J Syst Sci Complex 20XX, XX( ): 1–14

Modelling and Optimizing Motherboard Functional Testing in Laptop Manufacturing*

KANG Yu · BAI Peng · WANG Kangcheng · ZHAO Yun-Bo

DOI:
Received: x x 20xx / Revised: x x 20xx

Abstract Functional testing is key to fulfill quality control in laptop manufacturing which, however, has barely been touched from the academic community. For the first time, this paper provides technical understanding of the key principles of functional testing, mathematically models the general framework, elucidates existing testing strategy under the proposed framework and model, and finally proposes a specified optimization strategy which outperforms existing strategies.

Keywords Modelling, Motherboard Functional Testing, Laptop Manufacturing, Optimization.

1 Introduction

Functional testing is key to fulfill quality control in laptop manufacturing [1, 5, 12], whose general workflow is depicted in Fig. 1. In the so-called end-to-end functional testing workflow[7], there exist two main functional testing stages: Printed Circuit Boards (PCB) first go through the Surface Mount Technology (SMT) production line to be produced as the motherboards, which are functionally tested and the defective ones are repaired; Finished laptops are also functionally tested, with the defective ones being repaired as well. Due to harsh quality control requirement, typically 100% finished laptops are functionally tested, meaning that we have to focus on the motherboard functional testing stage for any optimizing purpose [6, 9, 15].

For the motherboard functional testing stage, the testing line is installed following immediately after the SMT production line [8]. In a typical laptop manufacturing factory considered in this work (referred to as "Factory L" hereafter for privacy consideration), which is one of

---

KANG Yu · ZHAO Yun-Bo
Department of Automation & Institute of Advanced Technology, University of Science and Technology of China, Hefei, China; Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, China.
BAI Peng
Department of Automation, University of Science and Technology of China, Hefei, China.
WANG Kangcheng
Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, China.
Corresponding author: ZHAO Yun-Bo. Email: ybzhao@ustc.edu.cn

*This research was supported by the Key Research and Development Program of Anhui under Grant No. 202104a05020064.

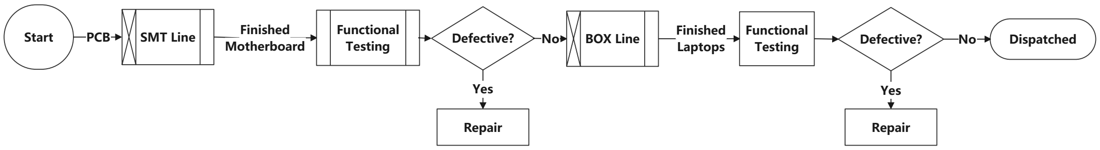

Figure 1: General framework of end-to-end functional testing in laptop manufacturing.

the largest all over the world, it has more than 30 such production lines, thus consequently accompanying more than 30 motherboard functional testing lines. Besides the space taken by the testing lines, the equipment of one automated testing line costs nearly 10^7 CNY, which includes 48 testing machines with each costing approximate 10^5 CNY, and necessary accessories costing approximate 4×10^6 CNY. Hence, in total the equipment only costs nearly 3×10^8 for motherboard functional testing in Factory L. This high cost justifies the economic value of optimizing functional testing in Factory L.

Though practically significant, such a seemingly urgent need remains barely touched from the academic community. In fact, such a need is not public to the academic community at all, as it is regarded as a business secret, only open to those trusted technical staff within Factory L. However, the technical staff may not even realize that the testing lines can be optimized and hence will not seek for external help. This is because, they intend to more focus on the specific technical issues of particular equipment and operation, but as an global optimization problem (as will be shown later in Section 3), one has to dig into the whole process and look at the problem from a global perspective [2–4, 10, 11, 13, 14, 16], which is clearly not easy for such a huge manufacturing factory involving too many manufacturing operations.

In this work, we spend more than two years to go deep into the production line of laptop manufacturing in Factory L, model the general framework of functional testing, and propose new testing strategies to optimize the testing line. Specifically, our contributions include,

- Technical understanding of the key principles in functional testing. In Section 2, three key principles, previously ambiguously comprehended by the technical staff, are explicitly stated and explained, which lay the foundation of the general motherboard functional testing framework.

- Mathematical modelling of the general framework of motherboard functional testing. In Section 3, the motherboard functional testing problem is for the first time mathematically modelled as an optimization problem, which lay the foundation of any further optimizing strategies for functional testing.

- Practical elucidating of Factory L's testing strategy under the general framework. In Section 4, we are able to elucidate why Factory L's existing strategy is designed in such a way, under our proposed general framework and model.

- A specified optimization strategy which outperforms existing strategies. In Section 5, under the general framework, we propose a specified optimization strategy for motherboard functional testing, which is shown to outperform existing strategies with high economic value.

