ANDERSON LAÉCIO GALINDO TRINDADE

Contribuições para o controle *on-line*
de processos por atributos

SÃO PAULO
2008

ANDERSON LAÉCIO GALINDO TRINDADE

Contribuições para o controle *on-line*
de processos por atributos

Tese apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do
título de Doutor em Engenharia.

SÃO PAULO
2008

ANDERSON LAÉCIO GALINDO TRINDADE

Contribuições para o controle *on-line*
de processos por atributos

Tese apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do
título de Doutor em Engenharia.

Área de Concentração:
Engenharia de Produção

Orientadora:
Prof. Livre Docente Linda Lee Ho

SÃO PAULO
2008

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 23 de abril de 2008.

Anderson Laécio Galindo Trindade (Autor)

Linda Lee Ho (Orientadora)

FICHA CATALOGRÁFICA

Trindade, Anderson Laécio Galindo
Contribuições para o controle on-line de processos por
atributos / A.L.G. Trindade. - ed. rev. -- São Paulo, 2008.
155 f.

Tese (Doutorado) - Escola Politécnica da Universidade de
São Paulo. Departamento de Engenharia de Produção.

1. Controle de processos I. Universidade de São Paulo.
Escola Politécnica. Departamento de Engenharia de Produção
II. t.

*Dedico este trabalho aos
meus pais, em especial à
minha mãe, que um dia sonhou
em ter um médico na família,
e agora terá um Doutor.*

Agradecimentos

- À professora, orientadora, amiga e mãe adotiva, Linda Lee Ho, pela dedicação e paciência na execução deste trabalho.
- Ao professor Roberto da Costa Quinino, um grande responsável pelo começo de minha incursão pela Engenharia de Produção.
- À professora Maria Laura, minha orientadora no mestrado, de quem trago grande parte de minha formação.
- À Família e Amigos que suportaram longas ausências.
- Aos professores e funcionários do Departamento de Engenharia de Produção.
- À Capes pela bolsa de estudos concedida.

Resumo

O procedimento de controle *on-line* de processos por atributos, proposto por Taguchi *et al.* (1989), consiste em amostrar um item a cada m produzidos e decidir, a cada inspeção, se houve ou não aumento na fração de itens não-conformes produzidos. Caso o item inspecionado seja não-conforme, pára-se o processo para ajuste supondo-se que tenha ocorrido uma mudança para a condição fora de controle. Como i) o sistema de inspeção pode estar sujeito a erros de classificação e o item inspecionado ser submetido à classificações repetidas; ii) a fração de itens não-conformes no estado fora de controle pode variar em função do número de itens produzidos (x) segundo uma função $\psi(x)$; iii) e a decisão de parar o processo pode ser tomada com base no resultado das últimas h inspeções, desenvolve-se um modelo que engloba estes pontos. Utilizando-se as propriedades de uma cadeia de Markov ergódica, obtém-se a expressão do custo médio do sistema de controle, que é minimizada por parâmetros que vão além do intervalo de inspeção m : o número de classificações repetidas r ; o número mínimo de classificações conformes para declarar um item como conforme s , o comprimento do histórico de inspeções considerado h e o critério de parada para ajuste ν . Os resultados obtidos mostram que: o uso de classificações repetidas pode ser uma alternativa econômica quando apenas um item é considerado na decisão sobre o ajuste do processo; uma cadeia de Markov finita pode ser utilizada para representar o sistema de controle na presença de uma função $\psi(x)$ não-constante; tomar a decisão de ajuste com base na observação de uma seqüência de itens inspecionados é a alternativa de maior impacto sobre o custo de controle do processo.

Palavras-chave: Controle de processos por atributos. Cadeia de Markov. Classificações repetidas. Deterioração da qualidade.

Abstract

The quality control procedure for attributes, proposed by Taguchi *et al.* (1989), consists in inspecting a single item at every m produced items and, based on the result of each inspection, deciding whether the non-conforming fraction has increased or not. If an inspected item is declared non-conforming, the process is stopped and adjusted, assuming that it has changed to out-of-control condition. Once: i) the inspection system is subject to misclassification and it is possible to carry out repetitive classifications in the inspected item; ii) the non-conforming fraction, when the process is out-of-control, can be described by $\psi(x)$; iii) the decision about stopping the process can be based on last h inspections, a model which considers those points is developed. Using properties of ergodic Markov chains, the average cost expression is calculated and can be minimized by parameters beyond m : number of repetitive classifications (r); minimum number of classifications as conforming to declare an item as conforming (s); number of inspections taken into account (h) and stopping criteria (ν). The results obtained show that: repetitive classifications of the inspected item can be a viable option if only one item is used to decide about the process condition; a finite Markov chain can be used to represent the control procedure in presence of a function $\psi(x)$; deciding about the process condition based on last h inspections has a significant impact on the average cost.

Keywords: Control procedure for attributes. Markov chain. Repetitive classifications. Quality deterioration.

Lista de Figuras

2.1	Representação do sistema de inspeção e respectivos estados da cadeia de Markov.	28
2.2	Transições entre os estados da cadeia de Markov	30
2.3	Gráfico de custo médio C em função de m e r	38
3.1	Representação do sistema de inspeção e respectivos estados da cadeia de Markov.	44
3.2	Fração de itens não-conformes em um processo sujeito a mudança de estado e deterioração da qualidade.	46
3.3	Transições entre estados da cadeia de Markov.	47
3.4	Representação ponto de mudança do estado sob controle para o estado fora de controle.	50
3.5	Representação do truncamento da cadeia para diferentes funções de deterioração.	54
3.6	Custo médio (a), tamanho da cadeia (b), número de classificações repetidas (c) e fração de defeitos (d) para diferentes intervalos entre inspeções. . . .	56
3.7	Valores de m° e r° em função dos erros α e β	58

3.8	Redução do custo médio: Cenário 3 versus Cenários 1 e 2.	59
3.9	Distribuição amostral do intervalo ótimo de inspeção.	60
4.1	Estados da cadeia e transições quando o critério de parada utilizado é uma seqüência de $h = 3$ inspeções não-conformes.	65
4.2	Custo médio versus política de parada utilizando o modelo Exponencial. .	68
4.3	Custo médio versus política de parada utilizando o modelo Linear.	68
4.4	Custo médio versus política de parada utilizando o modelo Constante. . . .	69
4.5	Representação de uma seqüência de inspeções considerando $h = 4$ e $\nu = 3$.	70
4.6	Representação da transição entre seqüências de inspeção quando: a) $h = 3$ e $\nu = 3$; b) $h = 3$ e $\nu = 2$	71
4.7	Custo versus política de parada para o modelo Exponencial e o Critério II.	79
4.8	Custo versus política de parada para o modelo Linear e o Critério II. . . .	79
4.9	Custo versus política de parada para o modelo Constante e o Critério II. .	80
4.10	Representação das matrizes esparsas utilizadas na solução do sistema linear quando $n = 5$, $h = 4$ e $\nu = 3$	82
4.11	Custo médio versus h ($\nu = 3; 4$) para o modelo Exponencial e o Critério II.	83
4.12	Custo médio versus h ($\nu = 3; 4$) para o modelo Linear e o Critério II. . . .	83
4.13	Custo médio versus h ($\nu = 3; 4$) para o modelo Constante e o Critério II. .	83
4.14	Custo médio versus política de parada utilizando o modelo Exponencial e o Critério II quando há custo relativo ao histórico.	84
4.15	Custo médio versus política de parada utilizando o modelo Linear e o Cri- tério II quando há custo relativo ao histórico.	85
4.16	Custo médio versus política de parada utilizando o modelo Constante e o Critério II quando há custo relativo ao histórico.	85

Lista de Tabelas

2.1	Parâmetros-base para os exemplos numéricos.	37
2.2	Custo e intervalo de inspeção ótimos em função do número de classificações repetidas.	38
2.3	Impacto de diferentes políticas para declarar o item inspecionado conforme sobre α^* e β^* quando $\alpha = 10\%$ e $\beta = 10\%$	39
2.4	Determinação de m° , r° , s° e C para a variação individual dos parâmetros do modelo.	40
3.1	Parâmetros-base para os exemplos numéricos.	52
3.2	Custo em função do tamanho da cadeia n para três diferentes funções de deterioração da qualidade.	53
3.3	Política ótima de controle para diferentes funções de deterioração.	57
3.4	Efeito dos componentes de custo sobre o custo médio do sistema de controle.	60
3.5	Análise de sensibilidade para diferentes parâmetros.	61
4.1	Comparação entre as políticas ótimas de controle utilizando o Critério I para as diferentes funções de deterioração.	69

4.2	Comparação entre os custos ótimos obtidos no Capítulo 3 ($h = 1$) e com o Critério I ($h = 2$).	70
4.3	Viabilidade das seqüências e transições para $h = 3$ e $\nu = 3$	73
4.4	Viabilidade das seqüências e transições para $h = 3$ e $\nu = 2$	73
4.5	Número de seqüências de inspeções válidas para diferentes h e ν	74
4.6	Equações das probabilidades de transição utilizadas na obtenção da matriz $\mathbf{P}_n$ para $n = 3$, $h = 3$ e $\nu = 2$	77
4.7	Comparação entre as políticas ótimas de controle utilizando os Critérios I e II para as diferentes funções de deterioração, com $h \leq 7$	80
4.8	Comparação entre os custos ótimos obtidos com os Critérios I e II.	80
4.9	Custo ótimo em função de h e ν para as diferentes funções de deterioração.	81
4.10	Comparação entre as políticas ótimas de controle utilizando os Critérios I e II para as diferentes funções de deterioração	84
4.11	Comparação entre os custos ótimos obtidos com os Critérios I, II, e II com custo relativo ao histórico.	85

Lista de Algoritmos

3.1	Algoritmo de Busca Direta $(C^\circ, m^\circ, r^\circ, s^\circ)$	55
4.1	Algoritmo para determinação da viabilidade e transições entre sequências .	72

Nomenclatura

Letras Romanas

bin	distribuição binomial
c_{ins}	custo de uma classificação do item inspecionado
c_{nc}	custo de um item defeituoso não detectado, que segue para o consumidor ou próximas etapas do processo
c_a	custo de ajustar o processo
c_{s_nc}	custo de descartar um item defeituoso
c_{s_c}	custo de descartar um item perfeito
$CB(z)$	função que conta o número de bits 1 da representação binária do número z
geo	distribuição geométrica
h	comprimento da sequência de inspeções considerado na estratégia de controle
I	matriz identidade
m	intervalo de inspeções

n	número mínimo de inspeções após a mudança de estado do processo para que a cadeia de Markov finita represente adequadamente o sistema de inspeção proposto
$\mathbf{P}$	matriz das probabilidades de transição da cadeia de Markov
$P_{(\cdot),(\cdot)}$	probabilidade de transição entre estados da cadeia de Markov
p_1	probabilidade de produção de itens defeituosos no Estado I
p_2	probabilidade de produção de itens defeituosos no Estado II
$PB(z)$	função que indica o primeiro bit (aquele à extrema direita) da representação binária do número z
r	número de classificações repetidas a que cada item inspecionado é submetido
s	número mínimo de classificações com resultado “conforme” para que um item submetido a r classificações seja declarado conforme
w	primeiro número inteiro do par ordenado que representa o estado da cadeia de Markov e está associado ao estado de produção do processo
$w_i z_j$	estado da cadeia de Markov no qual w_i está associado ao estado de produção do processo e z_j à classificação final do item inspecionado
$\mathbf{y}$	vetor linha das probabilidades estacionárias dos estados da cadeia de Markov
z	segundo número inteiro do par ordenado que representa o estado da cadeia de Markov e está associado à classificação do item recém-inspecionado

Letras Gregas

α	probabilidade de declarar um item conforme como defeituoso
α^*	probabilidade de declarar um item conforme como defeituoso após a realização de r classificações repetidas, considerado o critério s
β	probabilidade de declarar um item defeituoso como conforme
β^*	probabilidade de declarar um item defeituoso como conforme após a realização de r classificações repetidas, considerado o critério s

$\gamma(h, \nu)$	número de seqüências válidas que podem ser obtidas considerando o comprimento h e critério de parada ν
δ_n	Diferença relativa do custo obtido por matrizes de transição com tamanhos $n - 1$ e n
$\Delta_{\max}$	Critério de convergência para o custo obtido por uma matriz de transição n
ζ	classificação final do item inspecionado ($\zeta = 0$, item conforme; $\zeta = 1$, item defeituoso)
$\theta_{(w_i, z_j)}$	é o custo, no estado w_i, z_j , relacionado ao item inspecionado (somente a classificação final do item é conhecida, sendo necessário considerar a presença dos erros α^* e β^*)
ν	número de inspeções declaradas não-conformes para que o processo seja interrompido para ajuste
$\xi_{(w_i, z_j)}$	o custo, no estado w_i, z_j , decorrente dos itens defeituosos dentre os $(m - 1)$ itens não inspecionados que são enviados ao consumidor ou próximas etapas da produção
π	probabilidade do processo sair da condição sob controle a cada item produzido (parâmetro da distribuição geométrica)
ρ	número de classificações “conforme” apresentado por um item inspecionado r vezes
Θ	real condição do item inspecionado ($\Theta = 0$, item conforme; $\Theta = 1$, item defeituoso)
$\varphi_{(w_i, z_j)}$	é o custo, no estado w_i, z_j , para ajustar o processo
ϕ	fração média de itens não-conformes produzidos pelo processo
$\phi_{\max}$	fração média máxima de itens não-conformes que podem ser produzidos pelo processo
$\psi(x)$	função que descreve a probabilidade de produção de itens defeituosos no estado II em função do número de itens produzidos neste estado
$\bar{\psi}(x)$	$1 - \psi(x)$

$\Psi(z, \nu)$ função indicadora que informa se o inteiro z , em sua representação binária, corresponde a uma sequência viável com o item recém inspecionado declarado conforme (0), declarado não-conforme mas não leva ao ajuste (1), declarado não-conforme e leva a ajuste (2) ou trata-se de uma sequência inviável (-1).

Ω espaço de estados da cadeia de Markov

ω_i estado no qual o i -ésimo item foi produzido ($\omega_i = 1$, Estado I; $\omega_i = 2$, Estado II)

Sobrescritos

$^{\circ}$ parâmetro ótimo

T transposta

Sumário

1	Introdução	18
1.1	Controle Estatístico de Processo	18
1.2	Modelos de Taguchi para controle <i>on-line</i>	20
1.3	O modelo de Taguchi para controle <i>on-line</i> de processos por atributos . . .	21
1.3.1	Caso I	21
1.3.2	Caso II	22
1.4	Os Modelos Derivados de Taguchi	22
1.5	Objetivos Propostos	24
2	Controle <i>on-line</i> para atributos considerando classificações repetidas	25
2.1	Erros de inspeção e classificações repetidas	26
2.2	A cadeia de Markov	27
2.3	Probabilidades estacionárias	31
2.4	Custo dos estados da cadeia	32

2.5	Custo médio do sistema de controle	35
2.6	Exemplos numéricos e discussões	37
2.7	Considerações Finais	41
3	Controle <i>on-line</i> para atributos considerando deterioração da qualidade	42
3.1	A cadeia de Markov	43
3.1.1	Probabilidades de transição	44
3.1.2	Determinação de n	46
3.2	Custo dos estados da cadeia	48
3.3	Custo médio do sistema de controle	51
3.4	Exemplos numéricos e discussões	51
3.4.1	Truncamento da cadeia	52
3.4.2	Determinação da política ótima	54
3.4.3	Determinação da política ótima para o exemplo numérico	55
3.4.4	A política ótima para diferentes funções de deterioração	57
3.4.5	A presença dos erros de classificação	57
3.4.6	Incerteza nos componentes do custo	58
3.4.7	Análise de sensibilidade para os principais componentes de custo . .	60
3.5	Considerações Finais	61
4	Controle <i>on-line</i> para atributos considerando o histórico das h últimas inspeções	63
4.1	Critério I: Ajustar quando uma seqüência de h inspeções resultar em h itens não-conformes	64
4.1.1	Probabilidades de Transição	64
4.1.2	Custos dos estados da cadeia	66

4.1.3	Determinação da política ótima para as diferentes funções de deterioração aplicando o Critério I	67
4.2	Critério II: Ajustar quando forem observados ν itens não-conformes em uma sequência de h inspeções	70
4.2.1	A cadeia de Markov	74
4.2.2	Probabilidades de Transição	75
4.2.3	Custos dos estados da cadeia	76
4.2.4	Determinação da política ótima para as diferentes funções de deterioração aplicando o Critério II	79
4.2.5	Problemas na aplicação do Critério II	80
4.2.6	Custos na aplicação dos Critérios I e II	84
4.3	Considerações finais	86
5	Conclusões	87
	Referências Bibliográficas	90
	Apêndice A Programa desenvolvido	92
	Apêndice B Artigo publicado na revista <i>Pesquisa Operacional</i>	122
	Apêndice C Artigo publicado na revista <i>Applied Stochastic Models in Business and Industry</i>	135

CAPÍTULO 1

Introdução

1.1 Controle Estatístico de Processo

Em função da grande oferta de novos produtos e serviços, a qualidade tem sido um critério de decisão empregado tanto pelo consumidor como pelo produtor. Desta forma, as empresas têm feito esforços para aprimorar a qualidade de seus produtos através de programas de melhoria de qualidade.

Uma grande parte destes esforços está centrada no Controle Estatístico de Processo (CEP), que tem como objetivo atuar em processos para evitar uma produção grande de itens não conformes. Geralmente utilizam-se gráficos de controle que têm como função detectar desvios do comportamento das características específicas de um processo. De modo bem sucinto, um gráfico de controle é uma representação gráfica progressiva de uma característica de qualidade medida (pode ser atributo ou variável) ou calculada a partir de observações extraídas do processo periodicamente ao longo do tempo, utilizando-se as seguintes retas de referência:

$$LS(\text{limite superior de controle}) = \mu_D + d\sigma_D$$

$$LC(\text{linha central}) = \mu_D$$

$$LI(\text{linha inferior de controle}) = \mu_D - d\sigma_D$$

em que D é uma estatística associada a uma característica de qualidade; d é uma constante previamente escolhida, μ_D e σ_D são respectivamente a média e o desvio padrão de D . Caso o valor da estatística D esteja registrado dentro dos limites de controle é dito que o processo está sob controle estatístico; caso contrário é dito que o processo está fora de controle. Nesse caso, geralmente pára-se a produção para investigar (ajuste ou falso alarme). Costuma-se supor que itens são produzidos um a um e amostras de tamanho n são coletadas em intervalos regulares de duração m . O planejamento de um gráfico de controle é feito através da determinação dos valores de n , d e m , satisfazendo critérios tanto estatísticos como econômicos. Os critérios estatísticos estão associados às probabilidades do falso alarme ou número de amostras após a ocorrência do desvio (da característica específica do processo) até a sua detecção enquanto que critérios econômicos procuram valores dos parâmetros (ótimos) tal que minimize uma função-objetivo geralmente associada aos custos de operação (inspeção, de falso alarme, ajuste, retrabalho, etc.).

Shewhart, do Bell Telephone Laboratories, é apontado na literatura como o proponente do conceito estatístico do gráfico de controle nos idos anos 20. Este momento é considerado, em geral, como o começo formal do controle estatístico da qualidade. Os gráficos de controle têm sido planejados em relação apenas a critérios estatísticos. O planejamento de um gráfico de controle tem conseqüências econômicas, já que os custos de amostragem e de testes, os custos associados à investigação de sinais fora de controle e possivelmente à correção de causas atribuíveis, e custos da chegada ao consumidor de unidades não-conformes são todos afetados pela escolha dos parâmetros do gráfico de controle. Um dos trabalhos pioneiros em empregar critérios econômicos para planejamento de gráficos de controle foi de Duncan em 1956 (MONTGOMERY, 2004 apud DUNCAN, 1956, p. 307).

A fim de formular um modelo econômico para o planejamento de um gráfico de controle, são necessárias algumas suposições sobre o comportamento do processo. Em geral, supõe-se que o processo seja caracterizado por um único estado sob controle. Por exemplo, se o processo tem uma única característica da qualidade mensurável, então o estado sob controle corresponde à média dessa característica, quando não há quaisquer causas atribuíveis presentes. Analogamente, quando a característica da qualidade é um atributo, o estado sob controle é representado pela fração de não-conformes produzida pelo processo quando não estão presentes quaisquer causas atribuíveis. O processo pode ter $s \geq 1$ estados fora de controle, cada um destes estados associados a um tipo particular de causa atribuível. Outra suposição diz respeito ao tempo que o processo permanece no estado sob controle, que é geralmente descrito por uma variável aleatória exponencial (caso o

a produção seja considerada em tempo contínuo) ou geométrica (caso a produção seja considerada em tempo discreto). Esta suposição permite uma simplificação na elaboração de modelos econômicos, e em algumas situações resulta em uma estrutura de modelo de cadeia de Markov. Supõe-se também que o processo não seja autocorretivo, ou seja, uma vez ocorrida a transição para um estado fora de controle, o processo só pode retornar à condição sob controle com uma intervenção no processo. Em alguns casos admitem-se transições entre diferentes estados fora de controle, desde que as transições sejam sempre consistentes com uma posterior deterioração da qualidade (MONTGOMERY, 2004).

1.2 Modelos de Taguchi para controle *on-line*

Em planejamentos econômicos, muitos trabalhos já foram escritos, porém o procedimento econômico *on-line* mais conhecido por sua simplicidade e de fácil implantação foi o proposto por Taguchi (TAGUCHI *et al.*, 1989), desenvolvido para monitorar tanto características medidas como variáveis assim como atributos. O procedimento consiste em retirar um único item da linha produção a cada intervalo fixo de m itens produzidos. Se o valor da característica interesse do item examinado estiver fora dos limites de controle (ou se o item for não-conforme, no caso de atributos), pára-se o processo para investigação. Ainda segundo Taguchi, este sistema de controle de qualidade *on-line* deve ser empregado de modo que os valores alvos desejados da característica de qualidade possam ser economicamente controlados. O problema consiste em determinar o intervalo ótimo m e a constante ótima d (esta última apenas para o controle de variáveis) tal que minimize o custo médio do sistema de controle. A proposição de Taguchi para monitorar um processo *on-line* foi alvo de muitas críticas em função das suposições (muitas vezes implícitas) e aproximações empregadas. Por exemplo, no controle *on-line* para variáveis, Taguchi fez a suposição (implícita) que a característica de interesse segue uma distribuição uniforme no intervalo $[-d, +d]$, o que nem sempre descreve o comportamento real em situações práticas. Além disso, o modelo de Taguchi não deixa claro como ocorre uma mudança do parâmetro do processo.

Este tipo de planejamento foi estudado por diversos autores, como Adams e Woodall (1989), Srivastava e Wu (1991, 1995, 1994, 1996, 1997, 1999), Box e Kramer (1992), Box e Luceno (1994), Chou e Wang (1996), Nandi e Sreehari (1997, 1999), Wang e Yue (2001). Exemplos em que este tipo de procedimento é empregado incluem o processo automático de solda, produção de semi-condutores, de diodos, de placas de circuito impresso e processos químicos. Em geral, sistemas produtivos que empregam algum tipo de

controle automático com observações individuais podem ser monitorados por este tipo de procedimento.

1.3 O modelo de Taguchi para controle *on-line* de processos por atributos

Taguchi propôs um planejamento econômico para monitorar processos *on-line* por atributos considerando dois casos: i) a mudança de 0% de defeitos para 100% de defeitos; e ii) a mudança de 0% de defeitos para $\pi\%$ de defeitos (TAGUCHI *et al.*, 1989).

Os detalhes para obtenção das equações dos Casos I e II, descritos a seguir, também podem ser vistos em Souza (2002).

1.3.1 Caso I

O ciclo de produção inicia-se com uma fração inicial de defeitos igual a zero e, em algum instante aleatório, muda para uma fração de 100% de defeitos. Os itens são produzidos independentemente e um deles é examinado de cada m produzidos. Se o item examinado for conforme a produção continua; caso contrário interrompe-se a produção para fazer ajuste. Entre o instante da sinalização e a parada total do processo para o ajuste, são produzidos mais l itens (atraso). Após a parada, o processo volta a ter fração de defeitos igual a zero e o ciclo descrito se repete.

O intervalo de inspeção é escolhido tal que minimize a função custo L proposta por Taguchi:

$$L = \frac{C_i}{m} + \left(\frac{m+1}{2} + l \right) \frac{C_d}{\bar{u}} + \frac{C_a}{\bar{u}} \quad (1.1)$$

em que C_i é o custo de inspeção, C_d o custo decorrente de um item não conforme, C_a o custo de ajuste, $\bar{u}$ o número médio de itens produzidos entre ajustes e l o número de itens produzidos entre o instante da inspeção e a parada para o ajuste do processo (também chamado de atraso). Desprezando a dependência entre $\bar{u}$ e m e considerando que $\bar{u} \gg l$ e $C_d \gg C_a/\bar{u}$, Taguchi derivou (1.1) em relação à m e obteve o intervalo ótimo

$$m^\circ = \sqrt{\frac{2\bar{u}C_i}{C_d}}. \quad (1.2)$$

Note que (1.2) não depende de l e C_a . Taguchi sugeriu que se as condições estabelecidas

para obtenção de (1.2) não forem satisfeitas, $\bar{u}$ seja substituído por $\bar{u} + \frac{m}{2}$ em (1.1) e o intervalo ótimo seja obtido utilizando aproximação por séries de Taylor, resultando em

$$m^\circ = \sqrt{\frac{2(\bar{u} + l)C_i}{C_d - \frac{C_a}{\bar{u}}}}. \quad (1.3)$$

1.3.2 Caso II

Quando o processo muda sua fração de defeitos de 0% para $\pi\%$, Taguchi introduz um novo componente de custo: C_D , custo de um item defeituoso que segue sem detecção para as próximas etapas do processo, enquanto redefine C_d como o custo do item defeituoso que é detectado, sendo $C_D \gg C_d$. Utilizando as simplificações adotadas no Caso I, Taguchi obteve como intervalo ótimo de inspeção

$$m^\circ = \sqrt{\frac{2(\bar{u} + l)C_i}{C_D - \frac{C_a}{\bar{u}}}}. \quad (1.4)$$

As equações (1.2), (1.3) e (1.4) mostram a grande virtude do modelo proposto por Taguchi: a simplicidade. E são justamente as razões que proporcionam esta simplicidade que abrem precedente para os trabalhos subseqüentes.

1.4 Os Modelos Derivados de Taguchi

A proposta de Taguchi foi questionada em trabalhos posteriores, sendo que o trabalho de Nayeypour e Woodall (1993) representa a principal referência crítica. As observações realizadas por estes autores incluem: i) a ausência de um mecanismo explícito, na modelagem, para descrever a mudança do estado sob controle para o estado fora de controle; ii) a série de simplificações adotadas para obtenção das expressões analíticas para m° .

Nayeypour e Woodall (1993) apresentaram um modelo para controle *on-line* de atributos assumindo que a mudança para a condição fora de controle ocorre segundo uma distribuição geométrica de parâmetro π . O ciclo de produção é definido como o período de tempo (em unidades produzidas) a partir do início da produção (ou após um ajuste) até a detecção e remoção da causa atribuível. A sequência de produção, controle e ajuste, com os custos contabilizados a cada ciclo, é modelado pela Teoria da Renovação (detalhes em ROSS, 2003). Foram definidos C , T e L como variáveis aleatórias que representam o

custo por ciclo, o comprimento do ciclo e o custo por item produzido, respectivamente, e

$$E(L) = \frac{E(C)}{E(T)}. \quad (1.5)$$

Nayebpour e Woodall (1993) empregaram os mesmos componentes de custo utilizados por Taguchi para desenvolver modelos correspondentes aos Casos I e II. Porém, com as considerações adotadas não foi possível obter uma expressão analítica para o intervalo ótimo de inspeção a partir de (1.5), que é determinado por busca computacional. Os resultados obtidos pelos autores mostram que o Caso II apresenta a maior discrepância entre as abordagens, quando considerado o custo médio por item produzido. Os autores recomendaram aumentar a frequência de inspeção caso a proporção de itens defeituosos seja inaceitavelmente alta para o intervalo de inspeção ótimo. Em outras palavras, sugeriram que algum tipo de restrição estatística seja associada ao modelo econômico, sem contudo considerá-la em seu modelo. Este tipo de recomendação também é feita por Montgomery (2004).

As premissas adotadas por Nayebpour e Woodall (1993) tornaram-se referência para trabalhos mais recentes. Nandi e Sreehari (1997) inovaram ao introduzir a existência de duas causas atribuíveis: uma causa “menor” e outra causa “maior”, o que proporciona dois níveis diferentes de qualidade após a mudança para o estado fora de controle. Nandi e Sreehari (1999) modificaram o modelo anterior, introduzindo uma função linear para descrever a deterioração da fração de itens conformes após a mudança do processo. Para esta função obtiveram as expressões do custo médio e também determinaram computacionalmente o intervalo ótimo para as inspeções. Já a obtenção das expressões do custo médio para outras funções de deterioração como a exponencial, Weibull, log-normal ou algum outro modelo em particular pode não ser tarefa das mais simples.

Borges *et al.* (2001) propuseram que os trabalhos baseados no modelo de Taguchi podem apresentar erros de classificação, comprometendo a determinação do intervalo ótimo entre inspeções. Os autores consideraram os erros α (classificar um item conforme como defeituoso) e β (classificar um item defeituoso como conforme) e concluíram que, ainda que pequenos (da ordem de 1%), alteram o custo e política ótima de inspeção. Este resultado era esperado, já que a decisão de parar o processo é baseada na classificação de um único item inspecionado a cada m produzidos. Os erros de classificação associados ao controle de qualidade também são abordados em Johnson *et al.* (1991), Ranjan *et al.* (2003) e Wang (2007), para citar alguns exemplos.

1.5 Objetivos Propostos

Considerando que a produção de itens pelo processo é discreta, e que a transição do estado sob controle para o estado fora de controle segue uma distribuição geométrica, propõe-se neste trabalho o desenvolvimento de um modelo utilizando cadeia de Markov com estados discretos que inclui:

1. A classificação repetida e independente do item inspecionado.
Supõe-se, neste caso, que o item inspecionado pode ser submetido a classificações repetidas, o que pode proporcionar redução no custo médio quando há presença de erros de classificação. Esta hipótese foi aventada por Borges *et al.* (2001), porém não foi implementada.
2. O uso de uma função $\psi(x)$ para descrever a deterioração da fração de itens conformes em função do número de itens produzidos quando o processo opera fora de controle.
3. O uso do histórico das h últimas inspeções para a tomada de decisão sobre a necessidade de intervir sobre o processo.
Neste caso, mantém-se a essência do procedimento proposto por Taguchi (observações individuais) mas o ajuste é decidido com base em uma sequência de inspeções.

Neste trabalho, o modelo que contemple as três propostas será apresentado. Porém, o desenvolvimento será incremental, por capítulos, sendo que conceitos previamente apresentados podem ser apenas referenciados nos capítulos seguintes e, a cada novo capítulo, o modelo anterior se tornará um caso particular. O Capítulo 2 apresentará o modelo que considera o uso de classificações repetidas, com a fração de itens não-conformes passando de p_1 (processo sob controle) para p_2 (processo fora de controle). No Capítulo 3 a fração de itens não-conformes, quando o processo opera fora de controle, será descrita por uma função $\psi(x)$ e uma restrição à quantidade de itens não-conformes produzidos será aplicada. O Capítulo 4 modificará o modelo obtido até então para que o histórico das últimas h inspeções sejam consideradas no momento de decidir sobre a necessidade de ajuste do processo. O Capítulo 5 apresentará as conclusões e algumas sugestões para trabalhos futuros.

CAPÍTULO 2

Controle *on-line* para atributos considerando classificações repetidas [†]

Considere um processo que opera sob controle e, devido a uma causa atribuível, passa a condição fora de controle. O monitoramento deste processo é feito como aquele proposto por Taguchi, e as seguintes suposições são estabelecidas:

- o processo inicia (ou reinicia) a produção no Estado I (sob controle, com fração de itens defeituosos constante $p_1 \geq 0$) e, após a mudança, a fração de itens não-conformes passa ao valor constante $p_2 > p_1$;
- a mudança do Estado I para o Estado II ocorre segundo uma distribuição geométrica com parâmetro π ;
- o processo de monitoramento consiste na inspeção de um único item a cada m produzidos;
- o sistema de inspeção está sujeito a erros de classificação: α é a probabilidade de classificar um item conforme como defeituoso e β a probabilidade de classificar um item defeituoso como conforme;

[†] O texto base deste Capítulo foi publicado na revista *Pesquisa Operacional*, sob o título “Controle *on-line* por atributos com erros de classificação: uma abordagem econômica com classificações repetidas”, Trindade *et al.* (2007a). O artigo é apresentado no Apêndice B.

- o item inspecionado pode ser submetido a r classificações independentes (sendo classificado em cada uma como conforme ou defeituoso) e o tempo necessário para realizar estas classificações é desprezível;
- tão logo o item inspecionado seja declarado defeituoso (segundo um critério a ser estabelecido), a parada do processo ocorre imediatamente (sem atraso) e o item inspecionado é descartado.

2.1 Erros de inspeção e classificações repetidas

Como proposto, cada item inspecionado é submetido a r classificações repetidas com o objetivo de reduzir o impacto dos erros α e β . Como os resultados das classificações do item inspecionado podem divergir, deve-se estabelecer um critério para a classificação final. Sejam: ρ uma variável aleatória que representa o número de classificações com resultado “conforme” que um item inspecionado pode produzir, $\rho = \{0, 1, \dots, r\}$; s o número mínimo de classificações com resultado “conforme” para que um item seja declarado conforme; e ζ uma variável aleatória que representa a classificação final do item inspecionado, definida como:

$$\zeta = \begin{cases} 0, & \text{se } \rho \geq s \\ 1, & \text{se } \rho < s. \end{cases}$$

Suponha que o item inspecionado seja de fato conforme (i.e., $\rho \sim \text{bin}(r, 1 - \alpha)$). Neste caso, em cada uma das r classificações a que ele é submetido o erro α pode ocorrer. Seja Θ uma variável aleatória que representa a real condição do item inspecionado ($\Theta = 0$, item conforme; $\Theta = 1$, item defeituoso). A probabilidade de classificar corretamente um item conforme submetido à r classificações é dada por

$$\Pr(\zeta = 0 \mid \Theta = 0) = \Pr(\rho \geq s \mid \Theta = 0) = \sum_{i=s}^r \binom{r}{i} (1 - \alpha)^i \alpha^{r-i}. \quad (2.1)$$

O complementar de (2.1) pode ser definido como

$$\alpha^* = \Pr(\zeta = 1 \mid \Theta = 0) = \Pr(\rho < s \mid \Theta = 0) = \sum_{i=0}^{s-1} \binom{r}{i} (1 - \alpha)^i \alpha^{r-i}.$$

Suponha agora que o item inspecionado seja de fato defeituoso (i.e., $\rho \sim \text{bin}(r, \beta)$). A

probabilidade de classificar corretamente este item é dada por

$$\Pr(\zeta = 1 \mid \Theta = 1) = \Pr(\rho < s \mid \Theta = 0) = \sum_{i=0}^{s-1} \binom{r}{i} \beta^i (1 - \beta)^{r-i},$$

enquanto o erro cometido em sua classificação pode ser definido como

$$\beta^* = \Pr(\zeta = 0 \mid \Theta = 1) = \Pr(\rho \geq s \mid \Theta = 1) = 1 - \sum_{i=0}^{s-1} \binom{r}{i} \beta^i (1 - \beta)^{r-i}$$

Portanto, α^* e β^* são, efetivamente, as probabilidades dos erros que podem ser cometidos na classificação final do item inspecionado.

