國立交通大學

工業工程與管理學系

博 士 論 文

液晶顯示器良率配對方法之研究

Effective Approaches for Liquid Crystal Displays Yield Mapping

研 究 生：王鵬森

指 導 教 授：蘇朝墩　博士

中 華 民 國 九十三 年 七 月

液晶顯示器良率配對方法之研究

Effective Approaches for Liquid Crystal Displays Yield Mapping

研 究 生：王 鵬 森　　Student: Peng-Sen Wang

指導教授：蘇 朝 墩 博士　　Advisor: Dr. Chao-Ton Su

國 立 交 通 大 學

工業工程與管理學系

博 士 論 文

A Dissertation

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

Doctor of Philosophy

in Industrial Engineering and Management

July 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年七月

液晶顯示器良率配對方法之研究

研究生：王鵬森　　　　　　　　　　指導教授：蘇朝墩 博士

國立交通大學
工業工程與管理學系

摘　要

液晶顯示器（Liquid crystal display, LCD）屬於經濟部積極推動的兩兆雙星產業之一，而良率是LCD產業的重要競爭決定因素。TFT-LCD面板的製造過程，主要有三個階段：Array製程、Cell製程、Module製程；其中在Cell製程的良率最低，是業界提高良率的關鍵所在，其過程為將上下玻璃（陣列面板與彩色濾光片）貼合，並於其內灌入液晶。良率配對是選擇一片TFT（Thin film transistor）Array玻璃與另一片彩色濾光片（Color filter, CF）玻璃貼合以形成LCD面板，每一片玻璃基板內含有許多數目的面板（Panel），有些面板可能是不良的，只有當上下玻璃基板內相對應的面板都是好的，配對後才能生產出合格的LCD面板。本研究提出利用線性規劃的方法來最佳化TFT與CF的良率配對，以產生最多數目的LCD面板，提高Cell製程的良率。

Sorter是一個自動化的機器，通常有幾個ports，能夠轉移CF cassettes內的CF面板去和TFT面板配對，以增加TFT與CF的配對良率，本研究使用線性規劃的方法進行比較Sorter上的port數目對於配對良率之影響，這個方法是一個最佳解並提供LCD廠商重要的良率資訊。其次，本研究提出一個演算法處理大量cassettes的良率配對問題，這演算法提供一個極佳的解並避免過長的運算時間。實務上顯示所提出的方法，能夠被有效的運用與解決實際的問題。

關鍵詞：液晶顯示器、良率配對、匈牙利法、組合最佳化

Effective Approaches for Liquid Crystal Displays Yield Mapping

Student: Peng-Sen Wang Advisor: Dr. Chao-Ton Su

Department of Industrial Engineering and Management
National Chiao Tung University

Abstract

The ability to improve yield in the manufacturing process is a very important competitiveness determinant for LCD manufacturing firms. LCD contains three major manufacturing sectors: the array, cell and module assembly processes. The yield loss from the cell process is one of the most critical steps. The mapping operation matches one thin-film-transistor (TFT) and one color-filter (CF) glass-plate together to form one LCD plate. Each plate contains a certain number of cells. The matched LCD cell is good only when the TFT and CF cell match is good. When only a TFT or CF cell is good, there is a yield loss. We propose a linear programming (LP) formulation to maximize the yield rate through an optimal matching process to improve the cell process yield.

The sorter is a robot used in LCD manufacturing systems to achieve higher yield for matching TFT and CF plates. This sorter contains several ports that can transfer CF glasses from CF cassettes to match TFT glasses. This research proposes a linear programming formulation to compare the performance of the various ports in the post-mapping yield. This method provides an optimal solution and offers LCD manufacturers important yield information. Next, we propose an algorithm to reduce the number of ways for choosing different matched objects when the number of matched cassettes is large. This algorithm avoids computer over load and provides an excellent solution. The empirical results illustrate the efficiency and effectiveness of the proposed approaches.

Key Words

LCD, Matching, Yield Mapping, Hungarian Method, Combinatorial Optimization.

誌謝

能夠順利獲得博士學位，首先要感謝恩師蘇朝墩教授不辭辛勞對於研究論文的督促與悉心指導，從蘇老師的身上自己不僅學會如何獨立研究的方法，更學習到對於學術研究持續不斷的堅持與突破。於論文審查期間，承陳穆臻教授、沙永傑教授、駱景堯教授與魏秋建教授不吝指正，惠予寶貴意見，使本論文更臻完備，於此致上由衷感謝。

感謝群龍企管劉逸群總經理與專案總經林采羚小姐、統寶光電李光洲副理與林永淙工程師提供論文個案，感謝成功大學楊大和教授在研究期間對於本論文的支持與協助，也感謝本系梁馨科教授與三益制動科技吳証富協理務實的協助。

感謝同窗好友何境峰、謝松益與學弟許志華在求學期間的幫助與鼓勵，也感謝學弟許俊欽在論文上的協助，深厚情誼永記在心。

最後，感謝我的父母及妻子桂美這些年來的支持與諒解並照顧兩個可愛的小孩，使我能專心向學。在此謹以本論文獻給我最親愛的父母、妻子與家人。

王 鵬 森

2004 夏，于新竹交通大學

CONTENTS

1. Introduction

1.1 Research motivation

The market for liquid crystal displays (LCDs) is growing rapidly and impacting new fields. LCD applications include: personal digital assistants (PDA), cellular phones, digital cameras, computers, notebook computers, flat panel TVs, and various computer game units. During the past twelve years the market for LCDs has grown at over 20% on average per annum. The price for LCD products is significantly reduced due to both the technology maturity and ample manufacturing capacity. The downward pricing trend further promotes LCD applications.

LCDs can be divided into three major products: TN (twisted nematic), STN (super twisted nematic), and TFT (thin film transistor). The most widely used LCD for high information content displays is the TFT-LCD. In the TFT-LCD each picture pixel is controlled using a thin film transistor. In the 1980s, market demand forced a transition from twisted nematic displays to super twisted nematic displays. This higher-performance display is expected to grow rapidly and have a major market share in the display market. This led to today's amorphous silicon and low temperature poly silicon (LTPS) thin film transistor liquid crystal display. LTPS technology has gathered much attention from many display manufacturers because it has several advantages over amorphous displays, as shown below [1]:

1. The feasibility of integrating the peripheral drive IC circuit onto the substrate.
2. Faster TFT response time, smaller dimensions, fewer contacts and components.
3. Simplified system design.
4. Increasing panel reliability.
5. High Aperture Ratio and Resolution.

LTPS production technology aimed at manufacturing small and medium sized LCD panels for rapid growth in the digital still cameras (DSC), digital video camcorders (DVC), car automotive global positioning system (GPS), cellular phones and personal digital assistants (PDA) markets. In the future, this production technology will be applied not only to large size displays but also to large glass substrates.

The manufacturing technology, capital investment and industrial infrastructure are key factors affecting LCD industry competition [2]. The ability to improve yield in the manufacturing process is an important competitiveness determinant for LCD factories due to the significant yield loss ranging from 5 to 25%. This loss is attributed to three major manufacturing sectors: the array, cell and module assembly processes. The yield loss from the cell process is one of the most critical steps. To increase cell process yield, more conforming LCD panels must be produced from one glass substrate.

1.2 Research objectives

There are two options for cell process yield improvement. The first is to improve the TFT and/or CF plate yield. This approach requires improvement in the manufacturing processes, technology, tooling, etc., and may be costly and have technological constraints. For example, Kim and Choi [3] developed a megasonic cleaner to remove very small particles from the LCD panels to improve the manufacturing yield rate. The second option is to use a judicious mapping policy to optimize yield mapping. The matching technique plays an important role in TFT and CF yield mapping. We propose a linear programming (LP) formulation to solve the problems. This approach could be very efficient and does not alter the cell process or add equipment. We will:

1. Using LP formulation to minimize the yield loss or maximize the yield rate through an optimal matching process to obtain a greater number of acceptable LCD panels to improve the cell process yield.
2. Using the proposed reduction algorithm to avoid computer over load when the number of matched cassettes is large.
3. The sorter is a robot used in LCD manufacturing systems to achieve higher yield for matching TFT and CF plates. We use LP formulation to compare the performance of the number of sorter ports in the post-mapping yield. This method provides an optimal solution and offers LCD manufacturers important yield information.

1.3 Organization

The rest of this dissertation will summarize the TFT-LCD production process and linear programming formulation in Chapter 2. Chapter 3 of this dissertation will introduce an assignment model to provide insight into yield mapping in TFT-LCD fabrication, and then apply those models to TFT-LCD's using the Hungarian method. A case study adapted from LCD manufacturing to illustrate the effectiveness of the proposed optimal solution approach. Chapter 4 of this dissertation will discuss above case study from LCD manufacturing firm by using a sorter. We propose a linear programming formulation to compare various ports in the post-mapping yield control problem. In addition, we propose a reduction algorithm to reduce the number of ways for choosing different matched objects when the number of matched cassettes is large. Chapter 5 of this dissertation will compare the greedy algorithm, genetic algorithm (GA), and proposed reduction algorithms for the computation results. We also compare the performance of different algorithms for four defects types of LCD plates. Finally, Chapter 6 will summarize the key conclusions of the proposed approaches.

2. Background Information

2.1 The a-Si TFT-LCD manufacturing process

The manufacturing process for LCD may be likened to making a sandwich [4, 5]. The bottom substrate is the TFT array. The top substrate is the color filter plate. Color filter (CF) glasses are usually purchased from outside vendors. There are three main production sequences for TFT-LCD. They are shown schematically in Figure 2.1 [4, 6, 7]. For a concise presentation, readers are referred to O'Mara [4] for a detailed discussion of the manufacturing process.

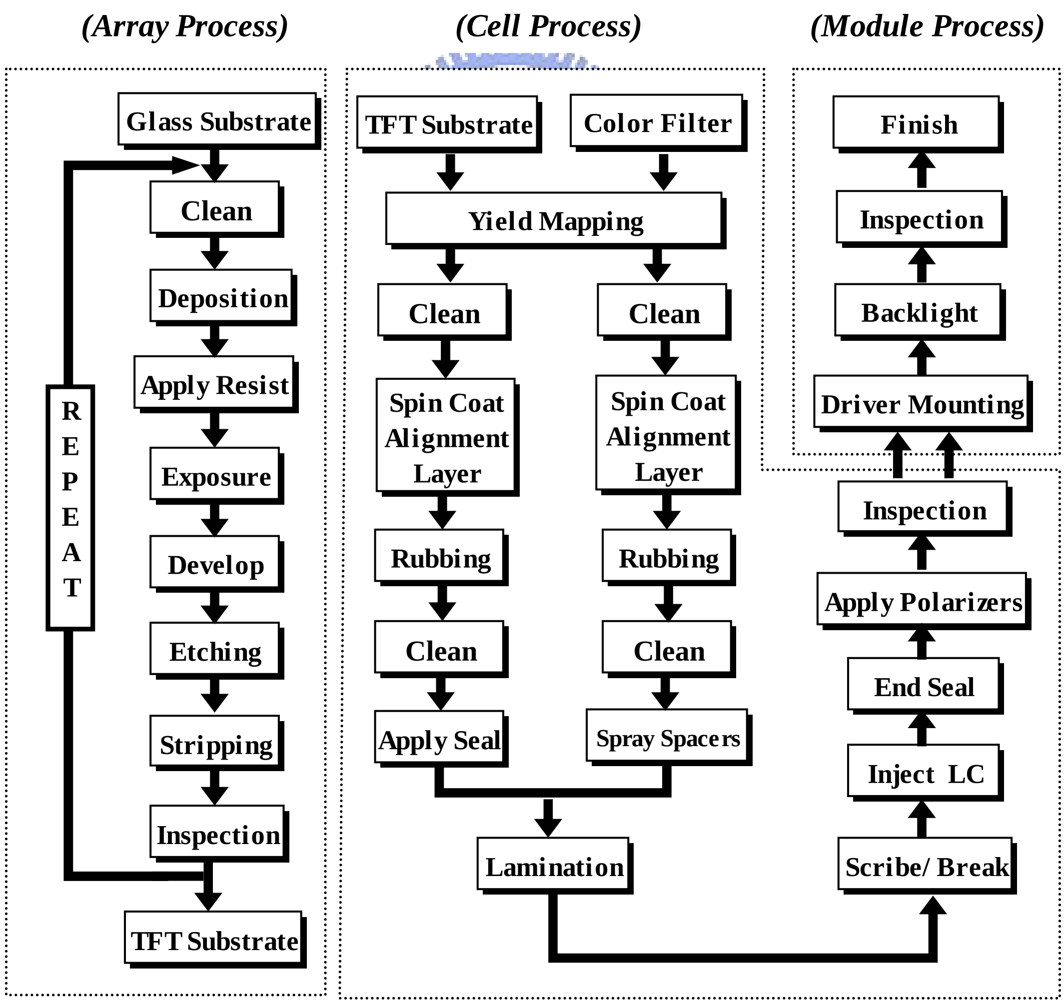

Figure 2.1 TFT-LCD manufacturing process

1. TFT array process: Thin film transistor fabrication

The TFT fabrication process sequence is a series of deposition and etching sequences, as with integrated circuit fabrication. Deposition processes are used first to form the thin film transistors onto the glass substrate. A photoresist is then applied and imaged to allow thin film etching to the appropriate dimensions.

There are various sizes in the glass substrates, depending on the generation of manufacturing equipment. Substrate size can have critical impact on display size. For example, for 550x650mm substrates could accommodate two 17 inch displays. Increase this size to 650x850mm could accommodate four 17 inch displays. The cost per display for the larger size substrate is much lower as a result of the four-up layout.

2. Cell process: Liquid crystal fill and seal

A TFT-LCD assembly line consists of one TFT and color filter line each, usually in parallel production steps. This critical step differs from semiconductor wafer fabrication. Both the TFT plate and color filter plate are first coated with a thin layer of polyimide [8]. The polyimide layers are then rubbed in prearranged directions to align the liquid-crystal director. The color filter plate is then sprayed with spherical plastic spacers. To obtain good display quality, these spacers have precise dimensions and are used to produce uniform spacing (4-10 μm) between the glass plates.

An epoxy seal material is applied to the TFT plate, which is then aligned to the color filter plate. The two substrates are laminated together and the glass plate is scribed to the appropriate display panel. Finally, a liquid crystal material is injected into the gap between the glass plates. Polarizers are applied to both sides of the liquid-crystal cell.

