方向	22
參考文獻	23

表目錄

表 2.1 單目標函數的基板配對結果	8
表 2.2 雙目標函數的基板配對結果	8
表 4.1 單位裝卸卡匣時間為 0.02T 的模擬結果	18
表 4.2 單位裝卸卡匣時間為 0.05T 的模擬結果	19



圖目錄

圖 1.1	TFT-LCD製程流程	1
圖 1.2	TFT基板和CF基板配對良率	2
圖 1.3	排序機組成	3
圖 1.4	卡匣組成關係圖	5
圖 3.1	染色體表示法	13
圖 3.2	染色體表示法	15
圖 3.3	突變示意圖	16



第一章 緒論

1.1 研究背景

薄膜電晶體液晶顯示器(thin film transistor-liquid crystal display，以下簡稱 TFT-LCD)是近年來蓬勃發展的產業，由於 TFT-LCD 廠的投資相當高，如何有效提高製程良率，縮短生產週期時間實在非常重要[1]，過去已有許多研究[7][10][12]專注於其生產過程中排程的問題。

TFT-LCD 的製程主要可以分成三部份[11]：陣列(array)製程、組立(cell)製程及模組(module)製程(圖 1.1)。陣列製程是製作薄膜電晶體基板(TFT-plate)，組立製程是將製作完成的薄膜電晶體基板和自製或外購的彩色濾光片(color filter plate, CF-plate)壓合，注入液晶形成 TFT-LCD 基板(TFT-LCD plate)，再將基板切割成一般常見的面板(panel)。模組製程是將面板與背光模組、驅動 IC、印刷電路板等其他附件進行組立。

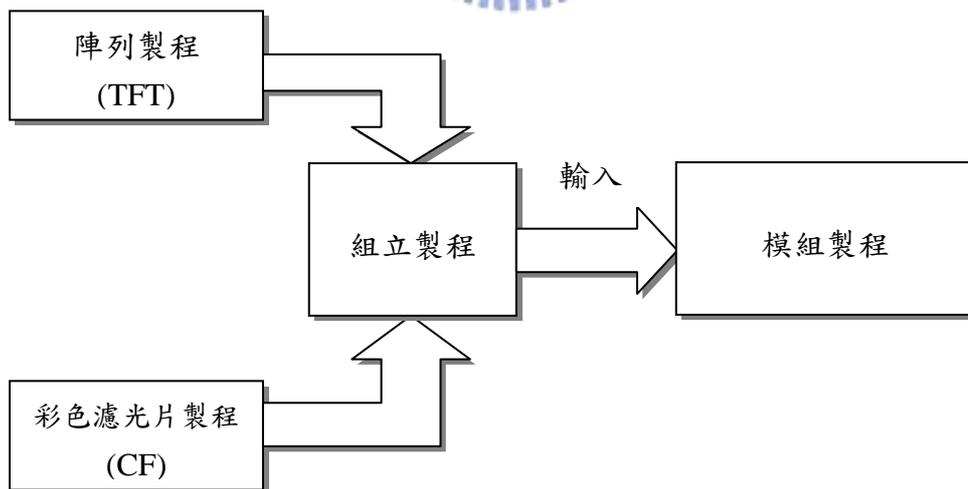


圖 1.1 TFT-LCD 製程流程

在組立製程中，TFT 基板和 CF 基板的配對方法是影響該製程良率的關鍵要素。如圖 1.2 所示，在組立過程中，一個基板(plate)是由許多個面板(panel)所構

成，若要生產一良品的 TFT-LCD 面板(panel)，TFT 面板和 CF 面板必須都是良品；在兩面板中，只要有一為不良品，則壓合後的面板即為不良品。如該圖所示，TFT 基板和 CF 基板的良率原各為 50%，但組合之後良率只有 25%。本研究稱此組合後的良率為配對良率(mapping yield)。