2.2 A cadeia de Markov

O sistema de inspeção proposto pode ser modelado como uma cadeia de Markov estacionária considerando o espaço de estados discretos

$$\Omega = \{(0,0), (0,1), (1,0), (1,1), (2,0), (2,1)\},$$

em que cada estado é descrito por um par de números inteiros w e z . O inteiro w está relacionado ao estado do processo em que os itens que compõem o ciclo de inspeção foram produzidos. Entenda-se por ciclo de inspeção cada conjunto de m itens produzidos, sendo o m -ésimo item o item inspecionado. Quando $w = 0$ (w_0), todos os itens (inclusive o inspecionado) foram produzidos no Estado I. Quando $w = 2$ (w_2) todos os itens do ciclo (inclusive o inspecionado) foram produzidos no Estado II. Quando $w = 1$ (w_1) necessariamente ocorreu uma mudança do Estado I para o Estado II no ciclo considerado e pelo menos o item inspecionado foi produzido no Estado II. O inteiro z , por sua vez, indica se o item inspecionado foi declarado conforme (z_0) ou defeituoso (z_1). A matriz $\mathbf{P}$ apresentada em (2.2) fornece as probabilidades de transição entre os estados em ciclos subseqüentes de inspeção.

$$\mathbf{P} = \begin{matrix} & w_0 z_0 & w_0 z_1 & w_1 z_0 & w_1 z_1 & w_2 z_0 & w_2 z_1 \\ \begin{matrix} w_0 z_0 \\ w_0 z_1 \\ w_1 z_0 \\ w_1 z_1 \\ w_2 z_0 \\ w_2 z_1 \end{matrix} & \left(\begin{array}{cccccc} P_{(0,0),(0,0)} & P_{(0,0),(0,1)} & P_{(0,0),(1,0)} & P_{(0,0),(1,1)} & P_{(0,0),(2,0)} & P_{(0,0),(2,1)} \\ P_{(0,1),(0,0)} & P_{(0,1),(0,1)} & P_{(0,1),(1,0)} & P_{(0,1),(1,1)} & P_{(0,1),(2,0)} & P_{(0,1),(2,1)} \\ P_{(1,0),(0,0)} & P_{(1,0),(0,1)} & P_{(1,0),(1,0)} & P_{(1,0),(1,1)} & P_{(1,0),(2,0)} & P_{(1,0),(2,1)} \\ P_{(1,1),(0,0)} & P_{(1,1),(0,1)} & P_{(1,1),(1,0)} & P_{(1,1),(1,1)} & P_{(1,1),(2,0)} & P_{(1,1),(2,1)} \\ P_{(2,0),(0,0)} & P_{(2,0),(0,1)} & P_{(2,0),(1,0)} & P_{(2,0),(1,1)} & P_{(2,0),(2,0)} & P_{(2,0),(2,1)} \\ P_{(2,1),(0,0)} & P_{(2,1),(0,1)} & P_{(2,1),(1,0)} & P_{(2,1),(1,1)} & P_{(2,1),(2,0)} & P_{(2,1),(2,1)} \end{array} \right) \end{matrix} \quad (2.2)$$

Como exemplo para ilustrar a notação utilizada, $P_{(0,0),(1,1)}$ representa a probabilidade de que a inspeção i ocorra sob o estado da cadeia $w_1 z_1$ dado que a inspeção $(i-1)$ ocorreu sob o estado $w_0 z_0$. Esta probabilidade de transição corresponde à quarta coluna da primeira linha da matriz $\mathbf{P}$. A Figura 2.1 relaciona o processo de produção aos estados da cadeia.

Figura 2.1: Representação do sistema de inspeção e respectivos estados da cadeia de Markov.

A probabilidade $P_{(0,0),(0,0)}$ indica que: a) o ciclo atual de inspeção iniciou-se sob o Estado I e todos os itens do ciclo foram produzidos neste estado; b) o item inspecionado foi declarado conforme. Para obter as probabilidades de transição é necessário definir uma variável aleatória não observável, ω_i , com $i \geq 0$, que representa o estado no qual o i -ésimo item foi produzido (1, Estado I; 2, Estado II). A probabilidade de que o processo permaneça no Estado I durante a produção dos m itens do ciclo pode ser expressa como

$$\Pr\{\omega_m = 1\} = \Pr\{\omega_1 = 1, \omega_2 = 1, \dots, \omega_m = 1\} = (1 - \pi)^m. \quad (2.3)$$

Ainda para se obter as probabilidades de transição deve-se calcular $\Pr\{\zeta = 0\}$ no Estado I.

Condicionando à Θ e utilizando as definições de α^* e β^* , pode-se escrever

$$\begin{aligned}\Pr\{\zeta = 0\} &= \Pr\{\zeta = 0 \mid \Theta = 0\} \Pr\{\Theta = 0\} + \Pr\{\zeta = 0 \mid \Theta = 1\} \Pr\{\Theta = 1\} \\ &= (1 - \alpha^*)(1 - p_1) + \beta^* p_1\end{aligned}\quad (2.4)$$

Combinando (2.3) e (2.4)

$$P_{(0,0),(0,0)} = (1 - \pi)^m [(1 - p_1)(1 - \alpha^*) + p_1 \beta^*]. \quad (2.5)$$

A probabilidade $P_{(0,0),(0,1)}$ difere da anterior em relação ao item inspecionado, que é declarado defeituoso. Neste caso, $\Pr\{\zeta = 1\}$ é dada por

$$\begin{aligned}\Pr\{\zeta = 1\} &= \Pr\{\zeta = 1 \mid \Theta = 0\} \Pr\{\Theta = 0\} + \Pr\{\zeta = 1 \mid \Theta = 1\} \Pr\{\Theta = 1\} \\ &= \alpha^*(1 - p_1) + (1 - \beta^*)p_1\end{aligned}\quad (2.6)$$

Portanto,

$$P_{(0,0),(0,1)} = (1 - \pi)^m [(1 - p_1)\alpha^* + p_1(1 - \beta^*)]. \quad (2.7)$$

$P_{(0,0),(1,0)}$ e $P_{(0,0),(1,1)}$ indicam que no ciclo de inspeção atual ocorreu uma mudança do Estado I para o Estado II. A probabilidade desta mudança é dada pelo complementar de (2.3). As probabilidades de declarar o item inspecionado como conforme ou defeituoso são similares as (2.4) e (2.6), mas o item inspecionado foi produzido no Estado II ($\Pr\{\Theta = 1\} = p_2$). Com isso,

$$P_{(0,0),(1,0)} = (1 - (1 - \pi)^m) [(1 - p_2)(1 - \alpha^*) + p_2 \beta^*]$$

$$P_{(0,0),(1,1)} = (1 - (1 - \pi)^m) [(1 - p_2)\alpha^* + p_2(1 - \beta^*)].$$

Pela definição dos estados da cadeia, não é possível uma transição direta entre os estados $w_0 z_0$ e $w_2 z_i$, como mostra a Figura 2.2. Assim

$$P_{(0,0),(2,0)} = 0; \quad P_{(0,0),(2,1)} = 0$$

Após um ajuste, por definição, o processo retorna ao Estado I. Assim, as probabilidades de transição que partem de um estado da cadeia no qual ocorreu um ajuste (i.e., $z = 1$) são iguais àquelas que partem do estado $w_0 z_0$. Em outras palavras, todas as linhas pares

da matriz $\mathbf{P}$ são iguais à primeira linha. Também por definição, uma vez no Estado II o processo só retorna ao Estado I após ajuste, o que anula as seguintes probabilidades:

$$P_{(1,0),(0,0)} = 0; \quad P_{(1,0),(0,1)} = 0; \quad P_{(2,0),(0,0)} = 0; \quad P_{(2,0),(0,1)} = 0.$$

O estado $w_1 z_0$ só pode ocorrer uma única vez após a mudança $I \rightarrow II$, visto que representa a primeira inspeção nesta nova condição. Daí,

$$P_{(1,0),(1,0)} = 0; \quad P_{(1,0),(1,1)} = 0; \quad P_{(2,0),(1,0)} = 0; \quad P_{(2,1),(1,0)} = 0;$$

As probabilidades restantes, $P_{(1,0),(2,0)}$, $P_{(1,0),(2,1)}$, $P_{(2,0),(2,0)}$ e $P_{(2,0),(2,1)}$, devem considerar apenas a probabilidade de julgar o item inspecionado como conforme ou defeituoso, pois o ciclo já se inicia no Estado II (i.e., não há mudanças $I \rightarrow II$ dentro destes ciclos).

$$\begin{aligned} P_{(1,0),(2,0)} &= P_{(2,0),(2,0)} = (1 - p_2)(1 - \alpha^*) + p_2\beta^* \\ P_{(1,0),(2,1)} &= P_{(2,0),(2,1)} = (1 - p_2)\alpha^* + p_2(1 - \beta^*) \end{aligned}$$

A Figura 2.2 resume as possíveis transições entre os estados da cadeia. Considerando as probabilidades de transição estabelecidas, podemos reescrever a matriz $\mathbf{P}$ (2.2) como

$$\mathbf{P} = \begin{pmatrix} P_{(0,0),(0,0)} & P_{(0,0),(0,1)} & P_{(0,0),(1,0)} & P_{(0,0),(1,1)} & 0 & 0 \\ P_{(0,0),(0,0)} & P_{(0,0),(0,1)} & P_{(0,0),(1,0)} & P_{(0,0),(1,1)} & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(1,0),(2,0)} & P_{(1,0),(2,1)} \\ P_{(0,0),(0,0)} & P_{(0,0),(0,1)} & P_{(0,0),(1,0)} & P_{(0,0),(1,1)} & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(2,0),(2,0)} & P_{(2,0),(2,1)} \\ P_{(0,0),(0,0)} & P_{(0,0),(0,1)} & P_{(0,0),(1,0)} & P_{(0,0),(1,1)} & 0 & 0 \end{pmatrix}$$

Figura 2.2: Transições entre os estados da cadeia de Markov

2.3 Probabilidades estacionárias

Como $\mathbf{P}$ é uma matriz de uma cadeia ergódica (ROSS, 2003), então existe o limite

$$\lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y},$$

em que todas as linhas da matriz $\mathbf{Y}$ são iguais ao vetor linha

$$\mathbf{y} = \begin{bmatrix} y_{(0,0)} & y_{(0,1)} & y_{(1,0)} & y_{(1,1)} & y_{(2,0)} & y_{(2,1)} \end{bmatrix}.$$

O vetor $\mathbf{y}$ é um vetor de probabilidade no estado estacionário,

$$\sum_i y_i = 1, \quad (2.8)$$

com todos os valores y_i estritamente positivos. Cada elemento de $\mathbf{y}$ é interpretado como sendo a fração do número de inspeções que são realizadas em cada um dos estados da cadeia após um número suficientemente grande de inspeções. Como

$$\mathbf{P}^{(j+1)} = \mathbf{P}^{(j)} \mathbf{P} \quad \text{e} \quad \lim_{j \rightarrow \infty} \mathbf{P}^{(j+1)} = \lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y},$$

então no limite a equação $\mathbf{Y} = \mathbf{Y} \mathbf{P}$ é verdadeira. Uma vez que todas as linhas de $\mathbf{Y}$ são iguais à $\mathbf{y}$, a equação $\mathbf{y} = \mathbf{y} \mathbf{P}$ também é válida no limite, e pode ser rescrita como

$$\begin{aligned} \mathbf{y} (\mathbf{P} - \mathbf{I}) &= \mathbf{0} \\ (\mathbf{P}^\top - \mathbf{I}) \mathbf{y}^\top &= \mathbf{0}^\top \end{aligned} \quad (2.9)$$

em que $\mathbf{I}$ é a matriz identidade, $\mathbf{0}$ o vetor linha de zeros. Portanto, o vetor $\mathbf{y}$ pode ser obtido a partir da resolução do sistema linear (2.9) com auxílio da equação (2.8), utilizando-se as técnicas numéricas disponíveis. O tamanho do sistema linear obtido neste estágio do trabalho permite a manipulação simbólica, e o resultado pode ser expresso em função das probabilidades de transição, com os elementos de $\mathbf{y}$ dados por:

$$\begin{aligned} y_{(0,0)} &= \frac{P_{(0,0),(0,1)} P_{(1,0),(2,1)}}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} & y_{(0,1)} &= \frac{P_{(0,0),(0,0)} P_{(1,0),(2,1)}}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} \\ y_{(1,0)} &= \frac{P_{(0,0),(1,0)} P_{(1,0),(2,1)}}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} & y_{(1,1)} &= \frac{P_{(0,0),(0,1)} P_{(1,0),(2,1)}}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} \\ y_{(2,0)} &= \frac{P_{(0,0),(1,0)} (1 - P_{(1,0),(2,1)})}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} & y_{(2,1)} &= \frac{P_{(0,0),(1,0)} P_{(1,0),(2,1)}}{P_{(0,0),(1,0)} + P_{(1,0),(2,1)}} \end{aligned}$$

2.4 Custo dos estados da cadeia

Trabalhos anteriores (TAGUCHI *et al.*, 1989; NAYEBPOUR; WOODALL, 1993; NANDI; SREEHARI, 1997; BORGES *et al.*, 2001; DASGUPTA, 2003; NANDI; SREEHARI, 1999) consideraram três componentes principais de custo: o custo de inspeção, o custo de um item não-conforme e o custo de ajuste. Neste trabalho será considerada uma estrutura similar, com a seguinte notação:

c_{ins} , é o custo de uma classificação do item inspecionado;

c_{nc} , é o custo de um item defeituoso não detectado, que segue para o consumidor ou para as próximas etapas do processo;

c_{s_nc} , é o custo de descartar um item defeituoso;

c_{s_c} , é o custo de descartar um item perfeito;

c_a , é o custo de ajustar o processo.

Os componentes c_{s_nc} e c_{s_c} são utilizados para a eventualidade dos itens descartados serem submetidos a algum processo de reaproveitamento, no qual o custo de reaproveitamento de um item conforme pode diferir daquele defeituoso. Caso isso não seja possível, basta estabelecer $c_{s_nc} = c_{s_c}$, custo este que deve ser o custo unitário de produção (desconsiderado o sistema de controle).

O custo de cada estado (w_i, z_j) pode ser escrito como

$$V_{(w_i, z_j)} = rc_{ins} + \xi_{(w_i, z_j)} + \theta_{(w_i, z_j)} + \varphi_{(w_i, z_j)}$$

em que:

rc_{ins} , é o custo de realizar r classificações do item inspecionado (presente em todos os estados da cadeia);

$\xi_{(w_i, z_j)}$, é o custo decorrente dos itens defeituosos dentre os $(m-1)$ itens não inspecionados que são enviados ao consumidor ou para as próximas etapas da produção;

$\theta_{(w_i, z_j)}$, é o custo relacionado ao item inspecionado (somente a classificação final do item é conhecida, sendo necessário considerar a presença dos erros α^* e β^*);

$\varphi_{(w_i, z_j)}$, é o custo para ajustar o processo (incluído em todos os estados em que $z = 1$).

No estado w_0z_0 todos os itens foram produzidos com o processo operando no Estado I. Dos $(m - 1)$ itens não inspecionados, o número esperado de itens defeituosos é $p_1(m - 1)$. Portanto

$$\xi_{(0,0)} = c_{nc}(m - 1)p_1.$$

O item inspecionado neste estado foi declarado conforme ($\zeta = 0$) após as r classificações, e segue seu caminho no processo. Porém, há uma probabilidade de que este item seja, de fato, defeituoso ($\Theta = 1$). Esta probabilidade é dada por

$$\begin{aligned} \Pr\{\Theta = 1 \mid \zeta = 0\} &= \frac{\Pr\{\Theta = 1, \zeta = 0\}}{\Pr\{\zeta = 1\}} \\ &= \frac{\Pr\{\zeta = 0 \mid \Theta = 1\} \Pr\{\Theta = 1\}}{\Pr\{\zeta = 1\}} \\ &= \frac{\Pr\{\zeta = 0 \mid \Theta = 1\} \Pr\{\Theta = 1\}}{\Pr\{\zeta = 0 \mid \Theta = 0\} \Pr\{\Theta = 0\} + \Pr\{\zeta = 0 \mid \Theta = 1\} \Pr\{\Theta = 1\}} \\ &= \frac{\beta^* p_1}{(1 - \alpha)(1 - p_1) + \beta^* p_1} \end{aligned}$$

O custo relacionado ao item inspecionado é

$$\theta_{(0,0)} = c_{nc} \frac{p_1 \beta^*}{(1 - p_1)(1 - \alpha^*) + p_1 \beta^*} \quad (2.11)$$

Por fim, como não há ajuste no processo,

$$\varphi_{(0,0)} = 0.$$

O estado w_0z_1 também produz todos os seus itens no Estado I. Por isso $\xi_{(0,1)} = \xi_{(0,0)}$. Já o item inspecionado, declarado não-conforme ($\zeta = 1$), é descartado e o processo parado para ajuste. O item inspecionado pode ser de fato defeituoso e agregar o custo c_{s_nc} , ou ser na realidade um item conforme, o que implica no custo c_{s_c} . Estas possibilidades são consideradas em (2.12) e (2.13).

$$\begin{aligned} \Pr\{\Theta = 1 \mid \zeta = 1\} &= \frac{\Pr\{\Theta = 1, \zeta = 1\}}{\Pr\{\zeta = 1\}} \\ &= \frac{\Pr\{\zeta = 1 \mid \Theta = 1\} \Pr\{\Theta = 1\}}{\Pr\{\zeta = 1\}} \\ &= \frac{\Pr\{\zeta = 1 \mid \Theta = 1\} \Pr\{\Theta = 1\}}{\Pr\{\zeta = 1 \mid \Theta = 0\} \Pr\{\Theta = 0\} + \Pr\{\zeta = 1 \mid \Theta = 1\} \Pr\{\Theta = 1\}} \\ &= \frac{(1 - \beta^*) p_1}{\alpha^*(1 - p_1) + \beta^*(1 - p_1)} \end{aligned} \quad (2.12)$$

$$\begin{aligned}
\Pr\{\Theta = 0 \mid \zeta = 1\} &= \frac{\Pr\{\Theta = 0, \zeta = 1\}}{\Pr\{\zeta = 1\}} \\
&= \frac{\Pr\{\zeta = 1 \mid \Theta = 0\} \Pr\{\Theta = 0\}}{\Pr\{\zeta = 1\}} \\
&= \frac{\Pr\{\zeta = 1 \mid \Theta = 0\} \Pr\{\Theta = 0\}}{\Pr\{\zeta = 1 \mid \Theta = 0\} \Pr\{\Theta = 0\} + \Pr\{\zeta = 1 \mid \Theta = 1\} \Pr\{\Theta = 1\}} \\
&= \frac{\alpha^*(1 - p_1)}{\alpha^*(1 - p_1) + \beta^*(1 - p_1)} \tag{2.13}
\end{aligned}$$

Em face do exposto, os custos $\xi_{(0,1)}$, $\theta_{(0,1)}$ e $\varphi_{(0,1)}$ são dados por

$$\begin{aligned}
\xi_{(0,1)} &= c_{nc}(m - 1)p_1 \\
\theta_{(0,1)} &= c_{s-nc} \frac{p_1(1 - \beta^*)}{(1 - p_1)\alpha^* + p_1(1 - \beta^*)} + c_{s-c} \frac{(1 - p_1)\alpha^*}{(1 - p_1)\alpha^* + p_1(1 - \beta^*)} \tag{2.14} \\
\varphi_{(0,1)} &= c_a.
\end{aligned}$$

No estado $w_1 z_0$, tem-se a garantia de que o último item do ciclo (justamente o inspecionado) foi produzido no Estado II. Porém, nos $(m - 1)$ itens não inspecionados, as quantidades produzidas no Estado I e Estado II são desconhecidas. A mudança pode já se manifestar no primeiro item produzido (neste caso todos os $m - 1$ itens são produzidos no Estado II) ou mesmo ocorrer apenas no último item do ciclo (neste caso, todos os $m - 1$ itens são produzidos no Estado I). Sendo i indicadora do primeiro item produzido no Estado II, pode-se expressar $\xi_{(0,1)}$ como

$$\xi_{(1,0)} = c_{nc} \sum_{i=1}^m \frac{(1 - \pi)^{i-1} \pi}{1 - (1 - \pi)^m} [(i - 1)p_1 + (m - i)p_2] \tag{2.15}$$

O custo relacionado ao item inspecionado é semelhante à (2.11), com a ressalva de que o item inspecionado, neste caso, é produzido no Estado II. Considerando-se o fato de que em $w_1 z_0$ não ocorre ajuste do processo, pode-se escrever

$$\begin{aligned}
\theta_{(1,0)} &= c_{nc} \frac{p_2 \beta^*}{(1 - p_2)(1 - \alpha^*) + p_2 \beta^*} \\
\varphi_{(1,0)} &= 0.
\end{aligned}$$

Em relação aos $(m - 1)$ itens produzidos no estado $w_1 z_1$ e que não sofrem inspeção, o custo gerado é o mesmo dado por (2.15). Há ajuste do processo, e o custo decorrente de descartar o item inspecionado segue a estrutura de (2.14), com a substituição de p_1 por

p_2 . Então,

$$\begin{aligned}\xi_{(1,1)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [(i-1)p_1 + (m-i)p_2] \\ \theta_{(1,1)} &= c_{s-nc} \frac{p_2(1-\beta^*)}{(1-p_2)\alpha^* + p_2(1-\beta^*)} + c_{s-c} \frac{(1-p_2)\alpha^*}{(1-p_2)\alpha^* + p_2(1-\beta^*)} \\ \varphi_{(1,1)} &= c_a.\end{aligned}$$

Os custos dos estados w_2z_0 e w_2z_1 são correspondentes aos estados w_0z_0 e w_0z_1 , já que não há mudança $I \rightarrow II$ dentro destes ciclos. Porém, a probabilidade de produção de itens defeituosos deve ser alterada para p_2 . Assim,

$$\begin{aligned}\xi_{(2,0)} &= c_{nc}(m-1)p_2 \\ \theta_{(2,0)} &= c_{nc} \frac{p_2\beta^*}{(1-p_2)(1-\alpha^*) + p_2\beta^*} \\ \varphi_{(2,0)} &= 0. \\ \xi_{(2,1)} &= c_{nc}(m-1)p_2 \\ \theta_{(2,1)} &= c_{s-nc} \frac{p_2(1-\beta^*)}{(1-p_2)\alpha^* + p_2(1-\beta^*)} + c_{s-c} \frac{(1-p_2)\alpha^*}{(1-p_2)\alpha^* + p_2(1-\beta^*)} \\ \varphi_{(2,1)} &= c_a.\end{aligned}$$

2.5 Custo médio do sistema de controle

Nos trabalhos anteriores, a Teoria da Renovação é aplicada ao ciclo entre ajustes. Porém, a cadeia de Markov proposta trabalha com o ciclo entre inspeções. Esta seção visa mostrar que a abordagem com o ciclo entre inspeções também é válida, e formular a expressão do custo médio do sistema de controle.

Seja o processo de renovação $\{N(t), t \geq 0\}$ a contagem do número de ajustes já realizados até a produção do t -ésimo item. Seja a variável aleatória T^* definida como o número de itens produzidos entre ajustes sucessivos e T_i^* o número de itens produzidos entre os ajustes $i-1$ e i (com $i \geq 1$). Considere que a cada ajuste pode-se associar um custo ao sistema de controle, representado pela variável aleatória V^* , sendo V_i^* o custo associado ao i -ésimo ajuste. Definamos

$$V^*(t) = \sum_{i=1}^{N(t)} V_i^*$$

como o custo total do sistema de controle após t itens produzidos, $E[V^*] = E[V_i^*]$ e $E[T^*] = E[T_i^*]$. Então, pela Teoria da Renovação (detalhes em Ross (2003)),

$$\lim_{t \rightarrow \infty} \frac{V^*(t)}{t} = \lim_{t \rightarrow \infty} \frac{E[V^*(t)]}{t} = \frac{E[V^*]}{E[T^*]} \quad (2.16)$$

Em outras palavras, após um número suficientemente grande de itens produzidos, o custo médio do sistema de controle por item produzido ($E[V^*(t)]/t$) é igual ao custo médio do ciclo entre ajustes ($E[V^*]$) dividido pelo número médio de itens produzidos por ciclo entre ajustes ($E[T^*]$). Porém, as informações fornecidas pela cadeia de Markov construída dizem respeito aos itens produzidos *entre inspeções*, e não *entre ajustes*. Definamos V uma variável aleatória que representa o custo entre duas inspeções sucessivas, sendo V_j o custo entre as inspeções $j - 1$ e j . Considere l o número de inspeções ocorridas até que o k -ésimo ajuste tenha ocorrido. O custo total após k ajustes será o somatório dos custos das l inspeções realizadas ($l \geq k$), ou seja

$$\sum_{i=1}^k V_i^* = \sum_{j=1}^l V_j. \quad (2.17)$$

Seja T uma variável aleatória representando o número de itens produzidos entre inspeções, sendo T_j o número de itens produzidos entre as inspeções $j - 1$ e j . Utilizando o mesmo argumento, o número de itens produzidos até o k -ésimo ajuste será o somatório dos itens produzidos nas l inspeções realizadas,

$$\sum_{i=1}^k T_i^* = \sum_{j=1}^l T_j. \quad (2.18)$$

Quando $k \rightarrow \infty$, seguramente $l \rightarrow \infty$. Assim, utilizando as equações (2.17) e (2.18), pode-se rescrever a equação (2.16) como

$$\frac{E[V^*]}{E[T^*]} = \frac{\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k V_i^*}{\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k T_i^*} = \frac{\lim_{k \rightarrow \infty} \sum_{i=1}^k V_i^*}{\lim_{k \rightarrow \infty} \sum_{i=1}^k T_i^*} = \frac{\lim_{l \rightarrow \infty} \sum_{j=1}^l V_j}{\lim_{l \rightarrow \infty} \sum_{j=1}^l T_j} = \frac{\lim_{l \rightarrow \infty} \frac{1}{l} \sum_{j=1}^l V_j}{\lim_{l \rightarrow \infty} \frac{1}{l} \sum_{j=1}^l T_j} = \frac{E[V]}{E[T]}.$$

Portanto, o custo médio do sistema de controle por item produzido pode ser expresso como a razão entre o custo médio por ciclo de inspeção dividido pelo número médio de itens produzidos por ciclo de inspeção. Os valores assumidos pelas variáveis T e V são discretos e estão diretamente relacionados aos estados da cadeia de Markov.

Em cada um dos ciclos de inspeção, são produzidos m itens, logo $E[T] = m$. Já a esperança do custo dos ciclos é dada por

$$E[V] = \sum_{i=0}^2 \sum_{j=0}^1 y_{(w_i, z_j)} V_{(w_i, z_j)}$$

em que $y_{(w_i, z_j)}$ é a probabilidade estacionária de w_i, z_j (obtida na Seção 2.3) e $V_{(w_i, z_j)}$ o custo do estado w_i, z_j . Assim, a expressão do custo médio pode ser escrita como

$$\begin{aligned} C(m, r, s) &= \frac{E[V]}{E[T]} = \frac{1}{m} \sum_{i=0}^2 \sum_{j=0}^1 y_{(w_i, z_j)} V_{(w_i, z_j)} \\ &= \frac{1}{m} \left(\sum_{i=0}^2 \sum_{j=0}^1 y_{(w_i, z_j)} (rc_{ins} + \xi_{(w_i, z_j)} + \theta_{(w_i, z_j)} + \varphi_{(w_i, z_j)}) \right) \end{aligned}$$

2.6 Exemplos numéricos e discussões

Para ilustrar o modelo proposto com um exemplo numérico, considere um processo de manufatura com os parâmetros da Tabela 2.1, sendo que os custos estão expressos em unidades monetárias.

Tabela 2.1: Parâmetros-base para os exemplos numéricos.

$p_1 = 0,01$	$\pi = 0,0001$	$c_{ins} = 0,25$	$c_{s_nc} = 2,00$
$p_2 = 0,20$	$\alpha = 0,01$	$c_{nc} = 20$	$c_{s_c} = 2,00$
	$\beta = 0,01$	$c_a = 100$	

Para simplificar a execução do modelo, considere que o item será julgado conforme caso mais da metade das r classificações sejam conformes, ou seja, $s = \lfloor r/2 \rfloor + 1$. A Figura 2.3 mostra o custo médio do sistema de controle por item produzido em função do intervalo entre inspeções (m) e do número de classificações repetidas a que o item inspecionado é submetido (r), e a Tabela 2.2 apresenta os custos ótimos obtidos. O ótimo global corresponde a $m^\circ = 31$ e $r^\circ = 3$, que resulta em um custo médio de 0,3117. Caso uma única classificação do item inspecionado seja realizada ($r = 1$), o intervalo ótimo é $m^\circ = 36$ e o custo médio 0,3237 (3,8% maior que aquele com múltiplas classificações).

Considere agora que os erros α e β são desprezados durante a determinação da estratégia ótima de controle. Para esta situação, tem-se $m^\circ = 25$ e $r^\circ = 1$. Porém, adotar esta estratégia em um sistema de inspeção que possui tais erros, sendo $\alpha = \beta = 1\%$, gera um custo médio real $C = 0,3313$. Este custo é 2,4% maior que o custo da estratégia

Figura 2.3: Gráfico de custo médio C em função de m e r .

Tabela 2.2: Custo e intervalo de inspeção ótimos em função do número de classificações repetidas.

r	s	m°	C°
1	1	36	0,3237
2	2	47	0,3503
3	2	31	0,3117
4	3	34	0,3202
5	3	35	0,3260
6	4	38	0,3328

considerando os erros e com uma única classificação (e 6,3% maior que aquela obtida considerando classificações repetidas).

O critério $s = \lfloor r/2 \rfloor + 1$, utilizado na obtenção da Figura 2.3, é uma restrição a um parâmetro que também pode ser utilizado para minimizar o custo médio. Esta escolha produziu um comportamento não esperado: o uso de duas classificações repetidas gerou um custo médio maior que aquele com apenas uma classificação, enquanto o uso de três classificações repetidas gerou um custo médio menor que aquele com apenas uma classificação. Há dois comportamentos previsíveis. No primeiro, o aumento do número de classificações repetidas leva ao contínuo aumento do custo médio (esta situação será necessariamente observada quando $\alpha = \beta = 0\%$). No segundo, o aumento do número de classificações repetidas reduz o custo médio até um valor mínimo, a partir do qual o incremento no número de classificações repetidas não traz mais benefícios ao sistema de inspeção, implicando apenas no aumento do custo médio.

A Tabela 2.3 mostra o que acontece com α^* e β^* quando três critérios são utilizados: a) $s = 1$ (o mais permissivo, basta uma classificação conforme para que o item seja declarado conforme); b) $s = \lfloor r/2 \rfloor + 1$ (o critério da maioria das classificações conforme); c) $s = r$ (o mais exigente, somente declara um item como conforme se todas as classificações forem conformes). Observe que, nos critérios *a* e *c*, enquanto um dos erros tende para zero, o outro tende para 100%. Já aplicando-se o critério *b*, apesar de quedas e aumentos alternados em α^* e β^* , ambos tendem a zero com o aumento de r . A adoção do critério *b* leva ao aumento de α^* quando $r = 2$. Como no exemplo numérico empregado o processo opera a maior parte do tempo no Estado I (em que há mais itens conformes que não-conformes), a elevação de α^* quando $r = 2$ explica o aumento do custo médio quando comparado ao custo de uma única classificação. Tornando s também uma variável de otimização, a estratégia ótima torna-se $m^\circ = 28$, $r^\circ = 2$ e $s^\circ = 1$, e o custo médio $C = 0,3031$, (2,8% menor que melhor estratégia obtida até então).

Tabela 2.3: Impacto de diferentes políticas para declarar o item inspecionado conforme sobre α^* e β^* quando $\alpha = 10\%$ e $\beta = 10\%$.

	$s = 1$		$s = \lfloor r/2 \rfloor + 1$		$s = r$	
r	α^*	β^*	α^*	β^*	α^*	β^*
1	10,000%	10,000%	10,000%	10,000%	10,000%	10,000%
2	1,000%	19,000%	19,000%	1,000%	19,000%	1,000%
3	0,100%	27,100%	2,800%	2,800%	27,100%	0,100%
4	0,010%	34,390%	5,230%	0,370%	34,390%	0,010%
5	0,001%	40,951%	0,856%	0,856%	40,951%	0,001%

A Tabela 2.4 mostra os valores ótimos para m , r e s variando-se apenas um parâmetro por vez, com os demais mantidos fixos nos valores da Tabela 2.1. Apesar da faixa de variação escolhida ser arbitrária, respeitam-se as seguintes condições que tornam as combinações utilizadas razoáveis:

- a fração de itens defeituosos no Estado II é superior à do Estado I ($p_2 > p_1$);
- o custo de descartar um item conforme é o mesmo de um item defeituoso ($c_{s_nc} = c_{s_c}$);
- o custo de ajuste é muito maior que o custo de inspeção ($c_a \gg c_{ins}$); e
- o custo de enviar um item defeituoso para o mercado é maior que o custo de descartá-lo ($c_{nc} > c_{s_nc}$).

Tabela 2.4: Determinação de m° , r° , s° e C para a variação individual dos parâmetros do modelo.

$\pi \times 10^3$	C°	m°	r°	s°	p_1	C°	m°	r°	s°	p_2	C°	m°	r°	s°
0,100	0,3031	28	2	1	0,010	0,3031	28	2	1	0,200	0,3031	28	2	1
0,325	0,3962	16	2	1	0,045	1,0579	55	2	1	0,350	0,3008	28	2	1
0,550	0,4641	12	2	1	0,080	1,7767	82	2	1	0,500	0,2974	30	2	1
0,775	0,5219	10	2	1	0,115	2,4736	118	2	1	0,650	0,2934	31	2	1
1,000	0,5737	9	2	1	0,150	3,1482	179	2	1	0,800	0,2890	33	2	1
α	C°	m°	r°	s°	β	C°	m°	r°	s°	c_{ins}	C°	m°	r°	s°
0,010	0,3031	28	2	1	0,010	0,3031	28	2	1	0,250	0,3031	28	2	1
0,033	0,3063	29	2	1	0,033	0,3042	27	2	1	0,563	0,3234	34	2	1
0,055	0,3122	30	3	1	0,055	0,3054	26	2	1	0,875	0,3399	41	1	1
0,078	0,3131	31	3	1	0,078	0,3066	25	2	1	1,188	0,3473	43	1	1
0,100	0,3147	31	3	1	0,100	0,3078	24	2	1	1,500	0,3543	46	1	1
c_{nc}	C°	m°	r°	s°	c_a	C°	m°	r°	s°	$c_{s_{nc}=c_s_c}$	C°	m°	r°	s°
5	0,1089	60	2	1	100	0,3031	28	2	1	1,00	0,3027	28	2	1
10	0,1783	41	2	1	575	0,4540	61	2	1	2,00	0,3031	28	2	1
15	0,2422	33	2	1	1050	0,5635	82	2	1	3,00	0,3036	28	2	1
20	0,3031	28	2	1	1525	0,6583	100	2	1	4,00	0,3040	28	2	1
25	0,3620	24	2	1	2000	0,7449	116	2	1	5,00	0,3045	28	2	1

Os parâmetros r° e s° mostram-se insensíveis aos custos c_{nc} , c_a e $c_{s_{nc}}$ nas faixas utilizadas, enquanto o aumento do custo de cada classificação (c_{ins}) leva a uma redução do número de classificações a que o item inspecionado é submetido, como esperado. Já o parâmetro m° apresenta um comportamento distinto para cada um dos custos avaliados. Um aumento no custo de inspeção (c_{ins}) ou no custo de ajuste (c_a) leva ao aumento de m° , reduzindo a frequência das inspeções e a ocorrência de ajustes desnecessários (ajustar o processo que opera no Estado I). Um aumento no custo de envio de itens defeituosos para o mercado (c_{nc}) leva à redução do intervalo entre inspeções, ou seja, ao aumento na frequência das inspeções, com o objetivo de detectar mais prontamente a ocorrência da mudança de estado do processo. Para a faixa pesquisada, $c_{s_{nc}}$ não altera a estratégia ótima de inspeção.

Um aumento na fração de itens defeituosos produzidos no Estado I (p_1) leva a um aumento no intervalo entre inspeções. Isso ocorre porque, com mais itens não-conformes sendo produzidos no Estado I, mais itens não-conformes seriam amostrados, o que levaria ao ajuste desnecessário. Já o aumento na fração de itens defeituosos produzidos no Estado II (p_2) aumenta a probabilidade de detectar a ocorrência da mudança, pois mais itens não-conformes são amostrados no Estado II. Como resposta a isso, as inspeções tornam-se menos frequentes, ou seja, há um aumento no valor de m° . Estes comportamentos ficam claros quando o objetivo do sistema de controle é revisto: detectar a mudança de estado

do processo tão logo ela ocorra, sendo π o parâmetro que controla a mudança do processo. Como mostra a Tabela 2.4, o aumento deste parâmetro (π) leva à redução no intervalo entre inspeções, pois em média a mudança de estado ($I \rightarrow II$) passa a ocorrer após um menor número de itens produzidos pelo processo.