3. Module assembly: Bonding the driver ICs to the liquid crystal cell

Module assembly involves mounting the integrated circuits for driving the display and attaching the backlight to the module. Three most common bonding techniques for the external ICs to the LC cell are Chip-on-board, tape automated bonding (TAB), and chip on glass (COG). The most popular method for packaging the driver ICs to the LC cell is with the TAB technique. The display then undergoes a final test to complete the operation.

2.2 The LTPS TFT-LCD manufacturing process

Generally, the LTPS manufacturing process will be achieved while the process temperature is under 600 degrees centigrade. The LTPS process uses laser-annealing technology to shift amorphous silicon to polysilicon, which uses a different process from amorphous silicon (a-Si). The LTPS module assembly process involves applying polarizers to both sides of the liquid-crystal cell and integrating the peripheral drive IC circuit onto the substrate for driving the display. The process does not need any outer driver LSI, so the tape automated bonding (TAB) and chip on glass (COG) processes are eliminated. A typical LTPS TFT LCD process is shown in Figures 2.2, 2.3, 2.4 and 2.5 [1]. For a concise presentation, readers are referred to Blake [9] for a detailed discussion of the manufacturing process.

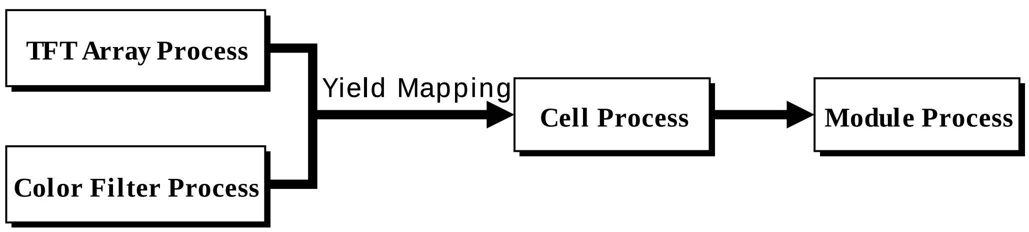

Figure 2.2 LTPS TFT LCD process flow

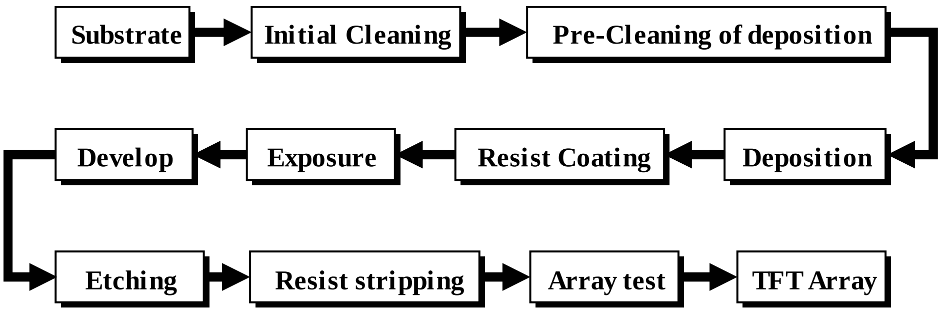

Figure 2.3 TFT array process flow

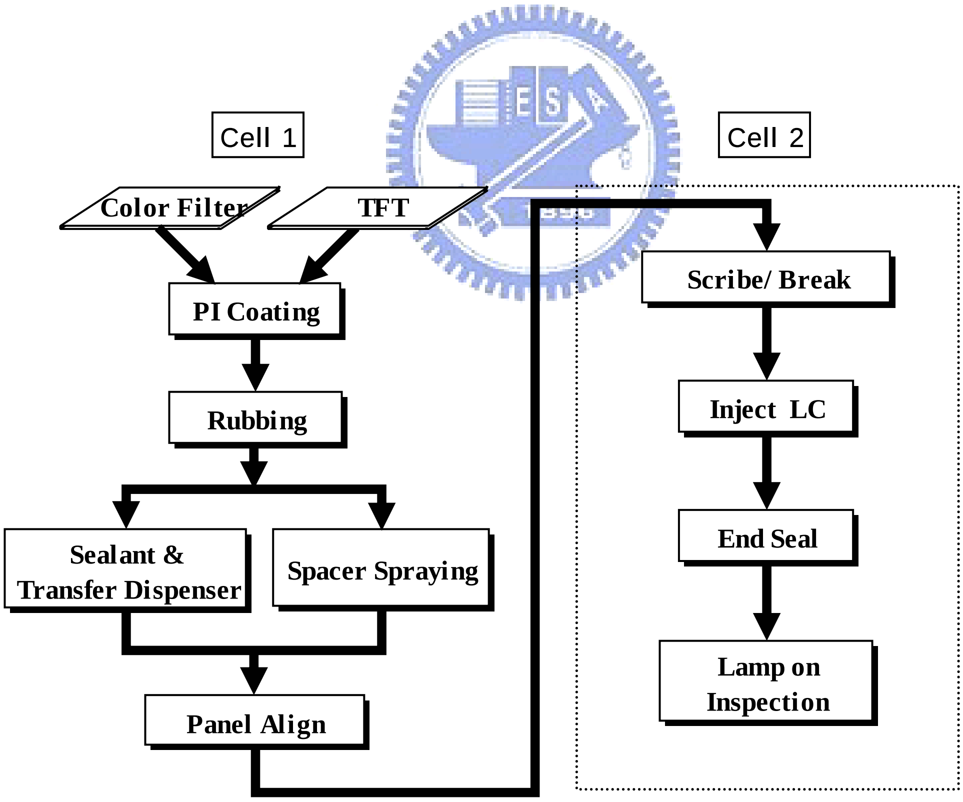

Figure 2.4 Cell process flow

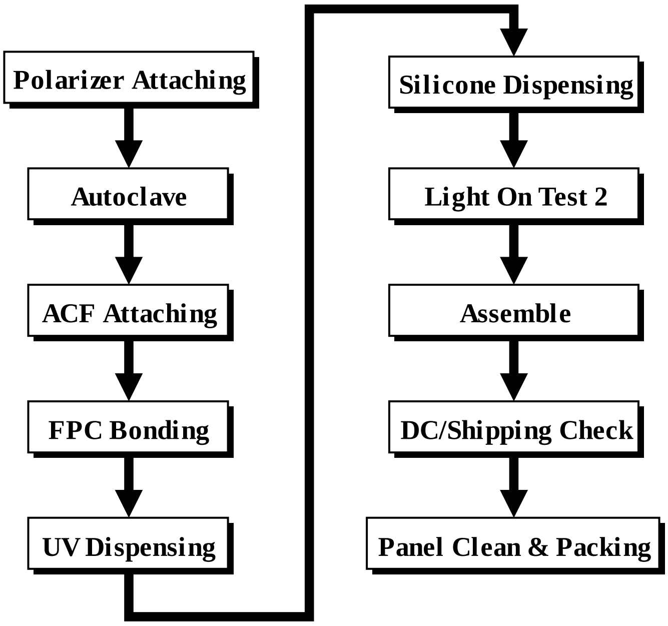

Figure 2.5 Module process flow

2.3 Linear programming

Linear programming (LP) is a tool for determining optimal solutions to problems that involves restrictions or constraints. An assignment problem is a special type of linear programming problem. Assigning *n* jobs to *n* machines with the least total cost is a common application of the assignment problem. If job *i* is assigned to machine *j* with a cost c_{ij}. The problem can be formulated in LP standard forms as follows:

$$\text{Maximize} \quad Z = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{2.1}$$

$$\text{Subject to} \quad \sum_{i=1}^{n} x_{ij} = 1 \quad \text{for} \quad j = 1,2,\cdots,n \tag{2.2}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \text{for} \quad i = 1,2,\cdots,n \tag{2.3}$$

and

$$x_{ij} \in \{0,1\} \tag{2.4}$$

where $x_{ij} = 1$ means job *i* is assigned to machine *j*, otherwise $x_{ij} = 0$

Due to its structure, some special algorithms have been developed that can solve the problem very efficiently. The most well-known is the Hungarian method, first proposed by Kuhn [10] in 1955. The method needs to cover all zeros in the reduced cost matrix by determining the minimum number of lines. To use the Hungarian method, a one-to-one matching is required. For example, each job must be assigned to only one machine. The basic procedure of the Hungarian method is [11]:

Step 1: Subtract the minimum element in each row from every element in the row. Construct the results in a new matrix.

Step 2: Subtract the minimum element in each column of the new matrix from every element in the column. Construct the results in another matrix (called the reduced cost matrix).

Step 3: Draw the minimum number of lines needed to cover all zeros in the reduced cost matrix. If the number of lines equals the number of rows, an optimal solution is available among the covered zeros in the matrix. Otherwise go on to Step 4.

Step 4: Subtract the minimum uncovered element from every uncovered element in the matrix. Add the minimum uncovered element to the elements that is covered by two lines. Return to Step 3.

As pointed out by Lotfi [12], finding the minimum number of lines to cover all of the zeros can become a tedious task. He developed a labeling algorithm for this task. Besides the Hungarian method, the Simplex method for linear programming was modified to solve the assignment problem (Paparrizos [13], Hung [14]). A scaling algorithm for the assignment problem was introduced by Goldberg [15]. Other relevant research solutions can be found from Balinski [16], Ji et al. [17], and Arora et al. [18].

The matching technique plays an important role in TFT and CF yield-matching. Several useful matching techniques have been proposed and implemented in various domains, such as string matching, dynamic programming matching, relaxation matching, heuristic matching and optimal combination matching. Among these methods, dynamic programming matching is usually used to compare two sequence of features. Lahmar et al. [19] use a dynamic programming algorithm to determine optimal feature assignment and vehicle sequences at an automotive manufacturer.

In the literature, the Hungarian method has been applied to solve matching problems. For example, Hsieh et al. [20] proposed a bipartite weighted matching method for online Chinese character recognition. Liu et al. [21] proposed a two-layer assignment method for online Chinese character recognition. Their method was stroke order and number-free. However, the former required a longer time to complete the recognition task. Fielding and Kam [22] applied the Hungarian method to stereo matching. The correspondence problem in stereo vision involves calculating matches between pixels (points) or features (e.g., lines) in stereo images. For a general review, readers are referred to Burkard [23] for a detailed discussion of the matching and assignment problem.

3. TFT-LCD Yield Mapping

3.1 Yield issues

A TFT-LCD cell (or panel) is an independent application unit, e.g., one piece of the PDA display. The glass plate contains a certain number of independent cells that belong to the same product. A given plate may contain different numbers of cells depending on its embedded cell size. The size of a cell varies from the small size used in a camera viewfinder to the large diagonal panel used in a television display.

The selected yield-matching CF glass in a TFT product can have a critical impact on cell process yield improvements. For example, each TFT and CF glass contains four panels. The yield-matching information is shown in Figure 3.1. In Figure 3.1, (*b*) and (*c*), the TFT glass contains a defective cell and the CF glass has two defective cells. Only one good panel is produced in (*c*), while (*b*) has two good panels.

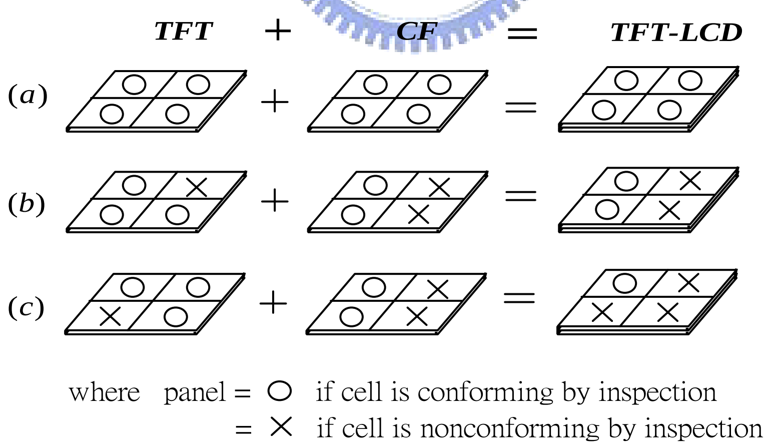

Figure 3.1 TFT and CF yield-matching

The matched LCD cell is "good" only when both the matching CF and TFT cells are "good". A "good" cell conforms to production specifications; otherwise, it is "bad" (or a defective cell). When one of the cells from either the TFT or CF plates is bad, the matched LCD cell is bad, resulting in a post-mapping yield loss. The status (good or bad) for a cell in either a TFT or a CF cell is determined by inspection before the mapping operation commences. Industrial experience has shown that the bad cell location on a plate is a direct result of the manufacturing process and material problems.

We considered improving the cell process in this work. Assume that there are *N* TFT and *N* CF cassettes in queue. Their sequences in the queues are a direct result of the manufacturing process. Each cassette is typically in lot sizes of 20 glass substrates [24]. The mapping process commences in two sequential stages: cassette and plate matching. A pair of cassettes from the *N* TFT and CF cassette lines must be chosen first. The matched pair of glass plates is then released into the cell process. The mapping process has a one-to-one match for both plates. The objective is to match the *N* TFT and corresponding *N* CF cassettes to obtain the greatest number of acceptable panels.

In the first stage, a cassette for moving twenty plates is used as a unit load. The mapping process first retrieves one cassette from each queue line. Assume that the *i*th and *j*th cassettes from the TFT and CF queue lines are selected. The *i*th TFT and *j*th CF cassettes are then matched. This is the "cassette-matching" step, as illustrated in Figure 3.2.

The next stage is to match the plates from the *i*th TFT cassette and the *j*th CF cassette to form LCD plates. Assume that the twenty plates from the TFT and CF lines are numbered $\mathrm{T}_{i1}, \mathrm{T}_{i2}, \ldots, \mathrm{T}_{i20}$ and $\mathrm{C}_{j1}, \mathrm{C}_{j2}, \ldots, \mathrm{C}_{j20}$, respectively. The plate matching chooses one TFT plate (T_{ik}) and one CF plate (C_{jl}) to form a matched LCD plate. This step is called "plate-matching", as illustrated in Figure 3.3.

Finally, T_{ik} and C_{jl} are cell mapped to form one LCD plate, as illustrated in Figure 3.4.

