International Journal of Production Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tprs20

Two-phased meta-heuristic methods for the post-mapping yield control problem

C.-T. Su [a] & C.-H. Yang [a]
[a] Department of Industrial Engineering and Engineering Management , National Tsing Hua University , Hsinchu, Taiwan
[b] Department of Industrial Engineering and Management , National Chiao Tung University , Hsinchu, Taiwan
Published online: 22 Feb 2007.

To cite this article: C.-T. Su & C.-H. Yang (2006) Two-phased meta-heuristic methods for the post-mapping yield control problem, International Journal of Production Research, 44:22, 4837-4854, DOI: 10.1080/00207540600619726

To link to this article: http://dx.doi.org/10.1080/00207540600619726

International Journal of Production Research,
Vol. 44, No. 22, 15 November 2006, 4837–4854

Taylor & Francis
Taylor & Francis Group

Two-phased meta-heuristic methods for the post-mapping yield control problem

C.-T. SU*† and C.-H. YANG‡

†Department of Industrial Engineering and Engineering Management,
National Tsing Hua University, Hsinchu, Taiwan
‡Department of Industrial Engineering and Management,
National Chiao Tung University, Hsinchu, Taiwan

(*Revision received February 2006*)

Yield control plays an important role in the TFT-LCD manufacturing firms, and the post-mapping operation is a crucial step. The post-mapping operation combines one TFT plate and one CF plate to form a LCD. Each TFT and CF plate is divided into a number of panels. The LCD panel is acceptable only when both TFT and CF panels are good. The TFT-LCD manufacturing firms use the sorter, a kind of robot, to increase the yield for matching TFT and CF plates. Evidently, there will be a great loss if a random mapping policy is executed. In this study, we first apply two of the most popular meta-heuristic methods to solve the post-mapping problem: Genetic Algorithm (GA) and Simulated Annealing (SA). However, when the number of matched cassettes is large, the number of ways for choosing different matched objects will become so enormous that the initial population in GA (or initial solution in SA) should be selected with a proper procedure. That is, we propose a two-phased GA and SA to improve the performance of the initial population. The basic concept of phase one is to generate an efficient initial population (or initial solution). In phase one, the initial population is created based on the optimal solution to the cassette-matching problem. In phase two, we perform GA (or SA) with the initial population created in phase one. The four different heuristic algorithms are tested for the same data to compare the various ports in the post-mapping yield control problem. The result shows that proposed two-phased algorithms provide a more excellent solution than GA and SA.

Keywords: Liquid crystal display (LCD); Post mapping; Thin film transistor (TFT); Color filter (CF); Linear programming; Genetic algorithm; Simulated annealing

1. Introduction

The LCD market has increased an average of 20% per annum over the past 12 years and is expected to grow rapidly. One of the operations of the manufacturing process for LCD can be equivalent to making a sandwich. This operation is called post-mapping operation and matches one thin film transistor (TFT) and one colour filter

*Corresponding author. Email: ctsu@mx.nthu.edu.tw

International Journal of Production Research
ISSN 0020–7543 print/ISSN 1366–588X online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540600619726

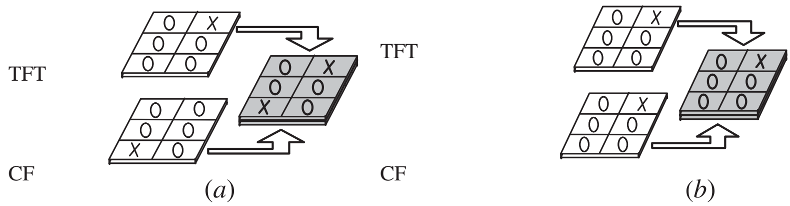

Figure 1. The post-mapping process. Panel O if cell is conforming by inspection; panel X if cell is non-conforming by inspection.

(CF) glass-plate together to form one LCD plate. A given TFT or CF plate could contain different numbers of cells (panels) with sizes ranging from small cells used in a camera viewfinder to large cells used in a television display. After mapping the TFT and CF, a liquid crystal material is injected into the gap between the glass plates to complete the post-mapping operation. Detailed discussion can be found in O'Mara (1993) and Blake *et al.* (1997).

In TFT LCD manufacturing firms, yield control, of course, plays an important role in increasing competitiveness. Reducing the loss from the post-mapping process is one of the most critical procedures. The post-mapping process, combining one TFT and one CF plate to complete a LCD plate, is shown in figure 1, where a good matched LCD cell is produced when both TFT and CF cell are good. In other words, if either the cell from the TFT or CF plate is bad, so is the matched LCD cell. The status (good or bad) of each cell for a TFT and a CF plate is known before the mapping process starts.