O uso de classificações repetidas (r) e de um critério para declarar um item conforme (s) atuam em resposta à existência do erro α no sistema de inspeção. Um aumento em α eleva a probabilidade de declarar um item conforme como não-conforme. A estratégia ótima contrapõe este aumento exigindo, proporcionalmente, que um número menor de classificações resulte em conformidade para declarar o item conforme. Para $\alpha = 1\%$, pelo menos $s^\circ/r^\circ = 50\%$ das classificações devem ser conformes para julgar o item conforme. Quando $\alpha = 10\%$, a razão diminui para $s^\circ/r^\circ = 33,3\%$.

2.7 Considerações Finais

Nos casos em que existem erros de classificação no sistema de inspeção por atributos, considerar o uso de classificações repetidas e um critério para o julgamento do item como conforme proporciona uma estratégia de controle tão ou mais eficiente que os modelos de inspeção simples. O que determina o quão mais eficiente será o modelo desenvolvido é a combinação específica dos fatores de custo, das probabilidades de produção nos Estados I e II, da probabilidade de mudança $I \rightarrow II$ e dos erros de inspeção do sistema de controle. Em linhas gerais, quanto menor for o custo de uma inspeção (c_{ins}) e quanto maior a incidência dos erros α e β , maior será a diferença observada entre a estratégia obtida com o modelo deste Capítulo e aqueles de inspeção simples.

Os exemplos calculados neste Capítulo podem ser obtidos com o programa apresentado no Apêndice A. Esta versão, implementada no Matlab e também disponível em www.anderson.trindade.nom.br/tese, corresponde ao modelo apresentado no Capítulo 4. A particularização de alguns dos parâmetros permite a simulação do modelo tal qual apresentado até aqui. Por isso, antes da execução do programa, recomenda-se a leitura dos próximos capítulos para que os novos parâmetros sejam definidos.

Até este ponto, p_1 e p_2 foram considerados constantes. Mas o que aconteceria se, uma vez operando na condição fora de controle, a qualidade da fabricação sofresse deterioração a cada item produzido? Este será o assunto tratado no próximo Capítulo.

CAPÍTULO 3

Controle *on-line* para atributos considerando deterioração da qualidade [†]

No modelo proposto neste Capítulo, a fração de itens defeituosos no Estado II torna-se função do número de itens produzidos neste estado. Além disso, incorpora-se uma restrição estatística à determinação dos parâmetros ótimos de controle. A seguir, as principais suposições adotadas:

- o processo inicia (ou reinicia) a produção no Estado I (sob controle, com fração de itens defeituosos constante $p_1 \geq 0$) e, após a mudança, a fração de itens defeituosos passa a ser descrita por uma função não-decrescente p_2 , $p_1 < p_2 = \psi(x) \leq 1$ (em que x é o número de itens produzidos após a mudança de estado do processo);
- a mudança do Estado I para o Estado II ocorre segundo uma distribuição geométrica com parâmetro π ;
- o processo de monitoramento consiste na inspeção de um único item a cada m produzidos;

[†] O texto base deste Capítulo foi publicado na revista *Applied Stochastic Models in Business and Industry*, sob o título “Monitoring process for attributes with quality deterioration and diagnosis errors”, Trindade *et al.* (2007b). O artigo é apresentado no Apêndice C.

- o sistema de inspeção está sujeito a erros de classificação: α é a probabilidade de classificar um item conforme como defeituoso e β a probabilidade de classificar um item defeituoso como conforme;
- o item inspecionado pode ser submetido a r classificações independentes (sendo classificado em cada uma como conforme ou defeituoso) e o tempo necessário para realizar estas classificações é desprezível;
- tão logo o item inspecionado seja declarado defeituoso (segundo um critério a ser estabelecido), a parada do processo ocorre imediatamente (sem atraso) e o item inspecionado é descartado.
- a obtenção dos parâmetros ótimos de controle estão sujeitas a uma fração máxima de itens defeituosos produzidos pelo processo, $\phi_{\max}$.

Note que não há modificações em relação aos erros de classificação, quando comparado ao Capítulo anterior, sendo então desnecessário redefinir os erros α^* e β^* . Por isso, parte-se para a definição da cadeia de Markov.

3.1 A cadeia de Markov

O sistema de inspeção pode ser modelado como uma cadeia de Markov estacionária considerando o espaço de estados discretos

$$\Omega = \{(0,0), (0,1), (1,0), (1,1), \dots, (w_i, z_0), (w_i, z_1), \dots, (n,0), (n,1)\}.$$

Cada estado é descrito por um par de números inteiros w e z , totalizando $(2n+2)$ estados. O inteiro z mantém o significado do Capítulo anterior: $z = 0$ (z_0) o item foi classificado como conforme, $z = 1$ (z_1) o item foi classificado como defeituoso. O inteiro w expressa agora o número de inspeções realizadas após a mudança para o Estado II ($w = 0, 1, \dots, n$). Quando $w = 0$ (w_0), o processo permanece no Estado I. Quando $w > 0$, w representa o número de inspeções (incluindo a atual) que foram realizadas no Estado II. Note que, teoricamente, n pode ser infinito. Por enquanto, suponha que há um n suficientemente grande, mas finito, que pode ser utilizado para a representação desta cadeia. Detalhes da obtenção de n serão apresentados nas seções 3.1.2 e 3.4.1. A Figura 3.1 relaciona o processo aos estados da cadeia.

Figura 3.1: Representação do sistema de inspeção e respectivos estados da cadeia de Markov.

Note que, ao contrário da Figura 2.1, na qual a fração de defeitos no Estado II é constante (p_2), na Figura 3.1 a fração de defeitos no Estado II é descrita por uma função não decrescente $\psi(x)$.

3.1.1 Probabilidades de transição

Com o desenvolvimento realizado no Capítulo anterior já se sabe que, na matriz de transição (3.1), as linhas pares (com ajuste) são iguais à primeira linha de (3.1). Pela nova definição de w , uma vez no estado $w_i z_0$, com $i \geq 1$, as transições só podem ocorrer para os estados $w_{i+1} z_0$ ou $w_{i+1} z_1$. Já as transições de $w_n z_0$ para $w_n z_0$ e $w_n z_1$ em (3.1) só podem ser definidas quando o critério para n for determinado.

$$\mathbf{P}_n = \begin{matrix} & w_0 z_0 & w_0 z_1 & w_1 z_0 & w_1 z_1 & w_2 z_0 & w_2 z_1 & \cdots & w_n z_0 & w_n z_1 \\ \begin{matrix} w_0 z_0 \\ w_0 z_1 \\ w_1 z_0 \\ w_1 z_1 \\ \vdots \\ w_{n-1} z_0 \\ w_{n-1} z_1 \\ w_n z_0 \\ w_n z_1 \end{matrix} & \begin{pmatrix} P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & \cdots & 0 & 0 \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ \vdots & & & & & & \cdots & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \end{matrix} \quad (3.1)$$

As probabilidades $P_{(0,0),(0,0)}$ e $P_{(0,0),(0,1)}$ são idênticas a (2.5) e (2.7), já que a diferença do modelo atual só se manifesta após a mudança para o Estado II.

$$P_{(0,0),(0,0)} = (1 - \pi)^m [(1 - p_1)(1 - \alpha^*) + p_1\beta^*].$$

$$P_{(0,0),(0,1)} = (1 - \pi)^m [(1 - p_1)\alpha^* + p_1(1 - \beta^*)].$$

Para determinar a probabilidade $P_{(0,0),(1,0)}$ deve-se considerar todos os possíveis pontos de mudança dentro do ciclo de inspeção. A particularidade agora é que a probabilidade do item inspecionado ser defeituoso é função do número de itens produzidos após a mudança de estado, ou seja, é função do ponto em que a mudança ocorre. Com isso, pode-se escrever:

$$P_{(0,0),(1,0)} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)(1 - \alpha^*) + \psi(m - i + 1)\beta^*]$$

em que $\bar{\psi}(x) = 1 - \psi(x)$, e $\psi(m - i + 1)$ corresponde à função ψ avaliada no ponto $(m - i + 1)$. Considere que a mudança ocorreu no primeiro item produzido no ciclo (i.e., $i = 1$), então a probabilidade do item inspecionado ser defeituoso corresponderá àquela do m -ésimo item produzido após a mudança (i.e., $\psi(m)$). Caso a mudança tenha ocorrido no último item produzido no ciclo (i.e., $i = m$), o item inspecionado será justamente o primeiro item produzido no Estado II (i.e., $\psi(1)$). Equação similar pode ser escrita para $P_{(0,0),(1,1)}$, com a ressalva de que agora o item inspecionado foi classificado como defeituoso.

$$P_{(0,0),(1,1)} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)\alpha^* + \psi(m - i + 1)(1 - \beta^*)].$$

As equações (3.2)-(3.5) indicam que um estado no qual há ajuste comporta-se, após o ajuste, de maneira similar ao estado w_0z_0 .

$$P_{(j,1),(0,0)|j=1\dots n} = P_{(0,0),(0,0)}; \quad (3.2)$$

$$P_{(j,1),(0,1)|j=1\dots n} = P_{(0,0),(0,1)}; \quad (3.3)$$

$$P_{(j,1),(1,0)|j=1\dots n} = P_{(0,0),(1,0)}; \quad (3.4)$$

$$P_{(j,1),(1,1)|j=j\dots n} = P_{(0,0),(1,1)}. \quad (3.5)$$

As probabilidades $P_{(j-1,0),(j,0)|j=2\dots n}$ e $P_{(j-1,0),(j,1)|j=2\dots n}$ correspondem às transições nas quais a mudança do estado do processo já aconteceu, mas o ponto no qual esta mudança ocorreu

influencia a probabilidade do item inspecionado. Assim,

$$P_{(j-1,0),(j,0)|j=2\dots n} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm-i+1)(1-\alpha^*) + \psi(jm-i+1)\beta^*] \quad (3.6)$$

$$P_{(j-1,0),(j,1)|j=2\dots n} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm-i+1)\alpha^* + \psi(jm-i+1)(1-\beta^*)] \quad (3.7)$$

3.1.2 Determinação de n

Resta definir as probabilidades $P_{(n,0),(n,0)}$ e $P_{(n,0),(n,1)}$ em (3.1), o que parece contrariar a maneira com que a cadeia de Markov foi proposta, onde infinitos estados podem ser estabelecidos. Entretanto, é fácil perceber que após a mudança para o Estado II e com $n \rightarrow \infty$,

$$\lim_{n \rightarrow \infty} \Pr\{(w_1 z_0), (w_2 z_0), (w_3 z_0), \dots, (w_n z_0)\} = 0. \quad (3.8)$$

Em outras palavras, quanto mais itens são produzidos após a mudança para o Estado II, mais inspeções são feitas e menos provável se torna a sequência (3.8), já que em algum instante um ajuste deve ocorrer. Com base nisso, uma cadeia finita, para um n suficientemente grande, deve produzir resultados numéricos similares ao de uma cadeia infinita.

A Figura 3.2 ilustra as inspeções em um processo sujeito à mudança aleatória e deterioração da qualidade.

Figura 3.2: Fração de itens não-conformes em um processo sujeito a mudança de estado e deterioração da qualidade.

Observe que as probabilidades de produzir um item defeituoso são muito próximas nas inspeções A e B . Seja ψ_A e ψ_B a função ψ avaliada respectivamente em A e B . Seja $\psi_A \approx \psi_B \equiv \psi_n$ e $\bar{\psi}_n = 1 - \psi_n$. Substituindo estes valores nas equações (3.6) e (3.7)

obtém-se

$$\begin{aligned}
P_{(j-1,0),(j,0)|j=n+1,\dots} &= \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}_n(1-\alpha^*) + \psi_n\beta^*] \\
&= [\bar{\psi}_n(1-\alpha^*) + \psi_n\beta^*] \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \\
&= \bar{\psi}_n(1-\alpha^*) + \psi_n\beta^* \\
P_{(j-1,0),(j,1)|j=n+1,\dots} &= \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}_n\alpha^* + \psi_n(1-\beta^*)] \\
&= [\bar{\psi}_n\alpha^* + \psi_n(1-\beta^*)] \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \\
&= \bar{\psi}_n\alpha^* + \psi_n(1-\beta^*).
\end{aligned}$$

Portanto, após n inspeções no Estado II, para um n suficientemente grande, as transições para estados subsequentes da cadeia tornam-se praticamente constantes, o que permite escrever

$$P_{(n,0),(n,0)} = P_{(n-1,0),(n,0)} \quad (3.9)$$

$$P_{(n,0),(n,1)} = P_{(n-1,0),(n,1)} \quad (3.10)$$

As inspeções $\{n+1, n+2, \dots\}$ podem ocorrer, mas estas inspeções se comportam probabilisticamente como aquela realizada na n -ésima inspeção. As equações (3.9) e (3.10) representam o truncamento que permite o uso de uma cadeia finita para o modelo proposto. A Figura 3.3 mostra um diagrama de transições entre estados para uma cadeia de tamanho n .

Figura 3.3: Transições entre estados da cadeia de Markov.

A equação (3.8) ou o argumento apresentado na Figura 3.2 são candidatos ao critério de

determinação de n . Por exemplo:

$$\min\{n: \Pr\{(w_1 z_0), (w_2 z_0), (w_3 z_0), \dots, (w_n z_0)\} < \epsilon_1\}, \quad \text{ou} \quad (3.11)$$

$$\min\{n: |\psi_n - \psi_{n-1}| < \epsilon_2\}, \quad (3.12)$$

em que ϵ_1 e ϵ_2 são parâmetros de convergência. No entanto, (3.11) e (3.12) não incluem a parte econômica do modelo. Por isso a formalização do critério para determinar n só será apresentado na Seção 3.4.1, após a definição do custo de cada estado da cadeia e da função custo médio.

De qualquer forma, uma vez definida a matriz de transição $\mathbf{P}_n$, o vetor $\mathbf{y}$ de probabilidades estacionárias é obtido como apresentado no Capítulo 2, a partir da resolução do sistema linear (2.9) com auxílio da equação (2.8).

3.2 Custo dos estados da cadeia

Os componentes e a estrutura do custo de cada estado da cadeia são os mesmos apresentados na Seção 2.4. Para o estado (w_0, z_0) , os seguintes custos são considerados: i) o custo relacionado aos itens defeituosos não-detectados (3.13); ii) o custo relacionado ao item inspecionado, que apesar de classificado como conforme, pode ser um item defeituoso enviado à próxima etapa do processo (3.14), e iii) o processo não é ajustado (3.15).

$$\xi_{(0,0)} = c_{nc}(m-1)p_1 \quad (3.13)$$

$$\theta_{(0,0)} = c_{nc} \frac{p_1 \beta^*}{(1-p_1)(1-\alpha^*) + p_1 \beta^*} \quad (3.14)$$

$$\varphi_{(0,0)} = 0. \quad (3.15)$$

No estado (w_0, z_1) , deve-se considerar que há ajuste do processo e que o item inspecionado será descartado, já que foi classificado como defeituoso (porém, a sua real condição é desconhecida).

$$\begin{aligned} \xi_{(0,1)} &= c_{nc}(m-1)p_1 \\ \theta_{(0,1)} &= c_{s-nc} \frac{p_1(1-\beta^*)}{(1-p_1)\alpha^* + p_1(1-\beta^*)} + c_{s-c} \frac{(1-p_1)\alpha^*}{(1-p_1)\alpha^* + p_1(1-\beta^*)} \\ \varphi_{(0,1)} &= c_a. \end{aligned}$$

O estado (w_1, z_0) corresponde à primeira inspeção após a mudança: parte dos itens são

produzidos no Estado I, parte no Estado II. Aqueles produzidos no Estado II têm a probabilidade serem defeituosos expressa em função do número de itens produzidos após a mudança. Daí a necessidade de considerar todos os possíveis pontos de mudança i . Com o item inspecionado declarado conforme e sem ajuste do processo, pode-se escrever

$$\begin{aligned}\xi_{(1,0)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \\ \theta_{(1,0)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)\beta^*}{\bar{\psi}(m-i+1)(1-\alpha^*) + \psi(m-i)\beta^*} \right] \\ \varphi_{(1,0)} &= 0.\end{aligned}\tag{3.16}$$

O estado (w_1, z_1) apresenta o mesmo custo (3.16) para os itens não inspecionados. Deve-se considerar agora o item inspecionado classificado como defeituoso e o ajuste do processo. Então,

$$\begin{aligned}\xi_{(1,1)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \\ \theta_{(1,1)} &= c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)(1-\beta^*)}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] \\ &\quad + c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(m-i+1)\alpha^*}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] \\ \varphi_{(1,1)} &= c_a.\end{aligned}$$

Nos estados $(w_j, z_0)|_{j=2\dots n}$, todos os itens são produzidos no Estado II, e dependem do ponto em que ocorreu a mudança no estado $w_1 z_0$. O item inspecionado é declarado conforme e não há ajuste do processo.

$$\xi_{(j,0)|_{j=2\dots n}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1)\tag{3.17}$$

$$\theta_{(j,0)|_{j=2\dots n}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm - i + 1)\beta^*}{\bar{\psi}(jm - i + 1)(1-\alpha^*) + \psi(jm - i + 1)\beta^*} \right]\tag{3.18}$$

$$\varphi_{(j,0)|_{j=2\dots n}} = 0.$$

A correção das equações (3.17) e (3.18) é de difícil conferência. Como ilustração, considere a representação da Figura 3.4: um item é inspecionado a cada $m = 5$ itens produzidos e a mudança de estado ocorreu no terceiro item produzido ($i = 3$).

Figura 3.4: Representação ponto de mudança do estado sob controle para o estado fora de controle.

Considerando o ciclo da segunda inspeção após a mudança ($w_2, j = 2$), 4 itens (i.e., $m - 1$) são enviados à próxima etapa do processo sem inspeção. As probabilidades destes itens serem defeituosos são dadas por $\psi(4)$, $\psi(5)$, $\psi(6)$ e $\psi(7)$. Portanto, o somatório interno da equação (3.17) deve refletir isso quando $j = 2$ e $i = 3$:

$$\begin{aligned} \left(\sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1) \right)_{\substack{j=2 \\ i=3}} &= \sum_{l=1}^{5-1} \psi((2-1) \times 5 - 3 + l + 1) \\ &= \sum_{l=1}^4 \psi(3 + l) \\ &= \psi(4) + \psi(5) + \psi(6) + \psi(7) \end{aligned}$$

A equação (3.18) diz respeito ao item inspecionado. Segundo a Figura 3.4, a probabilidade do item inspecionado ser defeituoso é $\psi(8)$, quando considerados $j = 2$ e $i = 3$. Calculando a probabilidade correspondente em (3.18),

$$\psi(jm - i + 1) = \psi(2 \times 5 - 3 + 1) = \psi(8)$$

O somatório $\sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m}$ nas equações (3.17) e (3.18) corresponde à probabilidade de ocorrência da mudança em cada um dos i itens produzidos em $w_1 z_0$ (ver Figura 3.4).

Os estados $(w_j, z_1)|_{j=2 \dots n}$ apresentam estrutura similar aos estados $(w_j, z_0)|_{j=2 \dots n}$. Porém, o item inspecionado é classificado como defeituoso e há ajuste do processo.

$$\begin{aligned} \xi_{(j,1)|_{j=2 \dots n}} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1) \\ \theta_{(j,1)|_{j=2 \dots n}} &= c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm - i + 1)(1 - \beta^*)}{\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i + 1)(1 - \beta^*)} \right] \\ &\quad + c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(jm - i + 1)\alpha^*}{\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i)(1 - \beta^*)} \right] \\ \varphi_{(j,1)|_{j=2 \dots n}} &= c_a. \end{aligned}$$

3.3 Custo médio do sistema de controle

Como mostrado no Capítulo anterior, o custo médio do sistema de controle pode ser expresso como a razão entre a esperança do custo dos estados da cadeia e a esperança do número de itens produzidos por ciclo de inspeção. Em cada um dos ciclos de inspeção são produzidos $E[T] = m$ itens. Já a esperança do custo dos ciclos é dada por

$$E[V] = \sum_{i=0}^n \sum_{j=0}^1 y_{(w_i, z_j)} V_{(w_i, z_j)}$$

Para um número suficientemente grande de inspeções, o custo médio do sistema de controle por item produzido pode ser expresso por

$$C(m, r, s, n) = \frac{E[V]}{E[T]} = \frac{1}{m} \left(\sum_{i=0}^n \sum_{j=0}^1 y_{(w_i, z_j)} (rc_{ins} + \xi_{(w_i, z_j)} + \theta_{(w_i, z_j)} + \varphi_{(w_i, z_j)}) \right) \quad (3.19)$$

A determinação da política ótima de controle consiste na determinação de m, r, s que minimizam (3.19). Ou seja,

$$(m^\circ, r^\circ, s^\circ) = \arg \min_{(m, r, s)} C, \text{ sujeito a } \phi < \phi_{\max}$$

em que ϕ é a proporção esperada de itens defeituosos produzidos pelo processo e $\phi_{\max}$ seu máximo valor permitido. A expressão de ϕ pode ser obtida a partir de (3.19), desde que ela seja relacionada apenas aos itens defeituosos produzidos em cada estado da cadeia. Para isso, basta fixar em 1 todos os componentes de custo associados aos itens defeituosos ($c_{nc} = c_{s_nc} = 1$) e fixar em 0 todos os demais ($c_{ins} = c_a = c_{s_c} = 0$). Portanto, ϕ pode ser escrito como

$$\phi(m, r, n) = C(m, r, n) \big|_{c_{nc}=c_{s_nc}=1; c_{ins}=c_a=c_{s_c}=0}.$$

3.4 Exemplos numéricos e discussões

Para ilustrar o procedimento proposto, considere o seguinte exemplo, adaptado de Taguchi et al. (TAGUCHI *et al.*, 1989, 2004). Um fabricante de circuitos integrados está enfrentando dificuldades no processo de soldagem. As conexões soldadas podem apresentar problemas como: falta ou excesso de solda, projeções ou posicionamento incorreto da solda. Dados históricos permitem adotar $p_1 = 0,01$ como a fração de defeitos no processo de soldagem (Estado I: sob-controle), e a mudança para o estado fora-de-controle

(Estado II) pode ser descrito por uma distribuição geométrica (i.e., $\pi = 0,0001$). Uma vez no Estado II, a degradação da fração de itens conformes segue o modelo exponencial $p_2 = 1 - 0,9e^{0,001(x-1)}$, em que x é o número de circuitos integrados produzidos após a mudança de estado. O sistema automático de inspeção por raios-x presente na linha de produção é imperfeito: apresenta erros de inspeção $\alpha = \beta = 0,01$. Os componentes de custo são estimados como $c_{ins} = \$0,25$; $c_{nc} = \$20$; $c_a = \$100$; $c_{s_nc} = \$2,0$; $c_{s_c} = \$2,5$ e a proporção máxima de circuitos integrados aceita é 3% (i.e., $\phi_{\max} = 0,03$). Estes valores estão resumidos na Tabela 3.1.

Tabela 3.1: Parâmetros-base para os exemplos numéricos.

$p_1 = 0,01$	$\pi = 0,0001$	$\phi_{\max} = 0,03$	$c_a = 100$
$p_2 = 1 - 0,9e^{-0,001(x-1)}$	$\alpha = 0,01$	$c_{ins} = 0,25$	$c_{s_nc} = 2,0$
	$\beta = 0,01$	$c_{nc} = 20$	$c_{s_c} = 2,5$

3.4.1 Truncamento da cadeia

Suponha que o procedimento de controle atual consiste em inspecionar 1 item a cada 25 itens produzidos ($m = 25$), cada item é classificado 3 vezes ($r = 3$) e é declarado conforme se mais da metade destas classificações for conforme ($s = \lfloor r/2 \rfloor + 1 = 2$). Considerando estes valores e aqueles na Tabela 3.1, somente o valor de n não está especificado. De acordo com a definição da variável w de cada estado da cadeia ($w = 0$, todos os itens do ciclo atual foram produzidos no Estado I; $w = 1$, o ciclo em que ocorre a mudança e pelo menos 1 item é produzido no Estado II; $w = 2$, segunda inspeção após a mudança e a primeira em que necessariamente todos os itens são produzidos no Estado II), é natural supor que $n = 2$ é o menor tamanho de cadeia para representar o processo

$$\text{Estado I} \xrightarrow{\text{mudança}} \text{Estado II}.$$

O procedimento para determinar n começa calculando a equação (3.19) para $\mathbf{P}_{n=2}$ e denotando o respectivo custo como $C_{n=2}$. Agora o custo $C_{n=3}$ é calculado. Seja δ_n a diferença relativa entre o custo obtido em dois tamanhos sucessivos de cadeia, definido por

$$\delta_n = \frac{|C_n - C_{n-1}|}{C_n},$$

e $\Delta_{\max}$ o critério de convergência estabelecido. O tamanho de cadeia $n = 3$ será adequado se $\delta_3 < \Delta_{\max}$. Caso contrário, o custo $C_{n=4}$ deve ser calculado e o procedimento para verificar a convergência é repetido. Portanto, o truncamento da cadeia pode ser escrito

como

$$\min \{n: \delta_n < \Delta_{\max}\}. \quad (3.20)$$

Ao contrário dos possíveis critérios de truncamento apresentados em (3.11) e (3.12), o critério (3.20) mede o desempenho da cadeia $\mathbf{P}_n$ em termos da variável objetivo: o custo. Além disso, os critérios (3.11) e (3.12) estão de alguma forma considerados em (3.20), já que a expressão do custo médio do sistema de controle depende de $\mathbf{P}_n$ e $p_2 = \psi(x)$.

Para ilustrar a aplicação de (3.20) sob diferentes condições de degradação no Estado II, três funções foram consideradas: a exponencial, a linear (adaptada de Nandi e Sreehari (1999)) e a constante. A Tabela 3.2 mostra os valores de C_n e δ_n para os diferentes tamanhos de cadeia. O procedimento de controle considerado foi ($m = 25, r = 3, s = 2$) e os demais parâmetros foram os especificados na Tabela 3.1.

Tabela 3.2: Custo em função do tamanho da cadeia n para três diferentes funções de deterioração da qualidade.

n	Exponencial $p_2 = 1 - 0,9e^{-0,001(x-1)}$		Linear $p_2 = \min \left\{ 0,1 + \frac{0,9(x-1)}{400}; 1 \right\}$		Constante $p_2 = 0,6$	
	C_n	δ_n	C_n	δ_n	C_n	δ_n
2	0,310763	-	0,308729	-	0,306953	-
3	0,311810	$3,36 \times 10^{-03}$	0,310587	$5,98 \times 10^{-03}$	0,306953	0
4	0,312467	$2,10 \times 10^{-03}$	0,311466	$2,82 \times 10^{-03}$	0,306953	0
5	0,312885	$1,34 \times 10^{-03}$	0,311888	$1,35 \times 10^{-03}$	0,306953	0
6	0,313152	$8,53 \times 10^{-04}$	0,312086	$6,35 \times 10^{-04}$	0,306953	0
7	0,313322	$5,44 \times 10^{-04}$	0,312175	$2,85 \times 10^{-04}$	0,306953	0
8	0,313430	$3,45 \times 10^{-04}$	0,312212	$1,20 \times 10^{-04}$	0,306953	0
9	0,313498	$2,16 \times 10^{-04}$	0,312226	$4,61 \times 10^{-05}$	0,306953	0
10	0,313540	$1,34 \times 10^{-04}$	0,312231	$1,61 \times 10^{-05}$	0,306953	0
11	0,313566	$8,23 \times 10^{-05}$	0,312233	$4,93 \times 10^{-06}$	0,306953	0
12	0,313582	$4,97 \times 10^{-05}$	0,312233	$1,30 \times 10^{-06}$	0,306953	0
13	0,313591	$2,95 \times 10^{-05}$	0,312233	$2,84 \times 10^{-07}$	0,306953	0
14	0,313596	$1,72 \times 10^{-05}$	0,312233	$4,90 \times 10^{-08}$	0,306953	0
15	0,313599	$9,85 \times 10^{-06}$	0,312233	$6,10 \times 10^{-09}$	0,306953	0
16	0,313601	$5,54 \times 10^{-06}$	0,312233	$4,64 \times 10^{-10}$	0,306953	0
17	0,313602	$3,05 \times 10^{-06}$	0,312233	$9,82 \times 10^{-11}$	0,306953	0
18	0,313603	$1,65 \times 10^{-06}$	0,312233	$1,40 \times 10^{-14}$	0,306953	0
19	0,313603	$8,71 \times 10^{-07}$	0,312233	$1,78 \times 10^{-16}$	0,306953	0
20	0,313603	$4,51 \times 10^{-07}$	0,312233	0	0,306953	0
$\vdots$						
100	0,313603	0	0,312233	0	0,306953	0

Considerando $\Delta_{\max} = 1 \times 10^{-6}$, as linhas pontilhadas na Tabela 3.2 indicam o tamanho mínimo da cadeia que atinge o critério de convergência: $n = 19$ para o modelo expo-

nencial, $n = 13$ para o modelo linear e $n = 3$ para o modelo constante. Para verificar se a convergência de fato ocorreu, calculou-se o custo para uma cadeia com $n = 100$, e os resultados indicam a convergência do custo dos três casos. Note, na Tabela 3.2, o comportamento de δ_n quando a função de deterioração é uma constante: para qualquer tamanho de cadeia, $\delta_n = 0$. A Figura 3.5 ilustra a razão deste comportamento.

Figura 3.5: Representação do truncamento da cadeia para diferentes funções de deterioração.

Se a deterioração segue uma função não-constante $p_2 = \psi(x)$, quanto maior o tamanho da cadeia, melhor será a representação da função deterioração (Figuras 3.5a,b - a linha pontilhada indica a função original, enquanto a linha cheia indica o truncamento). A Figura 3.5c mostra que se $p_2 = \text{constante}$, o truncamento em qualquer tamanho de cadeia ($n \geq 2$) coincide exatamente com a função p_2 .

3.4.2 Determinação da política ótima

O Algoritmo 3.1 mostra como a busca pela política ótima pode ser implementada. O programa apresentado no Apêndice A (e em www.anderson.trindade.nom.br/tese), mediante a especificação de alguns parâmetros, contempla o modelo proposto até aqui. Trindade *et al.* (2007b) disponibilizam versões específicas para este Capítulo, implementadas em Matlab (6.5 ou superior) e Microsoft Excel (97 ou superior).

Para manter os resultados coincidentes com aqueles em Trindade *et al.* (2007b), o critério $s = \lfloor r/2 \rfloor + 1$ foi adotado, o que implica alterar a linha 5 do Algoritmo 3.1 para:

para $s = \lfloor r/2 \rfloor + 1$ faça

Algoritmo 3.1 Algoritmo de Busca Direta ($C^\circ, m^\circ, r^\circ, s^\circ$)

Defina: $c_{ins}, c_{nc}, c_a, c_{s_nc}, c_{s_c}, \alpha, \beta, \pi, p_1, p_2, \phi_{\max}, \Delta_{\max}; m_{\min}, m_{\max}, r_{\min}, r_{\max}$

```

1:  $m \leftarrow m_{\min}, \phi_{new} \leftarrow 0, C^\circ \leftarrow \infty;$ 
2: repita
3:    $r \leftarrow r_{\min}, C_{new} \leftarrow \infty;$ 
4:   repita
5:     para  $s = 1$  até  $r$  faça
6:        $n \leftarrow 2$ ; Calcule  $C_{n=2}$ ;
7:       repita
8:          $n \leftarrow n + 1$ ; Calcule  $C_n$  e  $\delta_n$ ;
9:       até que  $\delta_n < \Delta_{\max}$ 
10:      se  $C_n < C_{new}$  então
11:         $C_{new} \leftarrow C_n, r_{new} \leftarrow r, s_{new} \leftarrow s;$ 
12:       $r \leftarrow r + 1$ ;
13:    até que  $r > r_{\max}$ 
14:    Calcule  $\phi$ ;  $m \leftarrow m + 1$ ;
15:    se  $\phi < \phi_{\max}$  e  $C_{new} < C^\circ$  então
16:       $C^\circ \leftarrow C_{new}, m^\circ \leftarrow m, r^\circ \leftarrow r_{new}, s^\circ \leftarrow s_{new};$ 
17:  até que  $m > m_{\max}$  ou  $\phi > \phi_{\max}$ 
18:  se  $C^\circ < \infty$  então
19:    retorne  $C^\circ, m^\circ, r^\circ, s^\circ$ ;
20:  senão
21:    imprima O critério  $\phi_{\max}$  é violado para o menor  $m$  testado;

```

3.4.3 Determinação da política ótima para o exemplo numérico

A Figura 3.6 apresenta o custo médio (C), tamanho da cadeia (n), número de classificações repetidas (r) e fração de defeitos produzidos pelo processo (ϕ) em função do intervalo entre inspeções (m) no intervalo $1 \leq m \leq 10^6$. Os parâmetros utilizados foram aqueles da Tabela 3.1. O custo ótimo ($C^\circ = 0,310873$) foi obtido para a política ($m^\circ = 32, r^\circ = 3$), que leva a uma fração de defeitos ($\phi = 1,27\%$). A restrição $\phi_{\max} = 3\%$ é satisfeita para $m < 300$. A curva do custo médio possui comportamento assintótico quando $m \rightarrow \infty$, pois todos os itens produzidos serão defeituosos com probabilidade $p_2 = \psi(\infty)$, e o custo médio será $\lim_{m \rightarrow \infty} C = \psi(\infty)c_{nc}$. Considerando o exemplo atual, $\psi(\infty) = 1$ (veja Figura 3.5d) e $\lim_{m \rightarrow \infty} C = c_{nc} = 20$, como mostra a Figura 3.5a. O comportamento geral do custo médio, para diversas simulações realizadas, é similar à Figura 3.6a: um rápido decréscimo do custo médio passando por um mínimo global, seguido de um gradual aumento do custo médio. Este formato também foi observado em trabalhos anteriores e permite agregar uma regra de parada ao Algoritmo 3.1:

- se um novo custo mínimo não foi obtido nos últimos k intervalos entre inspeções pesquisados, encerre a busca.

A Figura 3.6b mostra que, quanto maior o intervalo entre inspeções (m), menor é o tamanho da cadeia (n) necessário para satisfazer o critério $\delta_n < \Delta_{\max}$, já que mais itens

Figura 3.6: Custo médio (a), tamanho da cadeia (b), número de classificações repetidas (c) e fração de defeitos (d) para diferentes intervalos entre inspeções.

são produzidos neste intervalo. Porém, este comportamento só é observado para $m > 7$. Para $m \leq 7$, há um aumento do tamanho da cadeia com o aumento do intervalo entre inspeções. Isso pode ser explicado pela alta frequência de inspeções. Suponha o caso em que $m = 1$: todo o item produzido é inspecionado. A probabilidade do processo entrar e permanecer no Estado II é mínima, já que 100% dos itens são inspecionados. Portanto, a “curta existência” do Estado II pouco deve impactar o modelo. Quando $m = 2$, o processo permanecerá um maior tempo no Estado II, o que exigirá um aumento no tamanho da cadeia para representar o sistema. Com isso, observa-se um ponto de máximo no tamanho da cadeia (n) como mostra a Figura 3.6b. Este comportamento, também observado em todas as simulações realizadas, permite adicionar a seguinte regra ao Algoritmo 3.1:

- utilize como tamanho inicial da cadeia o maior tamanho já obtido durante a busca dos parâmetros ótimos

Esta regra evita que, para cada m pesquisado, seja determinado o tamanho de cadeia que atenda ao critério de convergência partindo-se de $n = 2$.

3.4.4 A política ótima para diferentes funções de deterioração

A Tabela 3.3 mostra a política ótima para as três funções apresentadas na Tabela 3.2, considerando dois níveis do componente de custo c_{nc} . Os demais parâmetros utilizados foram aqueles da Tabela 3.1. Os resultados mostram que diferentes esquemas de deterioração produzem diferentes políticas ótimas de controle. A magnitude desta diferença dependerá, obviamente, da combinação específica de parâmetros da Tabela 3.1 e das funções de deterioração. Os programas disponibilizados podem ser facilmente adaptados para outras funções $p_2 = \psi(x)$.