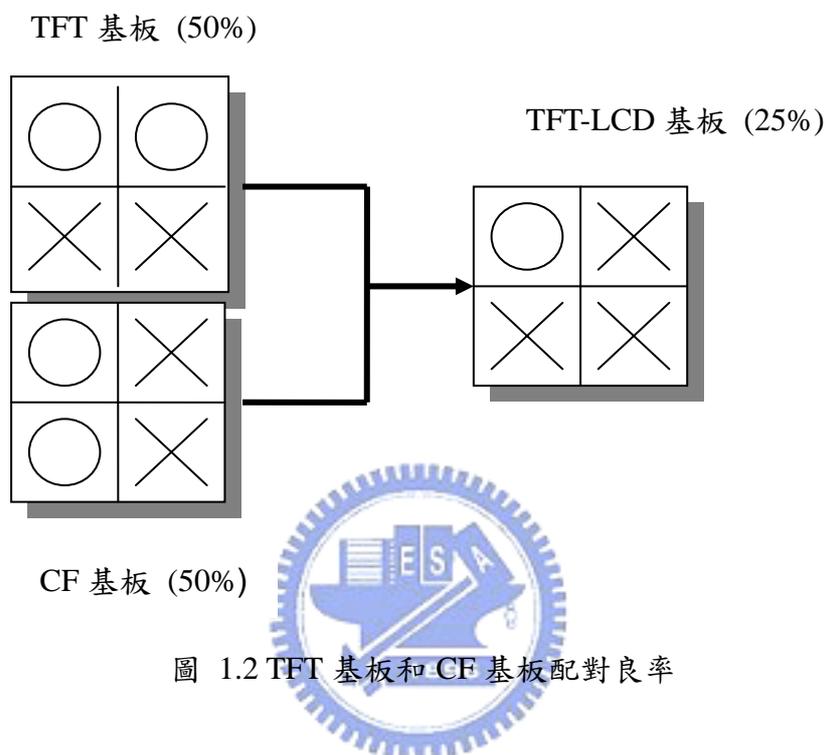


圖 1.2 TFT 基板和 CF 基板配對良率

在實際生產現場，基板是以卡匣(cassette)承載成批運送，一個卡匣可以承載約 10-20 個基板。假設一個卡匣可以承載 n 個基板，當 TFT 和 CF 各有 N 個卡匣時，則基板配對的組合有 $(N \times n)!$ 個方式。如何找尋最佳良率配對的決策，吾人稱為基板配對決策 (plate mapping decision)。楊毅臻[2]以線性規劃模型來描述此決策，並以匈牙利法(Hungarian method)[8]求解。

上述基板配對決策將得到 $(N \times n)$ 組配對。在每一組配對，其 TFT 基板和 CF 基板互稱為「匹配基板」(matched-plate)。為了方便 TFT 和 CF 基板的壓合作業，工廠通常以 CF 卡匣為基準，將 N 個 TFT 卡匣內的基板抽換，使每一個 CF 卡匣都有一個「匹配的」TFT 卡匣(matched-cassette)。亦即該 CF 卡匣內的所有基板，

其匹配的 TFT 基板都在此匹配的 TFT 卡匣內，如此抽換並重新安置基板後的卡匣稱為目標卡匣(target cassette)。如何將 $(N \times n)$ 個 TFT 基板重新抽換，分配在 N 個 TFT 卡匣內的作業，本研究稱之為排序作業(sorting operation)。此排序作業亦可以 TFT 卡匣為基準，重新抽換 CF 卡匣進行，為了便於討論，本研究皆以固定 CF 卡匣為基準討論排序決策。

排序作業是透過排序機(sorter)進行。如圖 1.3 所示，排序機的組成包括一個機器手臂(robot)，數個輸出埠(output port)，數個輸入埠(input port)。原始卡匣(original cassette)經由自動搬運車(automatic guided vehicle, AGV)，放置在輸入埠。輸出埠是放置空卡匣，機器手臂抽取原始卡匣中的 TFT 基板，逐一填入空卡匣中，使空卡匣成為目標卡匣。空卡匣變成目標卡匣之後，先以 AGV 搬離，然後再以 AGV 運上新的空卡匣。

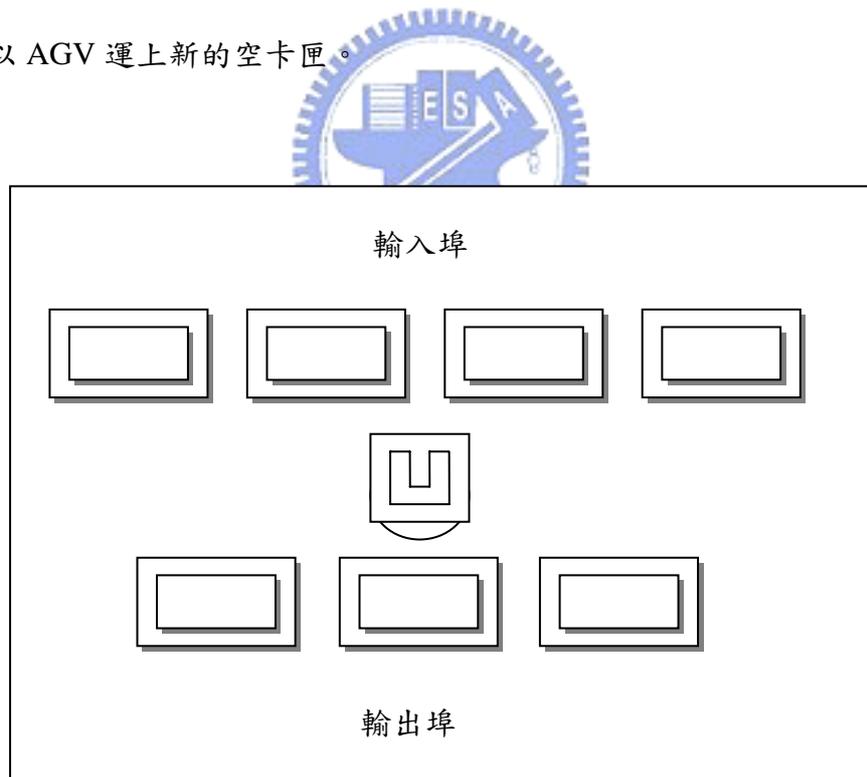


圖 1.3 排序機組成

楊佳翰[3]發展一基因演算法來處理基板配對決策和排序作業決策。該研究

假設排序機的終端埠和來源埠的卡匣數目相同，譬如說兩者都有 3 個卡匣。若生產現場有 12 組 TFT/CF 卡匣需要配對，該研究先將卡匣分成 4 群，亦即每一群有 3 個卡匣。該研究以基因演算法找出 TFT 群與 CF 群最佳的「群配對」(group mapping)方式，然後以線性規劃法找出每一配對群內(mapped group)最佳的基板配對(plate mapping)方法。此方法可節省排序機的作業時間，但是卻會犧牲基板配對的良率，亦即所求得的基板配對良率未必最佳。