Figure 1 also illustrates the importance of the selection of the yield-matching glass in the post-mapping operation. In figure 1, (a) and (b) indicate that one TFT glass can result in different yield loss by matching different CF glasses. Both the TFT glass and the CF glass contain a defective cell. Only one bad panel is produced in (b), while (a) has two. Therefore, if a random mapping approach is used, a great quantity of defective LCD display scrap may be produced. Hence, the post-mapping problem is an important determinant for LCD manufacturers.

To improve post-mapping yield, we can improve the TFT and CF plate yield, but it requires improvement in the manufacturing processes, technology, tooling, etc. This approach may be costly and have technological constraints. Another way to improve the post-mapping yield is to use a judicious mapping policy to optimise yield mapping. This approach could be very efficient and practicable.

Our objective is to maximise the number of good LCD panels to improve the post-mapping yield with a practicable way. In this study, we tested four different heuristic algorithms for the same data of the post-mapping problem. The results of this study may be of interest to managers attempting to develop a judicious mapping policy to increase the post-mapping yield.

2. Post-mapping problem

Assume that there are N TFT and N CF cassettes in a queue. Cassettes are used as a unit load and each cassette contains 20 pieces of TFT or CF plates. The queue

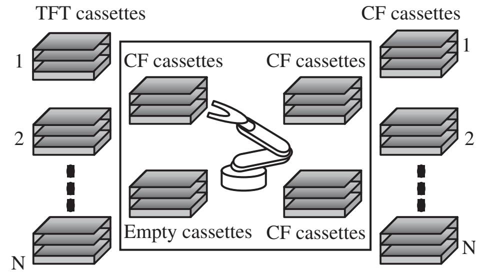

Figure 2. Mapping by using sorter (four ports).

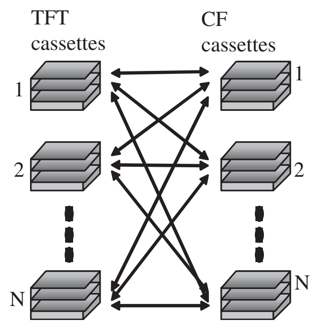

Figure 3. Cassette matching (a sample has one cassette).

sequences are the result of the manufacturing process. The sorter is a robot used to increase the yield for the post-mapping operation. The sorter usually contains several ports to load/unload CF glasses from CF cassettes into an empty cassette to match TFT cassettes. For example, three CF cassettes and one empty cassette are placed onto a sorter that has four ports, as shown in figure 2. First, 20 CF glasses will be transferred onto the empty cassette for matching the TFT cassette and the sorter transfers the remaining 40 CF glasses onto other two CF cassettes. Second, the sorter transfers another 20 CF glasses from the rest two CF cassettes onto an empty cassette to match another TFT cassette. Finally, the sorter transfers the last 20 CF glasses onto an empty cassette to match the third TFT cassette.

The mapping process involves two sequential stages: cassettes matching and plates matching. The objective is to match the N TFT to the corresponding N CF cassettes to obtain the greatest number of acceptable panels.

Figure 3 illustrates the first step in the mapping process. In this step, the mapping process retrieves $n-1$ TFT and CF cassettes (suppose the sorter has n ports) as one sample from each queue line. Assuming that the ith and jth sample cassettes from the TFT and CF queue lines are selected the ith TFT and jth sample CF cassettes are then matched. This is the 'cassettes-matching' step.

Figure 4 illustrates the next step that involves matching the plates from the ith sample TFT cassette and the jth sample CF cassette to form LCD plates, assuming that sixty plates from the TFT and CF lines are numbered $T_{i1}, T_{i2}, \ldots, T_{i60}$ and

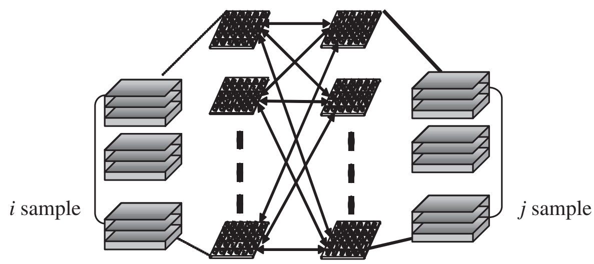

Figure 4. Plates matching (a sample has three cassettes).

$C_{j1}, C_{j2}, \ldots, C_{j60}$, respectively (that is, the sample contains three cassettes). 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'. Lastly, T_{ik} and C_{jl} are mapped to form one LCD plate, as illustrated in figure 1.

3. Mathematical model for the optimal post-mapping problem

In the existing literature, the optimal post-mapping problem has seldom received attention. In this study, we first formulate the post-mapping yield control problem in a set of linear programming (LP) problems so that the optimal solution can be found by solving all these LP problems (Su *et al*. 2005).