Tabela 3.3: Política ótima de controle para diferentes funções de deterioração.

$p_2(x)$	$c_{nc} = 20$			$c_{nc} = 5$		
	m°	r°	C°	m°	r°	C°
$p_2 = 1 - 0,9e^{-0,001(x-1)}$	32	3	0,3109	69	3	0,1120
$p_2 = \min \left\{ 0,1 + \frac{0,9(x-1)}{400}; 1 \right\}$	33	3	0,3085	75	3	0,1097
$p_2 = 0,6$	34	3	0,3023	72	3	0,1085

3.4.5 A presença dos erros de classificação

Suponha que os parâmetros apresentados na Tabela 3.1 são os reais parâmetros que governam o processo produtivo. Para ilustrar o efeito da presença dos erros de classificação, considere os cenários seguintes.

Cenário 1: A presença dos erros de classificação é desconhecida.

Nesta condição, a política ótima de controle é determinada assumindo $\alpha = 0\%$ e $\beta = 0\%$. Os valores obtidos são ($m^\circ = 26$, $r^\circ = 1$) e o custo médio do sistema que não possui erros de classificação é 0,292440. Porém, aplicando esta política no processo em questão (com erros $\alpha = \beta = 1\%$), o custo médio realmente obtido é 0,329149.

Cenário 2: Os erros são considerados, mas apenas uma classificação é feita.

O conhecimento dos erros $\alpha = \beta = 1\%$ faz com que sejam considerados. Resolvendo o modelo proposto, restrito a uma única classificação ($r = 1$), obtém-se o custo ótimo $C = 0,322724$ na política ($m^\circ = 36$, $r^\circ = 1$). O custo com esta política representa uma redução de 2,0% quando comparado ao custo do Cenário 1.

Cenário 3: Os erros são conhecidos e múltiplas classificações são consideradas.

Neste cenário, o custo ótimo é $C = 0,310873$ na política ($m^\circ = 32$, $r^\circ = 3$), o que representa redução de 5,6% and 3,7% quando comparado aos Cenários 1 e 2, respectivamente.

Mais uma vez ressalta-se que, o ganho com o uso de múltiplas classificações, se existir, dependerá da combinação específica dos parâmetros da Tabela 3.1.

A Figura 3.7 mostra o comportamento da política ótima em função dos erros de classificação, considerando os demais parâmetros fixos nos valores da Tabela 3.1. Como esperado, um aumento no número de classificações repetidas (r°) é necessário para minimizar o efeito do aumento dos erros de classificação. Com o aumento de α , aumenta a probabilidade de decisões erradas em relação aos itens conformes, o que levaria a ajustes desnecessários. Para contrapor este efeito, observa-se um aumento em m° , para reduzir o número de inspeções realizadas antes da mudança de estado (que é função de π). Com o aumento de β , aumenta a probabilidade de declarar itens defeituosos como conformes, o que dificulta a detecção da mudança de estado do processo. A reação do modelo é aumentar a frequência das inspeções (i.e., reduzir o intervalo entre inspeções m°) para acelerar a detecção. Note que tanto m° quanto r° são mais sensíveis às mudanças em α , o que pode ser explicado quando a probabilidade estacionária de $w_0 z_0$ é analisada: para todas as combinações (α, β) testadas, mais de 97% das inspeções são realizadas no estado $w_0 z_0$ (i.e., $y_{(w_0 z_0)} > 0,97$). Como neste estado a fração de itens conformes $(1 - p_1)$ é alta, a ocorrência do erro α é mais provável que a ocorrência do erro β , o que explica a sensibilidade observada.

Figura 3.7: Valores de m° e r° em função dos erros α e β .

3.4.6 Incerteza nos componentes do custo

Segundo Dasgupta (2003), a estimativa dos componentes do custo é uma tarefa complicada e não se pode esperar que tais estimativas sejam exatas. Entretanto, Montgomery (2004) argumenta que os componentes do custo não precisam ser determinados com

grande precisão. Ainda que os valores não sejam conhecidos exatamente, a proporção $c_{ins} : c_{nc} : c_a : c_{s_nc} : c_{s_c}$ pode ser utilizada para obter a política ótima. O fato é que, para utilizar um modelo econômico, os componentes do custo devem ser informados de alguma forma.

Suponha que os custos apresentados na Tabela 3.1 foram estimados com uma margem de erro de $\pm 25\%$ e que amostras aleatórias de $(c_i, c_{nc}, c_a, c_{s_nc}, c_{s_c})$ foram retiradas de uma distribuição uniforme considerando o intervalo $[75\% - 125\%]$ em torno do componente de custo especificado na Tabela 3.1 (os outros parâmetros foram mantidos fixos). Para cada amostra, o custo médio foi calculado considerando as políticas do Cenário 1 ($m^\circ = 26, r^\circ = 1$), Cenário 2 ($m^\circ = 36, r^\circ = 1$), e Cenário 3 ($m^\circ = 32, r^\circ = 3$). Este procedimento foi repetido 10.000 vezes e a Figura 3.8 mostra os resultados obtidos. Para todas as amostras geradas, a política do Cenário 3 reduziu o custo médio do sistema quando comparado às políticas dos Cenários 1 and 2. Portanto, os resultados obtidos na Seção 3.4.5 mantém-se mesmo que as estimativas dos custos tenham sido obtidas com uma margem de erro de $\pm 25\%$.

Figura 3.8: Redução do custo médio: Cenário 3 versus Cenários 1 e 2.

Para avaliar o efeito de cada um dos componentes sobre o custo médio do sistema de controle, realizou-se uma regressão múltipla considerando as 10.000 amostras de $(c_{ins}, c_{nc}, c_a, c_{s_nc}, c_{s_c})$ como variáveis preditoras (na realidade, utilizou-se o valor correspondente no intervalo $[75\% - 125\%]$, indicado por um asterisco na Tabela 3.4) e o custo médio como variável dependente. Com as variáveis preditoras na mesma escala, quando maior o coeficiente da regressão, maior a contribuição do componente no custo médio do sistema. A Tabela 3.4 mostra a estimativa dos coeficientes: c_{nc} , c_a e c_{ins} são, em ordem decrescente, os componentes de maior impacto. Este tipo de análise pode ser utilizada para determinar quais componentes devem ser estimados com maior precisão.

A Figura 3.9 mostra a distribuição do intervalo de inspeção ótimo quando, para cada

Tabela 3.4: Efeito dos componentes de custo sobre o custo médio do sistema de controle.

$$C = b_0 + b_1 c_{ins}^* + b_2 c_{nc}^* + b_3 c_a^* + b_4 c_{s_{nc}}^* + b_5 c_{s_c}^*$$

Cenário	b_0	b_1	b_2	b_3	b_4	b_5
Cenário 1	0,0000	0,0096	0,2327	0,0850	0,0009	0,0009
Cenário 2	0,0000	0,0069	0,2505	0,0639	0,0007	0,0007
Cenário 3	0,0000	0,0234	0,2451	0,0415	0,0008	0,0000

uma das 10.000 amostras geradas, os valores ótimos de m e r são determinados. Em todas as otimizações obteve-se $r = 3$ (por isso o gráfico não é apresentado). Os resultados mostram que a política baseada nos valores originais ($m = 32, r = 3$) corresponde à uma das mais frequentes políticas ótimas obtidas sob incerteza e representa uma boa escolha para monitorar o processo. Para obter uma distribuição mais estreita de m° é necessário melhorar a estimativa dos componentes de custo mais relevantes.

Figura 3.9: Distribuição amostral do intervalo ótimo de inspeção.

Em condições de incerteza sobre os componentes do custo, o procedimento de busca da política ótima pode ser alterado para determinar, por exemplo, qual a política de controle que minimiza a média dos custos médios, quando considerada uma amostra dos componentes de custo de tamanho, digamos, 10.000. Tal procedimento pode também ser utilizado quando há incerteza sobre qualquer parâmetro da Tabela 3.1.

3.4.7 Análise de sensibilidade para os principais componentes de custo

A Tabela 3.5 apresenta a política ótima, o custo médio e a fração de itens defeituosos para diferentes valores de c_{nc} , c_a e c_{ins} (com os demais parâmetros fixos nos valores da Tabela 3.1). Um aumento em c_a (custo de ajuste) ou em c_{ins} (custo de inspeção) reduz a frequência das inspeções (i.e., aumenta m°) enquanto um aumento em c_{nc} (custo do item defeituoso) reduz m° para acelerar a detecção da mudança, evitando a produção de

itens defeituosos no Estado II. As classificações repetidas mostram-se mais sensíveis a c_{ins} : quanto mais cara a inspeção, menos classificações repetidas são realizadas.

Tabela 3.5: Análise de sensibilidade para diferentes parâmetros.

		c_{ins}															
		0,25				0,50				0,75				1,00			
c_{nc}	c_a	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°
5	50	57	1	0,0984	1,4%	63	1	0,1025	1,5%	68	1	0,1063	1,5%	73	1	0,1098	1,6%
	100	69	3	0,1120	1,6%	82	1	0,1206	1,6%	87	1	0,1236	1,7%	91	1	0,1264	1,7%
	150	79	3	0,1236	1,6%	92	3	0,1325	1,7%	102	1	0,1386	1,8%	106	1	0,1410	1,8%
	200	88	3	0,1345	1,7%	100	3	0,1425	1,8%	110	3	0,1497	1,8%	119	1	0,1543	1,9%
10	50	39	1	0,1653	1,3%	43	1	0,1714	1,3%	47	1	0,1770	1,4%	50	1	0,1821	1,4%
	100	47	3	0,1832	1,4%	57	1	0,1957	1,4%	60	1	0,2000	1,5%	63	1	0,2041	1,5%
	150	54	3	0,1981	1,4%	63	3	0,2109	1,5%	70	1	0,2198	1,5%	73	1	0,2233	1,6%
	200	60	3	0,2118	1,5%	68	3	0,2235	1,5%	76	3	0,2339	1,6%	82	1	0,2405	1,6%
15	50	31	1	0,2274	1,2%	34	1	0,2350	1,3%	37	1	0,2420	1,3%	40	1	0,2484	1,3%
	100	37	3	0,2486	1,3%	45	1	0,2643	1,4%	48	1	0,2697	1,4%	50	1	0,2748	1,4%
	150	43	3	0,2661	1,4%	50	3	0,2822	1,4%	57	1	0,2931	1,4%	59	1	0,2975	1,5%
	200	48	3	0,2820	1,4%	55	3	0,2966	1,4%	61	3	0,3096	1,5%	66	1	0,3179	1,5%
20	50	26	1	0,2867	1,2%	29	1	0,2957	1,2%	32	1	0,3039	1,3%	34	1	0,3115	1,3%
	100	32	3	0,3109	1,3%	39	1	0,3294	1,3%	41	1	0,3357	1,3%	43	1	0,3416	1,3%
	150	37	3	0,3306	1,3%	43	3	0,3495	1,4%	48	1	0,3624	1,4%	50	1	0,3674	1,4%
	200	41	3	0,3486	1,3%	47	3	0,3657	1,4%	52	3	0,3809	1,4%	56	1	0,3905	1,4%

Todos os exemplos na Tabela 3.5 foram obtidos com a restrição $\phi_{\max} = 3\%$. Porém, este valor representa produzir 3 vezes mais itens defeituosos se comparado à produção sob controle (veja na Tabela 3.1 que $p_1 = 0,01$). Caso um critério mais restritivo fosse adotado, digamos $\phi_{\max} = 1,5\%$, diversos valores da Tabela 3.5 seriam modificados. Para ilustrar o efeito deste critério mais restritivo, considere os valores $c_{nc} = 5$, $c_a = 200$ and $c_{ins} = 0,5$ da Tabela 3.5, que levam a $\phi = 1,8\%$. Otimizando com a restrição $\phi_{\max} = 1,5\%$ obtém-se como política ótima $m^\circ = 63$, $r^\circ = 3$, $\phi = 1,4984\%$ e $C^\circ = 0,1503$ (um aumento de 5,5% sobre o custo correspondente na Tabela 3.5 para satisfazer o novo critério estatístico).

3.5 Considerações Finais

O modelo apresentado neste Capítulo é naturalmente mais complexo que o anterior, e tal complexidade decorre do uso da função $\psi(x)$. Apesar disso, o recurso “mais sofisticado” exigido para a resolução do modelo permanece o mesmo: um algoritmo para solução de sistemas lineares, utilizado na determinação de $\mathbf{y}$. Por isso, qualquer software que possua um ambiente de programação permite a implementação do modelo. O uso de diferentes funções $\psi(x)$ exige apenas a redefinição desta função no programa.

As conclusões sobre o uso de classificações repetidas se mantêm: podem representar uma opção mais econômica para um sistema de inspeção que possui erros de classificação. Caso variáveis do modelo sejam determinadas com algum grau de imprecisão, simulações de Monte Carlo podem ser utilizadas, em associação ao modelo proposto, para determinar a melhor política de controle sob tal condição.

O Capítulo 4 mostrará como o modelo proposto até aqui pode ser expandido para considerar uma seqüência de inspeções na decisão sobre o ajuste do processo, utilizando praticamente as mesmas equações desenvolvidas no Capítulo 3.

CAPÍTULO 4

Controle *on-line* para atributos considerando o histórico das h últimas inspeções

O modelo proposto por Taguchi pode ser considerado perfeito quando $p_1 = 0$, $p_2 = 1$ e não há erros de classificação. Nesta condição, o processo muda de um estado em que todos os itens produzidos são conformes para um estado no qual todos os itens produzidos são defeituosos, e um único item inspecionado fornecerá o diagnóstico exato da condição em que o processo está operando. Quando p_1 e p_2 se afastam de 0 e 1, respectivamente, e com a presença de erros de classificação, a decisão com base em apenas uma inspeção ocasionará falsos-alarmes e/ou retardo na detecção da mudança.

Nos Capítulos 2 e 3 a decisão sobre o ajuste do processo é realizada com base apenas no item inspecionado. Uma expansão natural é utilizar uma amostra de tamanho n , mas neste caso perde-se a conexão com as observações individuais. A alternativa então é considerar o que aconteceu nas últimas h inspeções para a tomada de decisão, o que será contemplado neste Capítulo.

Para considerar o resultado das h últimas inspeções para a tomada de decisão sobre a necessidade de ajuste, o modelo deverá claramente ser modificado, mas a estrutura básica se mantém. Vejamos dois possíveis critérios de parada apresentados a seguir.

4.1 Critério I: Ajustar quando uma seqüência de h inspeções resultar em h itens não-conformes

Com este critério, deve-se modificar a definição de z nos estados da cadeia de Markov. Agora, z será um contador de itens inspecionados e declarados não-conformes. Caso uma seqüência de h inspeções resulte em h itens declarados não-conformes, o processo é parado para ajuste.

Com isso, o sistema de inspeção pode ser modelado como uma cadeia de Markov estacionária considerando o espaço de estados discretos

$$\Omega = \{(0,0), \dots, (0,h), (1,0), \dots, (1,h), \dots, (w_i, z_j), \dots, (n, h-1), (n, h)\}.$$

Cada estado é descrito por um par de números inteiros w e z , totalizando $(h+1)(n+1)$ estados. Caso o comprimento da seqüência considerada seja $h = 1$ tem-se o modelo apresentado no Capítulo 3, com $2n+2$ estados. O inteiro z torna-se um contador do número de itens inspecionados e classificados como não-conformes sequencialmente. O inteiro w mantém o significado do Capítulo precedente: o número de inspeções realizadas após a mudança para o Estado II ($w = 0, 1, \dots, n$).

Como ilustração, considere a Figura 4.1, que mostra as transições entre estados da cadeia para uma seqüência de $h = 3$ itens não-conformes utilizado como critério de parada. Observe que se $z = 2$ inspeções resultarem em itens não-conformes e a inspeção subsequente declarar o item inspecionado como conforme, a transição ocorrerá para um estado no qual $z = 0$ (i.e., o “contador” z é zerado).

Não há alterações na essência das equações de probabilidade de transição e custo apresentadas no Capítulo 3. Porém, elas devem ser indexadas para que o ajuste só ocorra quando o estado $w_i z_h$ for atingido. Por esta razão, elas serão apresentadas nas subseções seguintes, mas sem o detalhamento visto anteriormente.

4.1.1 Probabilidades de Transição

As equações (4.1) e (4.2) correspondem às transições entre estados da cadeia nos quais o processo permanece operando no Estado I. Na equação (4.1), o item inspecionado é declarado conforme, o que leva ao estado da cadeia em que $z = 0$. Na equação (4.2), o

Figura 4.1: Estados da cadeia e transições quando o critério de parada utilizado é uma sequência de $h = 3$ inspeções não-conformes.

item inspecionado é declarado não-conforme, incrementando z .

$$P_{(0,l),(0,0)|l=0\dots h-1} = (1 - \pi)^m [(1 - p_1)(1 - \alpha^*) + p_1\beta^*] \quad (4.1)$$

$$P_{(0,l),(0,l+1)|l=0\dots h-1} = (1 - \pi)^m [(1 - p_1)\alpha^* + p_1(1 - \beta^*)]. \quad (4.2)$$

Nas equações (4.3) e (4.4) está representada a mudança do Estado I para o Estado II, sendo (4.3) quando o item inspecionado é declarado conforme e (4.4) quando é declarado não-conforme.

$$P_{(0,l),(1,0)|l=0\dots h-1} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)(1 - \alpha^*) + \psi(m - i + 1)\beta^*] \quad (4.3)$$

$$P_{(0,l),(1,l+1)|l=0\dots h-1} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)\alpha^* + \psi(m - i + 1)(1 - \beta^*)]. \quad (4.4)$$

Nas equações (4.5) e (4.6), a transição ocorre entre estados da cadeia nos quais o processo opera no Estado II.

$$P_{(j-1,l),(j,0)|j=2\dots n, l=0\dots h-1} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm - i + 1)(1 - \alpha^*) + \psi(jm - i + 1)\beta^*] \quad (4.5)$$

$$P_{(j-1,l),(j,l+1)|j=2\dots n, l=0\dots h-1} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i + 1)(1 - \beta^*)] \quad (4.6)$$

As equações (4.7) e (4.8) correspondem ao truncamento da cadeia na n -ésima inspeção

após a mudança para o Estado II.

$$P_{(n,l),(n,0)|l=0\dots h-1} = P_{(n-1,l),(n,0)} \quad (4.7)$$

$$P_{(n,l),(n,l+1)|l=0\dots h-1} = P_{(n-1,l),(n,l+1)} \quad (4.8)$$

Por fim, as probabilidades de transição dos estados representados pelas equações (4.9) a (4.12), em razão da ocorrência do ajuste, são idênticas às transições a partir do estado $w_0 z_0$.

$$P_{(j,h),(0,0)|j=1\dots n} = P_{(0,0),(0,0)} \quad (4.9)$$

$$P_{(j,h),(0,1)|j=1\dots n} = P_{(0,0),(0,1)} \quad (4.10)$$

$$P_{(j,h),(1,0)|j=1\dots n} = P_{(0,0),(1,0)} \quad (4.11)$$

$$P_{(j,h),(1,1)|j=1\dots n} = P_{(0,0),(1,1)} \quad (4.12)$$

4.1.2 Custos dos estados da cadeia

Os estados $w_i z_0$ ($i = 0 \dots n$) não sofreram alterações na forma com que foram definidos no Capítulo 3, e portanto seus componentes de custo ξ , θ e φ permanecem os mesmos, e estão aqui reproduzidos para conveniência do leitor.

$$\begin{aligned} \xi_{(0,0)} &= c_{nc}(m-1)p_1 \\ \theta_{(0,0)} &= c_{nc} \frac{p_1 \beta^*}{(1-p_1)(1-\alpha^*) + p_1 \beta^*} \\ \varphi_{(0,0)} &= 0. \end{aligned}$$

$$\begin{aligned} \xi_{(1,0)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1} \pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \\ \theta_{(1,0)} &= c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1} \pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)\beta^*}{\bar{\psi}(m-i+1)(1-\alpha^*) + \psi(m-i)\beta^*} \right] \\ \varphi_{(1,0)} &= 0. \end{aligned}$$

$$\xi_{(j,0)|j=2\dots n} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1} \pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1)$$

$$\theta_{(j,0)|j=2\dots n} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm-i+1)\beta^*}{\bar{\psi}(jm-i+1)(1-\alpha^*) + \psi(jm-i+1)\beta^*} \right]$$

$$\varphi_{(j,0)|j=2\dots n} = 0.$$

O custo dos estados $w_i z_j$ ($i = 0 \dots n, j = 1 \dots h$) são similares àqueles nos quais há ajuste e que foram definidos no Capítulo 3. Porém, o componente φ só corresponderá ao custo de ajuste c_a nos estados $w_i z_h$, o que leva à introdução de uma função indicadora nas equações $\varphi_{(w,z)}$.

$$\xi_{(0,l)|l=1\dots h} = c_{nc}(m-1)p_1$$

$$\theta_{(0,l)|l=1\dots h} = c_{s-nc} \frac{p_1(1-\beta^*)}{(1-p_1)\alpha^* + p_1(1-\beta^*)} + c_{s-c} \frac{(1-p_1)\alpha^*}{(1-p_1)\alpha^* + p_1(1-\beta^*)}$$

$$\varphi_{(0,l)|l=1\dots h} = c_a I_{(h)}(l).$$

$$\xi_{(1,l)|l=1\dots h} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right]$$

$$\theta_{(1,l)|l=1\dots h} = c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)(1-\beta^*)}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] +$$

$$c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(m-i+1)\alpha^*}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right]$$

$$\varphi_{(1,l)|l=1\dots h} = c_a I_{(h)}(l).$$

$$\xi_{(j,l)|j=2\dots n, l=1\dots h} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m-i+l+1)$$

$$\theta_{(j,l)|j=2\dots n, l=1\dots h} = c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm-i+1)(1-\beta^*)}{\bar{\psi}(jm-i+1)\alpha^* + \psi(jm-i+1)(1-\beta^*)} \right] +$$

$$c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(jm-i+1)\alpha^*}{\bar{\psi}(jm-i+1)\alpha^* + \psi(jm-i+1)(1-\beta^*)} \right]$$

$$\varphi_{(j,l)|j=2\dots n, l=1\dots h} = c_a I_{(h)}(l).$$

4.1.3 Determinação da política ótima para as diferentes funções de deterioração aplicando o Critério I

As Figuras 4.2-4.4 mostram o custo médio por item produzido considerando o Critério I, com as funções $\psi(x)$ empregadas no Capítulo 3 e demais parâmetros fixados nos valores

da Tabela 3.1. Utilizou-se também $\Delta_{\max} = 1 \times 10^{-4}$ e $s = \lfloor r/2 \rfloor + 1$. Como esperado, nos três casos há redução do custo médio por item produzido se comparado à estratégia de controle com apenas 1 item considerado no histórico (i.e. $h = 1$).

Figura 4.2: Custo médio versus política de parada utilizando o modelo Exponencial.

Figura 4.3: Custo médio versus política de parada utilizando o modelo Linear.

Na Figura 4.2 há pontos ausentes no gráfico para $(h = 7)$ e $(h = 8)$ porque a fração média de itens não-conformes (ϕ) produzidos com tais parâmetros superou o máximo estabelecido na Tabela 3.1 ($\phi_{\max} = 0,03$). A Tabela 4.1 mostra a estratégia ótima de controle obtida para as três funções de degradação. Nos três casos a estratégia consiste em parar o processo quando duas inspeções consecutivas resultarem em itens não-conformes (i.e. $h = 2$), classificando apenas uma vez cada item inspecionado. Para $h > 1$, nenhuma das três funções de deterioração empregadas utilizou-se de classificações repetidas. Quando apenas 1 item é considerado na tomada de decisão sobre ajustar ou não o processo, a única informação disponível provém deste item e o uso de classificações repetidas garante a precisão desta informação. Quando é possível considerar mais de 1 item para a tomada

Figura 4.4: Custo médio versus política de parada utilizando o modelo Constante.

da decisão, a solução econômica consiste em adicionar mais informação sobre a condição de operação do processo (i.e., mais itens inspecionados), ainda que a informação de cada item individualmente possua algum grau de incerteza (i.e., erros na classificação do item inspecionados).

Tabela 4.1: Comparação entre as políticas ótimas de controle utilizando o Critério I para as diferentes funções de deterioração.

Critério	Função	C^o	m^o	r^o	h^o
Capítulo 3 (Critério I, $h=1$)	Exponencial	0,3109	32	3	1
	Linear	0,3085	33	3	1
	Constante	0,3023	34	3	1
Critério I	Exponencial	0,2766	4	1	2
	Linear	0,2705	4	1	2
	Constante	0,2561	5	1	2

A Tabela 4.2 compara os custos ótimos obtidos no Capítulo 3 ($h = 1$) com aqueles obtidos utilizando o Critério I. A redução do custo médio por item produzido foi de pelo menos 11% quando a seqüência de itens inspecionados (e declarados não-conformes) foi considerada como critério de ajuste. A magnitude da redução dependerá da combinação específica dos parâmetros de entrada do modelo, mas uma vez que o modelo proposto nesta seção engloba àquele do Capítulo 3, recomenda-se sua utilização.

Tabela 4.2: Comparação entre os custos ótimos obtidos no Capítulo 3 ($h = 1$) e com o Critério I ($h = 2$).

Função	C° (Capítulo 3)	C° (Critério I)	Redução em C°
Exponencial	0,3109	0,2766	11,0%
Linear	0,3085	0,2705	12,3%
Constante	0,3023	0,2561	15,3%

4.2 Critério II: Ajustar quando forem observados ν itens não-conformes em uma seqüência de h inspeções

A formulação do critério de uma seqüência de h não-conformidades induz ao critério mais abrangente: por que não considerar $\nu \leq h$ não-conformidades em uma seqüência de h inspeções? Esta seção tratará deste critério. Partindo da estratégia utilizada na seção anterior, uma extensão natural seria utilizar o índice z na cadeia de Markov como variável contadora e, ao invés de reiniciá-lo quando um item for declarado conforme na inspeção, seu valor seria decrescido de uma unidade. A Figura 4.5 mostra que tal regra não funcionaria.

Considere que a parada do processo ocorra quando três itens forem declarados não-conformes nas últimas quatro inspeções (i.e., $\nu = 3$ e $h = 4$). A Figura 4.5 representa os instantes de inspeção em que símbolos cheios e vazios correspondem a itens não-conformes e conformes, respectivamente. Nos passos *a-c* a variável contadora funciona adequadamente: há a inclusão de 2 itens não-conformes na seqüência (i.e., $z = 2$). No passo *d*, um item é inspecionado e declarado conforme, o que com a regra presumida reduziria o contador em uma unidade. Porém, o valor da variável z não reflete mais o número de itens não-conformes que permanecem no histórico. Deste ponto em diante, há um descompasso entre z e o número de itens não-conformes na seqüência.

Figura 4.5: Representação de uma seqüência de inspeções considerando $h = 4$ e $\nu = 3$.

A Figura 4.6 ilustra a transição entre seqüências de inspeção de tamanho três, onde $\mathbf{0}$

representa um item considerado conforme e **1** aquele classificado como não-conforme.

Figura 4.6: Representação da transição entre seqüências de inspeção quando: a) $h = 3$ e $\nu = 3$; b) $h = 3$ e $\nu = 2$.

Na Figura 4.6a o critério de parada é a observação de três não-conformidades. Neste caso, há 8 seqüências possíveis e a única que leva ao ajuste é a **111**. Já a Figura 4.6b utiliza o critério de 2 não-conformidades para o ajuste e há 6 seqüências viáveis, com **011** e **101** levando ao ajuste do processo. As duas seqüências propositalmente esmaecidas, **110** e **111**, são seqüências inviáveis, pois pelo critério de parada o ajuste já deveria ter ocorrido. Com a ilustração percebe-se que, para registrar o histórico de inspeções de tamanho h , pode ser necessário representar até 2^h seqüências. Este valor máximo ocorrerá quando o número de não-conformidades utilizado como critério de parada (ν) coincidir com o tamanho da seqüência considerada (h).

Note que não há alteração na forma com que cada estado é tratado: se o item inspecionado for declarado conforme a produção continua; se o item inspecionado for declarado não-conforme a produção pode continuar ou o ajuste pode ocorrer. O âmag das equações permanece, e mais uma vez o problema recai em como indexar as equações de transição entre estados da cadeia e as equações de custo. A Figura 4.6 sugere o uso da base binária para representar as seqüências. No caso de uma seqüência de tamanho $h = 3$, a conversão para a base decimal leva aos valores inteiros de 0 à 7. Com isso, propõe-se o Algoritmo 4.1 para verificar a viabilidade e determinar as transições entre seqüências.

Os índices 2 e 10 presentes no algoritmo correspondem à representação de um número inteiro nas bases binária e decimal respectivamente. Quando um número for representado em base binária, o “primeiro dígito” refere-se ao dígito à extrema direita, enquanto “último dígito” refere-se àquele a extrema esquerda.

As Tabelas 4.3 e 4.4 mostram a aplicação do algoritmo aos critério $h = 3, \nu = 3$ e $h = 3, \nu = 2$. Os resultados podem ser confrontados com a Figura 4.6.

Algoritmo 4.1 Algoritmo para determinação da viabilidade e transições entre seqüências

Defina: h, ν ;

```

1: para  $i_{10} = 0$  até  $(2^h - 1)$  faça
2:    $i_2 \leftarrow$  representação de  $i_{10}$  em base binária de  $h$  dígitos ;
3:    $S \leftarrow$  soma dos dígitos de  $i_2$ ;
4:   se  $S < \nu$  então
5:     A seqüência  $i_2$  é viável;
6:     se O próximo item inspecionado é declarado conforme então
7:        $j_2 \leftarrow$  Descarte o último dígito de  $i_2$  e adicione o dígito 0 à sua direita;
       Transição:  $(i_2 \rightarrow j_2) = (i_{10} \rightarrow j_{10})$ 
8:     senão se O próximo item inspecionado é declarado não-conforme então
9:        $j_2 \leftarrow$  Descarte o último dígito de  $i_2$  e adicione o dígito 1 à sua direita;
       Transição:  $(i_2 \rightarrow j_2) = (i_{10} \rightarrow j_{10})$ 
10:      Se  $S = (\nu - 1)$  a seqüência  $j_2$  obtida leva ao ajuste;
11:    senão se  $S = \nu$  então
12:      se primeiro dígito de  $i_2 = 0$  então
13:        a seqüência  $i_2$  é inviável;
14:      senão
15:        A seqüência  $i_2$  é viável e leva ao ajuste;
16:        A próxima seqüência caso o item inspecionado seja declarado conforme será  $0_2$ ;
        Transição:  $(i_2 \rightarrow 0_2) = (i_{10} \rightarrow 0_{10})$ ;
17:        A próxima seqüência caso o item inspecionado seja declarado não-conforme será  $1_2$ :
         $(i_2 \rightarrow 1_2) = (i_{10} \rightarrow 1_{10})$ ;
18:    senão se  $S > \nu$  então
19:      a seqüência  $i_2$  é inviável;

```

Uma vez estabelecido o Algoritmo 4.1, resta definir a função $\gamma(h, \nu)$ que expressa o número de seqüências viáveis dados h e ν . Na equação (4.19), $k_1, k_2, \dots, k_h$ correspondem ao primeiro, segundo, $\dots$ h -ésimo dígitos da representação binária da seqüência, sendo o primeiro dígito aquele à extrema direita da seqüência. A primeira parte de (4.19) corresponde às seqüências nas quais o item recém-inspecionado foi declarado conforme. Assim, a soma dos dígitos da seqüência deverá ser sempre menor que ν para que seja viável. Por sua vez, a segunda parte da equação (4.19) indica que o item recém-inspecionado foi declarado não-conforme. Com isso, a soma de todos os dígitos da seqüência deverá ser menor ou igual à ν para que seja uma seqüência viável.

$$\begin{aligned}
\gamma(h, \nu) = & \sum_{k_1=0}^0 \sum_{k_2=0}^{\min(1, \nu-1-\sum k_l)} \sum_{k_3=0}^{\min(1, \nu-1-\sum k_l)} \dots \sum_{k_h=0}^{\min(1, \nu-1-\sum k_l)} 1 \\
& + \sum_{k_1=1}^1 \sum_{k_2=0}^{\min(1, \nu-\sum k_l)} \sum_{k_3=0}^{\min(1, \nu-\sum k_l)} \dots \sum_{k_h=0}^{\min(1, \nu-\sum k_l)} 1
\end{aligned} \tag{4.19}$$

A equação (4.19) pode ter suas partes combinadas para obter (4.20).

$$\gamma(h, \nu) = \sum_{k_1=0}^1 \sum_{k_2=0}^{\min(1, j-1+k_1-\sum k_l)} \sum_{k_3=0}^{\min(1, j-1+k_1-\sum k_l)} \dots \sum_{k_h=0}^{\min(1, j-1+k_1-\sum k_l)} 1 \tag{4.20}$$

Tabela 4.3: Viabilidade das seqüências e transições para $h = 3$ e $\nu = 3$.

Seqüência	S	Condição	Próxima Inspeção	Transição
$0 = 000_2$	0	viável	c: $\emptyset 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$000_2 \rightarrow 000_2 = 0 \rightarrow 0$
			n: $\emptyset 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$000_2 \rightarrow 001_2 = 0 \rightarrow 1$
$1 = 001_2$	1	viável	c: $\emptyset 01 \leftarrow 0 \Rightarrow 010_2 = 2_{10}$	$001_2 \rightarrow 010_2 = 1 \rightarrow 2$
			n: $\emptyset 01 \leftarrow 1 \Rightarrow 011_2 = 3_{10}$	$001_2 \rightarrow 011_2 = 1 \rightarrow 3$
$2_{10} = 010_2$	1	viável	c: $\emptyset 10 \leftarrow 0 \Rightarrow 100_2 = 4_{10}$	$010_2 \rightarrow 100_2 = 2 \rightarrow 4$
			n: $\emptyset 10 \leftarrow 1 \Rightarrow 101_2 = 5_{10}$	$001_2 \rightarrow 011_2 = 2 \rightarrow 5$
$3_{10} = 011_2$	2	viável	c: $\emptyset 11 \leftarrow 0 \Rightarrow 110_2 = 6_{10}$	$011_2 \rightarrow 110_2 = 3 \rightarrow 6$
			n: $\emptyset 11 \leftarrow 1 \Rightarrow 111_2 = 7_{10}$	$011_2 \rightarrow 111_2 = 3 \rightarrow 7$
$4_{10} = 100_2$	1	viável	c: $\cancel{1} 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$100_2 \rightarrow 000_2 = 4 \rightarrow 0$
			n: $\cancel{1} 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$100_2 \rightarrow 001_2 = 4 \rightarrow 1$
$5_{10} = 101_2$	2	viável	c: $\cancel{1} 01 \leftarrow 0 \Rightarrow 010_2 = 2_{10}$	$101_2 \rightarrow 010_2 = 5 \rightarrow 2$
			n: $\cancel{1} 01 \leftarrow 1 \Rightarrow 011_2 = 3_{10}$	$101_2 \rightarrow 011_2 = 5 \rightarrow 3$
$6_{10} = 110_2$	2	viável	c: $\cancel{1} 10 \leftarrow 0 \Rightarrow 100_2 = 4_{10}$	$110_2 \rightarrow 100_2 = 6 \rightarrow 4$
			n: $\cancel{1} 10 \leftarrow 1 \Rightarrow 101_2 = 5_{10}$	$110_2 \rightarrow 101_2 = 6 \rightarrow 5$
$7_{10} = 111_2$	3	ajuste	c: $\emptyset 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$111_2 \rightarrow 000_2 = 7 \rightarrow 0$
			n: $\emptyset 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$111_2 \rightarrow 001_2 = 7 \rightarrow 1$

c: conforme, n: não-conforme

Tabela 4.4: Viabilidade das seqüências e transições para $h = 3$ e $\nu = 2$.