2 Technical Understanding of the Key Principles in Functional Testing

2.1 Having 100% finished laptops tested is an inevitable requirement to ensure quality

In order to meet the harsh quality standards, it is understood that all finished laptops have to be tested for all the functions. That is, the laptop defectives in the market, is either hardly found in these functional testing, or occur after being dispatched from the factory. Such a defective rate is referred to as the Inherent Defective Rate (IDR) of laptops, denoted by r_0^l, which should be relatively small, near 0.

The above 100% functional testing operation for finished laptops are inevitable. We may interpret this point by the following simple calculations.

Denote the defective rate of all finished laptops (before testing and repairing) by r_D^l, and the ratio of the laptops to be repaired after testing by r_R^l. It is then held that

$$r_R^l = r_D^l - r_0^l \tag{1}$$

It is reasonable to assume that most defectives can be found by functional testing, meaning that $r_D^l \approx r_R^l$. At the same time, r_D^l should be a non-neglectable value, otherwise there will be no need for functional testing in the first instance. The above argument means

$$r_D^l \approx r_R^l \gg r_0^l \tag{2}$$

Now suppose we stop doing functional testing for 10% finished laptops, and then the defective rate of laptops in the market will become

$$\frac{0.1 r_D^l + 0.9 r_0^l}{r_0^l} \tag{3}$$

times of the inherent defective rate r_0^l. Then, for a typical case of $r_D^l / r_0^l > 100$, a decrease of 10% functional testing cost causes more than 10 times defective rate increase of laptops in the market. This is simply unacceptable.

2.2 Decreasing r_D^l is a fundamental requirement to decrease repair cost

We now understand that 100% finished laptops have to be tested effectively, which ensures the laptop quality in the market, but also means that the testing cost for each laptop, denoted by c_T^l, is inevitable.

On the average, the overall testing and repair cost for each laptop, can be written as follows,

$$\begin{aligned} c_A^l &= c_T^l + r_R^l c_R^l \\ &\approx c_T^l + r_D^l c_R^l \end{aligned} \tag{4}$$

where c_R^l is the average repair cost of laptops found defective in the testing operation. In practice, both r_D^l and c_R^l are relatively high, causing remarkable repair cost.

We notice that the average testing cost c_T^l and the average repair cost c_R^l should be some relatively fixed value and can not be manipulated. To decrease c_A^l, the only approach is to decrease r_R^l, and consequently to decrease the defective rate of finished laptops r_D^l, since $r_R^l \approx r_D^l$.

2.3 Motherboard functional testing is the key approach to decreasing r_D^l

The defective rate of finished laptops r_D^l is affected by the whole manufacturing production line. However, it is a fact that the repair cost can be much less if a defective is found right after its occurrence. Hence, having 100% testing only for finished laptops means the lowest testing cost but the highest repair cost; On the other hand, testing all laptops right after each manufacturing operation means the highest testing cost but the lowest repair cost.

The above analysis implies the existence of an optimal solution other than the above two extreme cases, which can balance between the testing and repair costs. The optimal solution should meet the following principles.

- Operations with sufficiently high yield should not have dedicated testing, since the testing cost can be much larger than the repair cost.
- Operations with sufficiently low yield should have dedicated 100% testing right after this operation, since this testing has to be part of the production line, and hence is not necessary to be partially tested. This also means that there is no further testing optimization possibility for such a low-yield operation.
- Operations with normal yield should be tested flexibly, balancing between the testing and repair costs. The above analysis already reveals that dedicated testing right after an operation should be 100%, it thus manes that flexible testing should not be right after the operation, and consequently, these flexible testing should be arranged in some break point of the production line, in order not to interrupt normal production.

The above principles lead immediately the conclusion that finished motherboards should be functionally tested for the defectives caused by operations with normal yield, since firstly, the SMT production line and BOX production line are independent from each other, and secondly, the defectives of motherboards contribute the majority defectives to finished laptops.

3 Mathematical Modelling of the General Framework of Motherboard Functional Testing

The general framework of motherboard functional testing is depicted in Fig. 2. In this framework, N_T different testing items are sequentially tested. For the ith testing item of a motherboard, denoted by $\mathcal{T}_i$, it is first checked whether testing is needed. If no then check the next testing item; if yes then do the testing, where the failed testing items are recorded. Only those passing all testing items or those successfully repaired go to the BOX production line.

