数理モデル化と問題解決 35－6
（２００１．６．２６）

不完全な部品の組合せによる合格率向上について

嘉村友作　中森眞理雄
東京農工大学
工学研究科 電子情報工学専攻　工学部 情報コミュニケーション工学科
〒184-8588 東京都小金井市中町2-24-16

概要

複数の部品から構成されるシステムの合格率向上について考察する．各部品のために複数の部品個体が提供される．各個体には固有の誤差があり，これを組合せて複数のシステムを作る．本論文では，これらのシステムの中の最大誤差を最も小さくする組合せ方を提案する．

この問題は整数計画問題として定式化されるが，本論文では生産現場で簡単に実行できるアルゴリズムを取り上げ，統計的に考察する．また，システムを構成する部品数が 2 と 3 の場合について考える．

部品数が 2 の場合には，k 番目に良い個体と k 番目に悪い個体を組合せるのが最適な組合せ方であることを示す．さらに，与えられた許容誤差値に対して合格するシステムの数が最大になる方法を示す．

部品数が 3 の場合については，最適な組合せ方のあらましを示す．

Combining Imperfect Components to Minimize the System's Error

Yuusaku Kamura and Mario Nakamori
Department of Computer Science
Tokyo A & T University
Koganei,Tokyo 184-8588,Japan

Abstract

In the present paper we consider the problem of minimizing systems' errors, where a system consists of many components. Many individuals are supplied for each component, so we are going to combine individuals such that the worst error of the combination is minimized.

This problem is formulated as an integer programming problem. We consider a simple algorithm which we can be carried out in a production site. We consider easier cases where the system consists of two or three components.

When the system consists of two components, we present a method of an optimal combination. We also show by probabilistic analysis how to maximize the number of systems whose error is less than a given bound.

When the system consists of three components, we outline an optimal method.

1 Introduction

The background of the problem with which we are concerned in this paper is to combine lenses for a semiconductor exposing equipment. Each equipment includes about 30 lenses, which are supplied in groups at fixed time intervals. Although lenses are made carefully, every lens has some error. Since lenses are very expensive it is required to make the number of rejected lenses as small as possible. Sometimes we have to combine "good" lenses with "bad" ones in order not to reject too much.

We suppose that the equipments consist of lenses (hereafter, we call components) $A_1, A_2, \ldots, A_m$ and for each component A_i there are n individuals. So we have n^m combinations in total. We denote the error of a combination C by $E(C)$. In order to make the analysis simple, we make an assumption:

Assumption 1 The error $E(C)$ of a combination C is the sum of the errors $\varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_m$, where ε_i is the error of the individual chosen for the component A_i.

We call a set of n combinations a *selection* when every individual appears only once. There are $(n!)^{m-1}$ selections. We denote the family of all selections by $\mathcal{S}$. For a selection S we call the maximum value of the error of the n combinations of S *the error of S*, and denote it by $E[S]$. Thus, $E[S] = \max_{C \in S} E(C)$. We would like to find a selection $\tilde{S}$ whose error is less than a given bound E_0, i.e., $E[\tilde{S}] < E_0$.

In Section 2 we formulate this problem as an integer programming problem.

In Section 3 we consider the simplest case $m = 2$, and show how to find the optimal selection for which the error is minimum. We analyse the error of the optimal selection statistically.

When a bound for the system's error is given, we can calculate the expected number of combinations that are under the bound. We also show a method of increasing the number of combinations falling under the bound when the bound is tight.

In Section 4 we consider the case $m \geq 3$ and propose a method of combining such that the maximum error is minimized and outline a systematic approach to get the optimal solution.