楊毅臻[2]先以匈牙利法 (Hungarian method) 求解基板配對決策，再發展數種啟發式規則 (heuristic rule) 來進行卡匣的排序作業，並以模擬驗證來比較各種規則所需的排序作業時間，希望能找出最小化排序總作業時間(total completion time)的排序規則。此研究可確保基板配對良率最佳化，但是 TFT-LCD 的生產是連續性生產(continuous production)，排序作業管制的重點應為及時供應目標卡匣予下游的連續性組立作業，藉以提高生產的效率。



基板配對決策的結果決定基板的配對關係，也同時決定 TFT 卡匣和 CF 卡匣之間的組成關係(composing relationship)，亦即可決定每一 CF 卡匣中，其目標卡匣需要哪些 TFT 卡匣來組成，所需的 TFT 卡匣數目稱為該 CF 卡匣的「組成數」(number of composing members)。圖 1.4 是某基板配對決策後，5 個 TFT 卡匣和 5 個 CF 卡匣的組成關係。該圖表示，CF-1 卡匣需要 TFT-2 和 TFT-5 兩卡匣才能組成，亦即 CF-1 卡匣的組成數為 2，並且其 TFT 匹配基板是存在 TFT-5 和 TFT-2 卡匣內。CF 卡匣的組成數如果越少，代表 TFT 卡匣上下排序機的次數也越少。

張東華[4]發現過去求解基板配對決策的線性規劃模型普遍存在多組解 (multiple solution)。他修正該線性規劃 (linear programming)模型，在基板配對的良率最佳的多組解中，找出 CF 卡匣加總的組成數最小的解，如此不但可以確保基板配對良率最佳，而且可以減少卡匣上下排序機的次數。

	TFT-1	TFT-2	TFT-3	TFT-4	TFT-5
CF-1	0	1	0	0	1
CF-2	0	0	1	0	1
CF-3	1	0	1	1	0
CF-4	1	1	0	0	1
CF-5	0	0	1	0	1

圖 1.4 卡匣組成關係圖

1.2 研究問題

本研究中，為配合下游組立作業的連續性製程，排序機設計成僅有一輸出埠及數個輸入埠。本研究先利用張東華[4]所發展的整數規劃模型來求解基板配對決策，然後發展一基因演算法，來求解 TFT 卡匣上下排序機的順序，目標是希望最大化下游組立作業的機台使用率，以提高生產的效率。



1.3 論文組織

本論文後續章節安排如下，第二章介紹張東華求解基板配對良率的整數規劃模型，第三章介紹排序機上下卡匣的順序問題，並探討求解的基因演算法，第四章是實例驗證，第五章則是結論及未來研究的方向。

第二章 基板配對決策之整數規劃模型

本章介紹張東華所發展之基板配對決策的整數規劃模型[4]。此模型可找出基板配對良率最佳，而且卡匣組成數最少的解。

2.1 模型符號

註標

k : TFT 卡匣 k 之編號

m : CF 卡匣 m 之編號

i : TFT 基板 i 之編號

j : CF 基板 j 之編號

參數

N : 每類卡匣的總數量

n : 每個卡匣的基板數

p : 每個基板的面板數

H : 為一個極大的正數

ω : 為一個極小的正數

a_{kimj} : 第 k 個 TFT 卡匣中的第 i 個基板配對到第 m 個 CF 卡匣中第 j 個基

板的良率，單位%， $1 \leq i \leq n$ ， $1 \leq j \leq n$ ， $1 \leq k \leq N$ ， $1 \leq m \leq N$ 。

變數

$x_{kimj} = 1$ ，若第 k 個 TFT 卡匣中的第 i 個基板被配對到第 m 個 CF 卡匣中

第 j 個基板，否則 $x_{kimj} = 0$ ， $1 \leq i \leq n$ ， $1 \leq j \leq n$ ， $1 \leq k \leq N$ ， $1 \leq m \leq N$ 。

$q_{km} = 1$ ，若 k 個 TFT 卡匣中有基板被配對到第 m 個 CF 卡匣中的基板，

否則 $q_{km} = 0$ ， $1 \leq k \leq N$ ， $1 \leq m \leq N$ 。亦即，若 $\sum_{j=1}^n \sum_{i=1}^n X_{kimj} \geq 1$ ，則 $q_{km} =$



1；若 $\sum_{j=1}^n \sum_{i=1}^n X_{kimj} = 0$ ，則 $q_{km} = 0$ 。

$M = \sum_{m=1}^N \sum_{k=1}^N q_{km}$ ，TFT 原始卡匣的總移動次數。

$Y = \sum_{k=1}^N \sum_{m=1}^N \sum_{i=1}^n \sum_{j=1}^n a_{kimj} x_{kimj} / N \times n$ ，卡匣配對的平均良率。

2.2 整數規劃模式

$$\text{Max } Z = Y - \omega M = \frac{\left(\sum_{k=1}^N \sum_{m=1}^N \sum_{i=1}^n \sum_{j=1}^n a_{kimj} x_{kimj} \right)}{N \times n} - \omega \left(\sum_{k=1}^N \sum_{m=1}^N q_{km} \right) \quad (1)$$

限制式：

$$\sum_{m=1}^N \sum_{j=1}^n x_{kimj} = 1, \quad i = 1, 2, \dots, n, \quad k = 1, 2, \dots, N \quad (2)$$

$$\sum_{k=1}^N \sum_{i=1}^n x_{kimj} = 1, \quad j = 1, 2, \dots, n, \quad m = 1, 2, \dots, N \quad (3)$$

$$H \times q_{km} \geq \sum_{i=1}^n \sum_{j=1}^n x_{kimj}, \quad k = 1, 2, \dots, N, \quad m = 1, 2, \dots, N \quad (4)$$

x_{kimj}, q_{km} : binary variables, $\forall k, \forall i, \forall m, \forall j$