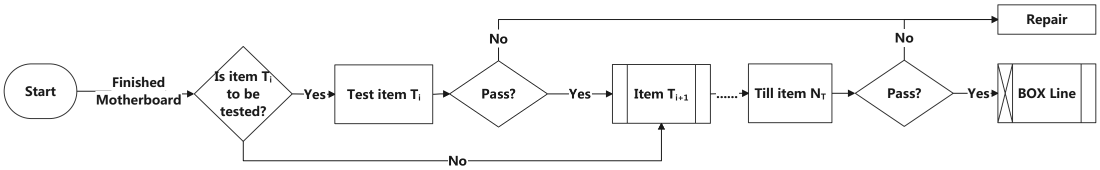

Figure 2: General framework of motherboard functional testing

In what follows, we first formulate the fundamental optimization problem of motherboard functional testing, and then detail each part of the optimization problem to form the general mathematical model. We also discuss the the basic solution to the optimization problem.

3.1 Formulating the fundamental optimization problem of motherboard functional testing

Notice that both the testing cost c_T^m and the repair cost r_R^m are dependent on specific testing strategies: A well-designed testing strategy can find more defective items at lower testing cost, while an ill-designed one may spend more testing cost to find only few defective items.

Hence, the fundamental optimization problem is to optimize the overall cost, denoted by $c_E(\mathcal{S})$, by designing appropriate testing strategy, i.e.,

$$\min_{\mathcal{S}\in\mathbb{S}} c_E(\mathcal{S}) \tag{5}$$

where $\mathcal{S} \in \mathbb{S}$ represents a specific testing strategy, $\mathbb{S}$ is the set of all possible testing strategies, and

$$c_E(\mathcal{S}) = c_T^m(\mathcal{S}) - \Delta c_R r_R^m(\mathcal{S}) \tag{6}$$

In practice, the average testing time for each motherboard can be a hard constraint. If so then the above optimization problem in (5) turns to

$$\begin{aligned} &\min_{\mathcal{S}\in\mathbb{S}} c_E(\mathcal{S}) \\ &\text{s.t.} \ \sum_{i=1}^{N_T} p_i \bar{t}_{Ti} \le \bar{t}_T \end{aligned} \tag{7}$$

where $\bar{t}_{Ti}$ is the average testing time for testing item $\mathcal{T}_i$, p_i is the percentage of testing item $\mathcal{T}_i$ being actually tested, and t_T is the time constraint for the testing.

3.2 Modelling the motherboard functional testing problem

3.2.1 The motherboard functional testing strategy $\mathcal{S}$

A motherboard functional testing strategy is to determine, either for a specific motherboard, whether each testing item $\mathcal{T}_i$ is tested or not; or for a volume of motherboards during a specific time interval, what percentage of motherboards is tested for each testing item. Clearly, the former is a more general interpretation but the latter may be more practical.

Based on the above understanding, a motherboard functional testing strategy, denoted by $\mathcal{S}$, can be written as follows,

$$\mathcal{S} : (p_1, p_2, \ldots, p_{N_T}) \tag{8}$$

where p_i, interpreted as the testing probability or percentage, is to be determined.

3.2.2 The testing cost $c_T^m(\mathcal{S})$ derived from the testing strategy $\mathcal{S}$

As mentioned earlier, the testing cost $c_T^m(\mathcal{S})$ is dependent on the testing strategy $\mathcal{S}$. Denote the testing cost of each testing item $\mathcal{T}_i$ by c_{Ti}, then, using testing strategy $\mathcal{S}$, the overall testing cost will be

$$c_T^m(\mathcal{S}) = \sum_{i=1}^{N_T} p_i c_{Ti}^m \tag{9}$$

3.2.3 The repair rate $r_R^m(\mathcal{S})$ derived from the testing strategy $\mathcal{S}$

In order to establish the relationship between the repair rate $r_R^m(\mathcal{S})$ and the testing strategy $\mathcal{S}$, we have to make two assumptions. Firstly, a motherboard will be repaired if it fails any one of the testing items (of course, there can be more than one testing items that it fails). Secondly, no correlations of any kind can be found in these testing items, meaning that one may not deduce anything on one testing item from any other testing item(s). This second assumption may not be always held, but should be generally true, since otherwise the correlated testing items have already been ignored in practice.

Based on the above assumptions, the probability of a motherboard being tested to be non-defective, i.e., $1 - r_R^m(\mathcal{S})$, can be written as the product of the probability of each testing item being non-defective, $1 - r_{Ri}^m(\mathcal{S})$, that is,

$$1 - r_R^m(\mathcal{S}) = \prod_{i=1}^{N_T} (1 - r_{Ri}^m(\mathcal{S})) \tag{10}$$

For any testing item $\mathcal{T}_i$ of any specific motherboard, it is reasonable to assume that being tested can always find whether it is defective, but it remains unknown without testing. Hence,

$$r_{Ri}^m(\mathcal{S}) = p_i r_{Di} \tag{11}$$

where r_{Di} is the actual defective rate of testing item $\mathcal{T}_i$. Therefore,

$$r_R^m(\mathcal{S}) = 1 - \prod_{i=1}^{N_T} (1 - p_i r_{Di}) \tag{12}$$

which builds the relationship between $r_R^m(\mathcal{S})$ and $\mathcal{S}$.