Seqüência	S	Condição	Próxima Inspeção	Transição
$0 = 000_2$	0	viável	c: $\emptyset 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$000_2 \rightarrow 000_2 = 0 \rightarrow 0$
			n: $\emptyset 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$000_2 \rightarrow 001_2 = 0 \rightarrow 1$
$1 = 001_2$	1	viável	c: $\emptyset 01 \leftarrow 0 \Rightarrow 010_2 = 2_{10}$	$001_2 \rightarrow 010_2 = 1 \rightarrow 2$
			n: $\emptyset 01 \leftarrow 1 \Rightarrow 011_2 = 3_{10}$	$001_2 \rightarrow 011_2 = 1 \rightarrow 3$
$2_{10} = 010_2$	1	viável	c: $\emptyset 10 \leftarrow 0 \Rightarrow 100_2 = 4_{10}$	$010_2 \rightarrow 100_2 = 2 \rightarrow 4$
			n: $\emptyset 10 \leftarrow 1 \Rightarrow 101_2 = 5_{10}$	$001_2 \rightarrow 011_2 = 2 \rightarrow 5$
$3_{10} = 011_2$	2	ajuste	c: $\emptyset 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$011_2 \rightarrow 000_2 = 3 \rightarrow 0$
			n: $\emptyset 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$011_2 \rightarrow 001_2 = 3 \rightarrow 1$
$4_{10} = 100_2$	1	viável	c: $\cancel{1} 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$100_2 \rightarrow 000_2 = 4 \rightarrow 0$
			n: $\cancel{1} 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$100_2 \rightarrow 001_2 = 4 \rightarrow 1$
$5_{10} = 101_2$	2	ajuste	c: $\emptyset 00 \leftarrow 0 \Rightarrow 000_2 = 0_{10}$	$101_2 \rightarrow 000_2 = 5 \rightarrow 0$
			n: $\emptyset 00 \leftarrow 1 \Rightarrow 001_2 = 1_{10}$	$101_2 \rightarrow 001_2 = 5 \rightarrow 1$
$6_{10} = 110_2$	2	inviável		
$7_{10} = 111_2$	3	inviável		

c: conforme, n: não-conforme

A Tabela 4.5 mostra o número de seqüências válidas para diferentes h e ν . A observação dos valores obtidos mostra alguma semelhança com o Triângulo de Pascal, o que permite escrever $\gamma(h, \nu)$ como

$$\gamma(h, \nu) = 2^h - \sum_{i=1}^{h-\nu} 2 \binom{h-1}{h-i}$$

Tabela 4.5: Número de seqüências de inspeções válidas para diferentes h e ν .

	ν									
h	1	2	3	4	5	6	7	8	9	10
1	2									
2	2	4								
3	2	6	8							
4	2	8	14	16						
5	2	10	22	30	32					
6	2	12	32	52	62	64				
7	2	14	44	84	114	126	128			
8	2	16	58	128	198	240	254	256		
9	2	18	74	186	326	438	494	510	512	
10	2	20	92	260	512	764	932	1004	1022	1024

4.2.1 A cadeia de Markov

Uma vez definido como as seqüências serão representadas e como determinar as seqüências viáveis, podemos descrever o sistema de inspeção atual como uma cadeia de Markov estacionária considerando o espaço de estados discretos

$$\Omega = \{(0,0), \dots, (0,2^h-1), (1,0), \dots, (1,2^h-1), \dots, (w_i, z_j), \dots, (n,2^h-2), (n,2^h-1)\}.$$

Cada estado é descrito por um par de números inteiros w e z , totalizando $2^h(n+1)$ estados dos quais $\gamma(h, \nu)(n+1)$ serão estados viáveis. Entenda-se por estado viável aquele estado acessível por algum outro estado da cadeia. O inteiro w continua descrevendo o número de inspeções realizadas no Estado II. O inteiro z , em sua representação binária, corresponde à seqüência de h itens inspecionados, onde o bit **0** indica um item declarado conforme e o bit **1** um item declarado defeituoso.

As equações de probabilidade de transição e custo não dependerão especificamente do valor de z , e sim de como aquela seqüência binária é classificada. Por isso, seja $CB(z)$ uma função que retorna o número de bits **1** do número z (acrônimo para Contador de Bits), $PB(z)$ uma função que indica qual é o primeiro bit do número z (acrônimo para Primeiro Bit), e $\Psi(z, \nu)$ uma função indicadora assim definida

$$\Psi(z, \nu) = \begin{cases} 0, & \text{se } CB(z) < \nu \text{ e } PB(z) = 0 \\ 1, & \text{se } CB(z) < \nu \text{ e } PB(z) = 1 \\ 2, & \text{se } CB(z) = \nu \text{ e } PB(z) = 1 \\ -1, & \text{caso contrário} \end{cases}$$

Quando $\Psi(z, \nu) = 0$, a seqüência é válida e o item recém-inspecionado foi declarado conforme. Quando $\Psi(z, \nu) = 1$ a seqüência é válida e, apesar do item recém-inspecionado ser declarado não-conforme, não há ajuste do processo (o número de inspeções não-conformes na seqüência não atingiu ν). Quando $\Psi(z, \nu) = 2$, o item recém-inspecionado é declarado não conforme e o número de itens não-conformes na seqüência atinge ν , o que leva ao ajuste do processo.

Nas equações das próximas subseções, o segundo valor do par ordenado não será o valor numérico de z , mas sim o resultado de $\Psi(z, \nu)$. Para simplificar a notação, o valor da função $\Psi(z, \nu)$ será apresentado em negrito e sublinhado.

4.2.2 Probabilidades de Transição

As equações (4.21) e (4.22) correspondem à transição entre estados da cadeia que produzem sob o Estado I.

$$P_{(0,k),(0,\underline{0})|k=\underline{0},\underline{1}} = (1 - \pi)^m [(1 - p_1)(1 - \alpha^*) + p_1\beta^*]. \quad (4.21)$$

$$P_{(0,k),(0,l)|k=\underline{0},\underline{1} \atop l=\underline{1},\underline{2}} = (1 - \pi)^m [(1 - p_1)\alpha^* + p_1(1 - \beta^*)]. \quad (4.22)$$

As equações (4.23) e (4.24) indicam que entre os estados de origem/destino da cadeia ocorreu a mudança Estado I $\rightarrow$ Estado II.

$$P_{(0,k),(1,\underline{0})|k=\underline{0},\underline{1}} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)(1 - \alpha^*) + \psi(m - i + 1)\beta^*] \quad (4.23)$$

$$P_{(0,k),(1,l)|k=\underline{0},\underline{1} \atop l=\underline{1},\underline{2}} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)\alpha^* + \psi(m - i + 1)(1 - \beta^*)]. \quad (4.24)$$

Nas equações (4.25) e (4.26) as transições são entre estados da cadeia que produzem sob o Estado II.

$$P_{(j-1,k),(j,\underline{0})|j=2\dots n \atop k=\underline{0},\underline{1}} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm - i + 1)(1 - \alpha^*) + \psi(jm - i + 1)\beta^*] \quad (4.25)$$

$$P_{(j-1,k),(j,l)|j=2\dots n \atop k=\underline{0},\underline{1} \atop l=\underline{1},\underline{2}} = \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} [\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i + 1)(1 - \beta^*)]. \quad (4.26)$$

O truncamento da cadeia é representado pelas equações (4.27) e (4.28).

$$P_{(n,k),(n,\underline{0})|k=\underline{0},\underline{1}} = P_{(n-1,k),(n,\underline{0})} \quad (4.27)$$

$$P_{(n,k),(n,l)} \Big|_{\substack{k=\underline{0},\underline{1} \\ l=\underline{1},\underline{2}}} = P_{(n-1,k),(n,l)} \quad (4.28)$$

Por fim, as equações (4.29)-(4.32) indicam as transições a partir de estados da cadeia nos quais há ajuste do processo.

$$P_{(j,\underline{2}),(0,\underline{0})} \Big|_{j=1\dots n} = P_{(0,\underline{0}),(0,\underline{0})} \quad (4.29)$$

$$P_{(j,\underline{2}),(0,k)} \Big|_{\substack{j=1\dots n \\ k=\underline{1},\underline{2}}} = P_{(0,\underline{0}),(0,k)} \quad (4.30)$$

$$P_{(j,\underline{2}),(1,\underline{0})} \Big|_{j=1\dots n} = P_{(0,\underline{0}),(1,\underline{0})} \quad (4.31)$$

$$P_{(j,\underline{2}),(1,k)} \Big|_{\substack{j=1\dots n \\ k=\underline{1},\underline{2}}} = P_{(0,\underline{0}),(1,k)} \quad (4.32)$$

A Tabela 4.6 ilustra a obtenção da matriz de transição $\mathbf{P}_n$ a partir das equações (4.21)-(4.32) quando $n = 3$, $h = 3$ e $\nu = 2$. O par ordenado (w, z) que representa cada estado tem o valor z representado de três formas: a decimal, a binária, e aquela de acordo com a função $\Psi(z, \nu)$, esta última utilizada para indexar as equações de probabilidade. No corpo da Tabela 4.6 é indicada a equação necessária para o cálculo da probabilidade de transição entre dois estados. Nas células em branco o valor desta probabilidade é nulo. As linhas e colunas com destaque em cinza na Tabela 4.6 correspondem aos estados inviáveis.

4.2.3 Custos dos estados da cadeia

As equações (4.33)-(4.41) correspondem aos componentes de custo para estados em que o item recém-inspecionado foi declarado conforme.

$$\xi_{(0,\underline{0})} = c_{nc}(m-1)p_1 \quad (4.33)$$

$$\theta_{(0,\underline{0})} = c_{nc} \frac{p_1\beta^*}{(1-p_1)(1-\alpha^*) + p_1\beta^*} \quad (4.34)$$

$$\varphi_{(0,\underline{0})} = 0. \quad (4.35)$$

$$\xi_{(1,\underline{0})} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \quad (4.36)$$

$$\theta_{(1,\underline{0})} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)\beta^*}{\bar{\psi}(m-i+1)(1-\alpha^*) + \psi(m-i)\beta^*} \right] \quad (4.37)$$

$$\varphi_{(1,\underline{0})} = 0. \quad (4.38)$$

$$\xi_{(j,\underline{0})|j=2\dots n} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m-i+l+1) \quad (4.39)$$

$$\theta_{(j,\underline{0})|j=2\dots n} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm-i+1)\beta^*}{\bar{\psi}(jm-i+1)(1-\alpha^*) + \psi(jm-i+1)\beta^*} \right] \quad (4.40)$$

$$\varphi_{(j,\underline{0})|j=2\dots n} = 0 \quad (4.41)$$

As equações (4.42)-(4.50) correspondem aos componentes de custo para estados em que o item recém-inspecionado foi declarado não-conforme. Para aqueles estados no qual há ajuste, $\varphi = c_a$.

$$\xi_{(0,l)|l=\underline{1},\underline{2}} = c_{nc}(m-1)p_1 \quad (4.42)$$

$$\theta_{(0,l)|l=\underline{1},\underline{2}} = c_{s-nc} \frac{p_1(1-\beta^*)}{(1-p_1)\alpha^* + p_1(1-\beta^*)} + c_{s-c} \frac{(1-p_1)\alpha^*}{(1-p_1)\alpha^* + p_1(1-\beta^*)} \quad (4.43)$$

$$\varphi_{(0,l)|l=\underline{1},\underline{2}} = c_a I_{(\underline{2})}(l). \quad (4.44)$$

$$\xi_{(1,l)|l=\underline{1},\underline{2}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \quad (4.45)$$

$$\begin{aligned} \theta_{(1,l)|l=\underline{1},\underline{2}} = & c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)(1-\beta^*)}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] + \\ & c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(m-i+1)\alpha^*}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] \end{aligned} \quad (4.46)$$

$$\varphi_{(1,l)|l=\underline{1},\underline{2}} = c_a I_{(\underline{2})}(l). \quad (4.47)$$

$$\xi_{(j,l)|j=2\dots n, l=\underline{1},\underline{2}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m-i+l+1) \quad (4.48)$$

$$\begin{aligned} \theta_{(j,l)|j=2\dots n, l=\underline{1},\underline{2}} = & c_{s-nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm-i+1)(1-\beta^*)}{\bar{\psi}(jm-i+1)\alpha^* + \psi(jm-i+1)(1-\beta^*)} \right] + \\ & c_{s-c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(jm-i+1)\alpha^*}{\bar{\psi}(jm-i+1)\alpha^* + \psi(jm-i+1)(1-\beta^*)} \right] \end{aligned} \quad (4.49)$$

$$\varphi_{(j,l)|j=2\dots n, l=\underline{1},\underline{2}} = c_a I_{(\underline{2})}(l). \quad (4.50)$$

4.2.4 Determinação da política ótima para as diferentes funções de deterioração aplicando o Critério II

Tal qual na aplicação do Critério I, utilizou-se os valores da Tabela 3.1, $\Delta_{\max} = 1 \times 10^{-4}$ e $s = \lfloor r/2 \rfloor + 1$. As Figuras 4.7, 4.8 e 4.9 mostram o custo médio obtido para as três funções de deterioração utilizadas quando o Critério II é aplicado para um histórico de até 7 itens. Nos três casos houve redução do custo médio obtido quando comparado àquele obtido utilizando o Critério I. As políticas ótimas de parada foram $(h = 6, \nu = 3)$ para a função Linear e $(h = 7, \nu = 3)$ para as funções Exponencial e Linear, como mostra a Tabela 4.7. Para os três casos, a política de inspeção ótima de inspeção foi $(m = 4, r = 1)$.

Figura 4.7: Custo versus política de parada para o modelo Exponencial e o Critério II.

Figura 4.8: Custo versus política de parada para o modelo Linear e o Critério II.

Os pontos ausentes na Figura 4.7 para as políticas de parada $(h = 7, \nu = 6)$ e $(h = 7, \nu = 7)$ indicam que a restrição da fração de itens defeituosos produzida foi violada. A Tabela 4.8 compara os custos ótimos obtidos com os Critérios I e II. A aplicação do Critério II produziu uma redução no custo ótimo entre 1,2% e 3,3%. A magnitude desta

Figura 4.9: Custo versus política de parada para o modelo Constante e o Critério II.

Tabela 4.7: Comparação entre as políticas ótimas de controle utilizando os Critérios I e II para as diferentes funções de deterioração, com $h \leq 7$.

Critério	Função	C°	m°	r°	h°	ν°
Critério I ($\nu = h$)	Exponencial	0,2766	4	1	2	2
	Linear	0,2705	4	1	2	2
	Constante	0,2561	5	1	2	2
Critério II ($h \leq 7$)	Exponencial	0,2678	4	1	7	3
	Linear	0,2617	4	1	7	3
	Constante	0,2529	4	1	6	3

redução é similar àquela obtida no Capítulo 2 quando as políticas de inspeção com e sem classificações repetidas foram comparadas.

Tabela 4.8: Comparação entre os custos ótimos obtidos com os Critérios I e II.

Deterioração	C° (Critério I)	C° (Critério II)	Redução em C°
Exponencial	0,2766	0,2678	3,2%
Linear	0,2705	0,2617	3,3%
Constante	0,2561	0,2529	1,2%

4.2.5 Problemas na aplicação do Critério II

Uma observação cuidadosa das Figuras 4.7-4.9 provavelmente fomentará duas dúvidas. A primeira relacionada ao h° nos modelos Exponencial e Linear, que coincide com o maior h utilizado na busca. A segunda, mais sutil, relacionada à redução do custo para valores $\nu = 3, 4$ e 5 com o aumento de h nos modelos Exponencial e Linear. Para explicitar estas questões, a Tabela 4.9 mostra o custo ótimo para os três modelos de degradação em função de h e ν . As células na cor cinza correspondem aos custos obtidos como ótimos

nas Figuras 4.7-4.9, enquanto os valores sublinhados mostram o menor custo obtidos para o ν especificado.

Tabela 4.9: Custo ótimo em função de h e ν para as diferentes funções de deterioração.

	C° (Exponencial)				C° (Linear)				C° (Constante)			
	ν				ν				ν			
h	2	3	4	5	2	3	4	5	2	3	4	5
2	0,2766	-	-	-	0,2705	-	-	-	<u>0,2561</u>	-	-	-
3	0,2722	0,3036	-	-	<u>0,2672</u>	0,2886	-	-	0,2569	0,2638	-	-
4	<u>0,2717</u>	0,2861	0,3207	-	0,2675	0,2745	0,3112	-	0,2598	0,2558	0,2797	-
5	0,2724	0,2761	0,3140	0,3298	0,2688	0,2682	0,2929	0,3183	0,2630	0,2534	0,2657	0,2966
6	0,2735	0,2708	0,2983	0,3209	0,2704	0,2641	0,2843	0,3097	0,2659	<u>0,2529</u>	0,2610	0,2776
7	0,2748	0,2678	0,2897	0,3169	0,2722	0,2617	0,2771	0,2978	0,2686	0,2531	0,2589	0,2691
8	0,2763	0,2652	0,2843	0,3110	0,2739	0,2604	0,2724	0,2906	0,2711	0,2534	0,2576	0,2652
9	0,2777	0,2637	0,2789	0,3016	0,2757	0,2597	0,2692	0,2859	0,2733	0,2540	0,2570	0,2633
10	0,2791	<u>0,2630</u>	<u>0,2747</u>	<u>0,2949</u>	0,2774	<u>0,2592</u>	<u>0,2668</u>	<u>0,2810</u>	0,2753	0,2547	<u>0,2567</u>	<u>0,2624</u>

Os valores na Tabela 4.9 mostram que, quando a busca foi ampliada até $h = 10$, apenas o modelo Constante manteve a política de parada ótima. Note que, fixado ν , o aumento do histórico h pode levar a um ponto de mínimo (e.g. modelo Constante, $\nu = 2$ ou 3) ou a valores decrescentes do custo que parecem convergir para um ponto de mínimo, sem contudo tê-lo alcançado (e.g. modelos Exponencial ou Linear, $\nu = 3, 4$ ou 5). A solução primária consiste em ampliar h até garantir que o mínimo global tenha sido alcançado. Porém, não é possível aumentar indefinidamente o tamanho h por questões computacionais.

Para exemplificar o problema, considere o cálculo do custo ótimo quando $h = 11$ e $\nu = 5$ e o modelo de degradação é o Exponencial. Neste caso, há 772 estados viáveis na cadeia de Markov e $n = 65$ para que ocorra a convergência do cálculo do custo, com a política ótima $m = 2, r = 1$. A matriz de transição para este caso possui $(772 \times (65 + 1)) = 50.952$ linhas e colunas, o que totaliza 2.596.106.304 elementos. Como a representação de cada valor real de dupla precisão consome 8 bytes, esta matriz de transição necessitaria de 19,34 Gigabytes livres na memória RAM para ser representada. Adicione a este valor a memória necessária para que o software resolva o sistema linear.

A estratégia utilizada para reduzir o impacto deste problema foi representar a matriz de transição como uma matriz esparsa no Matlab. Observe a matriz representada na Tabela 4.6. Cada linha da matriz de transição pode conter até 4 probabilidades não-nulas. O número máximo de elementos não nulos será $50.952 \times 4 = 203.808$. Na representação por

matrizes esparsas, cada elemento não nulo é representado por 3 valores: o índice da linha, o índice da coluna e o próprio valor a ser armazenado. Portanto, o tamanho máximo da matriz de transição será $203.808 \times 3 \times 8 \text{ bytes} = 4,66 \text{ Megabytes}$. Porém, no momento de resolver o sistema linear (2.9), a matriz esparsa será manipulada (e.g. Fatoração LU) e as matrizes triangulares resultantes, ainda que temporárias, possuirão um grande número de elementos não-nulos, exigindo uma significativa quantidade de memória RAM. Não é possível calcular exatamente a memória necessária, mas durante o cálculo com $h = 11$ e $\nu = 5$ e $n = 65$, a sessão do Matlab saltou de 150 Megabytes para mais de 900 Megabytes.

A Figura 4.10 mostra uma representação gráfica da matriz $(\mathbf{P}^T - \mathbf{I})$ e das matrizes triangulares inferior (L) e superior (U) resultantes da decomposição LU quando $n = 5$, $h = 4$ e $\nu = 3$.

Figura 4.10: Representação das matrizes esparsas utilizadas na solução do sistema linear quando $n = 5$, $h = 4$ e $\nu = 3$.

Os pontos nos gráficos indicam os elementos não-nulos nas matrizes, e nz corresponde ao total de valores não nulos. A matriz originalmente submetida à decomposição LU, com 84 linhas e colunas, possui apenas $389/84^2 = 5,5\%$ de elementos não-nulos. Já as matrizes triangular inferior e superior possuem 16,3% e 39,2% respectivamente. Se fossem representadas de maneira convencional, cada uma das matrizes da Figura 4.10 ocuparia 56.448 bytes, enquanto a representação esparsa exigiu menos memória: 5.116, 14.116 e 33.520 bytes, respectivamente. Estes valores não são necessariamente aqueles consumidos durante a solução do sistema linear pelo Matlab, mas servem como ilustração de que, mesmo usando a representação esparsa, não será possível aumentar h e ν indefinidamente.

As Figuras 4.11-4.13 mostram o custo ótimo para as três funções de degradação utilizadas com o aumento de h e $\nu = 3$ e 4. Apenas para a função de degradação Constante o mínimo havia sido encontrado com a restrição de $h \leq 7$, como pode ser visto na Tabela 4.10. Utilizando a função Exponencial, a política de parada $h = 13, \nu = 3$ leva ao custo $C^\circ = 0,2622$ (2,09% menor que aquele da Tabela 4.8 e 5,22% menor que aquele do Critério

I). Quando a função Linear foi utilizada, o custo $C^o = 0,2588$ foi obtido com a política de parada $h = 20, \nu = 4$ (1,12% menor que o ótimo da Tabela 4.8 e 4,35% menor que o obtido com o Critério I).

Figura 4.11: Custo médio versus h ($\nu = 3; 4$) para o modelo Exponencial e o Critério II.

Figura 4.12: Custo médio versus h ($\nu = 3; 4$) para o modelo Linear e o Critério II.

Figura 4.13: Custo médio versus h ($\nu = 3; 4$) para o modelo Constante e o Critério II.

Tabela 4.10: Comparação entre as políticas ótimas de controle utilizando os Critérios I e II para as diferentes funções de deterioração

Critério	Função	C^o	m	r	h	ν
Critério I	Exponencial	0,2766	4	1	2	2
	Linear	0,2705	4	1	2	2
	Constante	0,2561	5	1	2	2
Critério II ($h \leq 7$)	Exponencial	0,2678	4	1	7	3
	Linear	0,2617	4	1	7	3
	Constante	0,2529	4	1	6	3
Critério II	Exponencial	0,2622	5	1	13	3
	Linear	0,2588	5	1	20	4
	Constante	0,2529	4	1	6	3

4.2.6 Custos na aplicação dos Critérios I e II

Considerar um histórico de inspeções na tomada de decisão sobre a parada do processo, de acordo com a abordagem realizada até aqui, não representa um aumento no custo operacional do sistema. Suponha que armazenar a informação e gerenciá-la seja dispendioso, e que cada item adicional considerado no histórico tenha o custo $c_h = 0,05c_i$. Ou seja, há um custo linear $0,05c_i(h - 1)$ acrescido a cada estado da cadeia. Com isso, as grandes seqüências devem ser evitadas por tornarem-se mais custosas. As Figuras 4.14-4.16 mostram o efeito da existência deste custo, e as políticas ótimas são apresentadas na Tabela 4.11.

Figura 4.14: Custo médio versus política de parada utilizando o modelo Exponencial e o Critério II quando há custo relativo ao histórico.

O histórico considerado na política de parada foi significativamente reduzido quando com-

Figura 4.15: Custo médio versus política de parada utilizando o modelo Linear e o Critério II quando há custo relativo ao histórico.

Figura 4.16: Custo médio versus política de parada utilizando o modelo Constante e o Critério II quando há custo relativo ao histórico.

Tabela 4.11: Comparação entre os custos ótimos obtidos com os Critérios I, II, e II com custo relativo ao histórico.

Critério	Função	C^o	m	r	h	ν
Critério I (sem custo de armazenagem)	Exponencial	0,2766	4	1	2	2
	Linear	0,2705	4	1	2	2
	Constante	0,2561	5	1	2	2
Critério II (sem custo de armazenagem)	Exponencial	0,2622	5	1	7	3
	Linear	0,2588	5	1	13	3
	Constante	0,2529	4	1	20	3
Critério II (com custo de armazenagem $0,05c_i(h-1)$)	Exponencial	0,2765	4	1	2	2
	Linear	0,2713	7	1	3	2
	Constante	0,2583	6	1	2	2

parado com aqueles das Figuras 4.11-4.13. Portanto, se houver custo para gerenciar a informação do histórico (que no exemplo empregado foi de 5% do custo de uma inspeção), os problemas no uso do Critério II podem não ser experimentados.

4.3 Considerações finais

Como esperado, os critérios de parada considerados neste Capítulo levam a resultados mais econômicos que aqueles em que a decisão se baseia na inspeção de um único item. Para a implementação do Critério I, parar o processo quando uma seqüência de h itens não-conformes for obtida na inspeção, a reindexação das equações é simples e o modelo não apresenta problemas de memória durante o cálculo. Antes que h torne-se suficientemente grande para provocar tais problemas, ocorrerá a violação do critério $\phi_{\max}$.

A implementação do Critério II, apesar de mais engenhosa, consiste também da reindexação das equações já apresentadas no Capítulo 3. Porém, há um preço a ser pago por esta adaptação: o consumo de memória. Com o aumento de h e ν , há o aumento do número de estados viáveis $\gamma(h, \nu)$. Não bastasse isso, o aumento de ν tende a reduzir o intervalo entre inspeções m , o que exige o aumento da dimensão da matriz de transição (n) para que a convergência da cadeia de Markov ocorra. Como o número de elementos da matriz de transição é $((n+1)\gamma(h, \nu))^2$, a memória e tempo de execução devem aumentar, pelo menos, com o quadrado destas variáveis.

O Critério II, quando $\nu = h$, corresponde exatamente ao Critério I. Como mostrado neste Capítulo, o número de estados requeridos pelo Critério II com $\nu = h$ será $2^h(n+1)$, enquanto a implementação do modelo da forma apresentada como Critério I demanda $(h+1)(n+1)$ estados. Portanto, quando o critério $\nu = h$ for testado, é mais eficiente utilizar a representação da cadeia apresentada no Critério I. O programa apresentado no Apêndice A (e também disponibilizado em www.anderson.trindade.nom.br/tese) o faz, bem como descarta os estados inviáveis para minimizar o consumo de memória.

CAPÍTULO 5

Conclusões

No começo deste trabalho, propôs-se o desenvolvimento incremental de um modelo para controle *on-line* de processos por atributos utilizando cadeias de Markov com estados discretos. O espaço de estados foi sempre representado por um par de números inteiros (w, z) : o primeiro relacionado à condição de produção do processo, e o segundo relacionado à classificação do item inspecionado.

No Capítulo 2 apresentou-se a primeira versão do modelo, uma cadeia com seis estados, que introduziu o uso de classificações repetidas em um sistema de inspeção sujeito a erros de classificação do item inspecionado. Considerou-se que processo de produção muda sua proporção de itens não-conformes de p_1 para p_2 segundo uma distribuição geométrica com parâmetro π . Os resultados mostraram que, dependendo da combinação dos parâmetros que governam o processo, erros da ordem de 1% já justificariam o uso de classificações repetidas. Para processos que, sob controle, possuem alta probabilidade de produzir itens conformes, o erro α (classificar um item conforme como não-conforme) mostrou-se mais relevante que o erro β (classificar um item não-conforme como conforme).

As contribuições do Capítulo 3 consistiram em assumir que a fração de itens não-conformes produzidos com o processo fora de controle era descrita por uma função $\psi(x)$, e em considerar uma restrição estatística para a proporção média de itens não-conformes produzidos

pelo processo. Com isso, n inspeções com o processo operando na condição fora de controle precisariam ser representadas pelos estados da cadeia, o que totalizaria $2(n + 1)$ estados. Mostrou-se que n pode ser determinado, dentro de uma precisão $\Delta_{\max}$, utilizando como critério a variação relativa do custo obtido, e que o cálculo da proporção de itens defeituosos pode ser derivado da função custo.

Nos Capítulos 2 e 3, a decisão sobre a parada para ajuste é tomada como base um único item inspecionado. O Capítulo 4 teve por objetivo melhorar esta tomada de decisão ao considerar, no modelo, o que ocorreu nas últimas h inspeções. A primeira opção foi alterar a política de parada, ajustando o processo quando uma seqüência de h inspeções não-conformes fosse observada. Neste caso, $(h + 1)(n + 1)$ estados foram necessários para representar a cadeia de Markov, e uma pequena adaptação das equações desenvolvidas no Capítulo 3 fez-se necessária. A segunda opção generalizou a política de parada para: ajustar o processo quando ν inspeções não-conformes forem observadas em uma seqüência de h inspeções. Agora, o número de estados é função de h e ν e pode ser de até $2^h(n + 1)$. Mais uma vez, utilizou-se o cerne das equações do Capítulo 3. Como esperado, os resultados mostraram que tomar a decisão com base em um histórico de inspeções reduz significativamente o custo quando comparado àquele obtido no Capítulo 3. Porém, a redução pode ser menor caso exista algum custo em gerenciar a informação referente ao histórico de itens inspecionados.

O uso das cadeias de Markov fez com que a complexidade fosse transferida das equações para “como” as cadeias são definidas. Isso permitiu chegar ao modelo do Capítulo 4 com alterações apenas incrementais. Porém, há um efeito colateral: o aumento de h e ν será limitado pela quantidade de memória RAM disponível para representar e resolver o sistema linear resultante. Caso as funções $\psi(x)$ sejam de lenta convergência (i.e., exijam grandes valores de n , como a função Exponencial empregada) a restrição se fará para menores valores de h e ν . Caso $\psi(x) = \text{constante}$, $n = 2$ e maiores valores de h e ν poderão ser pesquisados.

Como em todo trabalho deste tipo, nem todas as possibilidades foram esgotadas. Assim, seguem algumas das possibilidades para trabalhos futuros:

- Quando classificações repetidas forem utilizadas, considerar no custo apenas as classificações necessárias.

Suponha que o item inspecionado seja submetido a $r = 3$ classificações repetidas e o critério para declarar o item conforme seja que $s = 1$ ou mais classificações

sejam conformes. O modelo atual contabiliza o custo das 3 classificações. Porém, se a primeira classificação já for conforme, não há necessidade de realizar as demais. Isso tornará o uso das classificações repetidas mais competitivo. As modificações deverão ser realizadas no custo de todos os estados das cadeias de Markov.

- Considerar um critério misto na política de parada quando o histórico é considerado.

Suponha que o histórico considerado seja de $h = 7$ inspeções. O processo poderia ser parado caso fossem observadas ν_1 inspeções não-conformes dentre as h inspeções ou uma sequência $\nu_2 < \nu_1$ inspeções não-conformes. As alterações deverão ser feitas na forma como as equações do Capítulo 4 são indexadas.

- Modificar o sistema de convergência da cadeia de Markov para a função de deterioração.

Acredita-se que seja possível obter a convergência representando a função $\psi(x)$ por um valor constante, levando à mesma política de controle. A questão seria como obter este valor constante.

Referências Bibliográficas

- ADAMS, B. M.; WOODALL, W. H. An analysis of Taguchi's on-line process-control procedure under a random-walk model. *Technometrics*, v. 31, n. 4, p. 401–413, 1989.
- BORGES, W.; HO, L. L.; TURNES, O. An analysis of Taguchi's on-line quality monitoring procedure for attributes with diagnosis errors. *Applied Stochastic Models in Business and Industry*, v. 17, p. 261–276, 2001.
- BOX, G. E. P.; KRAMER, T. Statistical process monitoring and feedback adjustment - a discussion. *Technometrics*, v. 34, p. 251–285, 1992.
- BOX, G. E. P.; LUCENO, A. Selection of sampling interval and action limit for discrete feedback adjustment. *Technometrics*, v. 36, p. 369–378, 1994.
- CHOU, S.; WANG, M. C. Taguchi's method and empirical procedure for on-line process control. *Communications in Statistics. Theory and Methods*, v. 25, n. 2, p. 301–320, 1996.
- DASGUPTA, T. An economic inspection interval for control of defective items in a hot rolling mill. *Journal of Applied Statistics*, v. 30, p. 273–282, 2003.
- DUNCAN, A. J. The economic design of $\bar{X}$ charts used to maintain current control of a process. *Journal of the American Statistical Association*, v. 51, p. 228–242, 1956.
- JOHNSON, N. L.; KOTZ, S.; WU, X. *Inspection Errors for Attributes in Quality Control*. London: Chapman and Hall, 1991.
- MONTGOMERY, D. C. *Introdução ao Controle Estatístico da Qualidade*. 4. ed. Rio de Janeiro: LTC, 2004.
- NANDI, S. N.; SREEHARI, M. Economy based on-line quality control method for attributes. *The Indian Journal of Statistics*, v. 59, p. 384–395, 1997.

- NANDI, S. N.; SREEHARI, M. Some improvements in Taguchi's economic method allowing continued quality deterioration in production process. *Communications in Statistics. Theory and Methods*, v. 28, n. 5, p. 1169–1181, 1999.
- NAYEBPOUR, M. R.; WOODALL, W. H. An analysis of Taguchi's on-line quality-monitoring procedures for attributes. *Technometrics*, v. 35, n. 1, p. 53–60, 1993.
- RANJAN, P.; XIE, M.; GOH, T. N. Optimal control limits for CCC charts in the presence of inspection errors. *Quality and Reliability Engineering International*, v. 19, p. 149–160, 2003.
- ROSS, S. M. *Introduction to Probability Models*. 8. ed. San Diego: Academic Press, 2003.
- SOUZA, N. R. de. *Controle estatístico de processos: uma perspectiva bayesiana para atributos*. 2002. Dissertação (Mestrado) — Instituto de Matemática e Estatística, São Paulo.
- SRIVASTAVA, M. S.; WU, Y. A second order approximation to Taguchi et al.'s on-line control procedure. *Communications in Statistics A*, v. 20, p. 2149–2168, 1991.
- SRIVASTAVA, M. S.; WU, Y. On-line control procedures under the random-walk model with measurement error and attribute observations. *Canadian Journal of Statistics*, v. 22, p. 377–386, 1994.
- SRIVASTAVA, M. S.; WU, Y. An improved version of Taguchi's on-line control procedure. *Journal of Statistical Planning and Inference*, v. 43, p. 133–145, 1995.
- SRIVASTAVA, M. S.; WU, Y. Economical quality control procedures based on symmetric random walk model. *Statistica Sinica*, v. 6, p. 389–402, 1996.
- SRIVASTAVA, M. S.; WU, Y. On-line quality control procedures for a random walk model with measurement error. *Sequential Analysis*, v. 16, p. 93–105, 1997.
- SRIVASTAVA, M. S.; WU, Y. Economical process adjustment with measurement error. *Communications in Statistics. Theory and Methods*, v. 28, p. 989–1003, 1999.
- TAGUCHI, G.; CHOWDHURY, S.; WU, Y. *Taguchi's Quality Engineering Handbook*. New Jersey: John Wiley & Sons, Inc, 2004.
- TAGUCHI, G.; ELSAYED, E. A.; HSIANG, T. *Quality Engineering in Production Systems*. New York: McGraw-Hill, 1989.
- TRINDADE, A. L. G.; HO, L. L.; QUININO, R. C. Controle on-line por atributos com erros de classificação: uma abordagem econômica com classificações repetidas. *Pesquisa Operacional*, v. 27, p. 105–116, 2007.
- TRINDADE, A. L. G.; HO, L. L.; QUININO, R. C. Monitoring process for attributes with quality deterioration and diagnosis errors. *Applied Stochastic Models in Business and Industry*, v. 23, p. 339–358, 2007.
- WANG, C.-H. Economic off-line quality control strategy with two types of inspection errors. *European Journal of Operational Research*, v. 179, p. 148–159, 2007.
- WANG, M. C.; YUE, J. Economic design of process adjustment for on-line control. *International Journal of Production Research*, v. 29, n. 5, p. 805–823, 2001.