目標函數包含兩個子函數，第一項是希望最大化配對良率 Y ，第二項是希望最小化 CF 卡匣的總組成數，期以減少上下排序機的次數。 w 為一個固定且極小的值，因此目標函數會先滿足第一項，再滿足第二項。公式 (2)、公式 (3) 限制每一 TFT/CF 基板只能對應到一 CF/TFT 基板。

公式 (4) 是用來計算第 k 個 TFT 卡匣基板是否為第 m 的 CF 卡匣的組成會員。在此 W 為一個固定且極大的值，此公式可確保在 $\sum_{i=1}^n \sum_{j=1}^n X_{kimj} > 0$ ， $q_{km} = 1$ ；

若 $\sum_{i=1}^n \sum_{j=1}^n X_{kimj} = 0$ ，則 $q_{km} = 0$ 。

在上述的模式當中，若將目標函數的第二項刪除，則可得楊毅臻所使用之線性規劃模型[2]。

2.3 實例說明

假設 CF 卡匣及 TFT 卡匣個數均為 $N = 10$ 個，每個卡匣當中可以承載 $n = 20$ 片基板，基板的平均良率為 90% 並且服從二項分配，每片基板的面板數 $p = 6$ 。根據張東華的實證研究，若利用楊毅臻[2]的線性規劃求解基板良率問題，可以得到最佳基板總良率的一組解(圖 2.1)。其 CF 卡匣的組成總數為 50。如前所述，基板配對決策有多組解，若採用張東華的整數規劃模式，所求得之解，不僅維持最佳的配對良率，而且其 CF 卡匣的組成總數降為 33 (圖 2.2)。



	TFT01	TFT02	TFT03	TFT04	TFT05	TFT06	TFT07	TFT08	TFT09	TFT10
CF01	1	0	0	0	1	1	0	0	1	0
CF02	1	0	1	0	0	1	1	1	0	0
CF03	0	0	0	1	1	1	1	1	1	1
CF04	0	1	0	0	1	1	0	0	1	0
CF05	0	0	1	0	0	0	1	0	1	0
CF06	1	1	1	1	1	0	1	0	0	1
CF07	1	1	0	0	0	1	0	1	0	1
CF08	0	1	1	0	0	0	1	0	0	0
CF09	0	1	0	1	1	1	1	1	1	0
CF10	0	1	1	0	1	0	0	0	1	1

表 2.1 單目標函數的基板配對結果

	TFT01	TFT02	TFT03	TFT04	TFT05	TFT06	TFT07	TFT08	TFT09	TFT10
CF01	0	0	0	0	1	0	0	0	1	0
CF02	1	0	1	0	0	1	1	0	0	0
CF03	0	0	0	0	1	1	1	0	1	0
CF04	0	0	0	0	1	1	0	0	0	0
CF05	0	0	1	0	0	0	1	0	1	0
CF06	0	1	1	1	0	0	1	0	0	1
CF07	1	1	0	0	0	0	0	1	0	1
CF08	0	1	1	0	0	0	1	0	0	0
CF09	0	1	0	1	0	0	1	1	0	0
CF10	0	0	0	0	0	0	0	0	1	1

表 2.2 雙目標函數的基板配對結果

第三章 研究方法

本章描述所要解決的問題，解構問題為兩個子題，最後探討該子題的求解方法。

3.1 問題分析與解構

本研究假設基板配對決策是採用張東華提出的方法[4]求得，在此前提下探討如何決定 TFT 卡匣上下排序機的順序，期使下游組立作業機台利用率的最大化。

為最大化組立製程連續性作業的機台利用率，本研究採用「先選先出」(first-select-first-out)的產出策略。亦即若吾人選定要產出某一 TFT 目標卡匣，排序機一定得先產出該卡匣；在產出該卡匣產出之前，不能改變生產目標。

本研究將此問題分成兩個子問題，第一個子問題是決定 TFT 目標卡匣的產出順序。第二個子問題是針對一待產出的 TFT 目標卡匣，決定其 TFT 輸入卡匣上下排序機的順序。如前所述，一個 TFT 目標卡匣是裝載某一 CF 卡匣所擬配對的基板，因此吾人可以此 CF 卡匣來代表該 TFT 目標卡匣。

此兩子問題可以進一步說明如下，第一個子問題是要將 CF 卡匣排序，假設有 N 個卡匣，則總共有 $N!$ 個排列順序可以選擇。第二個子問題是在某一 CF 卡匣順序中，決定 TFT 卡匣上下排序機的順序。假設每一個 CF 卡匣都有 k 個組成成員(composing member)，則有 $k!$ 個順序。因此決定 TFT 卡匣上下排序機順序的問題，其解空間的複雜度為 $(N \times k)!$ 。

本研究假設排序機系統有 1 支機器手臂，有 P 個埠可以放置卡匣，其中 1 個屬於輸出埠，用以放置空卡匣，目的是產出目標卡匣。其餘 $P-1$ 個屬於輸入埠，是用來放置原始卡匣。機器手臂將基板從將原始卡匣搬運至空卡匣，以產生目標卡匣。

排序機作業時，要先選定一 CF 卡匣，目的是經過排序機的作業後，產生該 CF 卡匣的 TFT 目標卡匣，亦即將該 CF 卡匣所對應的 TFT 基板搬運至輸出埠上的空卡匣，以產生 TFT 目標卡匣。