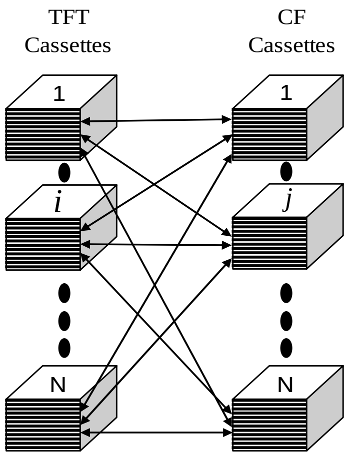

Figure 3.2 Cassette matching

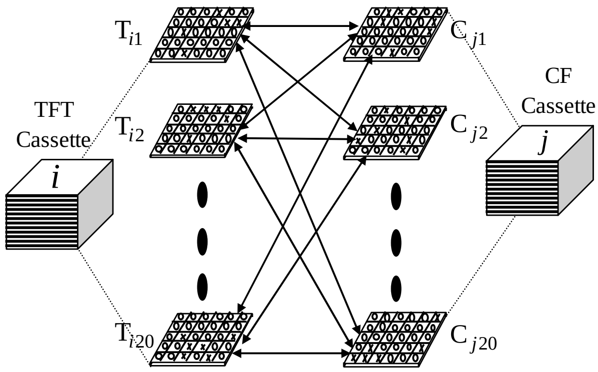

Figure 3.3 Plate matching

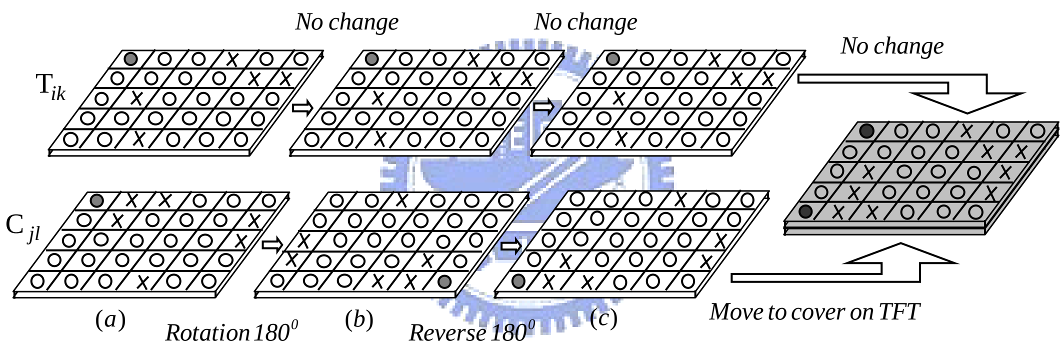

Figure 3.4 The cell mapping process (The CF glass goes through rotation and reverse then move to cover on TFT glass.)

In practice, the large-sized displays usually scribe glass in advance and then into cell process. For the small and medium size LCD panels, scribing glass in advance and then into cell process produces lower economy of scale. For example, per plate contains 50 panels scribing glass in advance that must perform cell process 50 times. An efficient method is desired to improve yield rate for the small and medium size LCD panels.

3.2 Proposed approaches

This research proposes a linear programming (LP) formulation approach to solve this problem. This approach could be very efficient and does not alter the cell process or add equipment. The results were benchmarked against two heuristics used in practice. The two heuristics will be discussed in the next section followed by the proposed LP approach.

1. Random heuristic

The simplest method is to match TFT and CF using a random approach. This approach randomly chooses a pair of cassettes and a pair of plates for cassette and plate matching. The advantages of random matching are that it is quick and easy to perform. A possible disadvantage is that not considering glass yield information might lead to LCD scrap and yield losses.

2. Best-first heuristic

This approach uses sorting techniques to improve the post-mapping yield as follows.

Step 1: Sort the N TFT cassettes in queue in descending order by yield rate.

Step 2: Sort the twenty TFT plates in each TFT cassette in descending order by yield rate.

Step 3: Based on the sequence from step 1, perform the "best" cassette-matching sequentially. "Best" indicates the highest yield. For example, the first TFT cassette in queue (after sorting) has the highest priority to choose the best matching CF cassette from those N CF cassettes in queue. Step 4 discusses how the cassette-matching yield is calculated. The second TFT cassette in queue then chooses its best matching CF cassette from those remaining $N-1$ CF cassettes. This procedure continues until the last TFT cassette in the queue is matched with the last CF cassette in queue.

Step 4: When the *i*th TFT cassette (after sorting) and the *j*th CF cassette are selected, the proposed procedure performs the "best" plate matching sequentially. It is similar to the above "best" cassette matching procedure. Based on the sequence from step 2, the first TFT plate in TFT cassette *i* has the highest priority to choose the "best" matching CF plate from those 20 CF plates in CF cassette *j*. When a TFT plate and a CF plate are chosen, their post-mapping yield is a direct compound as shown in Figure 3.4. The second TFT plate then chooses its "best" matching CF plate from those remaining 19 plates. This matching procedure continues until the last TFT plate is matched with the last CF plate.

Best-first search can potentially improve the post-mapping yield better than random heuristic, but cannot assure the optimal solution. The best yield-matching for any one TFT glass (or cassette) may not be consistent with a maximum-value when all TFT glasses (or cassettes) are considered. The best-first heuristic can be implemented on a program.

3. Linear programming formulation

Linear programming involves restrictions or constraints for determining optimal solutions to problems. The proposed LP formulation first solves the plate-matching problem for all of the possible cassette matches. The result then becomes the input to the cassette-matching problem. Notation is defined before the linear programming formulation as follows:

$N =$ the total number of cassettes in queue.

$r =$ the plate quantities of cassette.

ϕ_{ij} = the optimal matching yield of the *i*th TFT cassette and the *j*th CF cassette. This value is the result from the plate-matching LP solution.

f_{ikjl} = the mapping function represents the matching yield for the *k*th plate of the *i*th TFT cassette and the *l*th plate of the *j*th CF cassette.

$x_{ikjl} = 1$ when the *k*th plate from the *i*th TFT cassette is matched with the *l*th plate from the *j*th CF cassette. Otherwise, $x_{ikjl} = 0$. This is the decision variable of the plate-matching LP formulation.

$y_{ij} = 1$ when the *i*th TFT cassette is matched with the *j*th CF cassette. Otherwise, $y_{ij} = 0$. This is the decision variable of the cassette-matching LP formulation.

Then, the plate-matching problem can be formulated as equations (3.1) – (3.4).

Maximize $$\phi_{ij} = \sum_{k=1}^{r}\sum_{l=1}^{r} f_{ikjl}\, x_{ikjl} \tag{3.1}$$

Subject to $$\sum_{k=1}^{r} x_{ikjl} = 1 \qquad \text{for } l = 1,2,\cdots,r \tag{3.2}$$

$$\sum_{l=1}^{r} x_{ikjl} = 1 \qquad \text{for } k = 1,2,\cdots,r \tag{3.3}$$

and

$$x_{ikjl} \in \{0,1\} \tag{3.4}$$

Equation (3.1) is the objective function to maximize the yield when the *i*th TFT cassette and the *j*th CF cassette are chosen. Equation (3.2) assures that each CF plate has exactly one matching TFT plate. Equation (3.3) assures that each TFT plate has exactly one matching CF plate. Equation (3.4) is the {0, 1} constraints for the decision variables.

The proposed LP approach will solve the plate-matching LP formulation $N \times N$ times for all of the possible cassette-matching instances. Although this formulation is a combinatorial problem and for each pair matched cassettes there are $20! \approx 2.4 \times 10^{18}$ different matches, it has the special structure of a typical assignment problem that can be solved efficiently using a special algorithm, the Hungarian method. In the Hungarian method a one-to-one match is required. Readers are referred to Taha [25] and Winston [11] for a detailed discussion of the assignment problem and Hungarian method.

The proposed methodology then uses the ϕ_{ij} from the plate-matching solution results as the input to model the optimal cassette matching problem as shown in equations (3.5) – (3.8).

$$\text{Maximize} \qquad Z = \sum_{i=1}^{N}\sum_{j=1}^{N}\phi_{ij} y_{ij} \tag{3.5}$$

$$\text{Subject to} \qquad \sum_{i=1}^{N} y_{ij} = 1 \qquad \text{for} \quad j = 1,2,\cdots,N \tag{3.6}$$

$$\sum_{j=1}^{N} y_{ij} = 1 \qquad \text{for} \quad i = 1,2,\cdots,N \tag{3.7}$$

and

$$y_{ij} \in \{0,1\} \tag{3.8}$$

Equation (3.5) is the objective function that maximizes the yield through cassette matching. Equation (3.6) assures that each CF cassette is matched to exactly one TFT cassette. Equation (3.7) assures that each TFT cassette has exactly one matching CF cassette. Equation (3.8) is the {0, 1} constraint for the decision variables. The cassette matching formulation also has the special assignment problem structure and can be solved efficiently using the Hungarian method.

3.3 Illustration

To illustrate the effectiveness of the proposed optimal solution approach, a case study was adapted from a LCD manufacturing firm in Hsinchu, Taiwan. In this case study, the plate size was 620 mm × 750 mm. N is equal to 10. Five different cell sizes use the same plate. The larger the cell size, the fewer the number of cells used for a single plate. The corresponding number of cells for a given cell size is shown in Table 3.1.

Table 3.1 Cell size versus number of cells

glass substrate size: 620mm x 750mm					
Number of Panels	6	30	50	70	100
Size (inch)	14.1	6.7	5.2	3.9	3

The TFT average yield rate is about 90% for LCD factories. Color filter (CF) glasses are usually purchased from outside vendors. Therefore, the CF yield rate varies. The higher the CF yield rate, the higher the purchasing cost. Based on the company's historical data, three scenarios were investigated in this study. That is, the total average yield rates for TFT and CF plates were set at 90% and 85%, 90% and 90%, 90% and 95%, respectively.

In practice, the data can only be obtained through extra procedures with special equipment. Without losing this reality, random numbers were used to simulate the defective cells for a given yield rate. A random number generator output a value of 0 or 1 determined using the Bernoulli distribution. If the output value is 1, the cell is good. If the output value is 0, the cell is defective. Ten replications were performed to construct the 95% confidence interval [26] on the mean for each experimental scenario. The numerical results are summarized in Table 3.2 and Fig. 3.5, Table 3.3 and Fig. 3.6, Table 3.4 and Fig. 3.7.

Table 3.2 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 85% (N = 10)

Method / Panels	Random	Best-first	LP	Improvement Yield
6	76.4928 ± 0.0148	81.3333 ± 0.2810	83.3583 ± 0.1541	6.8655%, 2.0250%
30	76.4926 ± 0.0070	78.6417 ± 0.0943	80.4233 ± 0.0343	3.9307%, 1.7816%
50	76.4951 ± 0.0065	78.0530 ± 0.0755	79.5440 ± 0.0379	3.0489%, 1.4910%
70	76.4992 ± 0.0036	77.7793 ± 0.0734	79.0579 ± 0.0249	2.5587%, 1.2786%
100	76.4989 ± 0.0037	77.5630 ± 0.0453	78.6170 ± 0.0170	2.1181%, 1.0540%
Average	76.4957 %	78.6741 %	80.2001 %	3.7044%, 1.5260%

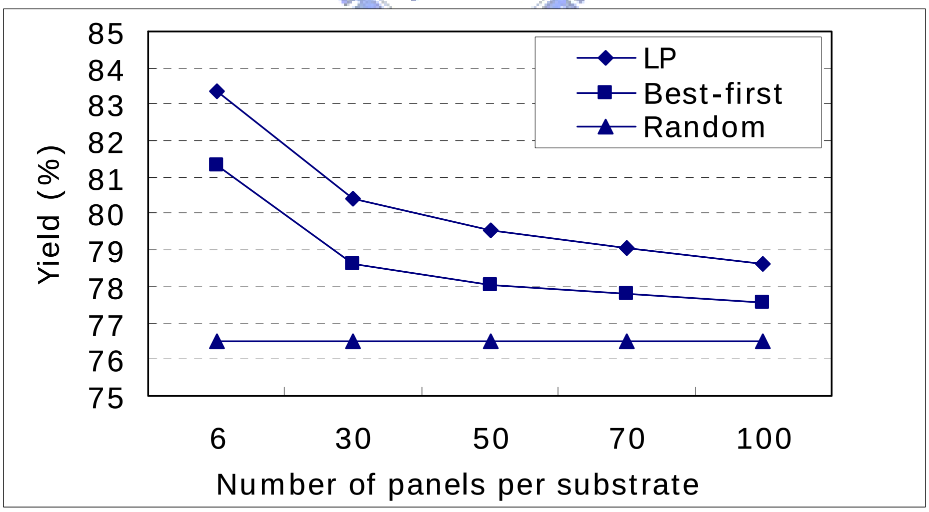

Fig.3.5 Comparison of effect with three different algorithms for various panels in 10 TFT and CF cassettes average yield rates are 90% and 85%, respectively.

Table 3.3 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 90% ($N = 10$)

Panels \ Method	Random	Best-first	LP	Improvement Yield
6	81.0035 ± 0.0092	85.4333 ± 0.2173	87.0083 ± 0.1895	6.0048%, 1.5750%
30	80.9986 ± 0.0067	82.8993 ± 0.0701	84.4567 ± 0.0304	3.4581%, 1.5584%
50	80.9998 ± 0.0048	82.3840 ± 0.0723	83.6350 ± 0.0194	2.6352%, 1.2510%
70	81.0024 ± 0.0040	82.1464± 0.0511	83.2307 ± 0.0172	2.2283%, 1.0843%
100	81.0000± 0.0022	81.9075 ± 0.0347	82.8365 ± 0.0148	1.8365%, 0.9290%
Average	81.0009 %	82.9539 %	84.2334 %	3.2325%, 1.2795%

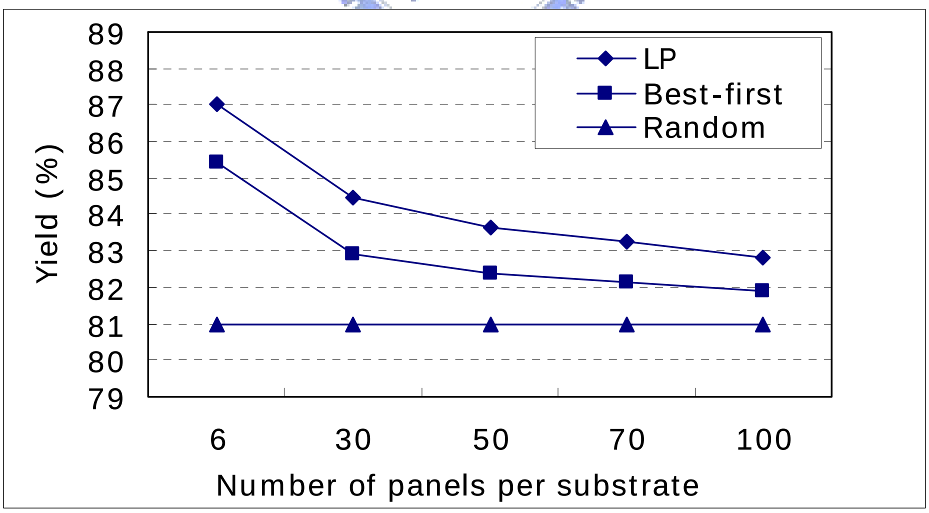

Fig.3.6 Comparison of effect with three different algorithms for various panels in 10 TFT and CF cassettes average yield rates are 90% and 90%, respectively.