In this section, LP formulation is provided. LP formulation involves constraints and objective function for determining optimal solutions to the problem. The proposed LP formulation solves the plates-matching problem, and the result from the LP is used as the input to the cassette-matching problem. The notations for the LP formulation are defined as the following:

N = The total number of cassettes in queue.
n = The cell quantities of plate (substrate).
a_{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.
f_{ikjl} = The matching yield for the *k*th plate from the *i*th sample TFT cassette and the *l*th plate from the *j*th sample CF cassette. We considered a set of zero-one variables for TFT T_{ik} and CF C_{jl} glasses. Let 1 represent good (O) and 0 represent bad (X).
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, $x_{ijkl} = 0$.
y_{ij} = 1. When the *i*th sample TFT cassette is matched with the *j*th sample CF cassette. Otherwise, $y_{ij} = 0$.
p = The number of ports onto a sorter.

The plates matching problem can then be formulated as equations (1)–(4).
Maximise

$$a_{ij} = \sum_{k=1}^{20(p-1)} \sum_{l=1}^{20(p-1)} f_{ikjl} x_{ikjl} \tag{1}$$

Subject to

$$\sum_{k=1}^{20(p-1)} x_{ikjl} = 1 \quad \text{for } l = 1, 2, \ldots, 20(p-1) \tag{2}$$

$$\sum_{l=1}^{20(p-1)} x_{ikjl} = 1 \quad \text{for } k = 1, 2, \ldots, 20(p-1) \tag{3}$$

and

$$x_{ikjl} \in \{0, 1\} \tag{4}$$

Equation (1) is the objective function for maximising the yield when the ith sample TFT cassette and the jth sample CF cassette are chosen. Equation (2) assures that each CF plate has exactly one matching TFT plate. Equation (3) assures that each TFT plate has exactly one matching CF plate. Equation (4) is the $\{0, 1\}$ constraint for the decision variables. The cassette-matching problem can then be formulated as equations (5)–(8).

Maximise

$$Z = \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} y_{ij} \tag{5}$$

Subject to

$$\sum_{i=1}^{N} y_{ij} = 1 \quad \text{for } j = 1, 2, \ldots, N \tag{6}$$

$$\sum_{j=1}^{N} y_{ij} = 1 \quad \text{for } i = 1, 2, \ldots, N \tag{7}$$

and

$$y_{ij} \in \{0, 1\} \tag{8}$$

Equation (5) is the objective function that maximises the yield through cassette matching. Equation (6) assures that each CF cassette is matched to exactly one TFT cassette. Equation (7) assures that each TFT cassette has exactly one matching CF cassette. Equation (8) is the $\{0, 1\}$ constraint for the decision variables.

Both formulations have typical assignment problem structure, so we can solve the problems efficiently by using the Hungarian method. However, when the number of TFT and CF cassettes is huge ($N \geq 10$), the number of LP problems will become so large that solving all the LP problems becomes impracticable. In the following sections, we propose other approaches to solve the post-mapping problem.

4. Meta-heuristic methods

Nowadays, the computer scientist uses meta-heuristic to solve many NP-complete problems. This trend has attracted many researchers to propose many meta-heuristic algorithms to the computing society, namely: genetic algorithm, simulated annealing, tabu search, ants colony optimisation and so on. In short, meta-heuristic methods are high level concepts for exploring search spaces by using different strategies (Blum and Roli 2003).

In the literature most researchers use one or two methods to compare their model performance. The potential of applying GA and SA for the post-mapping problem is explored in this study.

4.1 *Genetic algorithm*

Genetic algorithm (GA; Holland 1975) is a highly parallel mathematical algorithm that transforms a set of objects (population) into a new population by using operations patterned according to the Darwinian principle. It is a technique capable of solving difficult optimisation problems in a complex search space. The basic concept of GA emphasises that populations can survive and breed if they have a better environmental adaptability. The three typical steps in executing GA can be summarised as follows:

1. Randomly create an initial population of chromosomes.
2. Iteratively perform the following sub-steps on the population of chromosomes until the terminated criterion has been satisfied.
   (a) Evaluate the fitness of each chromosome in the population.
   (b) Apply at least the first two of the following three operations to generate the new population. The operations are chosen with the probability based on the fitness of each chromosome.
      (i) Copy the existing chromosome to the new population.
      (ii) Create two new chromosomes by recombining the chosen chromosomes genetically.
      (iii) Create a new chromosome from an existing one by mutating genes.
3. The best chromosome in any generation is designated as the result of the GA for the run. This result may represent a solution to the problem.

In recent years, GA has been applied to various industrial engineering problems (Renner and Ekart 2003, Su and Chiang 2002). Furthermore, many researchers have found that modified or hybrid GAs outperform simple GAs (Gong *et al*. 1997, Chen and Gen 1997). A detailed discussion of the foundation of GA can be found in Holland (1975) and Goldberg (1989).

4.2 *Simulated annealing*

Simulated annealing (SA) is another emerging technique used extensively to solve complex optimisation problems (Khan *et al*. 1997). It was first introduced in 1983 by Kirkpatrick, Gellat and Vecchi. This method has become popular because of its general applicability and ability to find the solution near optimum. The basic concept of SA is simple: create an initial solution, and perturb the initial solution to generate a new solution. Compute the change in the objective function value (ΔE). If the change results in a better solution, this new solution is accepted.