APÊNDICE A

Programa desenvolvido

O programa desenvolvido em Matlab (versão 7.0 ou superior) é apresentado neste anexo e consiste basicamente da função `f_otimo.m` e `f_det.m`. O primeiro arquivo contém uma série de sub-funções necessárias ao cálculo do modelo e à busca dos parâmetros ótimos m° , r° , s° e C° , fixados h e ν . No arquivo `f_det.m` estão definidas as funções de deterioração, e é este arquivo que deve ser modificado caso deseje definir sua própria função de deterioração. Por fim, são anexados dois programas ilustrando como chamar a função `f_otimo.m`.

Há uma série de operações repetitivas durante a busca de parâmetros ótimos, como o cálculo de α^* e β^* , e a determinação dos estados viáveis dados h e ν . Por isso o programa, quando executado, criará um diretório denominado **data** no qual serão armazenados arquivos de texto com dados a serem reutilizados (o que acelera o processo de busca). Portanto, execute o programa em um diretório em que dados possam ser salvos.

Por fim, recomenda-se visitar o site www.anderson.trindade.nom.br/tese para verificar atualizações e fazer o download dos arquivos, pois lá estão disponibilizados alguns programas que foram utilizados para gerar figuras e tabelas deste trabalho. Em caso de problemas no acesso ou no uso dos programas escreva para laecio1@yahoo.com.br.

f_optimum.m

```

1  function [mo,ro,so,Co,phio,chainlog,output]=f_optimum(param,
    modelparam,mrange,rrange,srange,flags,isisim)
2  %=====
3  % Contribuições para o controle de processos por atributos
4  % Tese - Doutorado em Engenharia de Produção
5  % Doutorando: Anderson Laecio Galindo Trindade
6  % Orientador: Linda Lee Ho
7  %
8  % versao: 2008-03-01 (aaaa-mm-dd)
9  %
10 % Sintaxe básica:
11 %
12 % [mo,ro,so,Co,phio,chainlog,output]=optimum(param,mrange,rrange,flags,
13 %                                           modelparam,isisim)
14 %
15 %     ENTRADA:
16 %     param    vetor dos parâmetros
17 %               [pi pl ci cnc ca cs_nc cs_c alpha beta deltamax phimax h nu ch]
18 %               (veja a tese para as definições, ou a função getdados)
19 %               ch é o custo de cada item extra adicionado ao histórico. Se
20 %               este valor for suprimido, não há custo extra.
21 %
22 %     modelparam vetor de parâmetros utilizados nos modelos de
23 %               deterioração. O primeiro elemento do vetor deve ser
24 %               necessariamente o número do modelo utilizado. Caso
25 %               deseje variar os parâmetros da função de deterioração,
26 %               use as demais posições no vetor para especificar os
27 %               parâmetros e programe a função adequadamente.
28 %
29 %               modelparam=[1], usa o Modelo Exponencial
30 %               modelparam=[2], usa o Modelo Linear
31 %               modelparam=[3], usa o Modelo Constante
32 %               modelparam=[4], usa a Função definida pelo usuário
33 %
34 %               Veja a função f_det.m para alterar/definir sua própria
35 %               função
36 %
37 %     mrange    vetor com os limites de busca para m
38 %               [mmin mmax]
39 %               Exemplos de uso:
40 %               [mmin mmax] = Busca entre os valores mmin e mmax
41 %               [] = Busca ilimitada (o programa decide quando parar a
42 %               busca) (padrão)
43 %
44 %     rrange    vetor com os limites de busca para r
45 %               [rmin rmax]
46 %               Exemplos de uso:
47 %               [rmin rmax] = busca entre os valores rmin e rmax

```


f_optimum.m

```

48 %      [] = Busca ilimitada. Neste caso, o programa decide quando
49 %      parar a busca, começando de r=1; (padrão)
50 %
51 %      srange  vetor com os limites de busca para s (número mínimo de
52 %      classificações conformes para que o item seja declarado
53 %      conforme)
54 %      Exemplos de uso:
55 %      [smin smax] = busca entre os valores smin e smax (use esta
56 %      opção observando os valores utilizados em rrange)
57 %      [] = busca entre s=1 e o r que estiver sendo utilizado;
58 %      0 = utiliza o critério "mais da metade das classificações
59 %      conformes para declarar o item conforme". (padrão)
60 %
61 %
62 %      flags  Parâmetro opcional
63 %      vetor com variáveis flag (indicadoras)
64 %      [gflag chainflag sflag schainlog step_big]
65 %
66 %      gflag = 1 para mostrar gráficos
67 %      gflag = 0 para não exibi-los. (padrão)
68 %
69 %      chainflag = (valor inteiro > 1) define o tamanho n da
70 %      cadeia para este valor. Esta opção, se escolhida,
71 %      ignora o critério deltamax.
72 %      chainflag = 0, o tamanho da cadeia é determinado a cada nova
73 %      combinação de m e r que for calculada, considerando o
74 %      deltamax. (padrão)
75 %      chainflag = 1, a cada passo da busca, o tamanho n da cadeia
76 %      será o maior já obtido na busca até então.
77 %
78 %      sflag = 1 pára a busca tão logo o mínimo seja encontrado.
79 %      (padrão)
80 %      sflag = 0 pára a busca apenas quando o critério phimax for
81 %      violado
82 %
83 %      schainlog = 1 para mostrar a evolução da convergência do custo
84 %      com o aumento da cadeia (não use esta opção quando a função
85 %      OPTIMUM for utilizada várias vezes por um programa externo,
86 %      pois o cálculo se tornará mais lento).
87 %      schainlog = 0 para não exibir a evolução da convergência
88 %      (padrão)
89 %
90 %      step_big >=1 acelera a busca pelo tamanho da cadeia. Valor
91 %      padrão = 5. Use step_big=1 quando estiver interessado em
92 %      analisar o tamanho da cadeia passo a passo. (parâmetro
93 %      opcional no vetor)
94 %
95 %      isim  Parâmetro opcional.

```

f_optimum.m

```

196 %          simulação atual no loop. Útil quando a função optimum for
197 %          utilizada dentro de um loop para a variação de parâmetros,
198 %          por exemplo
199 %
200 %
201 %          SAÍDA:
202 %          mo          Intervalo ótimo de amostragem
203 %          ro          Número ótimo de classificações repetidas
204 %          so          Número mínimo de classificações conformes para declarar
205 %                      o item inspecionado como conforme
206 %          Co          Custo ótimo - Se o valor de Co for Inf (infinito), isso
207 %                      significa que a restrição phi_max foi violada
208 %          phio        Fração de itens defeituosos quando o processo opera com
209 %                      os valores ótimos
210 %          chainlog     chain size and respective delta to mo, ro and Co
211 %          output      matriz que mostra o histórico da busca [m,c,r,s,phi,n]
212 %=====
213
214 % Declaração de variáveis globais
215 global pi p1 ci cnc ca cs_nc cs_c alpha beta alpha0 beta0
216 global Valfa Vbeta
217 global MODELPARAM ISIM
218 global F HIST CRIT INDEXHC NSTATESK
219 global CUSTO_HIST
220
221 tic;
222
223 %-----
224 % Limites de segurança / Configuração do Algoritmo
225 %-----
226 % nmax = tamanho máximo da cadeia
227 NMAX=1000;
228 % rmax = número máximo de classificações repetidas
229 RMAX=100;
230 % mmax = tamanho máximo do intervalo entre inspeções
231 MMAX=2e6;
232
233 % step_big = passo dado no tamanho n da cadeia quando, após 3 passos
234 % step_small sucessivos, não há convergência. Valor recomendado: 5
235 step_big=5;
236
237 % step_small = passo dado no tamanho n da cadeia em busca da convergência.
238 % Valor recomendado: 1
239 step_small=1;
240
241 % maxchain = ponto de partida para o tamanho da cadeia quando se utiliza
242 % a opção chainflag=1. Valor mínimo e recomendado: 2. Se a função de
243 % deterioração exigir altos valores do tamanho da cadeia, defina maxchain

```

f_optimum.m

```

144 % para um valor maior (como 50, por exemplo).
145 maxchain=2;
146
147
148 %-----
149 % Validando os parâmetros de entrada
150 %-----
151
152 if nargin < 2
153     error('esta função tem 2 parâmetros obrigatórios.');
```

154 end

```

155
156 if length(param)<13
157     error('param deve ser um vetor com 13 ou 14 elementos');
```

158 else

```

159     [pi,p1,ci,cnc,ca,cs_nc,cs_c,alpha0,beta0,deltamax,phimax,HIST,
CRIT,...
160     CUSTO_HIST]=getdados(param);
161 end
162
163 MODELPARAM=modelparam;
164
165 % Verificando como a busca pelo m foi definida
166 if nargin > 2
167     if length(mrange)==2;
168         temp=abs(round(mrange));
169         mmin=min(temp);
170         if max(temp)<MMAX;
171             mmax=max(temp);
172         else
173             mmax=MMAX;
174         end
175         if mmin<1
176             mmin=1;
177         end
178         if mmax<mmin
179             mmax=mmin;
180         end
181     else
182         mmin=1;
183         mmax=0;
184     end
185 else
186     mmin=1;
187     mmax=0;
188 end
189
190 % Verificando como a busca pelo r foi definida
```

f_optimum.m

```

191  if nargin > 3
192      if length(rrange)==2;
193          temp=abs(round(rrange));
194          rmin=min(temp);
195          if max(temp)<RMAX;
196              rmax=max(temp);
197          else
198              rmax=RMAX;
199          end
200          if rmin<1
201              rmin=1;
202          end
203          if rmax<rmin
204              rmax=rmin;
205          end
206      else
207          rmin=1;
208          rmax=0;
209      end
210  else
211      rmin=1;
212      rmax=0;
213  end
214
215  % Verificando como a busca pelo s foi definida
216  if nargin > 4
217      if length(srange)==2;
218          temp=abs(round(srange));
219          SMIN=min(temp);
220          if max(temp)<RMAX;
221              SMAX=max(temp);
222          else
223              SMAX=RMAX;
224          end
225          if SMIN<1
226              SMIN=1;
227          end
228          if SMAX<SMIN
229              SMAX=SMIN;
230          end
231      elseif length(srange)==1;
232          SMIN=-1;
233          SMAX=-1;
234      else
235          SMIN=0;
236          SMAX=0;
237      end
238  else

```

f_optimum.m

```

239     SMIN=0;
240     SMAX=0;
241 end
242
243 % Verificando como os flags foram definidos
244 if nargin > 5
245     if length(flags)<4
246         error('flags deve ser um vetor de pelo menos 4 elementos');
247     end
248     gflag = flags(1);
249     chainflag = flags(2);
250     if chainflag>NMAX
251         chainflag=NMAX;
252     end
253     sflag = flags(3);
254     sflagchainlog = flags(4);
255     if length(flags)==5
256         step_big=flags(5);
257     end
258 else
259     gflag = 0;
260     chainflag = 0;
261     sflag = 1;
262     sflagchainlog = 0;
263 end
264
265 % Verificando se foi fornecido o número da simulação atual
266 if nargin> 6
267     ISIM=isim;
268 else
269     ISIM=1;
270 end
271
272 % Fim da validação dos parâmetros de entrada
273 %-----
274
275
276 % Obtendo o número de stados válidos e as migrações
277 [INDEXHC,NSTATESK]=migracoes(HIST,CRIT);
278
279
280 %-----
281 % Pré-calculando alfa* e beta* para diferentes combinações de r e s.
282 % Caso os valores já tenham sido calculados e salvos em disco, eles serão
283 % carregados do arquivo texto
284 %-----
285
286 namefile=['data/Valfa_' num2str(RMAX) '_' num2str(alpha0,6) '.txt'];

```

f_optimum.m

```

287     if exist(namefile,'file')==2
288         % Se o arquivo com a tabela de migrações existe, então ele é
289         % carregado ao invés do cálculo ser realizado
290         Valfa=load(namefile);
291     else
292         Valfa=zeros(RMAX,RMAX);
293         for i=1:RMAX
294             vr=i*ones(1,i);
295             vs=1:i;
296             Valfa(i,vs)=binocdf(vs-1,vr,1-alpha0);
297         end
298         if not(exist('data','dir')==7)
299             mkdir('data');
300         end
301         save(namefile,'Valfa','-ascii');
302     end
303
304     namefile=['data/Vbeta_' num2str(RMAX) '_' num2str(beta0,6) '.txt'];
305     if exist(namefile,'file')==2
306         % Se o arquivo com a tabela de migrações existe, então ele é
307         % carregado ao invés do cálculo ser realizado
308         Vbeta=load(namefile);
309     else
310         Vbeta=zeros(RMAX,RMAX);
311         for i=1:RMAX
312             vr=i*ones(1,i);
313             vs=1:i;
314             Vbeta(i,vs)=1-binocdf(vs-1,vr,beta0);
315         end
316         if not(exist('data','dir')==7)
317             mkdir('data');
318         end
319         save(namefile,'Vbeta','-ascii');
320     end
321
322
323     % Pré-Cálculo da função fdet
324     FMAX=100000;
325     if mmax==0 || (mmax*NMAX) < FMAX
326         index=1:1:FMAX;
327         F=f_det(index);
328     else
329         index=1:1:max(mmax*5,FMAX);
330         F=f_det(index);
331     end
332
333
334     % Variáveis para entrada no loop de busca e inicialização de variáveis

```

f_optimum.m

```

335
336     OUTPUT=zeros(1000,5);    %Pre-allocando espaco para o armazenamento do OUTPUT
337
338     m=mmin-1;
339     contm=0;
340
341     Co=inf;
342     mo=0;
343     ro=0;
344     so=0;
345     phio=0;
346     no=0;
347
348     phinew=0;
349     mflag=0;
350
351     while phinew<phimax && (m < mmax || mmax==0) && (m<MMAX) && (mflag<3
        || sflag==0);
352
353         contm=contm+1;
354         m=m+1;
355
356         rflag=0;
357         r=rmin-1;
358
359         Cnew=inf;
360         rnew=0;
361         snew=0;
362
363         step_flag=0; %#ok<NASGU>
364
365         while (r<rmax || rmax==0) && r<RMAX && rflag<2;
366
367             r=r+1;
368
369             if SMIN==-1 && SMAX==-1
370                 smin=(floor(0.5*r)+1);
371                 smax=(floor(0.5*r)+1);
372             elseif SMIN==0 && SMAX==0
373                 smin=1;
374                 smax=r;
375             else
376                 smin=max(1,SMIN);
377                 smax=min(r,SMAX);
378             end
379
380             for s=smin:smax
381

```

f_optimum.m

```

382         step_flag=0;
383
384         if chainflag==0
385             n=2;
386         elseif chainflag==1
387             n=maxchain;
388         else
389             n=chainflag;
390         end
391
392         contn=1;
393
394         % Bloco try-catch para tentar fazer o cálculo com o tamanho da
395         % cadeia inicial (que pode ter sido obtido com o histórico da
396         % busca). Caso ocorra um erro (provavelmente devido a memória) a
397         % primeira providência é reduzir o tamanho da cadeia para 2 e
398         fazer
399
400         % a busca a partir deste valor
401         try
402             Cnold=Cost_mrn(m,r,s,n);
403         catch ME0
404             ME0.identifier
405             n=2;
406             Cnold=Cost_mrn(m,r,s,n);
407         end
408
409         delta=2*deltamax;
410
411         while (delta>deltamax) && (chainflag<2 || contn==1) && (n
412             <=NMAX)
413
414             % A busca do tamanho da cadeia é feita com 3 passos pequenos
415             % em n (step_small) seguido de 1 passo grande (step_big),
416             para
417
418             % acelerar o programa. Se deseja aumentar o tamanho da cadeia
419             % de 1 em 1, defina step_small=1 e step_big=1
420             if step_flag<3
421                 step_flag=step_flag+1;
422                 n=n+step_small;
423             else
424                 step_flag=0;
425                 n=n+step_big;
426             end
427
428             contn=contn+1;
429
430             % Bloco try-catch para tentar fazer o cálculo do modelo. Um
431             erro

```


f_optimum.m

```

426         % neste ponto indica que não será possível efetuar o cálculo
427         % com a precisão estabelecida (deltamax). O programa então
428         % aumenta o deltamax, redefine a cadeia para n=2 e reinicia
429         % novamente a busca com estes novos parâmetros
430         try
431             Cnnew=Cost_mrn(m,r,s,n);
432             delta=abs(Cnnew-Cnold)/Cnnew;
433             Cnold=Cnnew;
434         catch ME
435             ME.identifier
436             n=2;
437             step_flag=0;
438             contn=1;
439             deltamax=deltamax*2;
440             delta=2*deltamax;
441             Cnold=Inf;
442         end
443
444         status(mo,ro,so,Co,m,rnew,snew,Cnew,phnew,delta,n,
param(12),param(13));
445
446         end
447
448         if chainflag==1
449             maxchain=n-step_small;
450         end
451
452         if Cnnew < Cnew
453             Cnew=Cnnew;
454             rnew=r;
455             snew=s;
456         end
457
458         end
459
460         rflag=r-rnew;
461
462     end
463
464     % Calculando phnew com os parâmetros de controle obtidos
465     alpha=Valfa(rnew,snew);
466     if alpha<1e-8
467         alpha=1e-8;
468     end
469     beta=Vbeta(rnew,snew);
470     if beta<1e-8
471         beta=1e-8;
472     end

```

f_optimum.m

```

473     [P,y]=prob_model(m,n);
474     flagphi=1;
475     phinew=(cost_model(m,rnew,n,flagphi)*y)/m;
476
477     if phinew<phimax
478         if Cnew<Co
479             Co=Cnew;
480             ro=rnew;
481             so=snew;
482             phio=phinew;
483             mo=m;
484             no=n;
485         end
486     else
487         warning('O critério estatístico (phi_max) foi violado.')
488     end
489
490     mflag=m-mo;
491
492     if rem(contm,1)==0
493         status(mo,ro,so,Co,m,rnew,snew,Cnew,phinew,delta,n,param(12),
494 param(13))
495     end
496
497     OUTPUT(contm,1)=m;
498     OUTPUT(contm,2)=Cnew;
499     OUTPUT(contm,3)=rnew;
500     OUTPUT(contm,4)=phinew;
501     OUTPUT(contm,5)=n;
502     OUTPUT(contm,6)=deltamax;
503
504 end
505
506 OUTPUT=OUTPUT(1:contm,:);
507 output=OUTPUT;
508
509 % Calcula a evolução da cadeia passo a passo se assim foi especificado pelo
510 % usuário
511 if sflagchainlog==1
512     if no>0
513         CHAINLOG=zeros(NMAX-n+1,4);
514         n=2;
515         contn=1;
516         Cnold=Cost_mrn(mo,ro,so,n);
517         CHAINLOG(contn,1)=n;
518         CHAINLOG(contn,2)=Cnold;
519

```

f_optimum.m

```

520         for i=3:no
521             n=i;
522             contn=contn+1;
523             Cnnew=Cost_mrn(mo,ro,so,n);
524             delta=abs(Cnnew-Cnold)/Cnnew;
525             Cnold=Cnnew;
526
527             CHAINLOG(contn,1)=n;
528             CHAINLOG(contn,2)=Cnold;
529             CHAINLOG(contn,3)=delta;
530
531             if delta<deltamax
532                 CHAINLOG(contn,4)=1;
533             end
534         end
535         chainlog=CHAINLOG;
536     else
537         chainlog=[];
538     end
539 else
540     chainlog=[];
541 end
542
543 status(mo,ro,so,Co,m,rnew,snew,Cnew,phineu,delta,n,param(12),param(13
544 ))
545
546 if gflag==1
547     % Faz o gráfico de custo, r, phi e n em função de m.
548     figure;
549     set(gcf,'Color',[1,1,1]);
550
551     subplot(2,2,1);
552     plot(OUTPUT(:,1),OUTPUT(:,2),'b-',mo,Co,'ro');
553     xlabel('m');ylabel('Cost');
554     title(['C^o=' num2str(Co,6) ' ; r^o=' num2str(ro,6) ...
555           ' ; \phi^o=' num2str(phio,6) ' ; n^o=' num2str(no,6) ' ']);
556
557     subplot(2,2,2);
558     plot(OUTPUT(:,1),OUTPUT(:,3),'b-',mo,ro,'ro');
559     xlabel('m');ylabel('r');
560     title('r versus m');
561
562     subplot(2,2,3);
563     plot(OUTPUT(:,1),OUTPUT(:,4),'b-',mo,phio,'ro');
564     xlabel('m');ylabel('\phi');
565     title('\phi versus m');
566
567     subplot(2,2,4);

```

f_optimum.m

```

567     plot(OUTPUT(:,1),OUTPUT(:,5),'b-',mo,no,'ro');
568     xlabel('m');ylabel('n');
569     title('n versus m');
570 end
571
572 function C=Cost_mrn(m,r,s,n)
573 %=====
574 % Função auxiliar - Calcula o custo para dados m, r e n específicos
575 %=====
576 global alpha beta
577 global Valfa Vbeta
578
579 flagphi=0;
580 alpha=Valfa(r,s);
581 beta=Vbeta(r,s);
582
583 if alpha<1e-8;
584     alpha=1e-8;
585 end
586 if beta<1e-8;
587     beta=1e-8;
588 end
589
590 [P,y]=prob_model(m,n);
591 V=cost_model(m,r,n,flagphi);
592 C=(V*y)/m;
593
594
595 function [a,b]=alfabetaef(r,s,alpha0,beta0)
596 %=====
597 % Função auxiliar - Calculata alpha* e beta* (após r classificações)
598 %=====
599 a=binocdf(s-1,r,1-alpha0);
600 b=1-binocdf(s-1,r,beta0);
601
602
603 function [pi,pl,ci,cnc,ca,cs_nc,cs_c,alpha0,beta0,deltamax,phimax,
hist,...
604     crit,custo_hist]=getdados(mdados)
605 %=====
606 % Função auxiliar - Lendo os dados do vetor de parâmetros mdados
607 %=====
608 pi=mdados(1);           %parâmetro da distribuição geométrica relacionada
609                          %à mudança do estado I para o estado II
610 pl=mdados(2);           %fração de itens defeituosos no estado I
611 ci=mdados(3);           %custo de inspeção
612 cnc=mdados(4);          %custo de um item não conforme
613 ca=mdados(5);           %custo de ajuste

```

f_optimum.m

```

614   cs_nc=mdados(6);           %custo para descartar um item não conforme
615   cs_c=mdados(7);           %custo para descartar um item conforme
616   alpha0=mdados(8);         %probabilidade de classificar um conforme como
617                               %não-conforme
618   beta0=mdados(9);          %probabilidade de classificar não-conforme como
619                               %conforme
620   deltamax=mdados(10);       %diferença máxima entre o custo de duas matrizes de
621                               %transição subsequentes
622   phimax=mdados(11);         %fração máxima de itens não-conformes que podem ser
623                               %produzidos
624   hist=mdados(12);          %tamanho do histórico considerado
625   crit=mdados(13);          %número de itens não-conformes no histórico para que
626                               %ocorra ajuste
627   if length(mdados)==13
628       custo_hist=0;
629   else
630       %custo de gerenciar cada item adicional no histórico
631       custo_hist=mdados(14);
632   end
633
634
635   function [P,y]=prob_model(m,nw)
636   %=====
637   % Função auxiliar - Calcula a matriz de transição P e
638   %                     o vetor estacionário y
639   %
640   %     ENTRADA:
641   %     m   intervalo entre inspeções
642   %     nw  tamanho da cadeia
643   %=====
644   global p1 alpha beta pi
645   global F INDEXHC NSTATESK
646
647   % Declara e aloca o espaço máximo da matriz esparsa P
648   P=spalloc((nw+1)*(NSTATESK),(nw+1)*(NSTATESK),NSTATESK*(nw+1)*4);
649
650   % -----
651   % Probabilidades que partem do estado w0 e chegam em w0
652   % (com item inspecionado no estado destino julgado conforme)
653   % -----
654   prob=(1-pi)^m*((1-p1)*(1-alpha)+p1*beta);
655   for l=1:NSTATESK
656       i1=INDEXHC(l,1);
657       i2=INDEXHC(l,2);
658       P(i1,i2)=prob;
659   end
660
661   % -----

```

f_optimum.m

```

662 % Probabilidades que partem do estado w0 e chegam em w0
663 % (com item inspecionado no estado destino julgado não-conforme)
664 % -----
665 prob=(1-pi)^m*((1-p1)*alpha+p1*(1-beta));
666 for l=1:NSTATESK
667     i1=INDEXHC(l,1);
668     i2=INDEXHC(l,3);
669     P(i1,i2)=prob;
670 end
671
672 % -----
673 % Probabilidades que partem do estado w0 e chegam em w1
674 % (com item inspecionado no estado destino julgado conforme)
675 % -----
676 vi=1:1:m;
677 temp=((1-pi).^(vi-1))*pi;
678 f=F(m-vi+1);
679
680 prob=sum(temp.*((1-f)*(1-alpha)+f*beta));
681 for l=1:NSTATESK
682     i1=INDEXHC(l,1);
683     i2=INDEXHC(l,2)+NSTATESK;
684     if INDEXHC(l,5)==0
685         P(i1,i2)=prob;
686     end
687 end
688
689 % -----
690 % Probabilidades que partem do estado w0 e chegam em w1
691 % (com item inspecionado no estado destino julgado não-conforme)
692 % -----
693 prob=sum(temp.*((1-f)*alpha+f*(1-beta)));
694 for l=1:NSTATESK
695     i1=INDEXHC(l,1);
696     i2=INDEXHC(l,3)+NSTATESK;
697     if INDEXHC(l,5)==0
698         P(i1,i2)=prob;
699     else
700         % trata-se de um estado ajustado
701         P(i1,:)=P(1,:);
702     end
703 end
704
705
706 % -----
707 % Probabilidades que partem dos estados w1...w_{n-1}
708 % -----
709 temp=temp/(1-(1-pi)^m);

```

f_optimum.m

```

710     temp=temp/sum(temp);
711
712     for j=2:nw
713         f=F(j*m-vi+1);
714
715         % item do estado destino julgado conforme
716         prob=sum(temp.*((1-f)*(1-alpha)+f*beta));
717         for l=1:NSTATESK
718             i1=INDEXHC(1,1)+(j-1)*NSTATESK;
719             i2=INDEXHC(1,2)+j*NSTATESK;
720             if INDEXHC(1,5)==0
721                 P(i1,i2)=prob;
722             end
723         end
724
725         % item do estado destino julgado não-conforme
726         prob=sum(temp.*((1-f)*alpha+f*(1-beta)));
727         for l=1:NSTATESK
728             i1=INDEXHC(1,1)+(j-1)*NSTATESK;
729             i2=INDEXHC(1,3)+j*NSTATESK;
730             if INDEXHC(1,5)==0
731                 P(i1,i2)=prob;
732             else
733                 % trata-se de um estado ajustado
734                 P(i1,:)=P(1,:);
735             end
736         end
737     end
738
739
740     % -----
741     % Truncando a cadeia em w_n
742     % -----
743     for j=nw
744
745         f=F(j*m-vi+1);
746
747         % item do estado destino julgado conforme
748         prob=sum(temp.*((1-f)*(1-alpha)+f*beta));
749         for l=1:NSTATESK
750             i1=INDEXHC(1,1)+j*NSTATESK;
751             i2=INDEXHC(1,2)+j*NSTATESK;
752             if INDEXHC(1,5)==0
753                 P(i1,i2)=prob;
754             end
755         end
756
757         % item do estado destino julgado não-conforme

```

f_optimum.m

```

758     prob=sum(temp.*((1-f)*alpha+f*(1-beta)));
759     for l=1:NSTATESK
760         i1=INDEXHC(1,1)+j*NSTATESK;
761         i2=INDEXHC(1,3)+j*NSTATESK;
762         if INDEXHC(1,5)==0
763             P(i1,i2)=prob;
764         else
765             P(i1,:)=P(1,:);
766         end
767     end
768
769 end
770
771 tam=(nw+1)*(NSTATESK);
772 Identidade=sparse(1:tam,1:tam,1);
773 A=transpose(P)-Identidade;
774 A(tam,1:tam)=1;
775 A=sparse(A);
776
777 clear Identidade;           %liberando memória
778
779 B=sparse(tam,1);
780 B(tam,1)=1;
781
782 y=mldivide(A,B);           %resolve o sistema linear
783
784
785 function V=cost_model(m,r,nw,flagphi)
786 %=====
787 % Função auxiliar - Calcula a função Custo
788 %=====
789 global pi p1 ci cnc ca cs_nc cs_c alpha beta
790 global F INDEXHC NSTATESK
791 global CUSTO_HIST HIST
792
793 % flagphi=1 indica que deseja-se calcular a fração de itens não-conformes.
794 % Com isso, todos os custos associados à itens não-conformes são definidos
795 % como 1, e todos os demais são definidos como 0. No final, os valores são
796 % restaurados, já que se tratam de valores globais (também utilizados em
797 % outros pontos do código)
798 if flagphi==1
799     oci=ci;
800     ocnc=cnc;
801     oca=ca;
802     ocs_nc=cs_nc;
803     ocs_c=cs_c;
804
805     ci=0;

```


f_optimum.m

```

806     ca=0;
807     cnc=1;
808     cs_c=0;
809     cs_nc=1;
810 end
811
812 % Inicialização (alocando memória)
813 theta=zeros(1,(nw+1)*(NSTATESK));
814 epsilon=zeros(1,(nw+1)*(NSTATESK));
815 phi=zeros(1,(nw+1)*(NSTATESK));
816
817 % -----
818 % estados w0
819 % -----
820 for l=1:NSTATESK
821     ULTINSP=INDEXHC(1,6);
822     ii=1;
823     if ULTINSP==0
824         % item recém-inspacionado declarado conforme
825         epsilon(1,ii)=cnc*(m-1)*p1;
826         theta(1,ii)=cnc*p1*beta/((1-p1)*(1-alpha)+p1*beta);
827         phi(1,ii)=0;
828     else
829         % item recém-inspacionado declarado não-conforme
830         epsilon(1,ii)=cnc*(m-1)*p1;
831         theta(1,ii)=cs_nc*p1*(1-beta)/((1-p1)*alpha+p1*(1-beta))+...
832             cs_c*(1-p1)*alpha/((1-p1)*alpha+p1*(1-beta));
833         if INDEXHC(1,5)==0
834             phi(1,ii)=0;
835         else
836             % ocorre ajuste do processo
837             phi(1,ii)=ca;
838         end
839     end
840 end
841
842 % -----
843 % estados w1
844 % -----
845
846 index=1:1:m;
847 temp1=(pi*(1-pi).^(index-1))/(1-(1-pi)^m);
848 temp2=(index-1)*p1;
849
850 f=F(index);
851 temp4=temp2*0;
852 for i=1:1
853     if i<m

```

f_optimum.m

```

854         temp3=1:(m-i);
855         temp4(i)=sum(f(temp3));
856         temp2(i)=temp2(i)+temp4(i);
857     end
858 end
859
860 for i=2:m
861     if i<m
862         temp4(i)=temp4(i-1)-f(m-i+1);
863         temp2(i)=temp2(i)+temp4(i);
864     end
865 end
866
867 for l=1:NSTATESK
868
869     ULTINSP=INDEXHC(1,6);
870
871     ii=NSTATESK+1;
872
873     if ULTINSP==0
874         % item recém-inspacionado declarado conforme
875         epsilon(1,ii)=cnc*sum(temp1.*temp2);
876
877         f=F(m-index+1);
878         temp3=f*beta./((1-f)*(1-alpha)+f*beta);
879         theta(1,ii)=cnc*sum(temp1.*temp3);
880
881         phi(1,ii)=0;
882
883     else
884         % item recém-inspacionado declarado não-conforme
885         epsilon(1,ii)=cnc*sum(temp1.*temp2);
886
887         f=F(m-index+1);
888
889         temp3=f*(1-beta)./((1-f)*alpha+f*(1-beta));
890         temp4=(1-f)*alpha./((1-f)*alpha+f*(1-beta));
891         theta(1,ii)=cs_nc*sum(temp1.*temp3)+cs_c*sum(temp1.*temp4);
892
893         if INDEXHC(1,5)==0
894             phi(1,ii)=0;
895         else
896             % ocorre ajuste do processo
897             phi(1,ii)=ca;
898         end
899     end
900 end
901

```

f_optimum.m

```

902     end
903
904     % -----
905     % estados w2 até wn
906     % -----
907     for j=2:(nw)
908
909         temp2=zeros(1,m);
910         temp3=1:(m-1);
911         for k=1:1
912             if m>1
913                 temp4=m*(j-1)-k+temp3+1;
914                 f=F(temp4);
915                 temp2(k)=sum(f);
916             end
917         end
918
919         for k=2:m
920             if m>1
921                 i1=m*(j-1)-(k-1)+(m-1)+1;
922                 i2=m*(j-1)-(k)+(1)+1;
923                 temp2(k)=temp2(k-1)-F(i1)+F(i2);
924             end
925         end
926
927         for l=1:NSTATESK
928
929             ULTINSP=INDEXHC(l,6);
930
931             ii=j*NSTATESK+l;
932
933             if ULTINSP==0
934                 % item recém-inspacionado declarado conforme
935                 epsilon(1,ii)=cnc*sum(temp1.*temp2);
936
937                 f=F(j*m-index+1);
938                 temp3=f*beta./((1-f)*(1-alpha)+f*beta);
939                 theta(1,ii)=cnc*sum(temp1.*temp3);
940
941                 phi(1,ii)=0;
942
943             else
944                 % item recém-inspacionado declarado não-conforme
945                 epsilon(1,ii)=cnc*sum(temp1.*temp2);
946
947                 f=F(j*m-index+1);
948                 temp3=f*(1-beta)./((1-f)*alpha+f*(1-beta));
949                 temp4=(1-f)*alpha./((1-f)*alpha+f*(1-beta));

```

f_optimum.m

```

950
951         theta(1,ii)=cs_nc*sum(temp1.*temp3)+cs_c*sum(temp1.*temp4
952     );
953
954         if INDEXHC(1,5)==0
955             phi(1,ii)=0;
956         else
957             % ocorre ajuste do processo
958             phi(1,ii)=ca;
959         end
960     end
961
962 end
963
964 end
965
966 %Vetor custo completo (incluindo a inspeção e, se for o caso, o custo
967 %linear devido ao uso do histórico)
968
969 V=(epsilon+theta+phi)+r*ci+(CUSTO_HIST*(HIST-1))*(1-flagphi);
970 if flagphi==1
971     % Restaura os valores dos custos
972     ci=oci;
973     ca=oca;
974     cnc=ocnc;
975     cs_c=ocs_c;
976     cs_nc=ocs_nc;
977 end
978
979 function status(mo,ro,so,Co,m,rnew,snew,Cnew,phinew,delta,n,h,nu)
980 %=====
981 % Função Auxiliar - Mostra a evolução da busca
982 %=====
983 global ISIM
984 clc;
985 fprintf('%60s\n',
986     '=====');
987 fprintf('%60s\n','...: Search Log
988     :...');
989 fprintf('%15s\t %13.6f\n','simulac.= ',ISIM);
990 fprintf('%60s\n','');
991 fprintf('%15s\t %13.0f\n','m      = ',m);
992 fprintf('%15s\t %13.0f\n','r_new  = ',rnew);
993 fprintf('%15s\t %13.0f\n','s_new  = ',snew);
994 fprintf('%15s\t %13.6f\n','C      = ',Cnew);
995 fprintf('%15s\t %13.6f\n','phi    = ',phinew);
996 fprintf('%15s\t %13.6f\n','delta  = ',delta);