Table 3.4 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 95% (N = 10)

Method / Panels	Random	Best-first	LP	Improvement Yield
6	85.5038 ± 0.0063	88.8083 ± 0.2249	89.4417 ± 0.0845	3.9379%, 0.6334%
30	85.4991 ± 0.0059	87.0433 ± 0.0783	88.0083 ± 0.0528	2.5092%, 0.9650%
50	85.4991 ± 0.0037	86.6350 ± 0.0593	87.5000 ± 0.0241	2.0009%, 0.8650%
70	85.5005 ± 0.0028	86.3964 ± 0.0451	87.1871± 0.0161	1.6866%, 0.7907%
100	85.5019 ± 0.0017	86.2230± 0.0484	86.8870± 0.0108	1.3851%, 0.6640%
Average	85.5009 %	87.0212 %	87.8048 %	2.3039%, 0.7836%

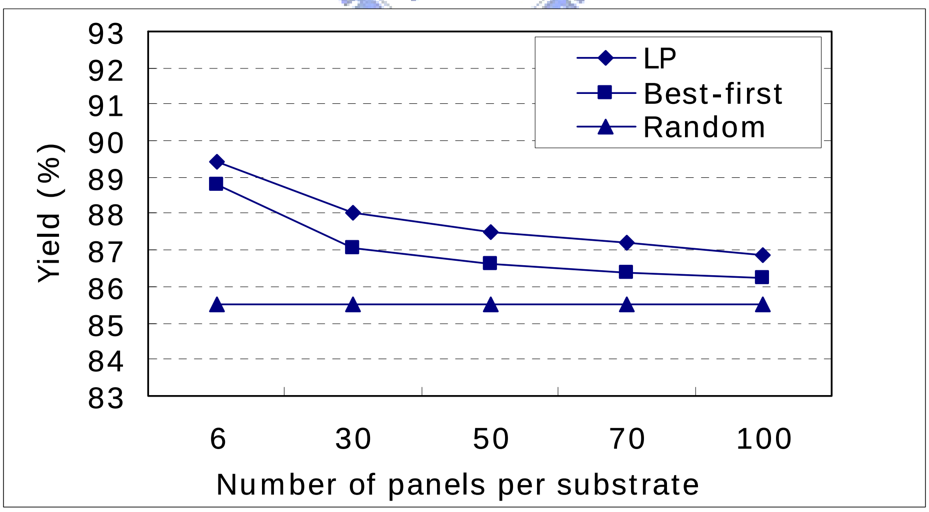

Fig.3.7 Comparison of effect with three different algorithms for various panels in 10 TFT and CF cassettes average yield rates are 90% and 95%, respectively.

To obtain solutions, we used the commercial software MATLAB and EXCEL. The average CPU time on a Pentium 4 workstation for the LP approach was about 1 minute. The best-first heuristic required about 0.8 minutes. The proposed linear programming method consistently generated superior solutions than the other heuristics. In Table 3.2, the average improvement yield from Random heuristic and Best-first heuristic were 3.7044% and 1.5260%, respectively. Considering the costly TFT and CF plates, the expected improvement represents a significant profit increase. In the case study example, the monthly throughput was 30,000 LCD plates. The average cost per LCD plate is about US$876. The expected monthly profit increases from random heuristic and best-first heuristic were about US$970,000 and US$400,000, respectively. Similarly, in Tables 3.3 and 3.4, the expected monthly profit increase to Random heuristic and Best-first heuristic were US$850,000 and US$340,000, US$610,000 and US$210,000, respectively.

In Figure 3.5, 3.6, and 3.7, the straight line at the bottom represents the average yield ratio from random matching, without respect to the panel quantities per substrate. This is unlike the other algorithms where average yield ratio increased as the panel quantities decreased.

4. TFT-LCD Yield Mapping by Using Sorter

4.1 Mapping using a sorter

Assume that there are N TFT and N CF cassettes in queue. The mapping process first places three CF cassettes and one empty cassette onto a sorter that has four ports, as illustrated in Figure 4.1. The sorter is a robot used in LCD manufacturing systems to achieve higher yield for matching TFT and CF plates. This sorter usually contains s ports that can transfer (load/unload) CF glasses from (s-1) CF cassettes into an empty cassette to match an indicated TFT cassette. The indicated TFT cassette and the empty CF cassette (filled with 20 CF glasses with the slot order the same as the matched TFT cassette glasses) will be transferred onto a loader for PI coating in the cell process. After the sorter transfers the remaining 40 CF glasses onto two other CF cassettes, one cassette becomes an empty cassette. The sorter transfers CF glasses from two CF cassettes onto the empty cassette to match another TFT cassette. The sorter then transfers the remaining 20 CF glasses onto an empty cassette to match the third TFT cassette. These steps are shown schematically in Figure 4.2.

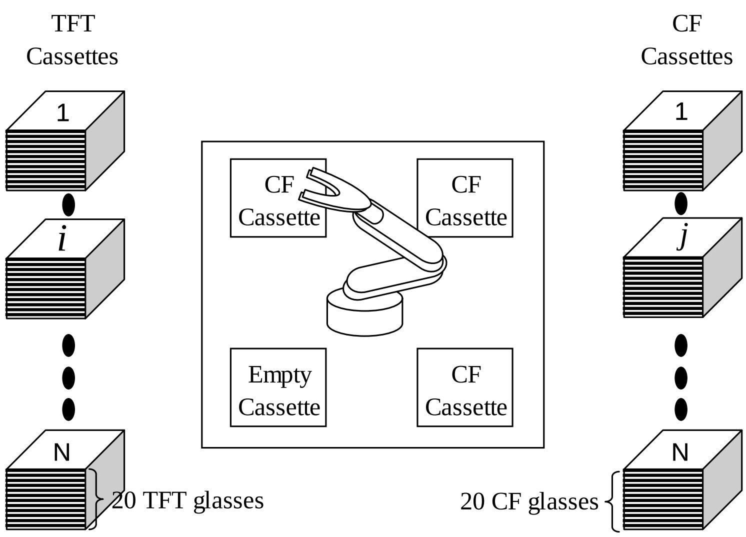

Figure 4.1 Mapping by using a sorter with 4 ports

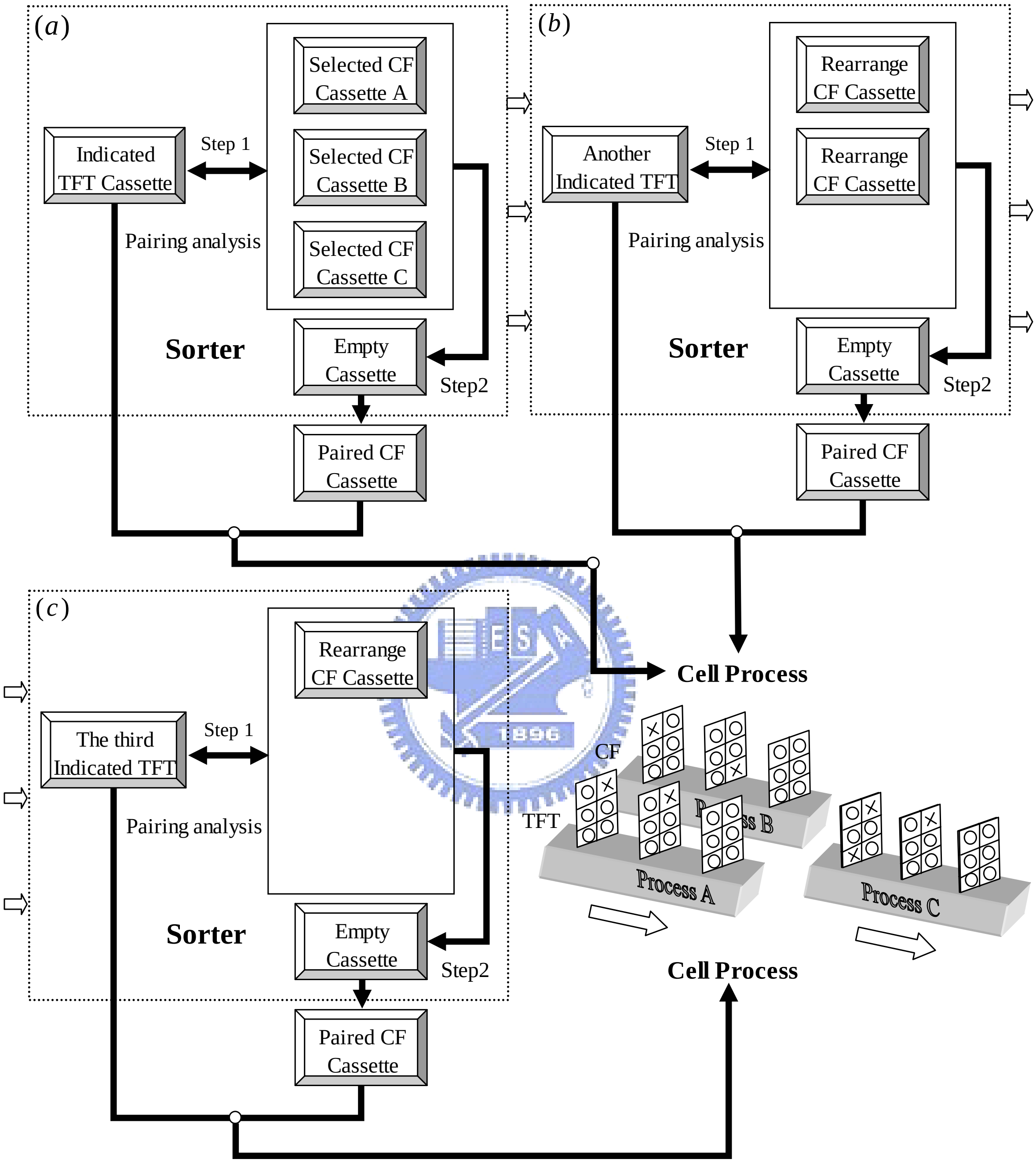

Figure 4.2 The sorter transfers CF glasses from CF cassettes into an empty cassette to match an indicated TFT cassette.

The mapping process involves two sequential stages: cassettes matching and plates matching. We suppose the sorter has 4 ports. In the first stage, the mapping process retrieves three cassettes as one sample from each queue line first. This sorter will transfer (load/unload) CF glasses from 3 CF cassettes into empty cassette to match TFT cassettes. Assume that the *i*th and *j*th sample cassettes from the TFT and CF queue lines are selected. The *i*th TFT and *j*th sample CF cassettes are then matched. This is the "cassettes-matching" step. If each sample contains only one cassette, this is "cassette-matching" as illustrated in Fig. 3.2.

The next stage involves matching the plates from the *i*th sample TFT cassette and the *j*th sample CF cassette to form LCD plates. Assume that sixty plates from the TFT and CF lines are numbered $T_{i1}, T_{i2}, \ldots, T_{i60}$ and $C_{j1}, C_{j2}, \ldots, C_{j60}$, respectively. The plate matching process chooses one TFT plate (T_{ik}) and one CF plate (C_{jl}) to form a matched LCD plate. This step is called "plates-matching" as illustrated in Figure 4.3. Similarly, if each sample contains only one cassette, this is "plate-matching" as illustrated in Figure 3.3.

Finally, T_{ik} and C_{jl} are cell mapped to form one LCD plate, as illustrated in Figure 3.4.

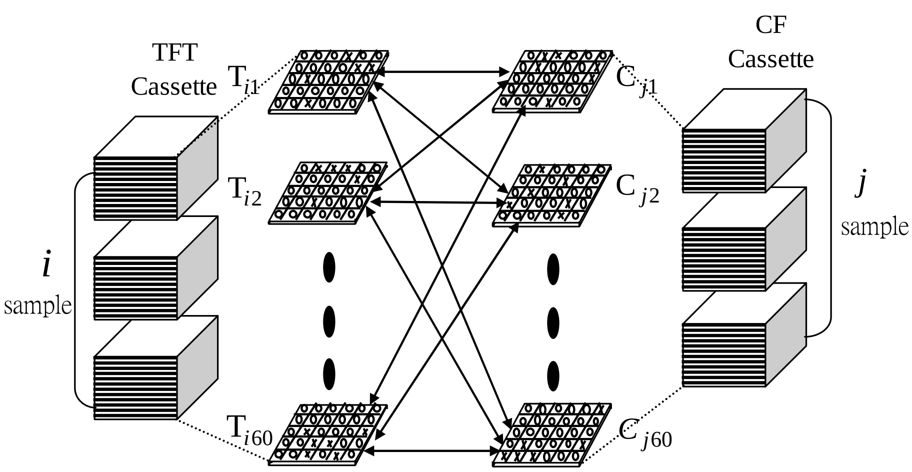

Figure 4.3 Plates matching (A sample has 3 cassettes)

4.2 Proposed approaches

Because the number of ports on the sorter is an important determinant in the post-mapping yield, this research first proposes a linear programming formulation to optimize the plates-matching problem for the various ports. This approach provides an optimal solution and offers LTPS manufacturers important yield information. Next, we propose an algorithm to reduce the number of ways for choosing different match objects when the number of matched cassettes is large. This algorithm avoids computer over-load and provides an excellent solution. A random heuristic will be discussed followed by the proposed two approaches.

1. Proposed LP formulation for solving the plates-matching problem

An assignment problem is a special type of linear programming problem. The usual assignment problem is given the same number of jobs and machines. In each assignment, assigning the job to the machine, has a fixed profit. This problem assigns each machine a unique job such that the sum of the profit from the machines is maximum. Without loss of generality, we will refer to jobs as TFT plates, machines as CF plates, and the profit as the matching yield for the TFT and CF plate. Therefore, plates-matching can be formulated as a linear programming problem. The notations are defined before the LP formulation as follows:

$N =$ the pair quantities of TFT and CF cassettes in queue.

