



An optimal mixed policy of inspection and burn-in and the optimal production quantity



出處：Applied Soft Computing 7 (2007) 211–228

作者：Int. J. Production Economics 105 (2007) 483–491

報告者：鄭欣卉

指導老師：童超塵教授



Contents

- Introduction
- Assumptions and the optimization problem
- The optimal plan
- A numerical example
- Conclusion

Introduction

- 為了達到製造商的品質要求，供應商必須從貨批中篩選出不良品。
- Tang and Tang (1994)調查文獻發現，產品品質檢查是篩選程序中一個很重要的方法。
- 供應商實際上面臨兩個問題：
 - 錯誤拒絕良品→需再生產更多產品→成本增加(型I誤差)
 - 錯誤接受壞產品→無法滿足品質要求→拒收(型II誤差)

Introduction

- 預燒測試也廣泛被用來篩選產品，其在足夠時間下的預燒能夠保證出貨批能夠滿足品質要求，但是因為浪費時間，所以成本高。
- 因此預燒測試要持續多久將是一個問題。
- 從經濟和品質觀點，兩種篩選方法沒辦法完全被取代。
- 本研究將要探討要如何使篩選在低成本且品質達到要求。



Introduction

- 本研究將進行發展檢查和預燒的最佳混合策略。
- 使用 average outgoing quality (AOQ) 來衡量績效。
- 在品質要求條件下達到最大期望利潤
 - 生產量
 - 檢查數量、預燒數量、不需檢查預燒數量
 - 如果需要預燒其最佳預燒時間

Assumptions and the optimization problem

■ Assumptions

- 製造商訂單每期數量： Q_0 (每週或每月)
- AOQ設置為 p
- 每個零件單價為 C 元
- 製造商付款後，零件配送到製造商
- 生產批量的平均不良率： $p_0(>p)$
- 每批生產量： Q
- 採用檢查和預燒兩種方法進行篩選
- 型I誤差： α ；型II誤差： β

Assumptions and the optimization problem

- 為了達到最大化利潤，供應商面臨的決策問題
 - 生產多少(Q)?
 - 檢驗數量(Q_s)?預燒數量(Q_b)?不檢驗也不預燒數量(Q_n)?
 - 在需要預燒時，其預燒的持續時間為多長(t_b)?

Assumptions and the optimization problem

■ 供應商的混合政策 — notations

- AOQ_v : 出貨批的平均出貨品質
- N_a : 經過檢查和預燒測試後的期望接收數量
(不管是正確接受或不正確接受)
- N_d : N_a 中期望不良品數

Assumptions and the optimization problem

■ The optimization problem

■ $P_v = P_v(Q, Q_s, Q_b, t_b)$: 每期供應商的期望利潤

■ $AOQ_v = N_d / N_a$

■ Max $P_v = P_v(Q, Q_s, Q_b, t_b)$

S.t $AOQ_v = N_d / N_a \leq p$

$N_a \geq Q_0$

$Q, Q_s, Q_b \in N = \{1, 2, \dots\}$

$t_b \geq 0$.

$t_b \leq t_0$ (供應商預先確定的容許界線) (1)

The optimal plan

- 解決最佳模型包含兩個主要步驟
 - 計算 N_a , N_d 和 AOQ_v
 - $P_v(Q, Q_s, Q_b, t_b)$

The optimal plan — 計算 N_a , N_d 和 AOQ_v

- First— Q_s
- Next — Q_b
 - $F(t)$ ：不良品壽命分配的CDF
 - $G(t)$ ：良品壽命分配的CDF
- Finally — Q_n

The optimal plan — 計算 N_a , N_d 和 AOQ_v

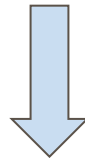
■ Table 1 Ten categories of items after inspection and burn-in tests

Q_s	$Q_s * p_0 * \beta$	expected number of false acceptance of defectives → False acceptance
	$Q_s * (1 - \beta) * p_0$	expected number of correct classification of defectives
	$Q_s * (1 - p_0) * (1 - \alpha)$	expected number of correct classification of nondefectives → Correct acceptance
	$Q_s * \alpha * (1 - p_0)$	expected number of false rejection of nondefectives
Q_b	$Q_b * p_0 * F(t_b)$	expected number of correct classification of defectives
	$Q_b * p_0 * [1 - F(t_b)]$	expected number of false acceptance of defectives → False acceptance
	$Q_b * (1 - p_0) * G(t_b)$	expected number of false rejection of nondefectives
	$Q_b * (1 - p_0) * [1 - G(t_b)]$	expected number of correct classification of nondefectives → Correct acceptance
Q_n	$Q_n * p_0$	expected number of defectives
	$Q_n * (1 - p_0)$	expected number of nondefectives