```

f_optimum.m

```

995 fprintf('%15s\t %13.0f\n','n      = ',n);
996 fprintf('%60s\n','');
997 fprintf('%15s\t %13.0f\n','h      = ',h);
998 fprintf('%15s\t %13.0f\n','nu     = ',nu);
999 fprintf('%60s\n','');
1000 fprintf('%15s\t %13.6f\n','m_o   = ',mo);
1001 fprintf('%15s\t %13.6f\n','r_o   = ',ro);
1002 fprintf('%15s\t %13.6f\n','s_o   = ',so);
1003 fprintf('%15s\t %13.6f\n','C_o   = ',Co);
1004 fprintf('%60s\n',
    '=====');

1005
1006
1007 function [newindex2,nstates]=migracoes(t,c)
1008 %=====
1009 % Função Auxiliar - Determina as migrações entre os estados em sua
1010 % representação decimal
1011 %
1012 % ENTRADA:
1013 % t=tamanho da história considerada
1014 % c=número de itens não-conformes para parar o processo e executar o ajuste
1015 %
1016 % SAIDA:
1017 % newindex2 = matriz com os índices de migração (6 colunas)
1018 % primeira coluna = z do estado de origem
1019 % segunda coluna = z destino se for julgado conforme
1020 % terceira coluna = z destino se for julgado não-conforme
1021 % quarta coluna = se o próximo item inspecionado for julgado
1022 %                não-conforme, haverá ajuste? 1 = sim, 0 = não; esta
1023 %                informação não está sendo utilizada atualmente
1024 %
1025 % quinta coluna = o z deste estado corresponde a um estado que sofre
1026 %                ajuste? 1 = sim; 0 = não
1027 % sexta coluna = diz se o último item inspecionado no z de origem é:
1028 %                1 = defeituoso, 0 = conforme
1029 %
1030 % Esta função gera os estados viáveis e as migrações entre estes estados.
1031 % Como fixados t e c, os estados viáveis e migrações não mudam, esta
1032 % rotina salva a tabela de migrações em um arquivo texto para que, numa
1033 % próxima simulação, ela possa ser lida diretamente do disco rígido,
1034 % poupando tempo de processamento. O nome do arquivo será mig_t_c.txt,
1035 % onde t e c são os valores fornecidos à função.
1036 %
1037 %=====
1038
1039 namefile=['data/mig_' num2str(t) '_' num2str(c) '.txt'];
1040
1041 if exist(namefile,'file')==2

```

f_optimum.m

```

1042     % Se o arquivo com a tabela de migrações existe, então ele é
1043     % carregado ao invés do cálculo ser realizado
1044     newindex2=load(namefile);
1045     nstates=size(newindex2,1);
1046
1047     else
1048
1049         if c==t
1050             % Se a sequência de itens não-conformes coincide com o tamanho do
1051             % histórico considerado, então o modelo implementado é aquele do
1052             % Critério I, pois há apenas (t+1) possibilidades para z
1053
1054             newindex2=zeros(t+1,6);
1055
1056             % Primeiro estado (item conforme)
1057             newindex2(1,1)=1;
1058             newindex2(1,2)=1;
1059             newindex2(1,3)=2;
1060             newindex2(1,4)=(t==1);
1061             newindex2(1,5)=0;
1062             newindex2(1,6)=0;
1063
1064             % Estados intermediários (item não-conforme, sem ajuste)
1065             for i=2:t
1066                 newindex2(i,1)=i;
1067                 newindex2(i,2)=1;
1068                 newindex2(i,3)=i+1;
1069                 newindex2(i,4)=((i+1)==(t+1));
1070                 newindex2(i,5)=0;
1071                 newindex2(i,6)=1;
1072             end
1073
1074             % Último estado (item não conforme, há ajuste)
1075             newindex2(t+1,1)=t+1;
1076             newindex2(t+1,2)=1;
1077             newindex2(t+1,3)=2;
1078             newindex2(t+1,4)=(t==1);
1079             newindex2(t+1,5)=1;
1080             newindex2(t+1,6)=1;
1081
1082             nstates=t+1;
1083
1084         else
1085
1086             n=2^t;
1087             passo=ceil(n/100);
1088             index=zeros(n,6);
1089             contnulo=0;

```

f_optimum.m

```

1090
1091     seq_ones=dec2bin(n-1);
1092
1093     for i=1:n
1094
1095         s=dec2bin(i-1,t);
1096
1097         %calculando a soma dos dígitos da representação binária do
1098         %inteiro
1099         temp=sum(s==seq_ones);
1100
1101         if temp>c
1102             % Estado Inviável
1103             contnulo=contnulo+1;
1104             index(i,1)=0;
1105         elseif temp==c
1106             if str2double(s(t))==0
1107                 % Estado Inviável
1108                 contnulo=contnulo+1;
1109                 index(i,1)=0;
1110             else
1111                 % Estado viável e há ajuste
1112                 index(i,1)=i;
1113                 index(i,5)=1;
1114
1115                 index(i,6)=str2double(s(t));
1116
1117                 s=dec2bin(0,t); % o estado é ajustado (parte do 0)
1118
1119                 s0=[s(2:t) '0']; % próximo estado caso conforme
1120                 s1=[s(2:t) '1']; % próximo estado caso não conforme
1121
1122                 i0=bin2dec(s0)+1;
1123                 i1=bin2dec(s1)+1;
1124
1125                 index(i,2)=i0;
1126                 index(i,3)=i1;
1127
1128                 temp=sum(s1==seq_ones);
1129
1130                 if temp==c
1131                     index(i,4)=1;
1132                 end
1133             end
1134         else
1135             % Estado viável e não há ajuste
1136             index(i,1)=i;
1137             index(i,6)=str2double(s(t));

```

f_optimum.m

```

1138
1139         s0=[s(2:t) '0'];
1140         s1=[s(2:t) '1'];
1141
1142         i0=bin2dec(s0)+1;
1143         i1=bin2dec(s1)+1;
1144         index(i,2)=i0;
1145         index(i,3)=i1;
1146
1147         temp=sum(s1==seq_ones);
1148
1149         if temp==c
1150             index(i,4)=1;
1151         end
1152
1153     end
1154
1155     if rem(i,passo)==0 || i==1
1156         status_mig(i,n,t,c)
1157     end
1158
1159 end
1160
1161 status_mig(n,n,t,c);
1162 newindex2=zeros(n-contrnulo,6);
1163
1164 % Constrói a matriz de transições apenas com os estados viáveis
1165 i0=0;
1166 for i=1:n
1167     if index(i,1)~=0
1168         i0=i0+1;
1169         newindex2(i0,:)=index(i,:);
1170     end
1171     if rem(i,passo)==0 || i==1
1172         status_mig(i,n,t,c)
1173     end
1174 end
1175 status_mig(n,n,t,c);
1176
1177 [nstates,temp]=size(newindex2);
1178
1179 % Faz a reindexação dos estados viáveis, para que eles variem entre
1180 % 1 e o número de estados viáveis
1181 for i=1:nstates
1182     for j=2:3
1183         ind=find(newindex2(:,1)==newindex2(i,j));
1184         newindex2(i,j)=ind;
1185     end

```


f_optimum.m

```

1186         end
1187         for i=1:nstates
1188             newindex2(i,1)=i;
1189         end
1190
1191     end
1192
1193     % Salva a matriz de transições para uso futuro
1194     try
1195         if not(exist('data','dir')==7)
1196             mkdir('data');
1197         end
1198         fid = fopen(namefile, 'wt');
1199         fprintf(fid, '%d %d %d %d %d %d\n', newindex2');
1200         fclose(fid);
1201     catch
1202         warning('Não foi possível salvar o arquivo com as migrações.'
1203 );
1204         warning('Verifique se há espaço em disco.');
```

1205 warning(['Verifique se o arquivo ' namefile ' está sendo
utilizado.']);

```

1206     end
1207 end
1208
1209
1210 function status_mig(iatual,itotal,tt,cc)
1211 %=====
1212 % Função Auxiliar - Display log
1213 %=====
1214 clc;
1215 fprintf('%60s\n',
1216 '=====');
1217 fprintf('%60s\n', '...::          Determinando as Migrações
1218 :...');
1219 fprintf('%60s\n', '');
1220 fprintf('%15s\t %13.0f\n', 'h          = ',tt);
1221 fprintf('%15s\t %13.0f\n', 'nu         = ',cc);
1222 fprintf('%60s\n', '');
1223 fprintf('%15s\t %13.0f\n', 'iatual    = ',iatual);
1224 fprintf('%15s\t %13.0f\n', 'itotal    = ',itotal);
1225 fprintf('%60s\n',
1226 '=====');
```

f_det.m

```

1  function y=fdet(z)
2  %=====
3  % Função auxiliar - Função de deterioração
4  %
5  % Defina a função de deterioração do sistema no bloco adequado. Implemente
6  % uma função que seja capaz de manipular o vetor z (a saída da função deve
7  % ser da mesma dimensão de z)
8  %=====
9  global MODELPARAM
10 y=0;
11
12 if MODELPARAM(1)==1
13     % modelo exponential
14     y=1-0.9*exp(-0.001*(z-1));
15     return;
16 elseif MODELPARAM(1)==2
17     % modelo linear
18     y=(0.1+0.9*(z-1)/400);
19     indice=find(y>1);
20     y(indice)=1;
21     return;
22 elseif MODELPARAM(1)==3
23     % modelo constante
24     y=(z*0)+0.6;
25     return;
26 elseif MODELPARAM(1)==4
27     % modelo definido pelo usuário
28     y=(z*0)+0.2;
29     return;
30 end
31

```

p_exemplo1.m

```

1  %=====
2  %
3  % Exemplo 1: Obtenção dos parâmetros ótimos do Capítulo 3
4  %
5  %=====
6  clc;
7  clear all;
8
9  %-----
10 % Parâmetros de entrada
11 %-----
12
13 param(1)=0.0001;    %pi;
14 param(2)=0.01;      %p1;
15 param(3)=0.25;      %ci;
16 param(4)=20;         %cnc;
17 param(5)=100;        %ca;
18 param(6)=2.0;        %cs_nc;
19 param(7)=2.5;        %cs_c;
20 param(8)=0.01;       %alpha;
21 param(9)=0.01;       %beta;
22 param(10)=1e-6;      %deltamax;
23 param(11)=1;         %phimax
24 param(12)=1;         %h = tamanho da sequência observada;
25 param(13)=1;         %nu = número de não-conformes para parar o processo;
26 param(14)=0;         %custo por item adicional no histórico;
27
28 %-----
29 % Parâmetros da simulação
30 %-----
31 MODELO=3;    %1=Exponencial, 2=Linear, 3=Constante (Cap. 3), 4=Def. pelo
    usuário
32
33 %-----
34 % Início da simulação
35 %-----
36 [mo,ro,so,Co,phio,chainlog,output]=f_optimum(param,MODELO,[],[],[0],[
    0 0 1 1],1);

```

p_exemplo2.m

```

1  %=====
2  %
3  % Exemplo 2: Obtenção dos parâmetros ótimos do Capítulo 4, com h=6, nu=3 e
4  % o modelo Linear. Para variar h e nu é necessário criar um loop com
5  % estes parâmetros.
6  %
7  %=====
8  clc;
9  clear all;
10
11 %-----
12 % Parâmetros de entrada
13 %-----
14
15 param(1)=0.0001;    %pi;
16 param(2)=0.01;      %pl;
17 param(3)=0.25;      %ci;
18 param(4)=20;        %cnc;
19 param(5)=100;       %ca;
20 param(6)=2.0;       %cs_nc;
21 param(7)=2.5;       %cs_c;
22 param(8)=0.01;      %alpha;
23 param(9)=0.01;      %beta;
24 param(10)=1e-4;     %deltamax;
25 param(11)=0.03;     %phimax
26 param(12)=6;        %h = tamanho da sequência observada;
27 param(13)=3;        %nu = número de não-conformes para parar o processo;
28 param(14)=0;        %custo por item adicional no histórico;
29
30 %-----
31 % Parâmetros da simulação
32 %-----
33 MODELO=3;    %1=Exponencial, 2=Linear, 3=Constante (Cap. 3), 4=Def. pelo
usuário
34
35 %-----
36 % Início da simulação
37 %-----
38 [mo,ro,so,Co,phio,chainlog,output]=f_optimum(param,MODELO,[],[],[0],[
0 0 1 1],1);

```

APÊNDICE B

Artigo publicado na revista
Pesquisa Operacional

CONTROLE *ON-LINE* POR ATRIBUTOS COM ERROS DE CLASSIFICAÇÃO: UMA ABORDAGEM ECONÔMICA COM CLASSIFICAÇÕES REPETIDAS

Anderson Laécio Galindo Trindade

Linda Lee Ho *

Departamento de Engenharia de Produção / EP
Universidade de São Paulo (USP)

São Paulo – SP

lindalee@usp.br

Roberto da Costa Quinino

Departamento de Estatística / ICEx

Universidade Federal de Minas Gerais (UFMG)

Belo Horizonte – MG

* *Corresponding author* / autor para quem as correspondências devem ser encaminhadas

Recebido em 03/2006; aceito em 02/2007

Received March 2006; accepted February 2007

Resumo

O procedimento de controle *on-line* de processos por atributos, proposto por Taguchi *et al.* (1989), consiste em amostrar um item a cada m produzidos e decidir, a cada inspeção, se houve ou não a redução da fração de itens conformes produzidos. Caso o item inspecionado for não conforme, pára-se o processo para ajuste. Como o sistema de inspeção pode estar sujeito a erros de classificação, desenvolve-se um modelo probabilístico que considera classificações repetidas e independentes do item amostrado em um sistema de controle com inspeção imperfeita. Utilizando-se as propriedades de uma cadeia de Markov ergódica, obtém-se uma expressão do custo médio do sistema de controle, que pode ser minimizada por três parâmetros: o intervalo entre inspeções; o número de classificações repetidas; e o número mínimo de classificações conformes (dentre as classificações repetidas), para julgar um item como conforme. Um exemplo numérico ilustra o procedimento proposto.

Palavras-chave: controle *on-line* de qualidade por atributos; erros de classificação; cadeias de Markov.

Abstract

The procedure to control on-line processes for attributes proposed by Taguchi *et al.* (1989) consists of sampling a single item at every m produced ones. If the examined item is non-conforming, the process is said to be out of control (the fraction of conforming item is reduced) and it is stopped for adjustment. However, the inspection procedure is subject to misclassification errors. So a probabilistic model was developed considering results of the repetitive and independent classifications of the examined item. Employing properties of an ergodic Markov chain, an expression of the expected cost function for the control system was obtained to be minimized by three parameters: sampling interval; number of the repetitive classifications; minor number of conforming classifications (among the repetitive classifications) to declare an item as conforming. A numerical example illustrates the proposed model.

Keywords: on-line process control for attributes; misclassification errors; Markov chain.

1. Introdução

O sistema de controle *on-line* de processos por atributo, apresentado por Taguchi *et al.* (1989), considera um processo que inicia sua operação produzindo itens com uma fração conforme $p_1 = 1$ (estado I). Após a ocorrência de uma causa especial, a fração de itens conformes passa a um valor p_2 ($0 \leq p_2 < p_1$) e o processo permanece produzindo nesta condição (estado II) até que esta mudança seja detectada e a causa especial removida. O sistema de controle consiste na inspeção periódica de um item a cada m produzidos e, uma vez que este item seja julgado como não conforme, admite-se que ocorreu uma mudança da fração de itens conformes e o processo é parado para ajuste. Após o ajuste, a fração de itens conformes retorna ao valor inicial $p_1 = 1$. Exemplos em que a utilização desta metodologia é bem sucedida incluem processo automático de solda, produção de semicondutores, produção de diodos utilizados em placas de circuito impressos e em processos químicos. De maneira geral, sistemas de produção que utilizam controle automático podem beneficiar-se da metodologia aqui discutida, como apresentado por Taguchi *et al.* (2004).

Taguchi *et al.* (1989) obtém expressões analíticas para o intervalo entre inspeções (m) que minimizam o custo médio do sistema de controle, porém não assumem um mecanismo explícito para a ocorrência da causa especial e tais expressões resultam de uma série de considerações e simplificações.

Nayebpour & Woodall (1993) desenvolvem um modelo de controle para atributos incluindo um mecanismo de falha para o processo (através de uma distribuição geométrica), o que representa uma maneira formal do processo mudar do estado I para o estado II. Com isso, afirmam que não pode ser obtida uma expressão analítica para intervalo ótimo entre inspeções e a busca deste parâmetro exige o uso de métodos computacionais.

No modelo de controle para atributos apresentado por Borges *et al.* (2001) admite-se que o sistema de inspeção está sujeito a erros de classificação. Pode-se julgar um item conforme dado que ele não o é (erro tipo II) ou julgar um item conforme como não conforme (erro tipo I). Suas conclusões indicam que o custo do sistema de controle é sensível à presença dos erros de classificação. Porém, nos três artigos mencionados não é considerada a hipótese de classificações repetidas do item inspecionado, o que pode reduzir o impacto dos erros de classificação e, conseqüentemente, o custo médio do sistema de controle por item produzido.

O uso de classificações repetidas em um sistema de inspeção com erros de classificação não é um procedimento novo. Alguns autores já propuseram o seu uso, empregando critérios diferenciados na classificação final do item inspecionado (o item examinado é classificado repetidamente e independentemente r vezes e em cada classificação, ele é avaliado em conforme ou não conforme). Em Greenberg & Stokes (1995) este procedimento é proposto, mas admite-se apenas a existência do erro do tipo I na determinação do número ótimo de classificações repetidas. Em Quinino & Suyama (2002), r classificações repetidas são realizadas no item inspecionado, que é declarado conforme se pelo menos forem observadas a classificações conformes. Os parâmetros a e r são determinados segundo uma abordagem econômica. Em Quinino & Ho (2004), os itens são examinados repetidamente até observar a classificações conformes ou b classificações não conformes (no primeiro caso, o item é declarado conforme e no segundo critério, o item é declarado não conforme). Para outros resultados, o procedimento de classificações repetidas continua. O objetivo foi determinar os parâmetros a e b ótimos também segundo uma abordagem econômica. Porém, tais trabalhos não consideram a mudança da fração de itens conformes durante a produção.

Este trabalho tem por objetivo desenvolver um modelo probabilístico que considera as classificações repetidas e independentes do item inspecionado através de cadeias de Markov de estados discretos para determinar uma estratégia ótima de controle (*on-line* por atributos) em um processo cujo sistema de inspeção está sujeito a erros de classificação. A estratégia ótima consiste na determinação do intervalo entre inspeções (m), do número de classificações repetidas do item inspecionado (r) e do número mínimo de classificações conformes (dentre as r) para julgar o item conforme (w) que minimizam o custo médio do sistema de controle. Este artigo está estruturado do seguinte modo: a Seção 2 apresenta o modelo probabilístico proposto. O custo médio do sistema de controle é obtido na Seção 3. Na Seção 4, um exemplo numérico ilustra o efeito da presença dos erros de classificação e das classificações repetidas sobre o custo médio do sistema de controle. As conclusões e sugestões são apresentadas na Seção 5.

2. Modelo Probabilístico

Considere que itens são produzidos em um processo que, no estado I, possui uma fração de itens conformes p_1 e, no estado II, a fração de itens conformes diminui para p_2 , com $0 \leq p_2 < p_1 \leq 1$. A probabilidade de mudança do estado I para o estado II, a cada item produzido, é dada por uma distribuição geométrica com parâmetro π , sendo $0 < \pi < 1$. O sistema de controle consiste em inspecionar o m -ésimo item a cada ciclo de m produzidos (aqui denominado ciclo de inspeção). O item inspecionado é submetido a r classificações (em cada classificação, o item é avaliado em conforme ou não conforme) e caso apresente pelo menos w classificações conforme, o item é julgado conforme. Como o sistema de inspeção é considerado imperfeito, cada classificação a que o item é submetido está sujeita a dois tipos de erro: classificar um item conforme como não conforme (α), e classificar um item não conforme como conforme (β). Caso o item inspecionado seja julgado não conforme, admite-se que ocorreu a mudança de estado e o processo de produção é interrompido, ajustado, e reiniciado sob o estado I. Entre o instante em que o item é inspecionado e o instante em que o processo é parado podem ser produzidos L itens, aos quais se denomina atraso. O processo produzindo sob o estado II só pode retornar ao estado I após a ocorrência de um ajuste.

Conforme a descrição, o processo de inspeção pode ser modelado como uma cadeia de Markov estacionária considerando o espaço de estados discretos $E = \{s_0k_1, s_0k_0, s_1k_1, s_1k_0, s_2k_1, s_2k_0\}$. A variável s indica o real estado do processo em que os itens que compõem o ciclo de inspeção foram produzidos. Quando $s = 0$ (s_0), todos os itens (inclusive o inspecionado) foram produzidos no estado I. Quando $s = 2$ (s_2) todos os itens do ciclo (inclusive o inspecionado) foram produzidos no estado II. Quando $s = 1$ (s_1) necessariamente ocorreu uma mudança do estado I para o estado II no ciclo considerado e pelo menos o item inspecionado foi produzido no estado II. A variável k , por sua vez, indica se o item inspecionado foi declarado conforme (k_1) ou não conforme (k_0). A Figura 1 relaciona o processo de produção aos estados da cadeia. A matriz de transição $\mathbf{P}$ em (1) apresenta as probabilidades de migração entre estes estados em ciclos subsequentes de inspeção.

Figura 1 – Fluxograma do processo e sistema de controle.

$$\mathbf{P} = \begin{matrix} & \begin{matrix} s_0k_1 & s_0k_0 & s_1k_1 & s_1k_0 & s_2k_1 & s_2k_0 \end{matrix} \\ \begin{matrix} s_0k_1 \\ s_0k_0 \\ s_1k_1 \\ s_1k_0 \\ s_2k_1 \\ s_2k_0 \end{matrix} & \begin{bmatrix} P_{(0,1),(0,1)} & P_{(0,1),(0,0)} & P_{(0,1),(1,1)} & P_{(0,1),(1,0)} & P_{(0,1),(2,1)} & P_{(0,1),(2,0)} \\ P_{(0,0),(0,1)} & P_{(0,0),(0,0)} & P_{(0,0),(1,1)} & P_{(0,0),(1,0)} & P_{(0,0),(2,1)} & P_{(0,0),(2,0)} \\ P_{(1,1),(0,1)} & P_{(1,1),(0,0)} & P_{(1,1),(1,1)} & P_{(1,1),(1,0)} & P_{(1,1),(2,1)} & P_{(1,1),(2,0)} \\ P_{(1,0),(0,1)} & P_{(1,0),(0,0)} & P_{(1,0),(1,1)} & P_{(1,0),(1,0)} & P_{(1,0),(2,1)} & P_{(1,0),(2,0)} \\ P_{(2,1),(0,1)} & P_{(2,1),(0,0)} & P_{(2,1),(1,1)} & P_{(2,1),(1,0)} & P_{(2,1),(2,1)} & P_{(2,1),(2,0)} \\ P_{(2,0),(0,1)} & P_{(2,0),(0,0)} & P_{(2,0),(1,1)} & P_{(2,0),(1,0)} & P_{(2,0),(2,1)} & P_{(2,0),(2,0)} \end{bmatrix} \end{matrix} \quad (1)$$

Como exemplo para ilustrar a notação utilizada, $P_{(2,1),(0,0)}$ representa a probabilidade de que a inspeção i ocorra sob o estado da cadeia s_2k_1 e que a inspeção $(i+1)$ ocorra sob o estado s_0k_0 , ou simplesmente $\Pr(E_{i+1} = s_0k_0 \mid E_i = s_2k_1)$. A seguir estão detalhadas as probabilidades de transição dadas em (1).

A probabilidade $P_{(0,1),(0,1)}$, $\Pr(E_{i+1} = s_0k_1 \mid E_i = s_0k_1)$, é o produto das probabilidades: a) de todos os m itens do ciclo serem produzidos no estado I, e b) do item inspecionado ser julgado conforme.

Para obter a) é necessário definir uma variável aleatória não observável, Θ_i , com $i \geq 0$, que representa a probabilidade de produção de um item conforme no instante em que é produzido o i -ésimo item. Com isso, a probabilidade de que o processo permaneça sob o estado I até o m -ésimo item produzido é

$$\Pr\{\Theta_m = p_1\} = \Pr\{\Theta_1 = p_1, \dots, \Theta_m = p_1\} = (1 - \pi)^m \quad (2)$$

Devido à existência de erros de classificação, a probabilidade de que a inspeção declare um item conforme é a soma de duas probabilidades: i) probabilidade de julgar um item conforme como conforme e ii) probabilidade de julgar um item não conforme como conforme. Julgar um item conforme, classificando-o repetidamente e independentemente é um processo binomial em que a probabilidade de sucesso é dada por $(1 - \alpha)$ caso, o item seja de fato conforme ou β caso o item seja, de fato, não conforme. Sendo Y uma variável aleatória que representa o

número de vezes que o item inspecionado é classificado como conforme e θ uma variável aleatória que define o estado real do item ($\theta = 1$, item conforme; $\theta = 0$, item não conforme) tem-se:

$$\Pr\{Y \geq w | \theta = 1\} = 1 - B_{w-1}(r, 1 - \alpha) \quad (3)$$

$$\Pr\{Y \geq w | \theta = 0\} = 1 - B_{w-1}(r, \beta) \quad (4)$$

em que $B_x(n, p)$ denota a probabilidade binomial acumulada até x sucessos em n tentativas, sendo p a probabilidade de sucesso. Note que (3) representa a probabilidade de uma decisão correta, que pode ser denotada como a probabilidade complementar de um erro α^* , enquanto (4) corresponde à probabilidade de uma decisão errada sobre o julgamento do item inspecionado que será denotado como um erro β^* . Assim

$$1 - \alpha^* = 1 - B_{w-1}(r, 1 - \alpha) \therefore \alpha^* = B_{w-1}(r, 1 - \alpha)$$

$$\beta^* = 1 - B_{w-1}(r, \beta)$$

Combinando (2), (3) e (4), tem-se:

$$\begin{aligned} P_{(0,1),(0,1)} &= \Pr\{\Theta_m = p_1\} \{p_1 \Pr\{Y \geq w | \theta = 1\} + (1 - p_1) \Pr\{Y \geq w | \theta = 0\}\} \\ &= (1 - \pi)^m \{p_1(1 - \alpha^*) + (1 - p_1)\beta^*\}. \end{aligned}$$

A probabilidade $P_{(0,1),(0,0)}$, por sua vez, implica em julgar o item inspecionado como não conforme, mesmo com todos os itens produzidos no ciclo sob o estado I. Portanto,

$$\begin{aligned} P_{(0,1),(0,0)} &= \Pr\{\Theta_m = p_1\} \{p_1 \Pr\{Y < w | \theta = 1\} + (1 - p_1) \Pr\{Y < w | \theta = 0\}\} \\ &= (1 - \pi)^m \{p_1\alpha^* + (1 - p_1)(1 - \beta^*)\}. \end{aligned}$$

A probabilidade $P_{(0,1),(1,1)}$, $\Pr(E_{i+1} = s_1 k_1 | E_i = s_0 k_1)$, consiste na mudança do estado I para o estado II desde a última inspeção, bem como no julgamento do item inspecionado como conforme. Note que agora a probabilidade do item inspecionado ser conforme é dada por p_2 . Com isso, tem-se:

$$\begin{aligned} P_{(0,1),(1,1)} &= (1 - \Pr\{\Theta_m = p_1\}) \{p_2 \Pr\{Y \geq w | \theta = 1\} + (1 - p_2) \Pr\{Y \geq w | \theta = 0\}\} \\ &= [1 - (1 - \pi)^m] \{p_2(1 - \alpha^*) + (1 - p_2)\beta^*\}. \end{aligned}$$

A probabilidade $P_{(0,1),(1,0)}$, $\Pr(E_{i+1} = s_1 k_0 | E_i = s_0 k_1)$, indica que há mudança do estado do processo ($I \rightarrow II$), sendo o item inspecionado declarado não conforme.

$$\begin{aligned} P_{(0,1),(1,0)} &= (1 - \Pr\{\Theta_m = p_1\}) \{p_2 \Pr\{Y < w | \theta = 1\} + (1 - p_2) \Pr\{Y < w | \theta = 0\}\} \\ &= [1 - (1 - \pi)^m] \{p_2\alpha^* + (1 - p_2)(1 - \beta^*)\}. \end{aligned}$$

As probabilidades restantes, $P_{(1,1),(2,1)}$, $P_{(1,1),(2,0)}$, $P_{(2,1),(2,1)}$ e $P_{(2,1),(2,0)}$, devem considerar apenas a probabilidade de julgar o item inspecionado conforme ou não conforme, pois o ciclo já se inicia no estado de produção II.

$$\begin{aligned} P_{(1,1),(2,1)} &= P_{(2,1),(2,1)} = \{p_2 \Pr\{Y \geq w | \theta = 1\} + (1 - p_2) \Pr\{Y \geq w | \theta = 0\}\} \\ &= \{p_2(1 - \alpha^*) + (1 - p_2)\beta^*\} \end{aligned}$$

$$\begin{aligned} P_{(1,1),(2,0)} &= P_{(2,1),(2,0)} = \{p_2 \Pr\{Y < w | \theta = 1\} + (1 - p_2) \Pr\{Y < w | \theta = 0\}\} \\ &= \{p_2\alpha^* + (1 - p_2)(1 - \beta^*)\}. \end{aligned}$$

As probabilidades $P_{(0,1),(2,1)} = P_{(0,1),(2,0)} = 0$. Em todos os estados nos quais há ajuste do processo, um novo ciclo de inspeções é iniciado necessariamente sob o estado I. Portanto, todas as linhas pares da matriz $\mathbf{P}$ são iguais à primeira linha da matriz $\mathbf{P}$. Como o processo só retorna ao estado de produção I após um ajuste, as probabilidades $P_{(1,1),(0,1)} = P_{(1,1),(0,0)} = P_{(2,1),(0,1)} = P_{(2,1),(0,0)} = 0$. Por fim, $P_{(1,1),(1,1)} = P_{(1,1),(1,0)} = P_{(2,1),(1,1)} = P_{(2,1),(1,0)} = 0$. A matriz $\mathbf{P}$ pode então ser reescrita como

$$\mathbf{P} = \begin{matrix} & \begin{matrix} s_0 k_1 & s_0 k_0 & s_1 k_1 & s_1 k_0 & s_2 k_1 & s_2 k_0 \end{matrix} \\ \begin{matrix} s_0 k_1 \\ s_0 k_0 \\ s_1 k_1 \\ s_1 k_0 \\ s_2 k_1 \\ s_2 k_0 \end{matrix} & \begin{bmatrix} P_{(0,1),(0,1)} & P_{(0,1),(0,0)} & P_{(0,1),(1,1)} & P_{(0,1),(1,0)} & 0 & 0 \\ P_{(0,1),(0,1)} & P_{(0,1),(0,0)} & P_{(0,1),(1,1)} & P_{(0,1),(1,0)} & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(1,1),(2,1)} & P_{(1,1),(2,0)} \\ P_{(0,1),(0,1)} & P_{(0,1),(0,0)} & P_{(0,1),(1,1)} & P_{(0,1),(1,0)} & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(2,1),(2,1)} & P_{(2,1),(2,0)} \\ P_{(0,1),(0,1)} & P_{(0,1),(0,0)} & P_{(0,1),(1,1)} & P_{(0,1),(1,0)} & 0 & 0 \end{bmatrix} \end{matrix} \quad (4)$$

Como a matriz $\mathbf{P}$ é uma matriz de uma cadeia regular (detalhes em Ross, 2003), então existe o $\lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y}$, em que todas as linhas da matriz $\mathbf{Y}$ são iguais ao vetor linha $\mathbf{y} = [y_{(0,1)} \ y_{(0,0)} \ \dots \ y_{(2,0)}]$. O vetor $\mathbf{y}$ é um vetor de probabilidade ($\sum_i y_i = 1$) no estado estacionário, com todos os valores y_i estritamente positivos. Cada elemento de $\mathbf{y}$ é interpretado como sendo a fração do número de inspeções que são realizadas em cada um dos estados da cadeia após um número suficientemente grande de inspeções. Como $\mathbf{P}^{(j+1)} = \mathbf{P}^{(j)}\mathbf{P}$ e $\lim_{j \rightarrow \infty} \mathbf{P}^{(j+1)} = \lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y}$, então no limite a equação $\mathbf{Y} = \mathbf{Y}\mathbf{P}$ é verdadeira. Como todas as linhas de $\mathbf{Y}$ são iguais à $\mathbf{y}$, a equação $\mathbf{y} = \mathbf{y}\mathbf{P}$ também é válida no limite, e pode ser escrita como

$$\mathbf{y} = \mathbf{y}\mathbf{P} \therefore \mathbf{y}(\mathbf{P} - \mathbf{I}) = \mathbf{0} \quad (5)$$

em que $\mathbf{I}$ é a matriz identidade. Portanto, o vetor $\mathbf{y}$ pode ser obtido a partir da resolução do sistema linear (5) com a restrição de que $\sum_i y_i = 1$. Resolvendo (5), os elementos de $\mathbf{y}$ são dados por

$$\begin{aligned} y_{(0,1)} &= \frac{P_{(0,1),(0,0)} P_{(1,1),(2,0)}}{P_{(0,1),(1,1)} + P_{(1,1),(2,0)}} \\ y_{(0,0)} &= \frac{P_{(0,1),(0,1)} P_{(1,1),(2,0)}}{P_{(0,1),(1,1)} + P_{(1,1),(2,0)}} \\ y_{(1,1)} = y_{(2,0)} &= \frac{P_{(0,1),(1,1)} P_{(1,1),(2,0)}}{P_{(0,1),(1,1)} + P_{(1,1),(2,0)}} \\ y_{(1,0)} &= \frac{P_{(0,1),(1,0)} P_{(1,1),(2,0)}}{P_{(0,1),(1,1)} + P_{(1,1),(2,0)}} \\ y_{(2,1)} &= \frac{P_{(0,1),(1,1)} (1 - P_{(1,1),(2,0)})}{P_{(0,1),(1,1)} + P_{(1,1),(2,0)}} \end{aligned}$$

3. Custo Médio do Sistema de Controle

Para obter a função custo, são necessárias algumas considerações adicionais. Tão logo o item inspecionado é declarado não conforme, o processo é parado para ajuste. Os itens produzidos, entre o instante em que o item inspecionado é amostrado (e declarado não conforme) e a completa parada do processo, são denominados atraso (L) e são descartados. O item inspecionado é classificado r vezes e inutilizado após a inspeção (por exemplo, pequenas partes do item inspecionado podem ser retiradas).

Considere os custos:

- c_0 – custo de uma classificação do item inspecionado;
- c_1 – custo de enviar um item não conforme para o mercado ou próximas etapas do processo;
- c_2 – custo de ajuste do processo;
- c_3 – custo de eliminar um item não conforme;
- c_4 – custo de eliminar um item conforme.

Os componentes c_3 e c_4 são utilizados para a eventualidade dos itens descartados serem submetidos a algum processo de reaproveitamento, no qual o custo de reaproveitamento de um item conforme pode diferir daquele não conforme. Caso isso não seja possível, basta estabelecer $c_3=c_4$, custo este que deve ser o custo unitário de produção (desconsiderado o sistema de controle).

O custo de cada estado (s,k) pode ser escrito como $V_{(s,k)} = rc_0 + \xi_{(s,k)} + \eta_{(s,k)} + \varphi_{(s,k)}$, em que

- rc_0 é o custo de realizar r classificações do item inspecionado (presente em todos os estados da cadeia);
- $\xi_{(s,k)}$, custo de enviar itens defeituosos para o consumidor ou estágios posteriores do processo;
- $\eta_{(s,k)}$, custo relacionado ao descarte do item inspecionado;
- $\varphi_{(s,k)}$, custo de ajuste e de descartar os itens que compõem o atraso.

A seguir estão detalhados os custos para os estados da cadeia de Markov. Para o estado (s_0k_1) , sabe-se que todos os m itens do ciclo foram produzidos no estado I. Dos $(m-1)$ itens que são enviados para o mercado, $(m-1)(1-p_1)$ é o número esperado de itens não-conformes. Portanto

$$\xi(0,1) = c_1(m-1)(1-p_1).$$

O item inspecionado foi declarado conforme, mas, devido à existência de erros de classificação, há uma probabilidade de que ele seja não conforme. Assim

$$\eta(0,1) = c_3 \frac{(1-p_1)\beta^*}{p_1(1-\alpha^*) + (1-p_1)\beta^*} + c_4 \frac{p_1(1-\alpha^*)}{p_1(1-\alpha^*) + (1-p_1)\beta^*}.$$

Como não há ajuste do processo então $\varphi_{(0,1)} = 0$.