$r =$ the plate quantities of cassette.

$s =$ the number of sorter ports.

$f_{ikjl} =$ the mapping function represents the matching yield for the kth plate from the ith sample TFT cassette and the lth plate from the jth sample CF cassette. Let two ordered n-tuples $\mathrm{p} = (p_1, p_2, \ldots, p_n)$ and $\mathrm{q} = (q_1, q_2, \ldots, q_n)$ represent TFT plate and corresponding CF plate panels (after rotation and reversing). Where $p_1, p_2, \ldots, p_n, q_1, q_2, \ldots, q_n =$ 0 (bad panel) or 1 (good panel). Then $f_{ikjl} = \mathrm{p} \cdot \mathrm{q} = p_1 q_1 + p_2 q_2 + \ldots + p_n q_n$.

$\phi_{ij} =$ the optimal matching yield from the *i*th sample TFT cassette and the *j*th sample CF cassette. This value is the result from the plates-matching LP solution.

$x_{ikjl} = 1$ when the *k*th plate from the *i*th sample TFT cassette is matched with the *l*th plate from the *j*th sample CF cassette. Otherwise, $x_{ikjl} = 0$. This is the decision variable from the plates-matching LP formulation.

The plates-matching problem can then be formulated as Equations (4.1) – (4.4).

Maximize $$\phi_{ij} = \sum_{k=1}^{r(s-1)} \sum_{l=1}^{r(s-1)} f_{ikjl} x_{ikjl} \tag{4.1}$$

Subject to $$\sum_{k=1}^{r(s-1)} x_{ikjl} = 1 \qquad \text{for } l = 1,2,\cdots,r(s-1) \tag{4.2}$$

$$\sum_{l=1}^{r(s-1)} x_{ikjl} = 1 \qquad \text{for } k = 1,2,\cdots,r(s-1) \tag{4.3}$$

and

$$x_{ikjl} \in \{0,1\} \tag{4.4}$$

Equation (4.1) is the objective function for maximizing the yield when the *i*th sample TFT cassette and the *j*th sample CF cassette are chosen. Equation (4.2) assures that each CF plate has exactly one matching TFT plate. Equation (4.3) assures that each TFT plate has exactly one matching CF plate. Equation (4.4) is the {0, 1} constraint for the decision variables. Using Equations (4.1) - (4.4), we can solve for various ports in the post-mapping yield problem.

The proposed LP approach will solve the plates-matching LP formulation $C_{s-1}^{N} \times C_{s-1}^{N}$ times for all of the possible cassettes-matching instances. Although this formulation is a combinatorial problem and for each sample matched cassettes there are $(r(s-1))!$ different matches. This is the typical assignment problem structure that can be solved efficiently using the Hungarian method. Another approach to solving the assignment problem is referred to Hung and Rom [27].

2. Proposed reduction algorithm for solving the cassettes-matching problem

Assume that there are *N* TFT and *N* CF cassettes in queue. The objective of the cassettes-matching problem is to find *N* TFT cassettes matching *N* CF cassettes such that the panel sum from the matching is maximized. In this subsection, we consider the situation when the sorter has four ports. The mapping process arbitrarily retrieves three cassettes each time from each queue line.

Given a set $S = \{1,2,3\ldots,N\}$, let sample space S_1, S_2 and S_3 be the set of all combinations of C_1^N, C_2^N and C_3^N, respectively, and *m* is a positive integer. We define

$\delta_{ij} = 1$ when the *i*th sample TFT cassette is matched with the *j*th sample CF cassette. Otherwise, $\delta_{ij} = 0$.

Let

$$\delta_{ij} = \begin{cases} x_{ij}, & \text{if each sample contains three cassettes} \\ y_{ij}, & \text{if each sample contains two cassettes} \\ z_{ij}, & \text{if each sample contains one cassette} \end{cases}$$

and

$$\phi_{ij} = \begin{cases} a_{ij}, & \text{if each sample contains three cassettes} \\ b_{ij}, & \text{if each sample contains two cassettes} \\ c_{ij}, & \text{if each sample contains one cassette} \end{cases}$$

Then, the maximum matching problem can be stated as follows:

1) when $N = 3m$

Maximize $$Z = \sum_{i=1}^{C_3^N} \sum_{j=1}^{C_3^N} a_{ij} x_{ij} \tag{4.5}$$

Subject to $$\sum_{j=1}^{C_3^N} x_{ij} = 1 \quad \text{for } i \in A \tag{4.6}$$

$$\sum_{i=1}^{C_3^N} x_{ij} = 1 \quad \text{for } j \in A \tag{4.7}$$

and

$$x_{ij} \in \{0,1\} \quad \text{for all } i \text{ and } j \tag{4.8}$$

where $A = \{A_1, A_2, \ldots, A_m \mid A_1, A_2, \ldots, A_m \in S_3,\ A_i \cap A_j = \phi \ \ \forall\, i \neq j,\ \bigcup_i^m A_i = S\}$

2) when $N = 3m + 2$

Maximize $$Z = \sum_{i=1}^{C_3^N} \sum_{j=1}^{C_3^N} a_{ij} x_{ij} + \sum_{i=1}^{C_2^N} \sum_{j=1}^{C_2^N} b_{ij} y_{ij} \tag{4.9}$$

Subject to $$\sum_{j=1}^{C_3^N} x_{ij} = 1 \quad \text{for } i \in D \tag{4.10}$$

$$\sum_{i=1}^{C_3^N} x_{ij} = 1 \quad \text{for } j \in D \tag{4.11}$$

$$\sum_{j=1}^{C_2^N} y_{ij} = 1 \quad \text{for } i \in E \tag{4.12}$$

$$\sum_{i=1}^{C_2^N} y_{ij} = 1 \quad \text{for } j \in E \tag{4.13}$$

and

$$x_{ij}, y_{ij} \in \{0,1\} \quad \text{for all } i \text{ and } j \tag{4.14}$$

where $D = \{B_1, B_2, \ldots, B_m\} \subset B$, $E = \{B_{m+1}\} \subset B$

$B = \{B_1, B_2, \ldots, B_m, B_{m+1} \mid B_1, B_2, \ldots, B_m \in S_3,\ B_{m+1} \in S_2,$

and $B_i \cap B_j = \phi \ \ \forall\, i \neq j,\ \bigcup_i^{m+1} B_i = S\}$

3) when $N = 3m + 1$

Maximize $$Z = \sum_{i=1}^{C_3^N} \sum_{j=1}^{C_3^N} a_{ij} x_{ij} + \sum_{i=1}^{C_1^N} \sum_{j=1}^{C_1^N} c_{ij} z_{ij} \qquad (4.15)$$

Subject to $$\sum_{j=1}^{C_3^N} x_{ij} = 1 \qquad \text{for } \; i \in F \qquad (4.16)$$

$$\sum_{i=1}^{C_3^N} x_{ij} = 1 \qquad \text{for } \; j \in F \qquad (4.17)$$

$$\sum_{j=1}^{C_1^N} z_{ij} = 1 \qquad \text{for } \; i \in G \qquad (4.18)$$

$$\sum_{i=1}^{C_1^N} z_{ij} = 1 \qquad \text{for } \; j \in G \qquad (4.19)$$

and

$$x_{ij}, z_{ij} \in \{0,1\} \qquad \text{for all } i \text{ and } j \qquad (4.20)$$

where $F = \{C_1, C_2, \ldots, C_m\} \subset C$, $G = \{C_{m+1}\} \subset C$

$C = \{C_1, C_2, \ldots, C_m, C_{m+1} \mid C_1, C_2, \ldots, C_m \in S_3$, $C_{m+1} \in S_1$,

and $C_i \cap C_j = \phi \;\; \forall\, i \neq j$, $\bigcup_{i}^{m+1} C_i = S \}$

Although these formulations have an assignment problem structure, the Hungarian method cannot be applied to obtain the optimal solution because the elements in the sample space S_2 and S_3 are not pairwise mutually exclusive events and they cannot satisfy a one-to-one match. However, if N is small, we can use the total enumeration method to find an optimal solution by computing all of the possible assignments. When N is large, the total enumeration method is not practical for solving the matching problem because of the very high computation time requirements. In this study, we present a reduction algorithm based on the Hungarian method for solving the large-sized TFT and CF cassettes matching problem.

The proposed approach uses the ϕ_{ij} from the plate-matching solution results as the input to model the optimal cassette-matching problem, as shown in Equations (4.21) – (4.24).

Maximize $$Z = \sum_{i=1}^{N}\sum_{j=1}^{N} c_{ij} z_{ij} \tag{4.21}$$

Subject to $$\sum_{\mathrm{i}=1}^{N} z_{ij} = 1 \qquad \text{for} \quad j = 1,2,\cdots,N \tag{4.22}$$

$$\sum_{j=1}^{N} z_{ij} = 1 \qquad \text{for} \quad i = 1,2,\cdots,N \tag{4.23}$$

and

$$z_{ij} \in \{0,1\} \tag{4.24}$$

Equation (4.21) is the objective function that maximizes the yield through cassette matching. Equation (4.22) assures that each CF cassette is matched to exactly one TFT cassette. Equation (4.23) assures that each TFT cassette has exactly one matching CF cassette. Equation (4.24) is the {0, 1} constraint for the decision variables. The cassette matching formulation also has the special assignment problem structure and can be solved efficiently using the Hungarian method.

Using Equations (4.21) - (4.24), each TFT cassette has exactly one matching CF cassette. This is the optimal solution for the cassette-matching. The proposed reduction algorithm based on the optimal solution for the cassette-matching produces an excellent solution and reduces the computational complexity for cassettes-matching. The procedure for the proposed reduction algorithm is listed as follows:

Step 1: Using Equations (4.1) - (4.4) find ϕ_{ij} for plate-matching.

Step 2: Using Equations (4.21) - (4.24) find Z for cassette-matching.

Step 3: Let TFT_i represent the *i*th TFT cassette and CF_j represent the *j*th CF cassette. By step 2, each TFT cassette has exactly one matching CF cassette. This can be denoted by $(\mathrm{TFT}_i) \leftrightarrow (\mathrm{CF}_j) \quad i, j = 1,2,\ldots,N$.

Adjusting CF cassette in order such that $j = i$. We have

$(\mathrm{TFT}_i) \leftrightarrow (\mathrm{CF}_i) \quad i = 1,2,\ldots,N$

Step 4: The assignment of the *i*th sample TFT cassette to the *j*th sample CF cassette must satisfy the following conditions:

$(\mathrm{TFT}_i, \mathrm{TFT}_j, \mathrm{TFT}_k) \leftrightarrow (\mathrm{CF}_i, \mathrm{CF}_j, \mathrm{CF}_k) \quad i, j, k = 1,2,\ldots,N \quad , \quad i \neq j \neq k$

$(\mathrm{TFT}_i, \mathrm{TFT}_j) \leftrightarrow (\mathrm{CF}_i, \mathrm{CF}_j) \quad i \neq j \quad \text{when} \quad N = 3m+2$

$(\mathrm{TFT}_i) \leftrightarrow (\mathrm{CF}_i) \quad \text{when} \quad N = 3m+1$

Step 5: Using Equations (4.1) - (4.4) calculate all assignments in step 4.

Step 6: Find maximum yield for *N* TFT cassettes matching *N* CF cassettes.

The proposed algorithm can be used to find an excellent solution with much less computational effort. Using step 2, an optimal solution for one-to-one cassette-matching can be obtained. When the sorter is used, selecting the corresponding cassettes as a sample group will produce better solutions. In step 4, the number of plates-matching sets will be reduced from $C_3^N \times C_3^N$ to C_3^N. Table 4.1 compares the optimal solution and proposed reduction algorithm for computational results when $N = 4, 5, 6$, and $n = 30$, where n represents the plate cell (panel) quantities. Under these test conditions, we can see that the solutions obtained using the proposed reduction algorithm were optimal or near optimal. When N is large, finding an optimal solution becomes complex and difficult. Table 4.2 compares the total enumeration method and proposed reduction algorithm in computational complexity for a sorter with 4 ports.

Table 4.1 A comparison of optimal and proposed reduction algorithm solutions when N = 4, 5, 6, and $n = 30$

Method \ Conditions		Proposed Reduction algorithm	Optimal solution (Total enumeration)	Difference (Panel)
TFT yield 90% CF yield 85% $s = 4, n = 30$	$N = 4$	1946	1946	0
	$N = 5$	2444	2444	0
	$N = 6$	2932	2932	0
TFT yield 90% CF yield 90% $s = 4, n = 30$	$N = 4$	2044	2044	0
	$N = 5$	2556	2556	0
	$N = 6$	3070	3070	0
TFT yield 90% CF yield 95% $s = 4, n = 30$	$N = 4$	2123	2123	0
	$N = 5$	2654	2655	1 panel
	$N = 6$	3191	3192	1 panel

Table 4.2 A comparison of total enumeration and proposed reduction algorithm for computational complexity (s = 4)

Cassettes \ Solution		Proposed Reduction algorithm	Total enumeration method
$N = 3m$	LP operation	$C_3^N + (N \times N) + 1$	$C_3^N \times C_3^N$
	The number of ways	$\frac{C_3^N \times C_3^{N-3} \times C_3^{N-6} \times \ldots \times C_3^3}{m!}$	$\frac{(C_3^N \times C_3^{N-3} \times C_3^{N-6} \times \ldots \times C_3^3)^2}{m!}$
$N = 3m + 2$	LP operation	$C_3^N + C_2^N + (N \times N) + 1$	$C_3^N \times C_3^N + C_2^N \times C_2^N$
	The number of ways	$\frac{C_2^N \times C_3^{N-2} \times C_3^{N-5} \times \ldots \times C_3^3}{m!}$	$\frac{(C_2^N \times C_3^{N-2} \times C_3^{N-5} \times \ldots \times C_3^3)^2}{m!}$
$N = 3m + 1$	LP operation	$C_3^N + (N \times N) + 1$	$C_3^N \times C_3^N + C_1^N \times C_1^N$
	The number of ways	$\frac{C_1^N \times C_3^{N-1} \times C_3^{N-4} \times \ldots \times C_3^3}{m!}$	$\frac{(C_1^N \times C_3^{N-1} \times C_3^{N-4} \times \ldots \times C_3^3)^2}{m!}$
$N = 10$	LP operation	221	14500
	The number of ways	2800	47040000

4.3 Illustration

1. Implementation results of the plates-matching

To illustrate the effectiveness of the proposed approaches, we conduct the same case study as illustrated in section 3.3, but a sorter with 2 to 5 ports is considered. Using proposed LP formulation, the implementation results (optimal solutions) for various ports on the sorter are summarized in Table 4.3 and Figure 4.4, Table 4.4 and Figure 4.5, Table 4.5 and Figure 4.6.