If the change results in a worse solution, this new solution is accepted with probability

$$p = \exp\left(\frac{-\Delta E}{T}\right),$$

where T is the system 'temperature'. The successes of simulated annealing have resulted in a surge of interest in the method. Simulated annealing has been applied to diverse areas (Salcedo-Sanz *et al*. 2004, Su and Fu 1998).

5. Proposed algorithms for the post-mapping problem

In this section, we first apply the conventional GA and SA to solve the post-mapping problem. Then we propose two-phased algorithms (GA-based and SA-based) to compare the solution with the conventional GA and SA.

5.1 *GA for the post-mapping problem*

A GA to solve the post-mapping problem is proposed in this section. The basic terms for the GA are defined as follows:

Chromosomes
Chromosomes represent every feasible solution. Let us assume that there are N TFT cassettes and CF cassettes in a queue and the sorter contains k ports. Both TFT and CF cassettes are numbered $0, 1, 2, \ldots, N-1$. The length of the chromosome is $2N$. The first N genes in the chromosome represent the TFT cassettes number and the other N genes represent the CF cassettes number. From the beginning, every $k-1$ gene can be grouped. For example, assume we have 10 TFT cassettes and 10 CF cassettes in queue. If the sorter has four ports, one of the chromosomes is presented as figure 5. We can see that Group 1, Group 2, Group 3 and Group 4 in TFT cassettes will be assigned to Group 1, Group 2, Group 3 and Group 4 in CF cassettes respectively. In the first three groups, all 60 CF plates can be matched with 60 TFT plates by using the sorter. The sum of yield rates from four groups is the total yield rate.

Fitness function
The fitness function for the proposed algorithm is given by

$$Z = \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} y_{ij}$$

which is defined the same as the equation (5).

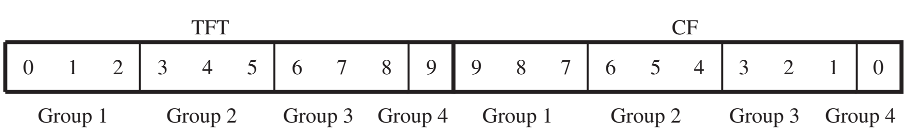

Figure 5. An example of matching chromosomes.

Crossover operator
As a crossover operator we use partially matched crossover (PMX) introduced by Goldberg (1989). This method creates two offspring from two parents. The parent chromosomes are selected with the probability based on the fitness. The operator chooses two crossover points at random, exchanges the elements between the crossover points and fill the rest of each chromosome by replacing corresponding elements.

Reproduction operator
As a reproduction operator we use ranking selection. The chromosomes are sorted according to their fitness values and the rank 1 is assigned to the best chromosome, the rank N assigned to the worst. For each chromosome, the reproduction operator is performed with probability

$$P_n = (N - n + 1) \div \sum_{n=1}^{N} n,$$

where n is the position of the chromosome after ranking.

Mutation operator
Mutation operator is used to create the opportunity to prevent the solution from local optimal. If the mutation operation is performed, it will generate a chromosome of the same length with the order of its elements reversed.

5.2 *SA for the post-mapping problem*

A SA to solve the post-mapping problem is proposed in this section. The basic terms for the SA are defined as follows:

Initial solution
Randomly generate one chromosome in GA as the initial solution.

Objective function
The objective function for conventional SA is given by

$$Z = \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} y_{ij}$$

which is defined the same as the equation (5).

Perturbing function
Randomly select two elements in the original solution and exchange these two elements.

Probability of acceptance
The probability function of accepting a worse solution is defined as

$$p = \exp\left(\frac{-\Delta E}{T}\right),$$

where ΔE is the change in the objective function value and T is the system 'temperature'.

Temperature function
If M successful changes or N total changes in the solution have occurred since the last change in temperature (T), then set the value of T to αT, where α is a constant.

5.3 *Two-phased algorithms*

Unfortunately, when the number of the TFT (or CF) cassettes is large, there will be so many feasible solutions that the result found by using conventional GA & SA might not be as good as expected. Consider the case in which $N = 10$ and $k = 4$, there will be 47 040 000 ways of combinations

$$\left(\frac{(C_3^{10} \times C_3^7 \times C_3^4)^2}{3!} = 47040000\right)$$

and 14 500 LP operations! We used the package LINGO with equipment AMD 1.6G and 256 DRAM in Windows XP environment to solve the LP operations. It will take us about 4 hours to solve just these LP operations, not to mention the immense number of ways of combinations. Figures 6 and 7 illustrate that both computational time and ways of combinations will start to rise dramatically when the TFT (or CF) cassette quantity exceeds 10.