如圖 1.4 所示，基板配對決策的結果會產生 CF 卡匣和 TFT 卡匣的對應關係。設若 CF 卡匣和 TFT 卡匣數目皆為 N 個，吾人可以一矩陣 $V = [v_{i,j}]$ $1 \leq i \leq N, 1 \leq j \leq N$ ，來表達 CF 卡匣和 TFT 卡匣的對應關係；若第 j 個 TFT 卡匣是第 i 個 CF 卡匣的組成成員，則 $v_{i,j} = 1$ ，否則 $v_{i,j} = 0$ 。若排序機的產出目標為 CF 卡匣 i 的配對卡匣，其組成成員 TFT 卡匣所構成的集合可表達成 $S_i = \{j \mid v_{i,j} = 1\}$ ；假設目前排序機上來源埠(source port)上現存的 TFT 卡匣為 $R_i = \{r_1, r_2, \dots, r_{p-1}\}$ ， $r_k = j$ 代表第 k 個來源埠目前存放第 j 個 TFT 卡匣，若 $r_k = 0$ 代表第 k 個來源埠目前沒有存放任何 TFT 卡匣。 $N(U)$ 代表任一集合 U 的元素個數。

如上所述，本研究問題可分成兩個子題，以下首先討論第二子題的解法，亦即在給定一 CF 卡匣 i ，排序機來源埠上現存的卡匣集合 R_i ，如何求解 S_i 集合內卡匣的上下排序機順序，以及 R_i 集合內卡匣下排序機的順序，並且計算完成該卡匣的時點(t_i)。其次討論如何以基因演算法求解第一子題，亦即找出最佳的目標卡匣產出順序。

3.2 第二子題的解法

S_i 和 R_i 集合內TFT卡匣上下排序機的順序依下列演算法Loading_Sequence決定，其中 i 是輸入變數，代表目標卡匣 i ， S_i 、 R_i 、 V 的定義如上所述。 t_i 是輸出變數，代表所欲產出目標卡匣的產出時點。

Procedure Loading_Sequence ($i, S_i, R_i, V, t_{i-1}, t_i$)

Step 0：設定變數的初始值 loading = 0, un_loading = 0

簡化變數表達法 $S = S_i, R = R_i$

Step 1: 將 R 集合的 TFT 卡匣分成兩群

$$R_A = \{j \mid j \in R, v_{i+1,j} = 1\}, R_B = \{j \mid j \in R, v_{i+1,j} = 0\}$$

$$S_A = \{j \mid j \in S, v_{i+1,j} = 1\}, S_B = \{j \mid j \in S, v_{i+1,j} = 0\}$$

Step 2：計算下列集合

$$D_1 = S_B \cap R, D_2 = S_B - D_1, C_1 = S_A \cap R, C_2 = S_A - C_1$$

Step 3：找出不必裝卸的會員卡匣

- IF $N(D_1) \geq 1$ ，將 D_1 集合內的TFT卡匣相關的基板放置於空卡匣 IF $N(C_1) \geq 1$ ，將 C_1 集合內的TFT卡匣相關的基板放置於空卡匣

Step 4：卸下 R 集合卡匣，裝上 D_2 集合的卡匣

- While $N(D_2) \geq 1$, Do
 - IF $N(R_A) + N(R_B) < p$,
 - Load a cassette $h, R_B = R_B + \{h\}, \text{loading} = \text{loading} + 1$
 - Elseif $N(R_B) \geq 1$,
 - Unload a cassette h in $R_B, R_B = R_B - \{h\}, \text{unloading} = \text{unloading} + 1$
 - Load a cassette k in $D_2, D_2 = D_2 - \{k\}, R_B = R_B + \{K\}, \text{loading} = \text{loading} + 1$
 - Elseif $N(R_A) \geq 1$,
 - Unload a cassette h in $R_A, R_A = R_A - \{h\}, \text{unloading} = \text{unloading} + 1$

Load a cassette k in D_2 , $D_2 = D_2 - \{k\}$, $R_B = R_B + \{K\}$, loading = loading + 1

Endif

End_while

Step 5: 卸下 R 集合卡匣，裝上 C_2 集合的卡匣

● While $N(C_2) \geq 1$, Do

IF $N(R_A) + N(R_B) < p$,

Load a cassette h , $R_A = R_A + \{h\}$, loading = loading + 1

Elseif $N(R_B) \geq 1$,

Unload a cassette h in R_B , $R_B = R_B - \{h\}$, unloading = unloading + 1

Load a cassette k in C_2 , $C_2 = C_2 - \{k\}$, $R_A = R_A + \{k\}$, loading = loading + 1

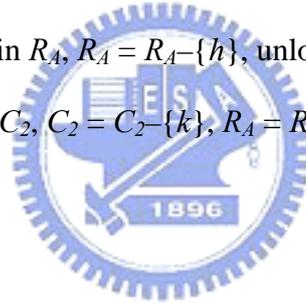
Elseif $N(R_A) \geq 1$,

Unload a cassette h in R_A , $R_A = R_A - \{h\}$, unloading = unloading + 1

Load a cassette k in C_2 , $C_2 = C_2 - \{k\}$, $R_A = R_A + \{K\}$, loading = loading + 1

Endif

End_while



Step 6: 輸出 $t_i = t_{i-1} + \lambda_1 \cdot \text{loading} + \lambda_2 \cdot \text{unloading}$ ，其中 λ_1 、 λ_2 是 loading/unloading 的時間。