In Table 4.3, the average yield ratio increases as the number of ports increases. The average improvement yield was 0.5908%, 0.2561% and 0.1935% every time the sorter added one port. Similarly, in Tables 4.4 and 4.5, the expected yield increase was 0.5151%, 0.2473% and 0.3124%, 0.4639%, 0.2080% and 0.1476%, respectively.

Figures 4.4, 4.5, and 4.6 show that the improvement decreases with the increase in the number of panels. Given the same defect rate, the number of defective patterns increased with the number of panels in a plate. Assume that the average defect rate is 20% for a plate. When the number of panels is 5, there is 1 defective panel that results in 5 defect patterns. This one defect has five possible panel locations. When the number of panels is 10, the number of defective patterns is $C_2^{10} = 45$. For both instances, the number of matching plates remains the same ($= 20 \times N$). The ratio of the number of matching plates to the number of defective patterns for the former and latter instances are $(20 \times N)/5$ and $(20 \times N)/45$, respectively. The greater the number of patterns, the smaller the mapping yield. This forms a constraint to the solution quality and explains why the improvement decreases with the increase in the number of panels.

Table 4.3 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 85%

Panels \ Method	2 ports/sorter	3 ports/sorter	4 ports/sorter	5 ports/sorter
6	82.9167 ± 0.4215	83.7917 ± 0.3568	83.8889 ± 0.3628	84.0625 ± 0.4457
30	80.2333 ± 0.2463	80.9333 ± 0.1219	81.2889 ± 0.1040	81.5292 ± 0.0675
50	79.3200 ± 0.1610	79.8400 ± 0.0731	80.1200 ± 0.0796	80.3875 ± 0.0567
70	78.8357 ± 0.1406	79.2893 ± 0.1113	79.5905 ± 0.0400	79.7482 ± 0.0440
100	78.3900 ± 0.0920	78.7950 ± 0.0375	79.0417 ± 0.0585	79.1700 ± 0.0702
Average	79.9391 %	80.5299 %	80.7860 %	80.9795 %

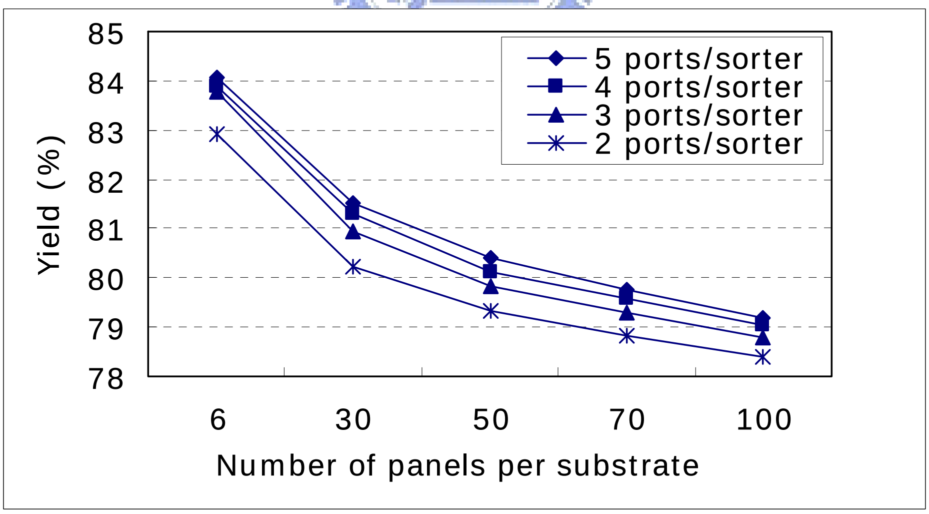

Fig.4.4 Comparison of the effect with different numbers of ports on the sorter for various panels with TFT and CF plates average yield rates are 90% and 85%, respectively.

Table 4.4 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 90%

Method / Panels	2 ports/sorter	3 ports/sorter	4 ports/sorter	5 ports/sorter
6	86.2500 ± 0.7567	86.7500 ± 0.6560	87.0833 ± 0.6016	87.8333 ± 0.4342
30	84.0833 ± 0.1013	84.8750 ± 0.1645	85.1278 ± 0.2070	85.4208 ± 0.1758
50	83.4800 ± 0.1501	84.0000 ± 0.1306	84.2167 ± 0.0828	84.4325 ± 0.0631
70	83.0143 ± 0.1315	83.4357 ± 0.0668	83.6619 ± 0.0722	83.8250 ± 0.0583
100	82.6600 ± 0.0980	83.0025 ± 0.0677	83.2100 ± 0.0491	83.3500 ± 0.0223
Average	83.8975 %	84.4126 %	84.6599 %	84.9723 %

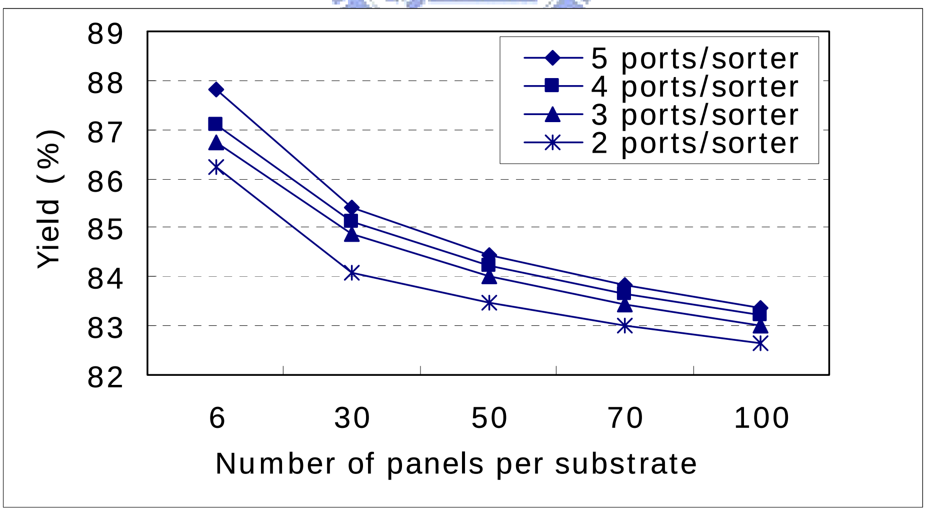

Fig.4.5 Comparison of the effect with different numbers of ports on the sorter for various panels with TFT and CF plates average yield rates are 90% and 90%, respectively.

Table 4.5 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 95%

Method / Panels	2 ports/sorter	3 ports/sorter	4 ports/sorter	5 ports/sorter
6	88.5833 ± 0.6315	89.2500 ± 0.4168	89.5000 ± 0.2616	89.6250 ± 0.1962
30	87.6333 ± 0.1760	88.2833 ± 0.1691	88.4833 ± 0.1634	88.6833 ± 0.1135
50	87.3200 ± 0.1253	87.6400 ± 0.0889	87.8633 ± 0.0307	88.0225 ± 0.0381
70	87.0357 ± 0.0909	87.3786 ± 0.0514	87.5952 ± 0.0501	87.7268 ± 0.0471
100	86.7100 ± 0.0980	87.0500 ± 0.0572	87.2000 ± 0.0329	87.3225 ± 0.0230
Average	87.4565 %	87.9204 %	88.1284 %	88.2760 %

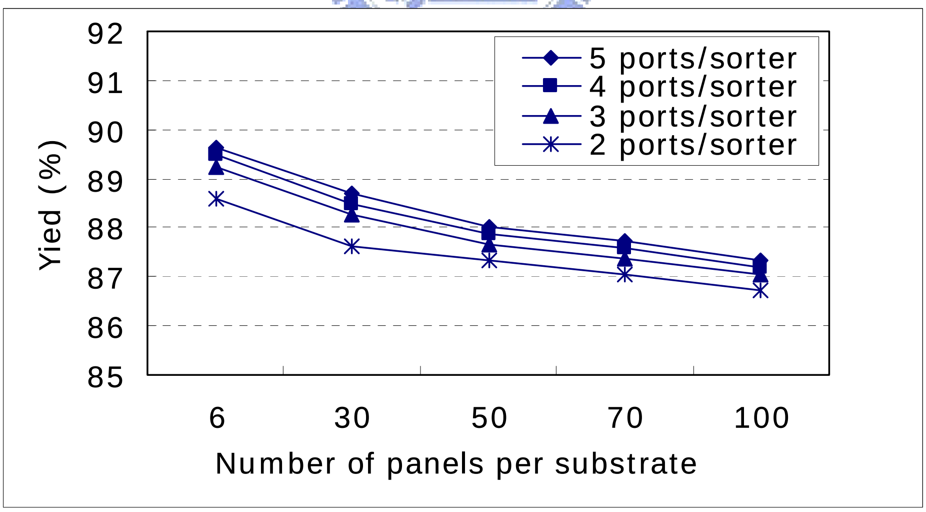

Fig.4.6 Comparison of the effect with different numbers of ports on the sorter for various panels with TFT and CF plates average yield rates are 90% and 95%, respectively.

2. Implementation results of the cassettes-matching

Figure 4.7 shows that 10 TFT cassettes and 10 CF cassettes in queue can be divided into four classes for yield mapping on a sorter with four ports. The optimal matching yield for the four classes is not practical to solve because of the very high computation time requirements.

10 TFT cassettes match 10 CF cassettes	=	3 TFT cassettes match 3 CF cassettes	+	3 TFT cassettes match 3 CF cassettes	+	3 TFT cassettes match 3 CF cassettes	+	1 TFT cassettes match 1 CF cassettes

Figure 4.7 TFT and CF match for a case involving $N = 10$ when the sorter has 4 ports.

In practice, the random mapping approach is frequently employed by engineers when n is large. This approach randomly chooses a pair of cassettes and a pair of plates for the cell process. This approach does not need to use the sorter. It is straightforward in implementation but the solution quality cannot be guaranteed. Another method is to use sorting techniques to improve the post-mapping yield. This approach uses the LP formulation according to the magnitude of the yield rate. However, it cannot assure the optimal solution. The sort approach procedures are listed as follows:

Step 1. Sort the N TFT cassettes in queue in descending order by yield rate.

Step 2. Sort the N CF cassettes in queue in descending order by yield rate.

Step 3. Using Equations (4.1) - (4.4) calculate all assignments:

$(\mathrm{TFT}_1, \mathrm{TFT}_2, \mathrm{TFT}_3) \leftrightarrow (\mathrm{CF}_1, \mathrm{CF}_2, \mathrm{CF}_3)$, $(\mathrm{TFT}_4, \mathrm{TFT}_5, \mathrm{TFT}_6) \leftrightarrow (\mathrm{CF}_4, \mathrm{CF}_5, \mathrm{CF}_6)$,

$(\mathrm{TFT}_7, \mathrm{TFT}_8, \mathrm{TFT}_9) \leftrightarrow (\mathrm{CF}_7, \mathrm{CF}_8, \mathrm{CF}_9)$, $(\mathrm{TFT}_{10}) \leftrightarrow (\mathrm{CF}_{10})$

Our proposed reduction algorithm can be used to find an optimal or near-optimal solution. To illustrate the effectiveness of the proposed reduction algorithm, we also conduct the same case study as illustrated in section 3.3, but the situation of N = 10 and s = 4 is considered in this subsection. To obtain solutions, we used the commercial software MATLAB and EXCEL. The numerical results for random mapping, sort approach and proposed reduction algorithm mapping are summarized in Tables 4.6, 4.7, 4.8, and Figures 4.8, 4.9, 4.10. The average CPU time on a Pentium 4 workstation for the proposed reduction algorithm was about 3 minutes. The sort approach required about 0.5 minutes.

In Table 4.6, the proposed reduction algorithm for the average improvement yield from random mapping and using sort approach mapping were 4.5309% and 0.3628%, respectively. Considering the costly TFT and CF plates, the expected improvement represents a significant profit increase. In the case study example, the monthly throughput was 30,000 LCD plates. The average cost per LCD plate is about US$876. The expected monthly profit increases from random mapping and using sort approach mapping were about US$1,200,000 and US$95,000, respectively. Similarly, in Tables 4.7 and 4.8, the expected monthly profit increase from random mapping and using sort approach mapping were about US$1,100,000 and US$97,000, US$740,000 and US$66,000, respectively.

For small and medium size LCD panels, if a random mapping approach is used, a great quantity of LCD display scrap is produced. Labor, material, and overhead costs are lost on scrapped displays. The proposed reduction algorithm can provide a better choice. However, if the displays size is very small or CF yield rate is very high, random mapping is feasible because the mapping average yield ratio decreases gradually as the panel quantity increases and the distances between the top curve and the bottom straight line for figures 4.8, 4.9, and 4.10 become smaller. For the large-sized displays, if TFT and CF glasses have higher yield, this proposed reduction algorithm could replace prior glass scribing. The mapping approach is more suitable for mass production than prior glass scribing.