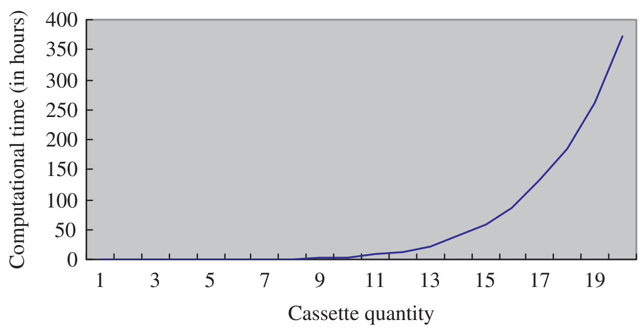

Figure 6. Computational time vs cassette quantity.

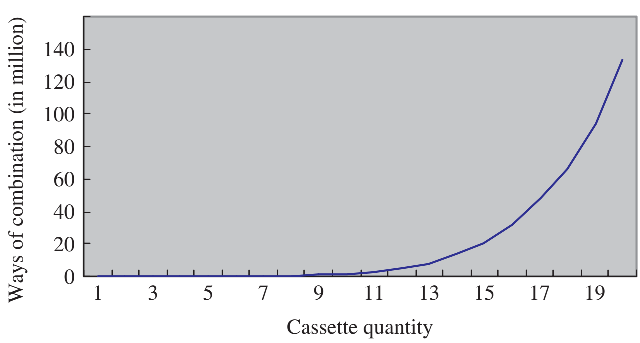

Figure 7. Ways of combinations vs cassette quantity.

When the number of combinations is huge, there will be too many feasible solutions which make it harder for GA and SA to achieve a better yield. For this reason, the initial population (or initial solution) should be selected by using proper procedures. We propose two-phased algorithms to modify our GA and SA. To generate initial population the procedures are provided in phase one. We can create more effective chromosomes by using phase one (initial solution). Next, we use the result of phase one to perform GA and SA in phase two.

5.3.1 Two-phased GA-based algorithm. The basic principles of phase one are divided into a few steps. First, overlook the sorter and find the optimal solution to the cassette-matching problem. That is, we want to determine which CF cassette should be assigned to each TFT cassette. Next, rearrange the order of the CF cassettes according to the optimal solution. The procedures are described in the following:

Phase one algorithm

BEGIN
DO {
1. Overlook the sorter, and find the optimal solution.
2. Rearrange the order of CF cassettes.
}END

In phase two, the procedures are the same as the conventional GA except the way to generate the initial population. All chromosomes are created by the following principle: if the *i*th TFT cassette is assigned to the group *k*, so is the corresponding CF cassette. In other words, the initial population should be created based on the result of phase one. For example, assume the order of TFT cassettes is (A, B, C, . . ., H, I, J) and the order of CF cassettes after phase one is (a, b, c, . . ., h, i, j). In some chromosome if the elements in group 1 of TFT cassettes are (B, G, J) the elements in group 1 of CF cassettes will be (b, g, j). The detailed procedures are described as such:

Phase two algorithm

BEGIN
DO {
1. Generate the initial population based on phase one.
2. Compute the fitness for each chromosome.
3. Use the crossover function and reproduction function to generate the next generation.
4. Compute the mutation probability for each chromosome and determine which chromosome should be mutated.
} WHILE (the maximum number of generations is satisfied)
END

5.3.2 Two-phased SA-based algorithm. In the two-phased SA-based algorithm, the basic concept is identical with the two-phased GA-based algorithm. In phase one we use the same procedures in two-phased GA to generate the initial solution. The detailed procedures are described in the following:

Phase one algorithm

```
BEGIN
DO {
   1. Overlook the sorter, and find the optimal solution.
   2. Rearrange the order of CF cassettes.
}
END
```

Phase two algorithm

```
BEGIN
DO {
   1. Generate the initial solution x based on phase one.
   2. Generate the next solution x*.
   3. Compute f(x) and f(x*)
          IF (f(x) < f(x*))
          {Let x = x*}
          ELSE
          {Compute the probability to determine whether x should be replaced.}
} WHILE (the maximum number of computations is satisfied)
END
```

We expect the performance of two-phased algorithms will be better than the conventional ones because two-phased algorithms provide a benchmark to efficiently reduce the number of different combinations. In figure 8, the queue sequences of TFT and CF cassettes are a direct result of the manufacturing process. Assume that they will obtain the yield rate, z. The CF cassettes rearranged after phase one are shown in figure 9, and they will obtain the yield rate, z^*. It is obvious that $z^* \geq z$.

It is impossible for the chromosomes (or solutions) created based on phase one to achieve worse yield rates than z^*. That is, in two-phased algorithms, z^* is used as the benchmark to assure the quality of the chromosomes; the idea of the

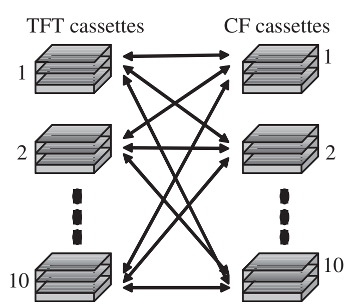

Figure 8. TFT and CF cassettes ranked by the manufacturing process.