3.3 第一子題的解法

如前所述，本研究排序機的產出策略是採用「先選先出」的策略，亦即選定一 CF 卡匣後，排序機就要將其匹配的 TFT 目標卡匣產出。假設有 N 個 CF 卡匣，則其目標卡匣產出的順序有 $N!$ 種可能。當 N 值變大時，解空間會迅速擴張，因此本研究發展一基因演算法[6]來求解目標卡匣產出順序的問題。

3.3.1 符號說明

N_p : 每個世代母體染色體(chromosome)數

P_c : 交配率

$CR = N_p \times P_c$: 交配染色體數

P_m : 突變率

$MU = N_p \times P_m$: 突變染色體數

End_Count : 最終搜尋的世代數

Epoch_length : 最佳解沒有改善則中止的世代數

3.3.2 染色體編碼方式及初始母體的產生

在此基因演算法，一組解代表一個目標卡匣輸出順序。本研究以一具有 N 個元素的橫向量 $X = [x_i]$, $1 \leq i \leq N$ 來代表一組解。此向量又稱為染色體，其元素稱為基因。基因值代表CF卡匣的編號。如圖 3.1 所示，此染色體表示目標卡匣的輸出順序。起始母體 P 是一個存放染色體的集合，其產生方式是隨機產生 N_p 條不同的CF卡匣完成順序。



6	8	5	3	2	9	1	10	7	4
---	---	---	---	---	---	---	----	---	---

圖 3.1 染色體表示法

3.3.3 適應函數(fitness function)

在基因演算法中，適應函數是用來評估一染色體的優劣或適合程度，據以決定該染色體的存活機會。本基因演算法的適應函數表達如下：

$$F(X) = Utilization(X)$$

上式中， $Utilization(X)$ 是一個函數，其目的是在給定一目標卡匣的產出順序(X)下，計算出下游組立作業的機台利用率，其執行步驟描述如下。

Function $Utilization(X)$

For $k = 1, N$

$i = x_k$

Call Procedure $Loading_Sequence(i, S_i, R_i, V, t_{i-1}, t_i)$

$t_i =$ 完成目標卡匣 i 的時點

Endfor

$$Utilization(X) \quad \rho = 1 - \frac{1}{N \cdot T} \sum_{i=1}^N [(t_i - t_{i-1}) - T]^+$$

3.3.4 子代產生方式

基因演算法是利用複製(reproduction)、交配(crossover)和突變(mutation)三種運算子(operator)來產生子代，其中交配和突變是用來產生新的染色體的方法。本研究的複製步驟是採取競爭式選擇(tournament selection) [5]，其基本概念是將母代的 N_p 條染色體兩兩配對形成 $\frac{N_p}{2}$ 對染色體，每對染色體適應函數值較佳者進入交配池中，複製的動作就是將這 $\frac{N_p}{2}$ 條適應函數值較佳者重製一次，再度回到 N_p 條染色體。

交配是隨機選取交配池中的兩條母代染色體，彼此交換基因，進而組成兩個新的染色體，藉由累積前代優秀的基因，期能產生更優秀的子代，本研究使用的方法[9]敘述如下：

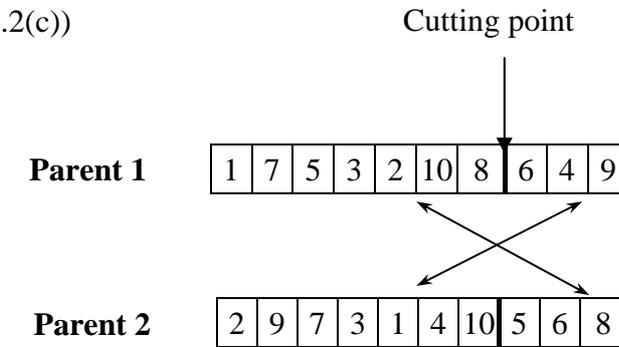
步驟一：對染色體隨機產生一交配點，將兩條挑選出來要進行交配的染色體依交配點一分為二，彼此交換一段基因。(參閱圖 3.2(a))

步驟二：檢視交換過後得到新的兩條染色體，是否有基因值缺少及重複的部份。

(參閱圖 3.2(b))

步驟三：由前往後將重複的基因值去掉，並將缺少的基因值依序號順序補上。

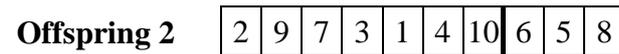
(參閱圖 3.2(c))



(a)



(b)



(c)

圖 3.2 染色體表示法

突變運算是從母代 N_p 條染色體中，挑選出 $MU = N_p \times P_m$ 條，針對每一條染色體，隨機挑選兩個基因，交換這兩個基因的基因值，如圖 3.3 所示：

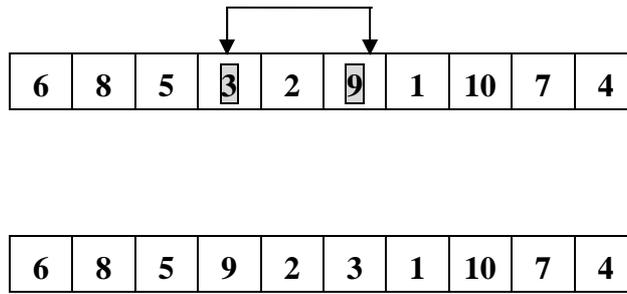


圖 3.3 突變示意圖

3.3.5 終止條件

本基因演算法有兩個終止條件，任滿足其一即終止運算，並以母體中的最佳解為最終解。

- 搜尋的世代數到達 end_count 即中止。
- 每代最好的解持續了 Epoch_length 代而沒有進步，則中止。



第四章 案例驗證

本章以 90 個工廠實際的案例來說明本研究所發展的方法及成果。此驗證所使用的電腦硬體是 AMD Athlon (TM) XP 3000+ 2.17GHz 1GB DDRAM，並以 C++ 撰寫演算法的程式。