3.2.4 The model for motherboard functional testing

From (5), (9) and (12), it can be derived that

$$\begin{aligned}
&\min_{\mathcal{S}\in\mathbb{S}} c_E(\mathcal{S})\\
=&\min_{\mathcal{S}\in\mathbb{S}}\{c_T^m(\mathcal{S})-\Delta c_R r_R^m(\mathcal{S})\}\\
=&\min_{\mathcal{S}\in\mathbb{S}}\{\sum_{i=1}^{N_T} p_i c_{Ti}^m-\Delta c_R(1-\prod_{i=1}^{N_T}(1-p_i r_{Di}))\}\\
=&\min_{\mathcal{S}\in\mathbb{S}}\{\sum_{i=1}^{N_T} p_i c_{Ti}^m+\Delta c_R\prod_{i=1}^{N_T}(1-p_i r_{Di})\}\\
=&\min_{\mathcal{S}\in\mathbb{S}} c(\mathcal{S})
\end{aligned} \tag{13}$$

where the Average Effective Cost (AEC) $c(\mathcal{S})$ can be obtained as follows,

$$c(\mathcal{S})=\sum_{i=1}^{N_T} p_i c_{Ti}^m+\Delta c_R\prod_{i=1}^{N_T}(1-p_i r_{Di}) \tag{14}$$

If we consider the time constraint as in (7), the above optimization model (13) then turns to

$$\begin{aligned}
&\min_{\mathcal{S}\in\mathbb{S}} c(\mathcal{S})\\
\text{s.t.}\ &\sum_{i=1}^{N_T} p_i \bar{t}_{Ti}\le \bar{t}_T
\end{aligned} \tag{15}$$

3.3 The optimal testing strategy

To obtain the solution to (14), we may let

$$\frac{\partial c(\mathcal{S})}{\partial p_i}=0 \tag{16}$$

which yields

$$\frac{c_{Ti}^m}{\Delta c_R r_{Di}}=\prod_{j\neq i}^{N_T}(1-p_j r_{Dj}) \tag{17}$$

or

$$\frac{c_{Ti}^m}{\Delta c_R r_{Di}}(1-p_i r_{Di})=\prod_{j=1}^{N_T}(1-p_j r_{Dj})=1-r_R^m(\mathcal{S}) \tag{18}$$

and further,

$$p_i=\frac{1}{r_{Di}}-\frac{\Delta c_R}{c_{Ti}^m}(1-r_R^m(\mathcal{S})) \tag{19}$$

Core to the testing strategy $\mathcal{S}$ is to determine the testing probability p_i of each testing item $\mathcal{T}_i$. (19) means that for the optimal testing strategy, the probability of testing item $\mathcal{T}_i$ being tested should be inversely proportional to its actual defective rate r_{Di} but is adjusted by its testing cost c_{Ti}^m, which is consistent with our intuitions.

4 Elucidating Factory L's Testing Strategy Under the General Framework

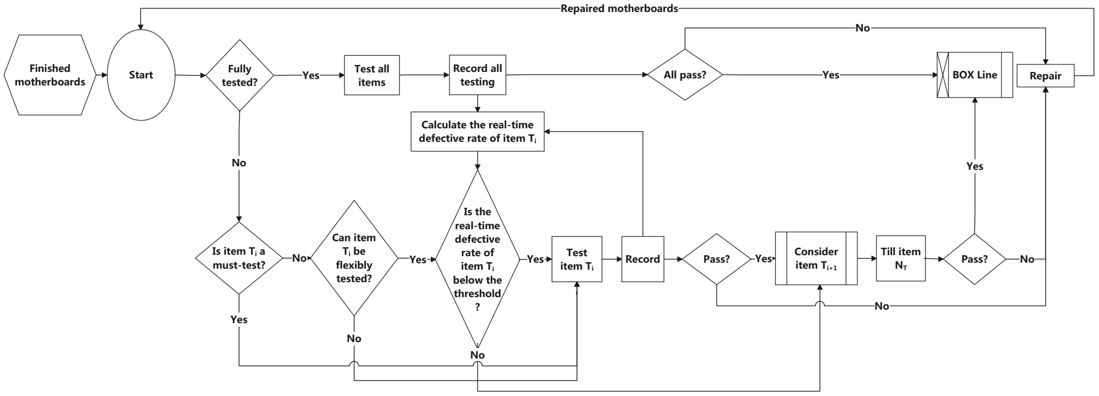

Figure 3: General framework of Factory L's motherboard functional testing strategy

The framework of the motherboard functional testing strategy of Factory L is depicted in Fig. 3. In this strategy, firstly, certain testing items are must-test due to their importance, i.e., being 100% tested regardless of other conditions; secondly, fully testing for a certain percentage of motherboards (typically 20%) are carried out, while for the left motherboards (typically 80%) the testing items are optional. An optional testing item is not tested, if 1) it is not a must-test item; 2) the estimated defective rate of this testing item is sufficiently low; and 3) the number of motherboards that can be used to estimate its defective rate is more than some threshold (typically 20000).