The optimal plan — 計算 N_a , N_d 和 AOQ_v

$$N_d = Q_s p_0 \beta + Q_b p_0 [1 - F(t_b)] + (Q - Q_s - Q_b) p_0$$

$$N_a = Q - (1 - \beta) Q_s p_0 - Q_s \alpha (1 - p_0) - Q_b p_0 F(t_b).$$



$$AOQ_v = \frac{Q_s p_0 \beta + Q_b p_0 [1 - F(t_b)] + (Q - Q_s - Q_b) p_0}{Q - (1 - \beta) Q_s p_0 - Q_s \alpha (1 - p_0) - Q_b p_0 F(t_b)}.$$

The optimal plan — $P_v(Q, Q_s, Q_b, t_b)$

■ $P_v(Q, Q_s, Q_b, t_b) = \text{收入} - \text{總成本}$

■ $\text{收入} = CQ_0$

■ 成本

■ $C_d * Q$

■ $C_s * Q_s$

■ $C_b t_b * Q_b$

■ 型I誤差 $\rightarrow C_r \alpha (1 - p_0) * Q_s$

$$C_r = C$$

The optimal plan — $P_v(Q, Q_s, Q_b, t_b)$

■ 期望總利潤：

$$P_v = CQ_0 - [C_d Q + C\alpha(1 - p_0)Q_s + C_s Q_s + C_b t_b Q_b]. \quad (3)$$

The optimal plan — The optimization model

Minimize $C_d Q + C\alpha(1 - p_0)Q_s + C_s Q_s + C_b t_b Q_b$

AOQ_v

Subject to $\frac{Q_s p_0 \beta + Q_b p_0 [1 - F(t_b)] + (Q - Q_s - Q_b) p_0}{Q - (1 - \beta) Q_s p_0 - Q_s \alpha (1 - p_0) - Q_b p_0 F(t_b)} \leq p,$

$Q - (1 - \beta) Q_s p_0 - Q_s \alpha (1 - p_0) - Q_b p_0 F(t_b) \geq Q_0,$

$Q_s + Q_b \leq Q,$

Na

$t_b \leq t_0$

$Q_s, Q_b \in \mathbf{N} \cup \{0\},$

$Q \in \mathbf{N},$

$t_b \geq 0.$

(4)

The optimal plan — Algorithm 1

- Q, Q_s, Q_b 的整數規劃問題，得到近似解 $(Q^*, Q_s^*, Q_b^*, t_b^*)$

Step1：分割 $(0, t_0)$ 區間為 m 個子區間，
設 $t_b(k) = k/m * t_0, k = 0, 1, 2, \dots, m$.

The optimal plan — Algorithm 1

Step 2 : 固定 k ，透過解整數方程式可得到 (Q, Q_s, Q_b) 的 $(Q^*, Q_s^*, Q_b^*, t_b^*)$ 相對最佳值

$$\begin{aligned} \text{Minimize} \quad & C_d Q + [C\alpha(1 - p_0) + C_s]Q_s \\ & + C_b t_b(k)Q_b \end{aligned}$$

$$\begin{aligned} \text{Subject to} \quad & [p_0\beta - p_0 + (1 - \beta)pp_0 + \alpha p(1 - p_0)]Q_s + [p_0pF(t_b(k)) \\ & - p_0F(t_b(k))]Q_b \leq (p - p_0)Q, \\ & Q - (1 - \beta)p_0Q_s - \alpha(1 - p_0)Q_s - p_0F(t_b(k))Q_b \geq Q_0, \\ & Q_s + Q_b \leq Q, \\ & Q_s, Q_b \in \mathbf{N} \cup \{0\}, \\ & Q \in \mathbf{N}. \end{aligned}$$

The optimal plan — Algorithm 1

Step 3 : 如果 t_b^* 滿如下式即得到 $(Q^*, Q_s^*, Q_b^*, t_b^*)$
近似最佳解

$$P_v(Q^*, Q_s^*, Q_b^*, t_b^*) = \min_{0 \leq k \leq m} P_v(Q(k), Q_s(k), Q_b(k), t_b(k)).$$

The optimal plan — Algorithm 2

■ 使用 $Q = r_q * Q_0$, $Q_s = r_s * Q_0$, and $Q_b = r_b * Q_0$.