4.1 輸入資料與參數設定

本驗證採用張東華[4]基板配對決策的求解結果，TFT 及 CF 卡匣總數以 N 表示，每個卡匣的基板個數是 n ，每個基板的面板個數以 p 表示。CF 基板的平均良率是 90%，TFT 基板的平均良率是 85%，兩者都是二項式分配(binomial distribution)。



排序機包含一個機器手臂、 P 個埠，其中一個是輸出埠，其他均為輸入埠，本案例討論五個情境分別為 $P=2、3、4、5、6$ 。排序機下游組立作業每完成一組卡匣的時間為 T 分鐘，TFT 卡匣裝上或卸下排序機的時間定義為相對於下游組立作業的時間，在此考慮每卸下或裝上一個卡匣須耗時 $0.02T$ 及 $0.05T$ 兩種情形，對於所有情境，皆做 15 次重複實驗取其平均值。本模擬比較的基準有二：業界所使用的隨機完成順序及考慮最少卡匣裝卸次數的完成順序。

本案例將基因演算法的參數設定如下：每個世代母體染色體數 $N_p = 100$ 、交配率 $P_c = 0.6$ 、突變率 $P_m = 0.06$ 、強制終止世代 $\text{End_Count} = 100000$ 、最佳解持續未改變則中止的世代數 $\text{Epoch_length} = 500$ 。

4.2 結果與分析

此案例的驗證結果如表 4.1、表 4.2 所示。表 4.1 是單位裝卸卡匣時間為 $0.02T$

的模擬結果，表 4.2 是單位裝卸卡匣時間為 $0.05T$ 的模擬結果。此二組參數的設計乃根據實際生產的情形而訂定，不同規模及代數(generation)的 LCD 廠，往往有不同組立時間對裝卸卡匣時間的比，但範圍大多介於 $0.02T-0.05T$ 。

表 4.1 單位裝卸卡匣時間為 $0.02T$ 的模擬結果

情境	埠數	隨機解	總完工時間最小的解	本研究所得的解
N=10, n=20, p=6	2	93.8%	90.3%	98.3%
	3	95.0%	97.8%	98.2%
	4	93.6%	92.5%	98.0%
	5	94.6%	97.1%	97.8%
	6	94.1%	92.4%	97.7%
	N=10, n=20, p=12	2	93.0%	92.9%
3		93.7%	92.9%	97.3%
4		93.4%	94.6%	97.1%
5		93.1%	92.9%	96.9%
6		93.6%	91.2%	96.7%
N=10, n=20, p=30		2	91.5%	96.5%
	3	90.7%	96.5%	94.8%
	4	91.3%	91.2%	95.0%
	5	91.1%	93.0%	95.4%
	6	91.3%	93.2%	94.8%
	N=15, n=20, p=6	2	94.9%	99.4%
3		94.4%	99.3%	99.4%
4		94.0%	99.2%	99.4%
5		94.9%	99.1%	99.3%
6		94.8%	90.1%	99.3%
N=15, n=20, p=12		2	93.6%	93.7%
	3	93.8%	93.2%	96.3%
	4	93.3%	94.3%	96.6%
	5	93.7%	96.0%	97.0%
	6	94.1%	95.5%	96.8%
	N=15, n=20, p=30	2	93.5%	93.6%
3		92.9%	93.3%	95.2%
4		93.0%	91.8%	95.3%
5		93.0%	94.6%	95.3%

	6	92.8%	92.9%	96.0%
N=20, n=20, p=6	2	95.5%	99.1%	99.2%
	3	95.2%	97.4%	99.2%
	4	95.5%	96.4%	99.2%
	5	96.6%	95.4%	99.1%
	6	96.4%	95.8%	99.1%
N=20, n=20, p=12	2	95.7%	94.7%	100.0%
	3	95.7%	97.9%	100.0%
	4	95.6%	92.7%	100.0%
	5	96.0%	97.7%	100.0%
	6	96.2%	94.9%	100.0%
N=20, n=20, p=30	2	94.3%	93.0%	100.0%
	3	94.6%	94.9%	100.0%
	4	97.0%	95.3%	100.0%
	5	95.8%	94.5%	100.0%
	6	93.9%	93.7%	100.0%

表 4.2 單位裝卸卡匣時間為 $0.05T$ 的模擬結果

情境	埠數	隨機解	總完工時間最小的解	本研究所得的解
N=10, n=20, p=6	2	89.9%	86.3%	97.7%
	3	91.2%	96.7%	97.5%
	4	90.0%	88.0%	97.1%
	5	89.7%	95.0%	96.8%
	6	88.7%	86.1%	96.6%
N=10, n=20, p=12	2	90.9%	90.4%	95.6%
	3	91.2%	90.4%	96.4%
	4	89.7%	92.6%	95.9%
	5	89.5%	90.2%	95.4%
	6	90.1%	88.0%	95.0%
N=10, n=20, p=30	2	88.5%	95.2%	93.3%
	3	89.1%	95.2%	93.3%
	4	88.6%	88.0%	93.4%
	5	86.3%	90.6%	93.6%
	6	85.6%	90.8%	92.4%
N=15, n=20, p=6	2	93.4%	99.3%	99.3%
	3	92.6%	99.1%	99.3%