Let μ be the percentage of the motherboards being fully tested. Define σ_{Ti} as the indicator that shows whether the optional testing item $\mathcal{T}_i$ is tested, that is, $\sigma_{Ti} = 1$ if $\mathcal{T}_i$ is tested, and $\sigma_{Ti} = 0$ if $\mathcal{T}_i$ is not tested. σ_{Ti} is determined by whether the estimated defective rate r_{Ti} of testing item $\mathcal{T}_i$ exceeds certain threshold r^0_{Ti}, i.e.,

$$\sigma_{Ti} = \begin{cases} 1, & r_{Ti} \geq r^0_{Ti} \\ 0, & r_{Ti} < r^0_{Ti} \end{cases} \tag{20}$$

r_{Ti} can be estimated as follows,

$$r_{Ti} = \frac{N_{Ti}(\mu)}{\mu M} \tag{21}$$

where $N_{Ti}(\mu)$ is the number of defective testing item $\mathcal{T}_i$ being found among M motherboards with the fully testing percentage being μ. It is understood that r_{Ti} approaches r_{Di} with the increase of μ (and sufficiently large M).

The testing strategy of Factory L, denoted by $\mathcal{S}_0$, can then be represented as follows,

$$\mathcal{S}_0 : (p_i = \mu + \sigma_{Ti}(1 - \mu), i = 1, 2, \ldots, N_T) \tag{22}$$

From (13) and (22), the effective average cost of this testing strategy is

$$\begin{aligned} c(\mathcal{S}_0) =& \sum_{i=1}^{N_T} p_i c_{Ti}^m + \Delta c_R \prod_{i=1}^{N_T} (1 - p_i r_{Di}) \\ =& \sum_{\mathcal{T}_i \in \mathbb{T}_0} \mu c_{Ti}^m + \sum_{\mathcal{T}_i \in \mathbb{T}_1} c_{Ti}^m + \Delta c_R \prod_{\mathcal{T}_i \in \mathbb{T}_0} (1 - \mu r_{Di}) \prod_{\mathcal{T}_i \in \mathbb{T}_1} (1 - r_{Di}) \end{aligned} \tag{23}$$

where

$$\mathbb{T}_0 = \{\mathcal{T}_i : \sigma_{Ti} = 0\} \tag{24}$$

$$\mathbb{T}_1 = \{\mathcal{T}_i : \sigma_{Ti} = 1\} \tag{25}$$

5 A Specified Optimization Strategy Validating the General Framework and Model for Functional Testing

In this section, we specify the general optimization framework for functional testing and verify its effectiveness numerically.

5.1 A specified optimization strategy

Consider the optimization strategy for an already founded testing line. Suppose, under strategy $\mathcal{S}$, there are in total $q_A(\mathcal{S})$ testing machines, each costing u_A, and the cost for all other accessories is u_A^a. Then the average cost for each testing machine (with consideration of its accessories) is

$$u'_A(\mathcal{S}) = u_A + \frac{u_A^a}{q_A(\mathcal{S})} \tag{26}$$

Let the average usage life for the testing machines be l_A, then for testing item $\mathcal{T}_i$, its cost in (14) can be specified as follows,

$$c_{T_i}^m(\mathcal{S}) = \frac{u'_A(\mathcal{S})}{l_A} \bar{t}_{Ti} \tag{27}$$

Let the running time for the testing machines be w_l per week, the working time and wage for the workers are w_R per week and v_R, respectively. The average repair costs (time) for the defective motherboard and laptop are t_R^m and t_R^l, respectively. Then, the two costs can be specified as follows,

$$c_R^m = t_R^m \frac{v_R}{w_R} \tag{28}$$

$$c_R^l = t_R^l \frac{v_R}{w_R} \tag{29}$$

Note that $\bar{t}_T$ is the average allowed testing time for each motherboard using $q_A(\mathcal{S}_0(\mu_0))$ testing machines. With $q_A(\mathcal{S})$ testing machines, this time turns to $\frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))} \bar{t}_T$. Thus, the constraint in (7) now turns to

$$\sum_{i=1}^{N_T} p_i \bar{t}_{Ti} \le \frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))} \bar{t}_T \tag{30}$$