變數轉換，改寫模式(4)為線性方程式

$$\begin{aligned} \text{Minimize } & Q_0 * \{C_d r_q + [C\alpha(1 - p_0) + C_s] * r_s \\ & + C_b t_b r_b\} \end{aligned} \quad (6)$$

$$\begin{aligned} \text{Subject to } & [p_0\beta - p_0 + (1 - \beta)pp_0 + \alpha p(1 - p_0)]r_s \\ & + [p_0pF(t_b) - p_0F(t_b)]r_b \leq (p - p_0)r_q, \\ & r_q - [(1 - \beta)p_0 + \alpha(1 - p_0)]r_s \\ & - p_0F(t_b)r_b \geq 1, \\ & r_s + r_b \leq r_q, \\ & t_b \leq t_0, \\ & r_q, r_s, r_b, t_b \geq 0. \end{aligned}$$



The optimal plan — Algorithm 2

Step 1 : $P_v(r_q, r_s, r_b, t_b) = P_v(Q, Q_s, Q_b, t_b)$

Step 2 : 分割 $(0, t_0)$ 區間為 m 個子區間，
設 $t_b(k) = k/m * t_0, k = 0, 1, 2, \dots, m$.

The optimal plan — Algorithm 2

Step 3 : 固定 k ，透過解線性方程式可得到
 (r_q, r_s, r_b) 的 $(r_q(k), r_s(k), r_b(k))$ 相對最佳值

$$\begin{aligned} \text{Minimize} \quad & C_d r_q + [C\alpha(1 - p_0) + C_s] * r_s \\ & + C_b t_b(k) r_b \\ \text{Subject to} \quad & [p_0\beta - p_0 + (1 - \beta)pp_0 + \alpha p(1 - p_0)]r_s + [p_0 p F(t_b(k)) \\ & - p_0 F(t_b(k))]r_b \leq (p - p_0)r_q, \\ & r_q - [(1 - \beta)p_0 + \alpha(1 - p_0)]r_s - p_0 F(t_b(k))r_b \geq 1, \\ & r_s + r_b \leq r_q, \\ & t_b \leq t_0, \\ & r_q, r_s, r_b, t_b \geq 0. \end{aligned}$$

The optimal plan — Algorithm 2

Step 4 : 如果 t_b^* 滿如下式即得到 $(Q^*, Q_s^*, Q_b^*, t_b^*)$
近似最佳解

$$P_v(r_q^*, r_s^*, r_b^*, t_b^*) = \min_{0 \leq k \leq m} P_v(r_q(k), r_s(k), r_b(k), t_b(k)).$$

Step 5 : 帶回 $Q = r_q * Q_0$, $Q_s = r_s * Q_0$, and $Q_b = r_b * Q_0$.

就可以得到 Q^*, Q_s^*, Q_b^*

The optimal plan — Algorithm 2

- 通常假設 r_q, r_s, r_b 為分數來滿足 Q, Q_s, Q_b 整數限制，取 Q^*, Q_s^*, Q_b^*, t_b^* 最佳解為最接近的整數，因此無法避免有誤差，但是可藉由夠大的 Q_0 來降低誤差。
- 相較於Algorithm 1，Algorithm 2不需要因為訂購量不同再重新計算。

A numerical example

- 根據Nelson(1990)得知電子產品壽命近似lognormal 分配。
- 假設
 - 良品壽命： $\tau_g \sim \text{lognormal}(\mu_g, \sigma^2)$
 - 不良品壽命： $\tau_d \sim \text{lognormal}(\mu_d, \sigma^2)$
 - Q_0 : 3,000/month
 - C : 3 dollars per item
 - C_s : 0.1 dollars per inspection
 - C_d : 1 dollars per item
 - t_0 : 300 h
 - $(\mu_d, \sigma^2) = (4.2, 1)$
 - $m=1000$

A numerical example

- Table 2 The optimal policies Q^*, Q_s^*, Q_b^*, t_b^* and the profits $P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$ under various combinations of (p_0, p, α, β)