	4	93.6%	98.9%	99.2%
	5	94.0%	98.8%	99.2%
	6	92.6%	85.9%	99.1%
N=15, n=20, p=12	2	92.6%	92.5%	95.5%
	3	92.5%	91.7%	95.6%
	4	92.7%	92.8%	95.9%
	5	93.1%	94.7%	96.4%
	6	92.9%	93.8%	96.1%
N=15, n=20, p=30	2	93.1%	92.7%	94.6%
	3	92.5%	92.3%	94.4%
	4	91.6%	90.2%	94.4%
	5	90.9%	93.4%	94.5%
	6	91.4%	91.0%	94.7%
N=20, n=20, p=6	2	94.9%	98.9%	99.1%
	3	95.1%	96.9%	99.1%
	4	95.7%	95.6%	99.0%
	5	95.5%	94.3%	99.0%
	6	96.2%	94.7%	99.0%
N=20, n=20, p=12	2	94.8%	94.0%	100.0%
	3	94.6%	97.6%	100.0%
	4	95.1%	91.6%	100.0%
	5	94.7%	97.3%	100.0%
	6	94.7%	94.0%	100.0%
N=20, n=20, p=30	2	92.8%	91.2%	98.8%
	3	93.1%	93.1%	99.3%
	4	93.1%	94.8%	100.0%
	5	93.3%	93.8%	100.0%
	6	93.5%	92.9%	100.0%

由模擬結果可以得知，本研究所提出的基因演算法，可以提升下游機台利用率高達 7%，對於資本密集，良率相當重要的面板產業來說，其相當於總利潤的 7%，又基因演算法的演算十分有效率，每一個情境的計算時間皆不到一分鐘。

另外，有兩點現象值得注意：

- (1) 若單位卡匣裝卸時間佔單位組立作業時間的比例越高，則下游組立作業機台利用率就會越低，在此情況下，本演算法所提升的機台利用率也越顯著。
- (2) 卡匣裝卸次數少在整個製程當中未必對組立作業機台利用率有所幫助，以實例驗證中卡匣裝卸次數最少的情形和隨機完成的情形比較，在機台利用率的表現上有大有小，沒有顯著的現象，此亦說明了排序作業的基效指標應以下游機台利用率為標竿，而非卡匣裝卸的次數。



第五章 結論與未來研究方向

5.1 結論

本研究探討 TFT-LCD 基板配對製程的卡匣排序作業問題，將此問題解構成兩個子題，第一子題是規劃目標卡匣的產出順序，第二子題在既定產出某目標卡匣下，規劃其組成卡匣的上機下機順序；目的是希望配合下游生產速率的需求，期使組立作業的產能利用率最大化，由於 TFT-LCD 屬於高度資本密集的產業，所以保持製程在高產能利用率是相當重要的課題，本研究所提出的方法，和業界現行使用的方法比較，可以提升高達 7% 的產能利用率。

本論文所提出的基因演算法以染色體來代表目標卡匣的產出順序，以適應函數來評估該解的優劣。亦即，第二子題是用來定義基因演算法的適應函數，第一子題則整合此適應函數於一基因演算法，來求解最佳的目標卡匣產出順序。此演算法是以「先選先產出」來規劃排序機的生產策略，亦即選定一目標卡匣為產出目標後，一定要產出，才考慮產出其他目標卡匣。

5.2 未來研究方向

本論文未來可以繼續研究的方向有下列幾點：

1. 考慮以動態規劃的方式來求解第二子題，修正啟發式演算法的不足。
2. 考慮建構一流量均衡的生產系統，整合上游下游的製程，來規劃排序機組態、AGV 數量及倉儲空間等問題。
3. 考慮以其他方式修正 GA 搜尋最佳解的效率。

參考文獻

- [1] 李秀玉 (1999) 應用賽局理論分析我國薄膜電晶體液晶顯示器產業之競爭策略，交通大學科技管理研究所碩士論文
- [2] 楊毅臻 (2003) 以基板配對為基礎的 LCD 廠排序系統規劃之研究，華梵大學工業管理研究所碩士論文
- [3] 楊佳翰 (2004) 應用改良型巨集式啟發式方法於 TFT-LCD 良率控制最佳化之研究，交通大學工業工程與管理研究所碩士論文
- [4] 張東華 (2005) TFT-LCD 組立製程配對作業之研究，交通大學工業工程與管理研究所碩士論文
- [5] Goldberg, D. E., Korb, B., and Deb, K., Messy genetic algorithms: motivation, analysis, and first results, *Complex Systems*, 1989, 3(4), 493-530.
- [6] Holland, J. H., *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, 1989 (University of Michigan Press: Michigan)
- [7] Jeong, B., Kim, S. W., and Lee, Y. J., An assembly scheduler for TFT LCD manufacturing, *Computers & Industrial Engineering*, 2004, 41(1), 37-58.
- [8] Kuhn, H. W., The Hungarian method for the assignment game, *Naval Research Logistics Quarterly*, 1955, 2, 83-97.
- [9] Reeves C. R., A genetic algorithm for flowshop sequencing, *Computers Operations Research*, 1995, 22(1), 5-13.

- [10] Shin, H. J., and Leon, V. J., Scheduling with product family set-up time: an application in TFT LCD manufacturing, *International Journal of Production Research*, 2004, 42(20), 4235-4248.
- [11] Toshihisa, T., TFT/LCD: liquid-crystal displays addressed by thin-film transistors, 1996 (Gordon and Breach Publishers: Australia).
- [12] Toba H., A tight flow control for job-shop fabrication lines with finite buffer, *IEEE Transactions on Automation Science and Engineering*, 2005, 2(1), 78-83.