Let the workers for repairing defective motherboards and laptops be $n_R^m(\mathcal{S})$ and $n_R^l(\mathcal{S})$, respectively. In order to make sure the functional testing operation is not slower than the BOX line, other than the constraint in (30), it is better that the repairing operation is not too slow, i.e.,

$$\frac{r_R^m(\mathcal{S})t_R^m}{n_R^m(\mathcal{S})} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \tag{31}$$

and

$$\frac{[r_D - r_R^m(\mathcal{S})]t_R^l}{n_R^l(\mathcal{S})} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \tag{32}$$

From (13) and (30–32), the optimization problem turns to

$$\begin{aligned} &\min_{\mathcal{S}\in\mathbb{S}} c(\mathcal{S}) \\ \text{s.t. } &\sum_{i=1}^{N_T} p_i\bar{t}_{Ti} \leq \frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \\ &\frac{r_R^m(\mathcal{S})t_R^m}{n_R^m(\mathcal{S})} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \\ &\frac{[r_D - r_R^m(\mathcal{S})]t_R^l}{n_R^l(\mathcal{S})} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S})}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \end{aligned} \tag{33}$$

With the above, Factory L's strategy can also be more specified, as follows,

$$\begin{aligned} &\min_{\mu} c(\mathcal{S}_0(\mu)) \\ \text{s.t. } &\sum_{\mathcal{T}_i\in\mathbb{T}_0} \mu\bar{t}_{Ti} + \sum_{\mathcal{T}_i\in\mathbb{T}_1} \bar{t}_{Ti} \leq \frac{q_A(\mathcal{S}_0(\mu))}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \\ &\frac{r_R^m(\mathcal{S}_0(\mu))t_R^m}{n_R^m(\mathcal{S}_0(\mu))} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S}_0(\mu))}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \\ &\frac{[r_D - r_R^m(\mathcal{S}_0(\mu))]t_R^l}{n_R^l(\mathcal{S}_0(\mu))} \leq \frac{w_R}{w_l}\frac{q_A(\mathcal{S}_0(\mu))}{q_A(\mathcal{S}_0(\mu_0))}\bar{t}_T \end{aligned} \tag{34}$$

where

$$r_R^m(\mathcal{S}_0(\mu)) = 1 - \prod_{\mathcal{T}_i\in\mathbb{T}_0}(1-\mu r_{Di})\prod_{\mathcal{T}_i\in\mathbb{T}_1}(1-r_{Di}) \tag{35}$$

With specified parameters, the above optimization problem can be solved in the following process. First, select $q_A(\mathcal{S})$ such that $q_A(\mathcal{S}) \leq q_A(\mathcal{S}_0(\mu_0))$. Second, determine $n_R^m(\mathcal{S})$ and $n_R^l(\mathcal{S})$ for the convergent solution under the constraints. Finally, by trying different values of $q_A(\mathcal{S})$, one may determine the values of $q_A(\mathcal{S})$, $n_R^m(\mathcal{S})$ and $n_R^l(\mathcal{S})$, corresponding to the optimal objective value.

5.2 Validating the Specified Strategy

Due to business privacy requirement, we are not allowed to show the real data here. We confirm that our mathematical model as well as its solution have been tested effectively in the real production line. Here, to show the effectiveness of our approach, we adopt a modified data set from the real one, which should be sufficient for the validation purpose.

Table 1: Optimal solution vs. Existing solution

Testing Item	$\bar{t}_{Ti}$	r_{Ti}	$\mathcal{S}_0(\mu_0)$	$\mathcal{S}_0(\mu^*)$	$\mathcal{S}^*$
1	8.4132	0.0004	0.2000	0.0000	0.0001
2	5.6211	0.0001	0.2000	0.0000	0.0001
3	4.9850	0.0015	1.0000	1.0000	1.0000
4	6.6460	0.0001	0.2000	0.0000	0.0000
5	5.7984	0.0006	1.0000	1.0000	0.0002
6	3.3483	0.0019	1.0000	1.0000	1.0000
7	9.3973	0.0014	1.0000	1.0000	0.9998
8	5.2658	0.0009	1.0000	1.0000	0.9998
9	6.0082	0.0002	0.2000	0.0000	0.0001
10	9.0318	0.0001	0.2000	0.0000	0.0000
11	8.9430	0.0018	1.0000	1.0000	0.9999
12	5.6829	0.0001	0.2000	0.0000	0.0001
13	7.1624	0.0001	0.2000	0.0000	0.0000
14	1.4172	0.0010	1.0000	1.0000	1.0000
15	2.1517	0.0003	0.2000	0.0000	0.9992
16	8.7972	0.0018	1.0000	1.0000	0.9999
17	6.4684	0.0009	1.0000	1.0000	0.2608
18	8.8059	0.0002	0.2000	0.0000	0.0000
19	6.4338	0.0004	0.2000	0.0000	0.0001
20	9.8463	0.0001	0.2000	0.0000	0.0000
21	2.8260	0.0007	1.0000	1.0000	0.9999
22	2.5504	0.0016	1.0000	1.0000	1.0000
23	8.9067	0.0002	0.2000	0.0000	0.0000
24	4.7249	0.0001	0.2000	0.0000	0.0001
25	3.9446	0.0007	1.0000	1.0000	0.9998
26	8.0647	0.0006	1.0000	1.0000	0.0001
27	3.0057	0.0006	1.0000	1.0000	0.9998
28	3.9100	0.0012	1.0000	1.0000	1.0000
29	9.9752	0.0007	1.0000	1.0000	0.0001
30	8.6386	0.0016	1.0000	1.0000	0.9999