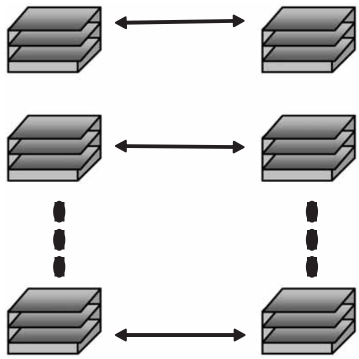

Figure 9. CF cassettes rearranged after phase one.

Table 1. Parameter settings of the post-mapping problem.

Parameter	Value
Number of cassettes (N)	10, 15, 20
Number of ports (p)	4, 5
Number of panels (n)	30, 50, 70, 100

Table 2. GA operation settings.

Parameter	Value
Population size	50
Number of iteration	20
Crossover rate	0.8
Mutation rate	0.1

two-phased algorithms is to guarantee the sorter actually helps us increase the post-mapping yield.

6. Numerical analysis

The proposed algorithms are coded in MATLAB 6.5 on AMD 1.6G and 512 DRAM PC in Windows XP environment. In this case, the total average yield rates for TFT and CF plates are 90% and 85% respectively. We used random data to simulate the detective cells on each plate, generating 0 to present detective cells and 1 to present good cells. The parameters settings of the post-mapping problem are listed in table 1. The GA and SA operation settings are listed in tables 2 and 3. The numerical results for various ports on the sorter are summarised in tables 4–9 and figures 10–15.

Table 3. SA operation settings.

Parameter	Value
Number of iteration	1000
Initial temperature	500
Temperature changing rate (α)	0.8

Table 4. Average mapping yield for $N = 10$ and $p = 4$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.60%	81.28%	81.40%	81.24%	81.40%
50	76.56%	80.22%	80.28%	80.21%	80.28%
70	76.57%	79.53%	79.69%	79.52%	79.69%
100	76.67%	79.03%	79.11%	79.01%	79.11%

Table 5. Average mapping yield for $N = 10$ and $p = 5$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.60%	81.62%	81.73%	81.65%	81.73%
50	76.56%	80.20%	80.47%	80.22%	80.47%
70	76.57%	79.78%	79.81%	79.77%	79.83%
100	76.67%	79.21%	79.24%	79.22%	79.24%

Table 6. Average mapping yield for $N = 15$ and $p = 4$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.41%	81.35%	81.50%	81.32%	81.50%
50	76.50%	80.20%	80.45%	80.18%	80.40%
70	76.42%	79.61%	79.73%	79.51%	79.72%
100	76.48%	79.14%	79.19%	79.13%	79.20%

Table 7. Average mapping yield for $N = 15$ and $p = 5$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.41%	81.70%	81.77%	81.70%	81.78%
50	76.50%	80.52%	80.57%	80.51%	80.55%
70	76.42%	79.80%	79.83%	79.71%	79.82%
100	76.48%	79.23%	79.31%	79.17%	79.32%

The results indicate that the number of ports and cassettes are both significant determinants for the post-mapping problem. The average mapping yield's increase is proportional to the increase of the number of ports or cassettes, and it seems that the number of ports can affect the improvement in the yield much greater than the

Table 8. Average mapping yield for $N=20$ and $p=4$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.48%	81.35%	81.51%	81.32%	81.52%
50	76.40%	80.27%	80.48%	80.31%	80.45%
70	76.57%	79.59%	79.78%	79.62%	79.79%
100	76.47%	79.05%	79.17%	79.07%	79.18%

Table 9. Average mapping yield for $N=20$ and $p=5$.

	Random	Simple GA	Two-phased GA	Simple SA	Two-phased SA
30	76.48%	81.68%	81.80%	81.50%	81.79%
50	76.40%	80.53%	80.62%	80.53%	80.60%
70	76.57%	79.84%	79.96%	79.82%	79.96%
100	76.47%	79.23%	79.31%	79.22%	79.32%

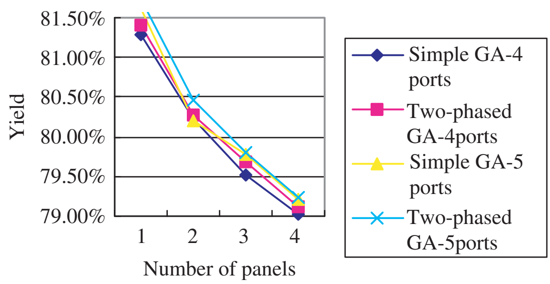

Figure 10. Comparison of the effect with different proposed GA algorithms and numbers of ports for various panels with $N=10$.