$(p_0, p, \alpha, \beta, C_b)$	$(Q^*, Q_s^*, Q_b^*, t_b^*)$	$P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$
(0.10, 0.02, 0.05, 0.10, 0.002)	(3394, 2546, 848, 84.3)	4864.750
(0.10, 0.04, 0.05, 0.10, 0.002)	(3217, 347, 2870, 84.6)	5215.458
(0.10, 0.06, 0.05, 0.10, 0.002)	(3133, 0, 3121, 55.5)	5520.194
(0.10, 0.08, 0.05, 0.10, 0.002)	(3067, 0, 1752, 49.2)	5760.947
(0.10, 0.02, 0.05, 0.10, 0.005)	(3424, 3137, 0, 0)	4839.222
(0.10, 0.04, 0.05, 0.10, 0.005)	(3318, 2353, 0, 0)	5129.416
(0.10, 0.06, 0.05, 0.10, 0.005)	(3212, 1569, 0, 0)	5419.616
(0.10, 0.08, 0.05, 0.10, 0.005)	(3106, 784, 0, 0)	5709.811
(0.10, 0.02, 0.05, 0.15, 0.002)	(3415, 2963, 452, 99.6)	4798.907
(0.10, 0.04, 0.05, 0.15, 0.002)	(3201, 13, 3188, 91.5)	5212.905
(0.10, 0.06, 0.05, 0.15, 0.002)	(3133, 0, 3121, 55.5)	5520.188
(0.10, 0.08, 0.05, 0.15, 0.002)	(3067, 0, 1752, 49.2)	5760.945
(0.10, 0.02, 0.05, 0.15, 0.005)	(3433, 3333, 0, 0)	4783.415
(0.10, 0.04, 0.05, 0.15, 0.005)	(3325, 2500, 0, 0)	5087.502
(0.10, 0.06, 0.05, 0.15, 0.005)	(3217, 1667, 0, 0)	5391.670
(0.10, 0.08, 0.05, 0.15, 0.005)	(3108, 833, 0, 0)	5695.838
(0.10, 0.02, 0.10, 0.10, 0.002)	(3267, 0, 3266, 164.4)	4659.548
(0.10, 0.04, 0.10, 0.10, 0.002)	(3200, 0, 3198, 91.8)	5212.829
(0.10, 0.06, 0.10, 0.10, 0.002)	(3133, 0, 3121, 55.5)	5520.193
(0.10, 0.08, 0.10, 0.10, 0.002)	(3067, 0, 1752, 49.2)	5760.941

A numerical example

■ Result 1 :

■ Optimal

$(p_0, p, \alpha, \beta, C_b)$	$(Q^*, Q_s^*, Q_b^*, t_b^*)$	$P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$
(0.10, 0.02, 0.05, 0.10, 0.002)	(3394, 2546, 848, 84.3)	4864.750

■ Pure

Test	$(Q^*, Q_s^*, Q_b^*, t_b^*)$	$P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$
Inspection	(3469, 3469, 0, 0)	4715.785
Burn-in	(3267, 0, 3267, 164.3)	4659.464

A numerical example

■ Result 2 :

■ Optimal

$(p_0, p, \alpha, \beta, C_b)$	$(Q^*, Q_s^*, Q_b^*, t_b^*)$	$P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$
$(0.15, 0.02, 0.10, 0.15, 0.005)$	$(3628, 1691, 1937, 280.2)$	2057.878

■ pure

AOQ_v=0.02905>0.02

Test	$(Q^*, Q_s^*, Q_b^*, t_b^*)$	$P_v(Q^*, Q_s^*, Q_b^*, t_b^*)$
Inspection	No solution	—
Burn-in	$(3459, 0, 3459, 220.75)$	-32637.71

A numerical example

■ 從Table 2 發現

- $p_0 \uparrow$ 則 $Q^* \uparrow$
- $p \uparrow$ 則 $Q^* \downarrow$
- 如果 p 很低，那預燒測試對供應商是有利的。
- 如果 $C_b < C_s$ ，則預燒測試對供應商是有利的。
相反 $C_s < C_b$ ，則檢驗對供應商有利。
- 如果 α 和 β 很低，則檢驗對供應商是有利的。
如果 α 和 β 很高，則預燒測試對供應商是有利的。

Conclusion

- 供應商使用檢查和預燒測試來篩選產品，由於檢查誤差和預燒花費成本高，如何在低成本且達到品質要求下，權衡這兩個技術將是供應商的挑戰。
- 發展混合檢察和預燒最佳混合策略來達到利潤最大化，在滿足品質要求條件下，得到(a)生產量(b)檢查數量、預燒數量、不需檢查不需預燒數量(c)如果需要預燒其最佳預燒時間。
- 實際上，一個產品不僅需要檢查一個特性，而且如果買方與供應商有長期合約，通常會有折扣。因此上述兩點是未來研究一個很重要的課題，探討考慮折扣因子下的最佳混合檢察和預燒策略。