Let the number of testing items be $N_T = 30$, and $M = 10^5$ testing data. The testing time for each testing item is a random number between $[0.2, 10]$, which yields an average testing time for a motherboard of 186.7708 seconds, close to the real value. The native defective rate of each testing item is random between $[0.998, 1]$.

Other parameters are determined as follows. The threshold of defective rate in Factory L is $r_{Ti}^0 = 0.0005$. Out of all motherboards, $\mu_0 = 20\%$ are fully tested. It has $q_A(\mathcal{S}_0(\mu_0)) = 24$ testing machines, with each costing $u_A = 10^5$ CNY, and $u_A^a = 2 \times 10^6$ CNY. The average usage life $l_A = 5$. The running time $w_l = 168$ hours per week. The number of workers $n_R^m(\mathcal{S}_0(\mu_0)) = 10$ and $n_R^l(\mathcal{S}_0(\mu_0)) = 2$. The wages $v_R = 1200$. The work time $w_R = 40$ hours per week. The average repair time $t_R^m = 900$ seconds, and $t_R^l = 960$ seconds.

Solve the optimization problem in (33) and (34) using the `fmincon` function in MATLAB, with the optimization interval being $[0, 1]$, and the initial value being random within this interval.

Denote the strategy designed based on Factory L's existing strategy $\mathcal{S}_0(\mu_0)$ be $\mathcal{S}_0(\mu^*)$, and the optimal one be $\mathcal{S}^*$. The results are shown in Tab. 1, where the percentage of fully tested motherboards of $\mathcal{S}_0(\mu^*)$ is $\mu^* = 4.4448 \times 10^{-6}$.

From Tab. 1, it is seen that strategy $\mathcal{S}_0(\mu^*)$ requires hardly any testing for testing items with high yield rate; $\mathcal{S}^*$ effectively balances between the testing time and defective rate of all testing items, with the focus on those items that can either be tested fast or are of high defective rate.

Let Δ_0^* and Δ^* be the relative change between the two strategies $\mathcal{S}_0(\mu^*)$ and $\mathcal{S}^*$ (unit: percentage), and $t^m(\mathcal{S})$ be the average testing time of strategy $\mathcal{S}$, i.e.,

$$t^m(\mathcal{S}) = \sum_{i=1}^{N_T} p_i \bar{t}_{Ti} \tag{36}$$

Table 2: Other optimization indexes

Index	Unit	$\mathcal{S}_0(\mu_0)$	$\mathcal{S}_0(\mu^*)$	$\mathcal{S}^*$	$\Delta_0^*(\%)$	$\Delta^*(\%)$
$c(\mathcal{S})$	CNY/board	0.6242	0.6036	0.5739	-3.2934	-8.0558
$t^m(\mathcal{S})$	seconds/board	115.2229	97.3363	70.8662	-15.5235	-38.4964
$q_A(\mathcal{S})$	set	24	19	19	-20.8333	-20.8333
$n_R^m(\mathcal{S}) + n_R^l(\mathcal{S})$	person	12	14	14	16.6667	16.6667
$r_R^m(\mathcal{S})$	\	0.0043	0.0000	0.0170	-99.9978	292.1079

From Tab. 2, it is shown that strategies $\mathcal{S}_0(\mu^*)$ and $\mathcal{S}^*$ increase a certain amount of repair workers, but optimize at the indexes of average effective cost, average testing time and number of required testing machines. In particular, strategy $\mathcal{S}^*$ decreases the testing time by as more as 30%, as well as the cost. This means, by applying the designed strategy $\mathcal{S}^*$ to the production line, more than 1/3 testing time can be saved, which is of great economic value.

In addition, it can be seen from the last row of Table 2 that strategy $\mathcal{S}_0(\mu^*)$ has a significantly lower repair rate for defective motherboards due to its low testing percentage, compared to Factory L's testing strategy. Conversely, strategy $\mathcal{S}^*$ takes into account various factors such as the inherent defective rate of testing items, average testing cost, and average repair cost, and assigns different testing percentages to each testing item. As a result, it achieves a significantly higher repair rate for defective motherboards compared to Factory L's strategy. If the designed strategy $\mathcal{S}^*$ is implemented in actual testing production lines, it can greatly improve the repair rate of defective motherboards.