Table 4.6 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 85% (N = 10, s = 4)

Method / Panels	Random Mapping	Sort Approach	Proposed Reduction Algorithm	Improvement
6	76.4928 ± 0.0148	83.6083 ± 0.1378	84.4583 ± 0.1596	7.9655%, 0.8500%
30	76.4926 ± 0.0070	81.1500 ± 0.0592	81.4267 ± 0.0352	4.9341%, 0.2767%
50	76.4951 ± 0.0065	80.0630 ± 0.0392	80.3280 ± 0.0314	3.8329%, 0.2650%
70	76.4992 ± 0.0036	79.5129 ± 0.0371	79.7050 ± 0.0312	3.2058%, 0.1921%
100	76.4989 ± 0.0037	78.9850 ± 0.0237	79.2150 ± 0.0514	2.7161%, 0.2300%
Average	76.4957 %	80.6638 %	81.0266 %	4.5309%, 0.3628%

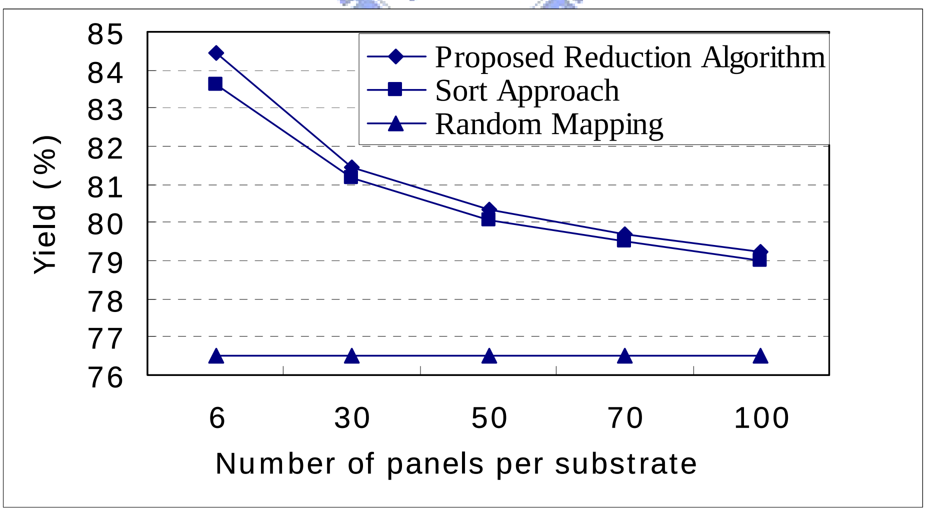

Fig.4.8 Comparison effect with three different algorithms for various panels using 10 TFT and CF cassettes with average yield rates of 90% and 85%, respectively.

Table 4.7 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 90% (N = 10, s = 4)

Panels \ Method	Random Mapping	Sort Approach	Proposed Reduction Algorithm	Improvement
6	81.0035 ± 0.0092	87.3250 ± 0.2291	88.2500 ± 0.1013	7.2465%, 0.9250%
30	80.9986 ± 0.0067	85.0350 ± 0.0905	85.3583 ± 0.0309	4.3597%, 0.3233%
50	80.9998 ± 0.0048	84.1190 ± 0.0450	84.3660 ± 0.0246	3.3662%, 0.2470%
70	81.0024 ± 0.0040	83.6107 ± 0.0234	83.8121 ± 0.0240	2.8097%, 0.2014%
100	81.0000± 0.0022	83.1655 ± 0.0352	83.3085 ± 0.0151	2.3085%, 0.1430%
Average	81.0009 %	84.6510 %	85.0190 %	4.0181%, 0.3679%

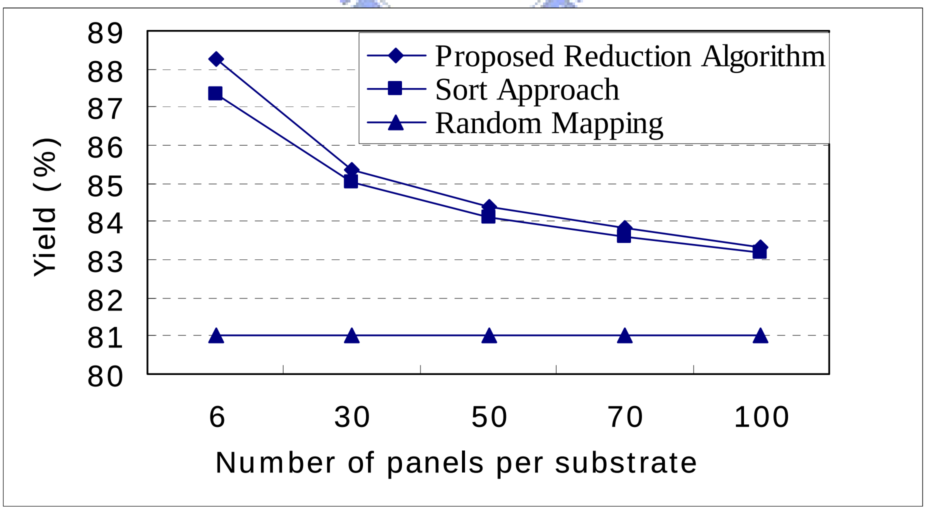

Fig.4.9 Comparison effect with three different algorithms for various panels using 10 TFT and CF cassettes with average yield rates of 90% and 90%, respectively.

Table 4.8 Mapping results in 95% confidence interval for TFT average yield 90% and CF average yield 95% (N = 10, s = 4)

Panels \ Method	Random Mapping	Sort Approach	Proposed Reduction Algorithm	Improvement
6	85.5038 ± 0.0063	89.3500 ± 0.1005	89.8750 ± 0.0644	4.3712%, 0.5250%
30	85.4991 ± 0.0059	88.3933± 0.1062	88.6467 ± 0.0798	3.1476%, 0.2534%
50	85.4991 ± 0.0037	87.8340 ± 0.0289	88.0450 ± 0.0308	2.5459%, 0.2110%
70	85.5005 ± 0.0028	87.5129 ± 0.0296	87.6621 ± 0.0152	2.1616%, 0.1492%
100	85.5019 ± 0.0017	87.1610 ± 0.0217	87.2815 ± 0.0163	1.7796%, 0.1205%
Average	85.5009 %	88.0502 %	88.3021 %	2.8012%, 0.2518%

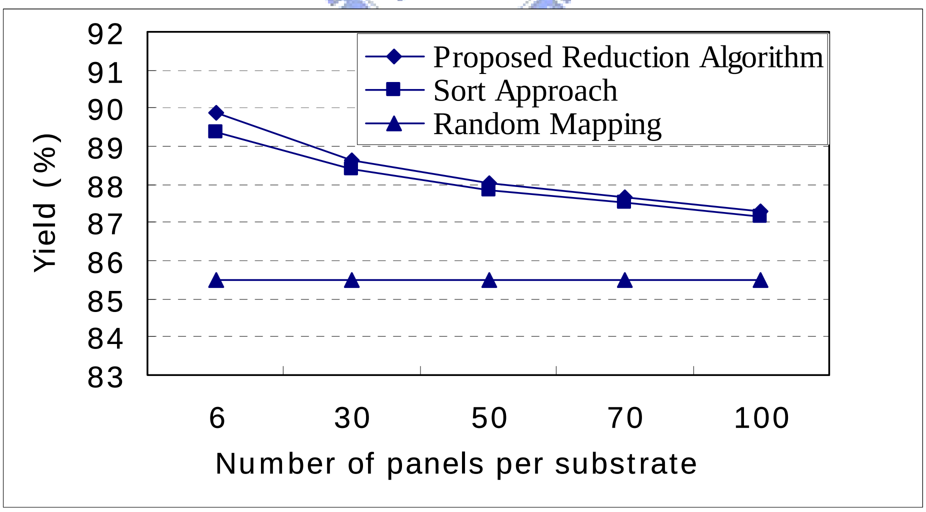

Fig.4.10 Comparison effect with three different algorithms for various panels using 10 TFT and CF cassettes with average yield rates of 90% and 95%, respectively.

5. Comparison

5.1 Comparison of the matching algorithms

In the literature, several heuristics have been developed for combinatorial optimization problems, such as: construction methods [28], limited enumeration methods [29], improvement methods [30], sampling and clustering [31], simulated annealing methods [32], genetic algorithms [33,34] and greedy randomized adaptive search procedure (GRASP) [35]. Among these, genetic algorithms (GAs) are a popular method for avoiding local optimal in improving the search. The GA attempts to parallel the biological evolution process to find better solutions. The genetic algorithm concept was introduced by Holland [33] in 1975. Recently, Ahuja et al. [34] produced very good results on large scale QAPs (quadratic assignment problems) from QAPLIB (a well-known library of QAP instances) by apply a hybrid algorithm called a greedy genetic algorithm. Out of the 132 total instances in QAPLIB, the greedy genetic algorithm obtained the best known solution for 103 instances

A greedy algorithm makes a locally optimal choice and hopes with a globally optimal solution. Hence, the algorithm does not always yield the optimal solution. However, the greedy algorithm is quite powerful for a large-sized combinatorial problem. We discuss the greedy algorithm for plates and cassettes-matching for a sorter with 4 ports as follows:

1. The greedy algorithm for plates-matching

Step 1: Sort the sixty TFT plates in descending order by yield rate.

Step 2: Based on the sequence from step 1, perform the "best" plates-matching sequentially. "Best" indicates the highest yield. For example, the first TFT plate has the highest priority for choosing the "best" matching CF plate from

60 CF plates. When a TFT plate and a CF plate are chosen, their post-mapping yield is a direct compound, as shown in Figure 3.4. The second TFT plate then chooses its “best” matching CF plate from the remaining 59 plates. This matching procedure continues until the last TFT plate is matched with the last CF plate.

Table 5.1 compares the greedy algorithm and Hungarian method for the plates-matching problems. Ten replications were performed to compare the mean for various panels using 3 TFT and CF cassettes with average yield rates of 90% and 85%, respectively. Under these test conditions, the differences in yield between the optimal solution and the solution obtained using the greedy algorithm has an average of 1.28 %.

Table 5.1 A comparison of the greedy algorithm and Hungarian method for plates-matching

Conditions \ Method		Greedy Algorithm	Hungarian Method (Optimal Solution)	Difference (%)
TFT yield 90% CF yield 85% $N = 3$ 10 replications	$n = 6$	82.6111 %	83.8889 %	1.2778 %
	$n = 30$	79.7000 %	81.2889 %	1.5889 %
	$n = 50$	78.8267 %	80.1200 %	1.2933 %
	$n = 70$	78.3524 %	79.5905 %	1.2381 %
	$n = 100$	78.0367 %	79.0417 %	1.0050 %
CPU time		About 1 second	About 1 second	

2. The greedy algorithm based on the Hungarian method for cassettes-matching

Step 1: Sort the N TFT cassettes in queue in descending order by yield rate.

Step 2: The first three TFT cassettes in queue (after sorting) have the highest priority to choose the best matching CF cassettes from those N CF cassettes in queue. The cassettes-matching yield is calculated using Equations (4.1) - (4.4).

The second three TFT cassettes in queue then chooses its best matching CF cassettes from the remaining $N - 3$ CF cassettes. This procedure continues until the last TFT cassette(s) in the queue is matched with the last CF cassette(s) in queue.

For comparison purposes, we randomly created an initial population of size 50 and used Equations (4.1) - (4.4) in the GA process. The genetic operation settings were length of string = 20 bits, crossover rate = 0.8 with PMX (Partial Matched Crossover) operator, mutation rate = 0.1 and 20 iterations. Nine samples were performed to compare the greedy algorithm, GA, and proposed reduction algorithms results for the case of $N = 10$. The results are shown in Table 5.2. As we can see, the proposed reduction algorithm consistently generated superior solutions than the other algorithms for the case using 10 TFT and CF cassettes with average yield rates of 90% and 85%, 90% and 90%, 90% and 95%, respectively.

Table 5.2 Comparison of the computation results for greedy, genetic, and proposed reduction algorithms when $N = 10$

Method / Conditions	Greedy Algorithm	Genetic Algorithm	Proposed Reduction Algorithm
TFT yield 90% CF yield 85% $s = 4, n = 30$	81.37 %	81.45 %	81.53 %
	81.17 %	81.32 %	81.40 %
	81.10 %	81.34 %	81.40 %
TFT yield 90% CF yield 90% $s = 4, n = 30$	85.22 %	85.30 %	85.42 %
	84.90 %	85.24 %	85.32 %
	84.97 %	85.24 %	85.30 %
TFT yield 90% CF yield 95% $s = 4, n = 30$	88.52 %	88.61 %	88.75 %
	88.37 %	88.63 %	88.70 %
	88.37 %	88.54 %	88.58 %
CPU time	About 3 minutes	About 30 minutes	About 3 minutes

5.2 Comparison of the matching algorithms for defect types

LCD plates have some defect types. The sources of defect types are from different stages of the manufacturing process. Materials, equipment, operations, etc., can cause the problems. We compare the performance of different algorithms for the following four defects types of LCD plates:

1. The defective panels scatter randomly on the TFT plate as illustrated in Fig. 5.1(a).
2. There are 80% defective panels gathered at the second quadrant of the TFT plate as illustrated in Fig. 5.1(b).
3. There are 80% defective panels gathered at the center of the TFT plate as illustrated in Fig. 5.1(c).
4. There are 80% defective panels gathered at the edge of the TFT plate as illustrated in Fig. 5.1(d).

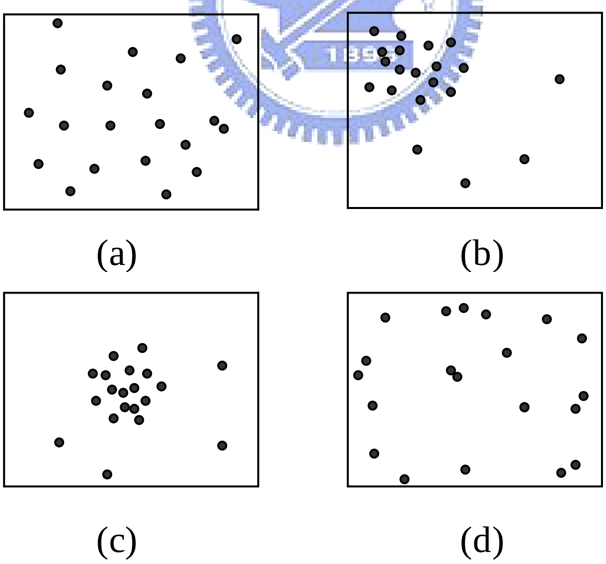

Figure 5.1 Defect types

The total average yield rates for four defect types of TFT and CF plates were set at 90% and 85%, respectively. The numerical results (N = 3) for random mapping, greedy algorithm and Hungarian method mapping using a sorter with 4 ports are summarized in Tables 5.3, 5.4, 5.5 and 5.6. The greedy algorithm is implemented on a program. The LP formulation is solved by a commercial mathematical programming solver, LINGO. Both greedy algorithm and LP formulation computation time is about 1 second.