2 Formulation of the problem

We slightly modify the above statement of the problem as follows: find a selection $\hat{S}$ whose error is the minimum among all selections, i.e., $E[\hat{S}] = \min_{S \in \mathcal{S}} E[S]$. This problem is stated as an integer programming problem as follows:

Problem 1 Minimize z

subject to

$$\sum_{i_m} \cdots \sum_{i_{k+1}} \sum_{i_{k-1}} \cdots \sum_{i_1} x_{i_1 i_2 \ldots i_m} = 1 \qquad (i_k = 1, 2, \ldots, n) \tag{1}$$

$$x_{i_1 i_2 \ldots i_m} = 0 \quad \text{or} \quad 1 \tag{2}$$

$$\sum_{i_m} \cdots \sum_{i_2} (a_{i_1} + a_{i_2} + \cdots + a_{i_m}) x_{i_1 i_2 \ldots i_m} \leq z \qquad (i_1 = 1, 2, \ldots, n) \tag{3}$$

Constraint (2) is essential, since otherwise we would usually obtain a noninteger solution.

3 The simplest case $m = 2$

We formulated the problem as a mixed integer programming problem(MIP) in Section 2. In the present paper, however, we deal with this problem from a different point of view.

In this section we consider a simple method which we can apply easily in producing sites. Then we analyse the method statistically.

3.1 The algorithm

If the equipment consists of only two components, i.e., $m = 2$, the problem becomes simple. First, note that

Lemma 1 If $a_1 < a_2 < \cdots < a_n$ and $b_1 < b_2 < \cdots < b_n$, then

$$\max(a_1 + b_n, a_2 + b_{n-1}, \ldots, a_n + b_1) \leq \max(a_1 + b_{\pi_1}, a_2 + b_{\pi_2}, \ldots, a_n + b_{\pi_n}) \tag{4}$$

for any permutation $(\pi_1, \pi_2, \ldots, \pi_n)$ on $(1, 2, \ldots, n)$. ∎

The algorithm for the optimal selection can be derived directly from this lemma:

Algorithm 1

Step1 : Sort individuals of component A_1 by the value.

Step2 : Sort individuals of component A_2 by the value.

Step3 : Combine the best individual of A_1 with the worst individual of A_2; combine the second best of A_1 with the second worst of A_2; combine the third best of A_1 with the third worst of A_2, and so on. ■

3.2 Statistical analysis

In order to perform a statistical analysis we assume that the error of each individual follows a uniform distribution between 0 and 1, and those values are distinct each other. Then we can derive the following theorems.

Theorem 1 The mean and the variance of the combination of the kth best individual of the component A_1 and the kth worst individual of the component A_2 are 1 and

$$\frac{2k(n+1-k)}{(n+2)(n+1)^2}$$

respectively. ■

Theorem 2 Algorithm 1 makes the means of all combinations equal and the maximum of variances the minimum. ■

3.3 The probability of falling within a given bound

We now calculate the probability that the system's error is less than the given bound E_0. We denote the error of the n individuals of the components A_1 and A_2 as $X_1, X_2, \ldots, X_n$ and $Y_1, Y_2, \ldots, Y_n$ in ascending order, respectively. According to Algorithm 1, the error of the system(combination) consisting of the kth best individual of A_1 and the kth worst individual of A_2 is $X_k + Y_{n-k+1}$. So we have

$$\begin{aligned} &\text{Prob}(X_k + Y_{n-k+1} \leq E_0) \\ &= 1 - \frac{(n!)^2}{(n-k)!(k-1)!} \int_0^1 S(y)dy \quad (5) \end{aligned}$$

where

$$S(y) = \sum_{i=1}^{k} \frac{y^{n-k}(1-y)^{k-1}(E_0-y)^{n-i}(1-E_0+y)^{n-k+i}}{(n-k+i)!(k-i)!}.$$

3.4 Improving the probability and the numerical experiment

In order to increase the probability, we have only to remove in turn the worst individuals of A_1 and A_2 respectively. After their removal, combining the remaining individuals for A_1 and A_2 in the same way yields a higher probability that the combination's error is less than E_0, while the total number of combinations decreases.