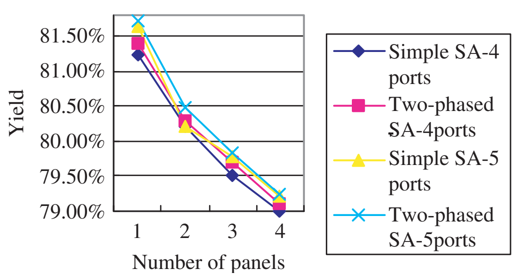

Figure 11. Comparison of the effect with different proposed SA algorithms and numbers of ports for various panels with $N=10$.

number of cassettes. For example, in tables 4 and 6, the average yield of 70 panels with two-phased GA increased from 79.69% to 79.73% while the number of cassettes increased from 10 to 15. However, the average yield increases from 79.69% to 79.81 in tables 4 and 5 where only the number of ports changed. Hence, solving

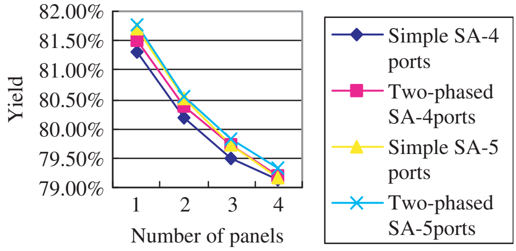

Figure 12. Comparison of the effect with different proposed GA algorithms and numbers of ports for various panels with $N = 15$.

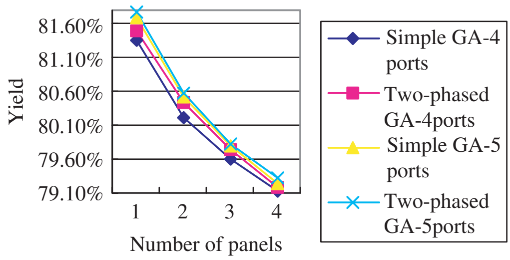

Figure 13. Comparison of the effect with different proposed SA algorithms and numbers of ports for various panels with $N = 15$.

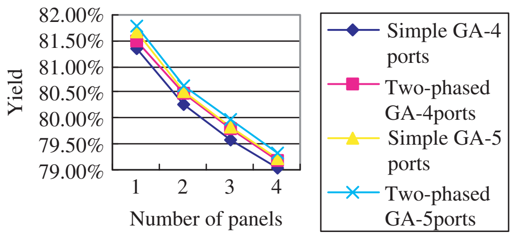

Figure 14. Comparison of the effect with different proposed GA algorithms and numbers of ports for various panels with $N = 20$.

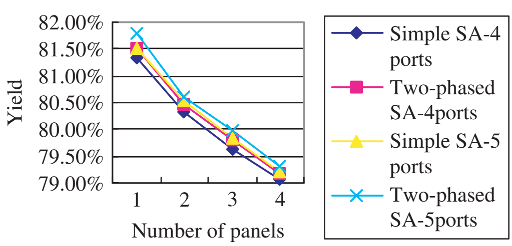

Figure 15. Comparison of the effect with different proposed SA algorithms and numbers of ports for various panels with $N = 20$.

the post-mapping problem with more ports can achieve a higher yield rate than with more cassettes.

Our results seem to provide evidence that two-phased algorithms outperform simple ones. In addition, our results also imply that there is no notable difference between the results of GA and SA. In this case study, the yearly throughput is 360 000 LCD plates. The average cost per LCD plate is about US$876. The profits from different approaches are shown in tables 10–13.

According to the data in tables 10–13, the post-mapping problem apparently should be solved with as many ports and cassettes as possible. However, the computation time will also increase. The comparison of CPU time with different scenario is shown in table 14.

Table 10. Profit caused by different proposed GA algorithms (four ports).

	Random mapping		Simple GA			Two-phased GA		
	Profit	Profit increased	Profit	Profit increased	Percentage profit increases	Profit	Profit increased	Percentage profit increases
$N=10$	241.57	–	252.34	10.77	4.5%	252.67	11.10	4.6%
$N=15$	241.10	–	252.52	11.42	4.7%	252.97	11.87	4.9%
$N=20$	241.19	–	252.49	11.3	4.7%	253.03	11.84	4.9%

In million US dollars.

Table 11. Profit caused by different proposed GA algorithms (five ports).

	Random mapping		Simple GA			Two-phased GA		
	Profit	Profit increased	Profit	Profit increased	Percentage profit increases	Profit	Profit increased	Percentage profit increases
$N=10$	241.57	–	252.93	11.36	4.7%	253.27	11.70	4.8%
$N=15$	241.10	–	253.27	12.17	5.0%	253.45	12.35	5.1%
$N=20$	241.19	–	253.30	12.11	5.0%	253.62	12.43	5.2%

In million US dollars.

Table 12. Profit caused by different proposed SA algorithms (four ports).