Table 5.3 Mapping results in 95% confidence interval for defective panels scatter randomly on the plate with TFT average yield 90% and CF average yield 85%

Method / Panels	Random Mapping	Greedy Algorithm	Hungarian method	Improvement Yield
30	76.4867 ± 0.0161	79.7000 ± 0.2393	81.2889 ± 0.1040	4.8022%, 1.5889%
50	76.4880 ± 0.0214	78.8267 ± 0.1010	80.1200 ± 0.0796	3.6320%, 1.2933%
70	76.4916 ± 0.0061	78.3524 ± 0.1169	79.5905 ± 0.0400	3.0989%, 1.2381%
100	76.5094 ± 0.0163	78.0367 ± 0.0909	79.0417 ± 0.0585	2.5323%, 1.0050%
Average	76.4939 %	78.7290 %	80.0103 %	3.5164%, 1.2813%

Table 5.4 Mapping results in 95% confidence interval for 80% defective panels gathered at the second quadrant of the plate with TFT average yield 90% and CF average yield 85%

Method / Panels	Random Mapping	Greedy Algorithm	Hungarian method	Improvement Yield
30	76.5082 ± 0.0870	78.8389 ± 0.1701	80.5611 ± 0.2373	4.0529%, 1.7222%
50	76.5190 ± 0.0273	78.2667 ± 0.1211	79.7333 ± 0.0665	3.2143%, 1.4666%
70	76.4912 ± 0.0321	78.0119 ± 0.1224	79.2048 ± 0.0618	2.7136%, 1.1929%
100	76.4895 ± 0.0495	77.6517 ± 0.0802	78.6917 ± 0.0832	2.2022%, 1.0400%
Average	76.5020 %	78.1923 %	79.5477 %	3.0458%, 1.3554%

Table 5.5 Mapping results in 95% confidence interval for 80% defective panels gathered at the center of the plate with TFT average yield 90% and CF average yield 85%

Method / Panels	Random Mapping	Greedy Algorithm	Hungarian method	Improvement Yield
30	76.5193 ± 0.0801	78.8222 ± 0.2270	80.5333 ± 0.2079	4.0140%, 1.7111%
50	76.4841 ± 0.0412	78.2367 ± 0.1033	79.7333 ± 0.0954	3.2492%, 1.4966%
70	76.5391 ± 0.0199	78.0833 ± 0.1293	79.3571 ± 0.0440	2.8180%, 1.2738%
100	76.5186 ± 0.0249	77.7733 ± 0.0554	78.7967 ± 0.0533	2.2781%, 1.0234%
Average	76.5153 %	78.2289 %	79.6051 %	3.0898%, 1.3762%

Table 5.6 Mapping results in 95% confidence interval for 80% defective panels gathered at the edge of the plate with TFT average yield 90% and CF average yield 85%

Method / Panels	Random Mapping	Greedy Algorithm	Hungarian method	Improvement Yield
30	76.4869 ± 0.0262	79.0611 ± 0.2048	81.1111 ± 0.1060	4.6242%, 2.0500%
50	76.5294 ± 0.0293	78.5033 ± 0.1392	80.0000 ± 0.0803	3.4706%, 1.4967%
70	76.4811 ± 0.0293	78.0786 ± 0.0779	79.4095 ± 0.0845	2.9284%, 1.3309%
100	76.4896 ± 0.0146	77.7200 ± 0.0878	78.8050 ± 0.0500	2.3154%, 1.0850%
Average	76.4968 %	78.3408 %	79.8314 %	3.3347%, 1.4907%

As we can see, the Hungarian method consistently generated superior solution than the other algorithms for the four defect types with TFT average yield 90% and CF average yield 85%. In Table 5.3, the Hungarian method for the average improvement yield from random mapping and greedy algorithm were 3.5164% and 1.2813%, respectively. Considering the costly TFT and CF plates, the expected improvement represents a significant profit increase. In the case study example, the monthly throughput was 30,000 LCD plates. The average cost per LCD plate is about US$876. The expected monthly profit increase from random mapping and greedy algorithm were about US$924,000 and US$337,000, respectively. Similarly, in Tables 5.4, 5.5, and 5.6, the expected monthly profit increase from random mapping and greedy algorithm were about US$800,000 and US$356,000, US$812,000 and US$362,000, and US$876,000 and US$392,000, respectively.

In Tables 5.3, 5.4, 5.5 and 5.6, the average yield ratio from random mapping, without respect to the panel quantities per substrate. This is unlike the others algorithm where average yield ratio increased as the panel quantities decreased. This implies that if the displays size is very small or if CF is purchased by a very high yield rate, random mapping is feasible.

Table 5.7 represents the numerical results for defective panels scatter randomly on the plate with TFT total average yield rates 90% and CF total average yield rates 95% to 99%. According to LCD firm estimation, the difference between random mapping and using sorter mapping does not exceed yield 1%. Therefore, LCD firms should purchase CF with yield rate no less than 99% for using random mapping with TFT yield 90%.

Table 5.7 Mapping results in 95% confidence interval for defective panels scatter randomly on the plate with TFT average yield 90% and CF average yield 95% to 99%

Conditions \ Method		Random Mapping	Hungarian method	Difference
TFT yield 90% CF yield 95%	$n = 30$	85.5009 ± 0.0194	88.4833 ± 0.1634	2.9824 %
	$n = 50$	85.5039 ± 0.0091	87.8633 ± 0.0307	2.3594 %
	$n = 70$	85.5095 ± 0.0097	87.5952 ± 0.0501	2.0857 %
	$n = 100$	85.5007 ± 0.0091	87.2000 ± 0.0329	1.6993 %
TFT yield 90% CF yield 96%	$n = 30$	86.3970 ± 0.0130	88.9389 ± 0.0687	2.5419 %
	$n = 50$	86.4057 ± 0.0085	88.5700 ± 0.0650	2.1643 %
	$n = 70$	86.3989 ± 0.0087	88.2333 ± 0.0513	1.8344 %
	$n = 100$	86.3974 ± 0.0073	87.9550 ± 0.0342	1.5576 %
TFT yield 90% CF yield 97%	$n = 30$	87.2971 ± 0.0131	89.4056 ± 0.0623	2.1085 %
	$n = 50$	87.3016 ± 0.0064	89.1567 ± 0.0374	1.8551 %
	$n = 70$	87.2980 ± 0.0093	88.8810 ± 0.0681	1.5830 %
	$n = 100$	87.2985 ± 0.0074	88.6583 ± 0.0370	1.3598 %
TFT yield 90% CF yield 98%	$n = 30$	88.2010 ± 0.0115	89.7444 ± 0.0598	1.5434 %
	$n = 50$	88.2019 ± 0.0064	89.5833 ± 0.0377	1.3814 %
	$n = 70$	88.2006 ± 0.0071	89.4262 ± 0.0498	1.2256 %
	$n = 100$	88.1981 ± 0.0059	89.3000 ± 0.0446	1.1019 %
TFT yield 90% CF yield 99%	$n = 30$	89.0996 ± 0.0079	89.9167 ± 0.0429	0.8171 %
	$n = 50$	89.0981 ± 0.0063	89.8933 ± 0.0293	0.7952%
	$n = 70$	89.1007 ± 0.0033	89.8524 ± 0.0287	0.7517 %
	$n = 100$	89.0998 ± 0.0039	89.8000 ± 0.0210	0.7002 %

6. Conclusions

There are two options for post-mapping yield improvement. The first is to improve the TFT and/or CF plate yield. This approach requires improvement in the manufacturing processes, technology, tooling, etc., and may be costly and have technological constraints. The second option is to use a judicious mapping policy to optimize yield mapping. The yield mapping problem can be a significant loss contributor. A judicious matching policy is very cost effective because it does not require a significant investment to produce yield improvement. This study uses the second approach to improve the yield. We first propose a linear programming formulation to optimally solve the problem. The results were compared with two heuristics utilized in practice and showed superior solution quality.

Next, we consider a mapping problem by using a sorter. We use LP formulation to compare the various ports in the yield mapping problem and a reduction algorithm to reduce the number of ways for choosing different matched objects when the number of matched cassettes is large. This LP method provides an optimal solution and offers LCD manufacturers important yield information. The proposed reduction algorithm avoids computer over-load and produces very good results on the large scale cassettes matching problem. This avoids a great quantity of LCD display scrap, reduces production costs and improves the production yield. The LTPS focuses on manufacturing small and medium size LCD panels, scribing glass prior to the cell process leads to a much lower economy of scale. The proposed reduction algorithm can provide a better choice. Implementation results revealed that proposed approaches are effective in solving a practical problem.

For the large scale cassettes matching problem, future research can consider developing a better classification method to reduce the number of ways for choosing different matched objects. In addition, the mapping costs should be investigated to obtain an overall optimal solution.

REFERENCES

[1] http://www.toppoly.com/product/index_main.htm

[2] S.-J. Liu, and J.-F. Lee, "Liquid crystal display industry in Taiwan," *International Journal of Technology Management*, vol. 13, no.3, pp.308-325, 1997.

[3] H. S. Kim and G. S. Choi, "Development of magasonic cleaner for TFT LCD manufacturing," *International Journal of Applied Electromagnetics and Mechanics*, vol. 18, pp. 41-51, 2003.

[4] W. C. O'Mara, *Liquid Crystal Flat Panel Displays: Manufacturing Science & Technology*. New York: Van Nostrand Reinhold, 1993.

[5] P. Singer, "Flat panel displays: An interesting test case for the U.S.," *Semiconductor International*, vol. 17, no. 7, pp. 78-88, 1994.

[6] S. Jurichich, S. C. Wood and K. C. Saraswat, "Manufacturing cost of active-matrix liquid-crystal displays as a function of plant capacity," *IEEE Transactions on Semiconductor Manufacturing*, vol. 9, no. 4, pp. 562-572, Nov. 1996.

[7] S. J. Lalama, "Flat panel display manufacturing overview," *IEEE/CPMT International Electronics Manufacturing Technology Symposium*, vol. 1, pp. 185-190, 1994.

[8] J. Varney, "Liquid Crystal Display Assembly, " *Solid State Technology*, vol. 35, no. 9, pp. 61-65, 1992.

[9] J. G. Blake, M. C. King, J. D. Stevens III and R. Young, "Low-temperature polysilicon reshapes FPD production," *Solid State Technology*, vol. 40, no. 5, pp. 151-161 & vol. 40, no. 7, pp.253-258, 1997.

[10] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol.2, pp. 83-97, 1955.

[11] W. L. Winston, *Operations Research: Applications and Algorithms*. Belmont, California: Duxbury Press, 1994.

[12] V. Lotfi, "A labeling algorithm to solve the assignment problem," *Computers and Operations Research*, vol. 16, pp. 397-408, 1989.

[13] K. Paparrizos, "An infeasible (exterior point) simplex algorithm for assignment problems," *Mathematical Programming*, vol. 51, pp. 45-54, 1991.

[14] M. S. Hung, “A polynomial simplex method for the assignment problem,” *Operations Research*, vol. 31, pp. 595-600, 1983.

[15] A. V. Goldberg and R. Kennedy, “An efficient cost scaling algorithm for the assignment problem,” *Mathematical Programming*, vol. 71, pp. 153-177, 1995.

[16] M. L. Balinski, “Signature methods for the assignment problem,” *Operations Research*, vol 33, pp. 527-536, 1985.

[17] P. Ji, W. B. Lee and H. Li, “A new algorithm for the assignment problem: an alternative to the Hungarian method,” *Computers and Operations Research*, vol. 24, pp. 1017-1023, 1997.

[18] S. Arora, A. Frieze, and H. Kaplan, “A new rounding procedure for the assignment problem with applications to dense graph arrangement problems,” *Mathematical Programming*, vol. 92, no. 1, pp. 1-36, 2002.

[19] M. Lahmar, H. Ergan, and S. Benjaafar, “Resequencing and feature assignment on an automated assembly line,” *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 89-102, 2003.

[20] A.-J. Hsieh, K.-C. Fan, and T.-I Fan, “Bipartite weighted matching for on-Line handwritten Chinese character recognition,” *Pattern Recognition*, vol. 28, no. 2, pp. 143-151, 1995.

[21] J. Z. Liu, K. Ma, W. K. Cham, and M. M. Y. Chang, “Two-layer assignment method for online Chinese character recognition,” *IEE proceedings-Vision, Image and Signal Processing* , vol. 147, no. 1, pp. 47-54, 2000.

[22] G. Fielding and M. Kam, “Applying the Hungarian method to stereo matching,” *Proceedings of the 36th IEEE Conference on Decision & Control*, Vol. 2, pp. 1928-1933, 1997.

[23] R. E. Burkard, “Selected topics on assignment problems,” *Discrete Applied Mathematics*, vol. 123, pp. 257-302, 2002.

[24] B. Jeong, S.-W. Kim, and Y.-J. Lee, “An assembly scheduler for TFT LCD manufacturing,” *Computers & Industrial Engineering*, vol. 41, no. 1, pp. 37-58, 2001.

[25] H. A. Taha, *Operations Research: An Introduction*. Upper Saddle River, New Jersey: Prentice-Hall, 1997.

[26] D. C. Montgomery, *Introduction to Statistical Quality Control*. New York: J.

Wiley, 1997.

[27] M. S. Hung and W. O. Rom, "Solving the assignment problem by relaxation," *Operations Research*, Vol. 28, pp. 969-982, 1980.

[28] J. Chakrapani, and J. Skorin-Kapov, "A constructive method for improving lower bounds for a class of quadratic assignment problems," *Operations Research*, vol. 42, pp. 837-845, 1994.

[29] D. H. West, "Algorithm 608: Approximate solution of the quadratic assignment problem," *ACM Transactions on Mathematical Software*, vol.9, pp.461-466, 1983.

[30] A. M. Frieze, J. Yadegar, S. El-Horbaty, and D. Parkinson, "Algorithms for assignment problems on an array processor," *Parallel Computing*, vol.11, pp. 151-162, 1989.

[31] C. G. E. Boender, A. H. G. Rinnooy Kan, G. T. Timmer and L. Stougie "A stochastic method for global optimization," *Mathematical Programming*, vol.22, no. 2, pp. 125-140, 1982.

[32] A. Misevicius, "A modified simulated annealing algorithm for the quadratic assignment problem," *Informatica*, vol.14, pp. 497-514, 2003.

[33] J. H. Holland, *Adaptation in Natural and Artificial Systems.* Ann Arbor: University of Michigan Press, 1975.

[34] R. K. Ahuja, J. B. Orlin, and A. Tiwari, "A greedy genetic algorithm for the quadratic assignment problem," *Computers and Operations Research*, vol.27, pp. 917-934, 2000.

[35] C. A. S. Oliveira, P. M. Pardalos, and M. G. C. Resende, "GRASP with path-relinking for the quadratic assignment problem," *Lecture Notes in Computer Science*, vol.3059, pp. 356-368, 2004.