6 Conclusions

Functional testing in laptop manufacturing is systematically addressed from the academic community for the first time. This includes the technical understanding of the key principles in functional testing, the mathematical modelling of the general framework, the practical elucidating of the existing strategy of a leading factory, and finally a specified optimization strategy outperforming existing ones. This work is a perfect example of how academia can contribute to industry, provided a real deep collaboration relationship is built between both sides.

Conflict of Interest

KANG Yu and ZHAO Yun-Bo are editorial board members for Journal of Systems Science and Complexity and were not involved in the editorial review or the decision to publish this article. All authors declare that there are no competing interests.

References

[1] Ait-El-Cadi, A., Gharbi, A., Dhouib, K. and Artiba, A. [2021], 'Integrated Production, Maintenance and Quality Control Policy for Unreliable Manufacturing Systems Under Dynamic Inspection', *International Journal of Production Economics* **236**, 108140.

[2] Bogle, I. D. L. [2017], 'A Perspective on Smart Process Manufacturing Research Challenges for Process Systems Engineers', *Engineering* **3**(2), 161–165.

[3] Evjemo, L. D., Gjerstad, T., Grøtli, E. I. and Sziebig, G. [2020], 'Trends in Smart Manufacturing: Role of Humans and Industrial Robots in Smart Factories', *Current Robotics Reports* **1**(2), 35–41.

[4] Jiyuan, Z., Baicun, W., Liu, M. and Yuan, Z. [2018], 'Brief Analysis on Three Basic Paradigms of Intelligent Manufacturing', *Chinese Journal of Engineering Science* **20**(4), 13.

[5] Jwo, J.-S., Lin, C.-S. and Lee, C.-H. [2021], 'Smart Technology–Driven Aspects for Human-in-the-Loop Smart Manufacturing', *The International Journal of Advanced Manufacturing Technology* **114**(5), 1741–1752.

[6] Kundu, S., Mak, T. and Galivanche, R. [2004], Trends in Manufacturing Test Methods and Their Implications, *in* '2004 International Conferce on Test', pp. 679–687.

[7] Li, D., Wang, L. and Huang, Q. [2019], A Case Study of Sos-Svr Model for Pcb Throughput Estimation in Smt Production Lines, *in* '2019 International Conference on Industrial Engineering and Systems Management (IESM)', pp. 1–6.

[8] Li, Y., Wang, K., Kang, Y., Zhao, Y. and Bai, P. [2023], Board-level Functional Test Selection Based on Fault Tree Analysis, *in* '2023 6th International Symposium on Autonomous Systems (ISAS)', IEEE, Nanjing, China, pp. 1–6.

[9] Liang, Y., Sun, Z., Song, T., Chou, Q., Fan, W., Fan, J., Rui, Y., Zhou, Q., Bai, J., Yang, C. and Bai, P. [2022], 'Lenovo Schedules Laptop Manufacturing Using Deep Reinforcement Learning', *INFORMS Journal on Applied Analytics* **52**(1), 56–68.

[10] Qian, F. [2017], 'Smart and Optimal Manufacturing: The Key for the Transformation and Development of the Process Industry', *Engineering* **3**(2), 151–1.

[11] Qian, F., Zhong, W. and Du, W. [2017], 'Fundamental Theories and Key Technologies for Smart and Optimal Manufacturing in the Process Industry', *Engineering* **3**(2), 154–160.

[12] Ribeiro, J., Lima, R., Eckhardt, T. and Paiva, S. [2021], 'Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review', *Procedia Computer Science* **181**, 51–58.

[13] Wang, B., Tao, F., Fang, X., Liu, C., Liu, Y. and Freiheit, T. [2021], 'Smart Manufacturing and Intelligent Manufacturing: A Comparative Review', *Engineering* **7**(6), 738–757.

[14] Wang, C., Wang, L., Xue, Y. and Li, R. [2022], 'Revealing Spatial Spillover Effect in High-Tech Industry Agglomeration from a High-Skilled Labor Flow Network Perspective', *Journal of Systems Science and Complexity* **35**(3), 839–859.

[15] Yang, J., Li, S., Wang, Z., Dong, H., Wang, J. and Tang, S. [2020], 'Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges', *Materials* **13**(24), 5755.

[16] Yang, YANG., Letian, ZHAO., Siyi, CHEN. and Dan, YU. [2023], 'Fiducial Approach for the Storage Reliability Assessment of Complex Repairable Systems', *Journal of Systems Science and Complexity* .