	Random mapping		Simple SA			Two-phased SA		
	Profit	Profit increased	Profit	Profit increased	Percentage profit increases	Profit	Profit increased	Percentage profit increases
$N=10$	241.57	–	252.27	10.70	4.4%	252.67	11.10	4.6%
$N=15$	241.10	–	252.40	11.30	4.7%	252.93	11.83	4.9%
$N=20$	241.19	–	252.54	11.35	4.7%	253.03	11.84	4.9%

In million US dollars.

Table 13. Profit caused by different proposed SA algorithms (five ports).

	Random mapping		Simple GA			Two-phased GA		
	Profit	Profit increased	Profit	Profit increased	Percentage profit increases	Profit	Profit increased	Percentage profit increases
$N = 10$	241.57	–	252.97	11.40	4.7%	253.29	11.72	4.9%
$N = 15$	241.10	–	253.15	12.05	5.0%	253.45	12.35	5.1%
$N = 20$	241.19	–	253.13	11.94	5.0%	253.60	12.41	5.1%

In million US dollars.

Table 14. Comparison of CPU time with different number of ports and cassettes.

	Four ports	Five ports	Equipment
$N = 10$	About 30 minutes	About 45 minutes	AMD 1.6 G
$N = 15$	About 45 minutes	About 60 minutes	Windows XP
$N = 20$	About 60 minutes	About 90 minutes	512 MB DRAM

The CPU time for GA and SA are almost the same.

7. Conclusions

One of the most important factors in TFT-LCD yield control problem is the post-mapping problem. A LCD plate is produced in the post-mapping operation by mapping one TFT plate and one CF plate. The objective of this study is to increase the yield in the post-mapping operation without changing any existing equipments. Although the post-mapping problem can be formulated in a set of typical linear programming problems, it is too unrealistic to solve all of them. In this study, we propose simple GA & SA and a two-phased GA & SA to solve this complicated problem. In phase one of the two-phased algorithms, we first neglected the sorter and find the optimal solution to the post-mapping problem. Second, we rearranged the order of CF cassettes. Finally, the initial solutions are properly selected and the simple GA & SA are executed in phase two.

The results indicate that it is helpful to increase the mapping yield with more cassettes or ports. The results also support that two-phased algorithms perform better than the simple ones because two-phased algorithms can efficiently reduce the number of combinations. Moreover, we also find that there is no significant difference between the results of GA and SA.

By using the proposed two-phased algorithms, the yield can be increased more than 4% in average; the firms can save millions of dollars per year. The results of this research might help managers to make more judicious mapping policies.

Acknowledgement

This research was supported in part by Grant No. NSC-92-2213-E-009-060 from the National Science Council (Taiwan).

References

Blake, J.G., King, M.C., Stevens III, J.D. and Young, R., Low-temperature polysilicon reshapes FPD production. *Solid State Tech*., 1997, **40**(5), 151–161 and **40**(7), 253–258.

Blum, C. and Roli, A., Metaheuristics in combinatorial optimisation: overview and conceptual comparison. *ACM Comp. Surv*., 2003, **35**(3), 268–308.

Chen, R. and Gen, M., Parallel machine scheduling problems using memetic algorithms. *Comp. & Indust. Eng*., 1997, **33**(3/4), 761–764.

Goldberg, D.E., *Genetic Algorithm in Search, Optimisation and Machine Learning*, 1989 (Addison-Wesley: New York, NY).

Gong, D., Gan, M., Yamazaki, M.G. and Xu, W., Hybrid evolutionary method for capacitated location-allocation problem. *Comp. & Indust. Eng*., 1997, **33**(3/4), 577–580.

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

Khan, Z., Prasad, B. and Singh, T., Machine condition optimisation by genetic algorithm and simulated annealing. *Comp. & Op. Res*., 1997, **24**(7), 647–657.

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

Renner, G. and Ekart, A., Genetic algorithms in computer aided design. *CAD*, 2003, **35**(8), 709–726.

Salcedo-Sanz, S., Santiago-Mozos, R. and Bousono-Calzon, C., A hybrid hopfield network-simulated annealing approach for frequency assignment in satellite communications systems. *IEEE Trans. on Syst., Man and Cybern*., 2004, **34**(2), 1108–1116.

Su, C.-T., Wang, P.-S. and Hsu, C.-C., Effective approaches for low temperature polysilicon TFT LCD post-mapping yield control problem. *IEEE Trans. on Autom. Sci. & Eng*., 2005, **2**(2), 198–206.

Su, C.-T. and Chiang, T.-L., Optimal design for a ball grid array wire bonding process using a neuro-genetic approach. *IEEE Trans. on Electron. Pack. Manuf*., 2002, **25**(1), 13–18.

Su, C.-T. and Fu, H.-P., A simulated annealing heuristic for robotics assembly using the dynamic pick-and-place model. *Prod. Plan. & Cont*., 1998, **9**(8), 795–802.