Example 1 Let $n = 100$ and $E_0 = 1.014$. The given bound E_0 is equal to the sum of the mean and the standard deviation of $X_1 + Y_{100}$. We seek the critical point at which the number of combinations falling within the bound is the maximum.

In result, we can find effect of removing indivuduals as given in Table 1, where r is the number the worst individuals removed from consideration for each of A_1 and A_2 and passing is the number of the number of combinations falling within the bound. From Table 1 we can conclude the critical point is 10.

Table 1: Effect of removing individuals

r	Value passing
0	61.40
5	82.56
10	87.19
11	87.01
12	86.61

4 The case $m \geq 3$

For $m \geq 3$, we do not have an exact algorithm like Algorithm 1. But we outline an approach for this case here. Again, we assume that each individual follows a uniform distribution between 0 and 1.

4.1 Fundamental case $n = 3$

For the case $m = 3$ and $n = 3$, namely, $A_1 : X_1, X_2, X_3$; $A_2 : Y_1, Y_2, Y_3$; $A_3 : Z_1, Z_2, Z_3$, we have the following theorem.

Theorem 3 In order to minimize the variances when $m = n = 3$, it is optimal to combine individuals such that all suffixes 1, 2, and 3 appear in each combination, that is, (X_1, Y_2, Z_3), (X_2, Y_3, Z_1), (X_3, Y_1, Z_2). ■

4.2 Case $m = 3$ and $n > 3$

We now extend the result of Section 4.1 for the case $n > 3$. For the sake of simplicity, we assume that $n = 3p\,(p = 2, 3, \ldots)$.

Our proposed idea is as follows : divide each set of components A_1, A_2 , and A_3, into three blocks of the same size. Let's denote these by A_{11}, A_{12}, A_{13}; A_{21}, A_{22}, A_{23}; A_{31}, A_{32}, A_{33}. Combine the blocks following Theorem 3 such as (A_{11}, A_{22}, A_{33}), (A_{12}, A_{23}, A_{31}), (A_{13}, A_{21}, A_{32}). Continue this operation recursively until each block has three elements.

4.3 The general case $m > 3$ and $n > 3$

For $m > 3$, we slightly extend the method considered in Section 4. When $m = n$, we simply combine the components like the method outlined in Section 4.1. When $m < n$, we first divide each component into m blocks of the same size, and then we combine the blocks as show in Section 4.2. We apply this procedure recursively until each block has m elements.

5 Further problems

In the present paper, we have assumed that the system's error is the sum of each individual's error and all errors are scalar. In most cases, however, errors are not scalar but vector. We have to extend our results to the case of error vectors.

Also, we have assumed that the components' error follow the uniform distribution. We have to extend to other types of distributions.

We also need to compare the solution obtained by our method and the one obtained by solving the MIP directly.

These problems are left for further research.

6 Conclusions

In the present paper, we dealt with the problem to improve the system's error. First we considered the case when the system consists of two components and assumed the individual's error follows a uniform distribution between 0 to 1. Then we demonstrated a method to minimize the system's error with an example.

We also considered the case when the system consists of three components. We outlined a recursive method of making the systems(combinations) such that the maximum variance is minimized.

The method of maximizing the number of systems(combinations) that fall under the bound for the case $m \geq 3$ is left for further research.

References

[1] Kiyonori Kunisawa. *Probability Theory and Its Applications*(in Japanese), Iwanami Shoten, 1982.

[2] Kiyonori Kunisawa. *Modern Probability Theory*(in Japanese), Iwanami Shoten, 1951.

[3] R.S.Burington and D.C.May. *Handbook of Probability and Statistics with Tables,* McGraw-Hill, 1970.

[4] M.Gen and R.Cheng *Genetic Algorithms & Engineering Design,* John Wiley & Sons, 1997.

[5] R.L.Haupt and S.E.Haupt *Practical Genetic Algorithms,* John Wiley & Sons, 1998.