Para o estado (s_0k_0) , todos os m itens do ciclo também são produzidos no estado I, logo $\xi_{(0,0)} = \xi_{(0,1)}$. Já o item inspecionado foi declarado não conforme, e o custo de seu descarte é

$$\eta(0,0) = c_3 \frac{(1-p_1)(1-\beta^*)}{p_1\alpha^* + (1-p_1)(1-\beta^*)} + c_4 \frac{p_1\alpha^*}{p_1\alpha^* + (1-p_1)(1-\beta^*)}.$$

Neste estado há ajuste do processo e L itens que compõem o atraso são descartados. Apesar do item inspecionado ter sido produzido no estado I, há uma probabilidade não nula de ocorrer a mudança de estado em algum dos itens que compõe o atraso, o que é considerado em

$$\begin{aligned}\varphi(0,0) &= c_2 + L(1-\pi)^L[(1-p_1)c_3 + p_1c_4 \\ &\quad + c_3 \sum_{i=1}^L (1-\pi)^{i-1} \pi[(i-1)(1-p_1) + (L-i+1)(1-p_2)] \\ &\quad + c_4 \sum_{i=1}^L (1-\pi)^{i-1} \pi[(i-1)p_1 + (L-i+1)(1-p_2)] \\ &= c_2 + L(1-\pi)^L[(1-p_1)c_3 + p_1c_4 \\ &\quad + \frac{(1-\pi)(1-(1-\pi)^L) - (1-\pi)^L \pi L}{\pi} [(1-p_1)c_3 + p_1c_4] \\ &\quad + \frac{(1-\pi)^{L+1} + \pi(L+1) - 1}{\pi} [(1-p_2)c_3 + p_1c_4]\end{aligned}$$

onde i é o instante em que o primeiro item é produzido no estado II.

Para o estado (s_1k_1) , uma parte dos m itens é produzida no estado I. Assim

$$\begin{aligned}\mathcal{E}(1,1) &= c_1 \sum_{i=1}^m \frac{\pi(1-\pi)^{i-1}}{1-(1-\pi)^m} [(i-1)(1-p_1) + (m-i)(1-p_2)] \\ &= c_1(1-p_1) \frac{(1-\pi)(1-(1-\pi)^m) - \pi m(1-\pi)^m}{\pi(1-(1-\pi)^m)} \\ &\quad + c_1(1-p_2) \frac{\pi m + (1-\pi)^m - 1}{\pi(1-(1-\pi)^m)} \\ \eta(1,1) &= c_3 \frac{(1-p_2)\beta^*}{p_2(1-\alpha^*) + (1-p_2)\beta^*} + c_4 \frac{p_2(1-\alpha^*)}{p_2(1-\alpha^*) + (1-p_2)\beta^*}.\end{aligned}$$

Como não há ajuste do processo segue que $\varphi_{(1,1)} = 0$.

Para o estado (s_1k_0) , $\xi_{(1,0)} = \xi_{(1,1)}$. O custo associado a descartar o item inspecionado é

$$\eta(1,0) = c_3 \frac{(1-p_2)(1-\beta^*)}{p_2\alpha^* + (1-p_2)(1-\beta^*)} + c_4 \frac{p_2\alpha^*}{p_2\alpha^* + (1-p_2)(1-\beta^*)}.$$

Como a mudança para o estado II já ocorreu em algum ponto do ciclo em questão, todos os itens que compõem o atraso são produzidos no estado II. Assim

$$\varphi(1,0) = c_2 + L[c_3(1-p_2) + c_4p_2].$$

Para o estado (s_2k_1) , todos os itens do ciclo são produzidos no estado II. Assim

$$\xi(2,1) = c_1(m-1)(1-p_2).$$

Já o custo de descartar o item inspecionado $\eta_{(2,1)} = \eta_{(1,1)}$ e como não há ajuste, $\varphi_{(2,1)} = 0$.

Para o estado (s_2k_0) , são válidas as igualdades de custo: $\xi_{(2,0)} = \xi_{(2,1)}$, $\eta_{(2,0)} = \eta_{(1,0)}$ e $\varphi_{(2,0)} = \varphi_{(1,0)}$.

Em cada ciclo de inspeção, $(m-1)$ itens são enviados para o mercado ou para os próximos estágios de produção. Para um número suficientemente grande de inspeções realizadas, y corresponde ao vetor de probabilidade de ocorrência de cada estado da cadeia. Portanto, o custo médio por item produzido e não descartado (em cada ciclo de inspeção) é:

$$C(m, r, w) = \frac{1}{m-1} \left(rc_0 + \sum_{i=0}^2 \sum_{j=0}^1 y(s_i, k_j) [\xi(s_i, k_j) + \eta(s_i, k_j) + \varphi(s_i, k_j)] \right). \quad (6)$$

O problema consiste em determinar m , r e w que minimizam (6), ou seja

$$(m^0, r^0, w^0) = \arg \min_{(m, r, w)} C \quad (7)$$

Como não é possível obter uma expressão analítica para (7), utiliza-se uma busca computacional exaustiva para determinação dos parâmetros ótimos. As versões do programa em Microsoft Excel e Matlab podem ser obtidas no endereço www.est.ufmg.br/~roberto. A seguir está um exemplo numérico para ilustrar o procedimento proposto.

4. Exemplos Numéricos

Para ilustrar o modelo proposto com um exemplo numérico, considere um processo de manufatura com os seguintes custos:

- Custo de inspeção $c_0 = \$0,25$;
- Custo de envio de um item não conforme $c_1 = \$20$;
- Custo de ajuste $c_2 = \$100$;
- Custo de eliminar um item conforme ou não conforme iguais, $c_3 = c_4 = \$2,0$.

E os seguintes parâmetros de processo:

- Fração conforme no estado I: $p_1 = 0,99$;
- Fração conforme no estado II: $p_2 = 0,80$;
- Probabilidade de mudança do estado I para estado II: $\pi = 0,0001$;
- Erros de classificação: $\alpha = \beta = 0,01$ e
- O número de itens produzidos entre a sinalização (item não conforme) e a completa parada do processo (atraso) $L = 10$.

sendo que os custos estão expressos em unidades monetárias. Para simplificar a execução do modelo, considere que o item será julgado conforme caso mais da metade das r classificações sejam conformes, ou seja, $w = \lceil 0,5r \rceil$. A Figura 2 mostra o custo médio do sistema de controle por item produzido (e não descartado) em função do intervalo entre inspeções (m) e do número de classificações repetidas que o item inspecionado é submetido (r). Os valores ótimos obtidos são $m^0 = 51$ e $r^0 = 3$, que resultam em um custo médio de \$0,3762. Caso uma única classificação do item inspecionado seja realizada ($r = 1$), o intervalo ótimo é $m^0 = 55$ e o custo médio \$0,3853, (2,42% maior que aquele com múltiplas classificações).

Figura 2 – Gráfico de custo médio (C) em função de m e r .

Considere agora que os erros Tipo I e II são desprezados durante a determinação da estratégia ótima de controle. Para esta situação, tem-se $m^o=47$ e $r^o=1$. Porém, adotar esta estratégia em um sistema de inspeção que possui tais erros, sendo $\alpha = \beta = 1\%$, gera um custo médio real $C=\$0,3876$. Este custo é 0,60% maior que o custo da estratégia com $r = 1$ (e 3,03% maior que aquela obtida considerando classificações repetidas).

O critério $w = \lceil 0,5r \rceil$ é uma restrição a um parâmetro que também pode ser utilizado para minimizar o custo médio. Sem esta imposição, a estratégia ótima torna-se $m^o=48$, $r^o=2$ e $w^o=1$, e o custo médio $C = \$0,3719$, (1,14% menor que melhor estratégia obtida até então).

A Tabela 1 mostra os valores ótimos para m , r e w variando-se apenas um parâmetro por vez, com os demais mantidos constantes nos valores utilizados no exemplo numérico do início desta seção. Apesar da faixa de variação escolhida ser arbitrária, respeitam-se as seguintes condições que tornam as combinações utilizadas razoáveis:

- a fração de itens conformes no estado II é inferior à do estado I ($p_2 < p_1$);
- o custo de descartar um item conforme é o mesmo de um item não conforme ($c_3 = c_4$);
- o custo de ajuste é muito maior que o custo de inspeção ($c_2 \gg c_0$) e
- o custo de enviar um item não conforme para o mercado é maior que o custo de descartá-lo ($c_1 > c_3$).

Os parâmetros r^o e w^o mostram-se insensíveis aos custos c_1 , c_2 e c_3 nas faixas utilizadas, enquanto o aumento do custo de cada classificação (c_0) leva a uma redução do número de classificações a que o item inspecionado é submetido, como esperado.

Já o parâmetro m^o apresenta um comportamento distinto para cada um dos custos avaliados. Um aumento no custo de inspeção (c_0) ou no custo de ajuste (c_2) leva ao aumento de m^o , reduzindo a frequência das inspeções e a ocorrência de ajustes desnecessários (ajustar o processo que opera no estado I). Um aumento no custo de envio de itens defeituosos para o mercado (c_1) leva à redução do intervalo entre inspeções, ou seja, ao aumento na frequência das inspeções, com o objetivo de detectar mais prontamente a ocorrência da mudança de estado do processo. Para a faixa pesquisada, c_3 não altera a estratégia ótima de inspeção.

Uma redução na fração de itens conformes produzidos no estado I (p_1) leva a um aumento no intervalo entre inspeções. Isso ocorre porque, com mais itens não-conformes sendo produzidos no estado I, aumenta a ocorrência de ajustes desnecessários, pois mais itens não-conformes são amostrados. Já o aumento na fração de itens conformes produzidos no estado II (p_2) reduz a probabilidade de detectar a ocorrência da mudança, pois mais itens conformes são amostrados. Como resposta a isso, as inspeções tornam-se mais frequentes, ou seja, há uma redução no valor de m^o . Estes comportamentos ficam claros quando o objetivo do sistema de controle é revisto: detectar a mudança de estado do processo tão logo ela ocorra, sendo π o parâmetro que controla a mudança do processo. Como mostra a Tabela 1, o aumento deste parâmetro (π) leva à redução no intervalo entre inspeções, pois em média a mudança de estado (I $\rightarrow$ II) passa a ocorrer após um menor número de itens produzidos pelo processo.

O uso de classificações repetidas (r) e de um critério para declarar um item conforme (w) atuam em resposta à existência dos erros Tipo I (α) e Tipo II (β) no sistema de inspeção. Um aumento de α eleva a probabilidade de declarar um item conforme como não conforme. A estratégia ótima contrapõe este aumento exigindo, proporcionalmente, que um número menor de classificações resulte em conformidade para julgar o item conforme. Para $\alpha = 1\%$, pelo menos $w^o/r^o = 50\%$ das classificações devem ser conformes para julgar o item conforme. Quando $\alpha = 10\%$, a razão diminui para $w^o/r^o = 33,3\%$. Comportamento oposto é observado com o aumento de β , classificar um item defeituoso como conforme. Quando $\beta = 1\%$, no mínimo $w^o/r^o = 50\%$ das classificações devem ser conformes para julgar o item conforme. Quando $\beta = 10\%$ este valor passa a $w^o/r^o = 66,6\%$ devido ao maior número de falsas conformidades.

Tabela 1 – Determinação de m^o , r^o , w^o e C para uma variação individual dos parâmetros do modelo, os demais constantes mantidos nos valores utilizados no exemplo numérico.

$\pi \times 10^3$	C	m^o	r^o	w^o	p_1	C	m^o	r^o	w^o	p_2	C	m^o	r^o	w^o
0,100	0,3719	48	2	1	0,850	3,1855	245	1	1	0,200	0,3474	56	2	1
0,325	0,5262	28	2	1	0,885	2,5221	155	2	1	0,350	0,3551	53	2	1
0,550	0,6383	22	2	1	0,920	1,8319	110	2	1	0,500	0,3621	51	2	1
0,775	0,7334	19	2	1	0,955	1,1178	78	2	1	0,650	0,3681	49	2	1
1,000	0,8186	17	2	1	0,990	0,3719	48	2	1	0,800	0,3719	48	2	1
α	C	m^o	r^o	w^o	β	C	m^o	r^o	w^o	c_0	C	m^o	r^o	w^o
0,010	0,3719	48	2	1	0,010	0,3719	48	2	1	0,250	0,3719	48	2	1
0,033	0,3739	49	2	1	0,033	0,3749	47	2	1	0,563	0,3847	52	2	1
0,055	0,3779	51	2	1	0,055	0,3768	50	3	2	0,875	0,3965	59	1	1
0,078	0,3786	50	3	1	0,078	0,3774	50	3	2	1,188	0,4018	61	1	1
0,100	0,3796	50	3	1	0,100	0,3781	50	3	2	1,500	0,4070	62	1	1
c_1	C	m^o	r^o	w^o	c_2	C	m^o	r^o	w^o	$c_3=c_4$	C	m^o	r^o	w^o
5	0,1400	98	2	1	100	0,3719	48	2	1	1	0,3456	40	2	1
10	0,2239	68	2	1	575	0,4954	73	2	1	2	0,3719	48	2	1
15	0,2999	56	2	1	1050	0,5967	93	2	1	3	0,3947	55	2	1
20	0,3719	48	2	1	1525	0,6872	110	2	1	4	0,4150	61	2	1
25	0,4414	43	2	1	2000	0,7709	125	2	1	5	0,4335	66	2	1

5. Conclusões e Sugestões

Nos casos em que existem erros de classificação no sistema de inspeção por atributos, o modelo desenvolvido considerando o uso de classificações repetidas e um critério para o julgamento do item conforme proporciona uma estratégia de controle tão ou mais eficiente que os modelos de inspeção simples. O que determina o quão mais eficiente será o modelo desenvolvido é a combinação específica dos fatores de custo, das probabilidades de produção nos estados I e II, da probabilidade de mudança de estado e dos erros de inspeção do sistema de controle em questão. Em linhas gerais, quanto menor for o custo de uma inspeção (c_0) e quanto maior a incidência dos erros Tipo I e II, maior será a diferença observada entre a estratégia obtida com o modelo deste trabalho e aqueles de inspeção simples. Para o exemplo numérico adotado, em que há erros de classificação da ordem de 1%, a estratégia de classificações repetidas gera uma redução de 3,48% no custo médio do sistema de controle por item produzido e não descartado, quando comparado ao sistema de inspeção simples.

Para trabalhos futuros sugere-se a inclusão no modelo da decisão de descartar ou enviar para o mercado os itens que compõe o atraso e o item inspecionado. Propõe-se também o uso de uma amostra com dois ou mais itens. Esta modificação deve aumentar significativamente a complexidade do modelo, pois passa a existir a probabilidade de ocorrência da mudança de estado entre os itens que compõem a amostra.

Agradecimentos

Os autores agradecem a Fapesp, CNPq e Capes pelo apoio financeiro.

Referências Bibliográficas

- (1) Borges, W; Ho, L.L. & Turnes, O. (2001). An analysis of Taguchi's on-line quality monitoring procedure for attributes with diagnosis errors. *Applied Stochastic Models in Business and Industry*, **17**, 261-276.
- (2) Greenberg, B.S. & Stokes, S.L. (1995). Repetitive testing in the presence of inspection errors. *Technometrics*, **37**, 102-111.
- (3) Nayeypour, M.R. & Woodall, W.H. (1993). An analysis of Taguchi's on-line quality monitoring procedure for attributes. *Technometrics*, **35**, 53-60.
- (4) Quinino, R. & Suyama, E. (2002). Número ótimo de classificações independentes com erro na avaliação da conformidade de produtos. *Pesquisa Operacional*, **22**, 1-8.
- (5) Quinino, R. & Ho, L.L. (2004). Repetitive tests as an economic alternative procedure to control attributes with diagnosis errors. *European Journal of Operation Research*, **155**, 209-225.
- (6) Ross, S.M. (2003). *Introduction to probability models*. 8th edn. Academic Press, San Diego.
- (7) Taguchi, G.; Elsayed, E.A. & Hsiang, T. (1989). *Quality Engineering in Production in Systems*. McGraw-Hill, New York.
- (8) Taguchi, G.; Chowdhury, S. & Wu, Y. (2004). *Taguchi's Quality Engineering – Handbook*. John Wiley & Sons, Inc. New Jersey.

APÊNDICE C

Artigo publicado na revista
Applied Stochastic Models in Business and Industry

Monitoring process for attributes with quality deterioration and diagnosis errors

Anderson Laécio Galindo Trindade¹, Linda Lee Ho^{1,*},[†] and Roberto da Costa Quinino²

¹*Departamento de Engenharia de Produção, EPUSP, SP, São Paulo, Brazil*

²*Departamento de Estatística, ICEX, UFMG, Belo Horizonte, Minas Gerais, Brazil*

SUMMARY

The aim of this paper is to present an online economical quality-control procedure for attributes in a process subject to quality deterioration after random shift and misclassification errors during inspections. The process starts in control (State I) and, in a random time, it shifts to out of control (State II). Once at State II, the non-conforming fraction increases according to a non-decreasing function $\psi(z)$, where z is the number of items produced after a shift. The monitoring procedure consists of inspecting a single item at every m produced items, which is examined r times independently to decide its condition. Once an inspected item is declared non-conforming, the process is stopped and adjusted. A direct search technique is used to find the optimum parameters which minimize the expected cost function. The proposed model is illustrated by a numerical example. Copyright © 2007 John Wiley & Sons, Ltd.

Received 10 July 2006; Revised 5 March 2007; Accepted 14 March 2007

KEY WORDS: quality control; on-line attribute monitoring; economic model; misclassification errors; Markov chain; quality deterioration

1. INTRODUCTION

Taguchi *et al.* [1] have presented online economical quality-control procedures for attribute quality characteristics. The aim was to propose an inspection policy that minimizes costs due to quality changes in the process. In a random time, the process shifts from the in-control state (producing

*Correspondence to: Linda Lee Ho, Departamento de Engenharia de Produção, EPUSP, SP, São Paulo, Brazil.

[†]E-mail: lindalee@usp.br

Contract/grant sponsor: Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)

Contract/grant sponsor: Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP)

Contract/grant sponsor: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES)

conforming items) to the out-of-control state (the percentage of non-conforming items greater than 0%). Using mathematical symbols, the probability of non-conforming items shifts from $p_1 = 0$ (State I) to $p_2 > 0$ (State II). The monitoring procedure consists of inspecting a single item at every m produced items. If the inspected item is judged conforming, the production process goes on; otherwise, the process is stopped for adjustment.

This type of procedure to monitor process was studied by many authors as Adams and Woodall [2], Srivastava and Wu [3–8], Box and Kramer [9], Box and Luceno [10], Chou and Wang [11], Nandi and Sreehari [12, 13], and Wang and Yue [14]. Examples that put into practice this kind of procedure may include the automatic process of soldering, production of semiconductors, production of diode, production of printed circuit boards and chemical processes [15–17]. Generally, production systems that employ some type of automatic control by collecting individual observations may be improved by the procedure here discussed.

In Taguchi *et al.* [1], the authors assumed implicitly a uniform distribution to describe the shift from State I to State II. Such a condition provided a simpler cost function; however, it does not cover most practical situations. Nayeypour and Woodall [18] developed an alternative procedure assuming a geometric distribution to model the random shifts. They concluded that their approach was more appropriate than Taguchi's proposal mainly for $p_2 > 0$. Nandi and Sreehari [12] developed a model under the assumption that the production process is subject to two assignable causes (in their paper, they are named minor and major).

Borges *et al.* [19] noted that the procedures discussed in Taguchi *et al.* [1] and further papers may present inspection errors, compromising the determination of the optimum sampling interval. To this effect, they not only developed an approach that incorporates inspection errors but also evaluated the impact when those errors are not taken into account. About the effects of inspection errors on quality control, see Johnson *et al.* [20], Ranjan *et al.* [21], and Wang [22].

Previous studies assume that the quality of process shifts from a constant high-quality level (State I) to a constant low-quality level (State II) [1, 12, 16, 18, 19] or follows a specific deterioration scheme as in [13]. In Nandi and Sreehari [13], the authors considered that the process shifts from producing conforming items to $p_2(100)\%$ non-conforming items, and from such a level the quality deterioration is described by a linear model (this function was stated in a specific time interval), until the shift is detected by the control procedure. Due to this specification (the linear function), it is not difficult to calculate the cost function; however, for other deterioration schemes like the exponential, Weibull or log normal, such determination is not easily reached.

In practical situations the reduction of quality level after a shift is a realistic portrayal of how the behaviour of production process and the deterioration function can take on different forms. The common point in [1, 12, 13, 16, 18, 19] is to get the optimum strategy based exclusively on economical criteria, which can result in a high proportion of non-conforming items manufactured. Montgomery [23] suggests that cost functions in economical process control must be optimized through the restriction to adequate statistical performance criteria.

In this paper the procedure was developed considering the following assumptions:

- the process starts in State I (in control, with a fraction of non-conforming items $p_1 \geq 0$) and after a shift, the fraction of non-conforming items increases progressively, described by a non-decreasing function $p_1 \leq p_2 = \psi(z) \leq 1$ (where z is the number of produced items after shift), until an intervention procedure in the process;
- the monitoring procedure consists of inspecting a single item at every m produced items;

- the inspection system is subject to misclassifications and it is possible to perform repetitive independent tests in the inspected item. If the inspected item is classified r times independently (each time as conforming or non-conforming) by an imperfect inspection system and the final judgment of the inspected item is decided based on the most frequent classification, this may reduce the impact of inspection errors. Such a procedure may be possible and economically feasible in many situations [24];
- once the inspected item declared non-conforming, the stoppage of the process follows immediately (no lag time) and the declared non-conforming item is scrapped;
- the optimum parameters are obtained subject to a maximum proportion of non-conforming items (i.e. a threshold value previously fixed at $\phi_{\max}$) that can be manufactured during the process.

For such, the model presented in this study is based on the Markov chain for discrete states. This approach was not considered previously and will allow the use of any deterioration function $\psi(z)$ (a non-decreasing function). The final aim is to obtain the sampling interval (m°) and the number of repeated classifications per inspected item (r°) that minimize the expected average cost per produced item, restricted to a statistical criteria $\phi_{\max}$.

This paper is organized as follows: Section 2 describes the probabilistic model. The cost model developed to determine the optimum values is presented in Section 3. Section 4 features a numerical example to illustrate the proposed model with discussions; conclusions and further suggestions are outlined in Section 5.

2. PROBABILISTIC MODEL

2.1. Inspection system with diagnosis errors

Consider items produced in a process and a control system that consists of inspecting a single item at every m items produced. The inspected item is examined r times independently and each time it is classified as conforming or non-conforming. Each classification may present two types of errors: one named β , in which a non-conforming item is classified as conforming, and the second one known as α , in which a conforming item is classified as non-conforming. The inspected item is declared conforming if more than half of r classifications results conforming, otherwise it is declared non-conforming. If the inspected item is declared conforming, the production goes on; otherwise, the process is stopped for adjustment. In this study, α and β are supposedly lower than 0.5, a reasonable level for practical situations mentioned in many examples (see Johnson *et al.* [25]).

Let ρ denote the number of classifications whose results are 'conforming'. Employing the criteria 'more than half' and, for example, the inspected item being classified $r = 5$ times, it will be declared conforming if $\rho > \lfloor 0.5r \rfloor$ (i.e. $\rho = 3, 4, 5$); otherwise, it will be declared non-conforming if $\rho \leq \lfloor 0.5r \rfloor$ (i.e. $\rho = 0, 1, 2$), where $\lfloor x \rfloor$ is the floor function that returns the largest integer less than or equal to x . For a conforming inspected item, ρ follows a binomial distribution with parameters $(r, 1 - \alpha)$, $\rho \sim \text{bin}(r, 1 - \alpha)$ and the probability of the examined item being wrongly classified after r classifications is

$$\alpha^* = \Pr(\rho \leq \lfloor 0.5r \rfloor \mid \text{inspected item is conforming}) = \sum_{i=0}^{\lfloor 0.5r \rfloor} \binom{r}{i} (1 - \alpha)^i \alpha^{r-i} \quad (1)$$

For a non-conforming inspected item, $\rho \sim \text{bin}(r, \beta)$ and the probability of being wrongly classified as conforming after r classifications is given by

$$\beta^* = \Pr(\rho > \lfloor 0.5r \rfloor \mid \text{inspected item is non-conforming}) = 1 - \sum_{i=0}^{\lfloor 0.5r \rfloor} \binom{r}{i} (\beta)^i (1 - \beta)^{r-i} \quad (2)$$

So, α^* and β^* are, respectively, the probability to classify erroneously a conforming item (α^*) and a non-conforming item (β^*) after r repeated classifications.

2.2. Process control with degenerative quality at State II

The duration of the process (at State I) is usually modelled by the exponential distribution for a continuous case. The geometric distribution is typically used for the discrete case in which the duration is measured by the number of units produced before the shift [18]. Following previous papers [12, 13, 16, 18, 19], in this study we depart from the assumption that the production process may shift randomly from State I to State II, according to a geometric distribution with parameter π . The value of $(\pi^{-1} - 1)$ can be viewed as the average volume of items produced at State I until a shift takes place (since adjustments only happen after a shift).

Once at State II, the probability $p_2 = \psi(z)$ increases progressively, where z ($z \geq 1$) denotes the amount of items produced after a shift to State II. To represent quality deterioration, $\psi(z)$ is a non-decreasing function such that if $z_2 > z_1$ then $\psi(z_2) \geq \psi(z_1)$ and $\psi(1) \geq p_1$; $\psi(\infty) \leq 1$. We assume that changes from State II to State I do not occur (probability equal to zero) if no adjustment is made in process.

The inspection system is described by a Markov Chain and the set of states space $\Omega = \{(0, 1), (0, 2), (1, 1), (1, 2), \dots, (w_i, k_1), (w_i, k_2), \dots, (n, 1), (n, 2)\}$ is used. Each state is described by a pair of integer w and k in a total of $(2n + 2)$ states. The integer k describes the final classification of the inspected item. For $k = 1$ (k_1) the inspected item is declared conforming; otherwise, $k = 2$ (k_2), the inspected item is declared non-conforming. The integer w expresses the number of inspections

Figure 1. The proposed process control.

performed after the process had shifted from State I ($w = 0, 1, \dots, n$). So, for $w = 0$ (w_0), the process is still in-control (at State I). For $w > 0$, w denotes the number of inspections (including the current one) performed at State II. Figure 1 illustrates the inspection process scheme and the states of the chain.

The transition matrix $\mathbf{P}$ for this inspection process is presented in (3). For example, $P_{(w_0, k_1), (w_1, k_2)}$, or, simply, $P_{(0,1), (1,2)}$ refers to the probability of the current inspection to be at state (w_1, k_2) conditioned on the previous inspection at (w_0, k_1) . This probability is represented by the fourth element of the first row of $\mathbf{P}$.

$$\mathbf{P}_n = \begin{matrix} & \begin{matrix} w_0k_1 & w_0k_2 & w_1k_1 & w_1k_2 & w_2k_1 & w_2k_2 & \cdots & w_nk_1 & w_nk_2 \end{matrix} \\ \begin{matrix} w_0k_1 \\ w_0k_2 \\ w_1k_1 \\ w_1k_2 \\ \vdots \\ w_{n-1}k_1 \\ w_{n-1}k_2 \\ w_nk_1 \\ w_nk_2 \end{matrix} & \left(\begin{matrix} P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & \cdots & 0 & 0 \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} \\ P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & P_{(\cdot),(\cdot)} & 0 & 0 & \cdots & 0 & 0 \end{matrix} \right) \end{matrix} \quad (3)$$

The probability $P_{(0,1), (0,1)}$ denotes that the item currently inspected is declared conforming and produced at State I (w_0, k_1) given that the item examined was also declared conforming and produced at State I (w_0, k_1) in a previous inspection. In this cycle, all m items are produced at State I (the probability of this event is $(1 - \pi)^m$); applying the total probability theorem and expressions (1) and (2), the probability of an examined item being declared conforming is $(1 - p_1)(1 - \alpha^*) + p_1\beta^*$. So,

$$P_{(0,1), (0,1)} = (1 - \pi)^m [(1 - p_1)(1 - \alpha^*) + p_1\beta^*] \quad (4)$$

The probability $P_{(0,1), (0,2)}$ is obtained similarly, but the inspected item is declared non-conforming. So,

$$P_{(0,1), (0,2)} = (1 - \pi)^m [(1 - p_1)\alpha^* + p_1(1 - \beta^*)] \quad (5)$$

The probability $P_{(0,1), (1,1)}$ indicates that it is the first inspection after a shift (from State I to State II) and the inspected item is declared conforming. The probability to classify the inspected item as conforming or non-conforming depends on $p_2 = \psi(z)$. Note that at least one item (the inspected one) was produced at State II and considering all possibilities to produce items at State II, its value is

$$P_{(0,1), (1,1)} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)(1 - \alpha^*) + \psi(m - i + 1)\beta^*] \quad (6)$$

where $\bar{\psi}(z) = 1 - \psi(z)$. The value of $P_{(0,1),(1,2)}$ is obtained similarly (but the inspected item was declared non-conforming), so

$$P_{(0,1),(1,2)} = \sum_{i=1}^m (1 - \pi)^{i-1} \pi [\bar{\psi}(m - i + 1)\alpha^* + \psi(m - i + 1)(1 - \beta^*)] \quad (7)$$

The probabilities $P_{(0,1),(j,1)}|_{j=2\dots n} = 0$, since the first inspection must be performed at State II before a second, a third, and other subsequent inspections take place at State II.

When an inspected item is declared non-conforming, the process is adjusted and a new cycle begins at State I. In this sense

$$P_{(j,2),(0,1)}|_{j=1\dots n} = P_{(0,1),(0,1)}, \quad P_{(j,2),(0,2)}|_{j=1\dots n} = P_{(0,1),(0,2)} \quad (8a)$$

$$P_{(j,2),(1,1)}|_{j=1\dots n} = P_{(0,1),(1,1)}, \quad P_{(j,2),(1,2)}|_{j=1\dots n} = P_{(0,1),(1,2)} \quad (8b)$$

Note that, due to the adjustment, all even rows of the matrix $\mathbf{P}$ are equal to the first row of $\mathbf{P}$. If the previous inspection was $(j - 1)$ th inspection at State II and the inspected item was declared conforming [that is, $(j - 1, 1)$], the possible states of the current inspection are $(j, 1)$ or $(j, 2)$ and their probabilities are, respectively

$$P_{(j-1,1),(j,1)}|_{j=2\dots n} = \sum_{i=1}^m \frac{(1 - \pi)^{i-1} \pi}{1 - (1 - \pi)^m} [\bar{\psi}(jm - i + 1)(1 - \alpha^*) + \psi(jm - i + 1)\beta^*] \quad (9)$$

$$P_{(j-1,1),(j,2)}|_{j=2\dots n} = \sum_{i=1}^m \frac{(1 - \pi)^{i-1} \pi}{1 - (1 - \pi)^m} [\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i + 1)(1 - \beta^*)] \quad (10)$$

The chain as stated above assumes infinite states. However, one can note that after a shift to State II and as $n \rightarrow \infty$, the next equality holds

$$\lim_{n \rightarrow \infty} \Pr\{(w_1 k_1), (w_2 k_1), (w_3 k_1), \dots, (w_n k_1)\} = 0 \quad (11)$$

That is, as more items are produced after a shift to State II (i.e. more inspections are performed at State II), more likely the occurrence of an adjustment (process stoppage) is. In this sense, a finite state chain, for n sufficiently large, can be employed and must yield similar numerical results as an infinite one does. Figure 2 illustrates the inspections in a process subject to random shift and quality deterioration.

Note that the probabilities of a process to produce non-conforming items are very close at inspections A and B . Let $\psi_A \approx \psi_B \equiv \psi_n$ and $\bar{\psi}_n = 1 - \psi_n$; replacing them in Equations (9) and (10), they result in

$$\begin{aligned} P_{(j-1,1),(j,1)}|_{j=n+1,\dots} &= \sum_{i=1}^m \frac{(1 - \pi)^{i-1} \pi}{1 - (1 - \pi)^m} [\bar{\psi}_n(1 - \alpha^*) + \psi_n \beta^*] \\ &= \bar{\psi}_n(1 - \alpha^*) + \psi_n \beta^* \end{aligned}$$

Figure 2. Non-conforming fraction in a process subject to random shift and quality deterioration.

Figure 3. The transitions of the states of the chain *versus* corresponding equations.

$$\begin{aligned}
 P_{(j-1,1),(j,2)} | j=n+1, \dots &= \sum_{i=1}^m \frac{(1-\pi)^{i-1} \pi}{1 - (1-\pi)^m} [\bar{\psi}_n \alpha^* + \psi_n (1 - \beta^*)] \\
 &= \bar{\psi}_n \alpha^* + \psi_n (1 - \beta^*)
 \end{aligned}$$

Thus, after the n th inspection, for n sufficiently large, the transition probabilities for the next states of the chain are close to a constant and they can be written as the probabilities $P_{(n,1),(n,1)}$ and $P_{(n,1),(n,2)}$, stated as

$$P_{(n,1),(n,1)} = P_{(n-1,1),(n,1)} \quad (12)$$

$$P_{(n,1),(n,2)} = P_{(n-1,1),(n,2)} \quad (13)$$

The inspections $\{n+1, n+2, \dots\}$ can take place, but they will be very similar to the n th inspection. Equations (12) and (13) are the cutoff that allows the use of a finite Markov chain to describe the proposed model in this study. Figure 3 shows a diagram connecting the transitions of the states of the chain and the number of expressions of the transition probabilities.

Equation (11) or the argument presented in Figure 2 are candidates to be used as criteria to determine the chain size n . For example

$$\min\{n: \Pr\{(w_1k_1), (w_2k_1), (w_3k_1), \dots, (w_nk_1)\} \leq \varepsilon\} \quad (14a)$$

$$\min\{n: |\psi_n - \psi_{n-1}| \leq \varepsilon\} \quad (14b)$$

where ε is a convergence parameter. Despite that, however, it is necessary to include some economical aspect to establish the criteria. For this, Section 3 presents the cost function, while the criteria are presented in Section 4.

2.3. Stationary distribution

As matrix $\mathbf{P}$ is a regular one (for more details, see Ross [26]), $\lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y}$ exists and all rows of $\mathbf{Y}$ are equal to a vector $\mathbf{y}$. Vector $\mathbf{y}$ is a vector of probabilities ($\sum_i y_i = 1$) at a stationary state; each value is strictly positive and can be viewed as the proportion of inspections performed at each state of chain in a large number of inspections. As $\mathbf{P}^{(j+1)} = \mathbf{P}^{(j)}\mathbf{P}$ and $\lim_{j \rightarrow \infty} \mathbf{P}^{(j+1)} = \lim_{j \rightarrow \infty} \mathbf{P}^{(j)} = \mathbf{Y}$, then $\mathbf{Y} = \mathbf{Y}\mathbf{P}$. As all rows of $\mathbf{Y}$ are equal to $\mathbf{y}$, then $\mathbf{y} = \mathbf{y}\mathbf{P}$ is valid. So

$$\mathbf{y} = \mathbf{y}\mathbf{P} \therefore \mathbf{y}(\mathbf{P} - \mathbf{I}) = \mathbf{0} \therefore (\mathbf{P}^T - \mathbf{I})\mathbf{y}^T = \mathbf{0}$$

where $\mathbf{I}$ is the identity matrix and this linear system can be solved by restricting $\mathbf{y}$ to $\sum_i y_i = 1$.

3. COST FUNCTION

Previous studies [1, 12, 13, 16, 18, 19] have considered three major costs: cost of inspection, cost of producing a non-conforming item, and adjustment cost. In this study, a similar structure is considered and the following notation was used:

- c_i the inspection cost (one classification);
- c_{nc} the cost of one non-conforming item which goes undetected to the customer or to later stages of production;
- $c_{s_{nc}}$ the cost to scrap one non-conforming item which was rightly classified as non-conforming during inspection;
- c_{s_c} the cost to scrap one conforming item which was wrongly classified as non-conforming during inspection;
- c_a the process adjustment cost.

The cost for a state (w, k) can be written as $V_{(w,k)} = rc_i + \xi_{(w,k)} + \theta_{(w,k)} + \varphi_{(w,k)}$ where:

- rc_i is the cost to perform the r classifications of the inspected item (included in all states of the chain);
- $\xi_{(w,k)}$ cost of non-conforming items in $(m - 1)$ non-inspected items which will be sent to the customer or to later stages of production;
- $\theta_{(w,k)}$ cost related to the inspected item (only its final classification as conforming or non-conforming is known and the errors α^* and β^* should be considered);
- $\varphi_{(w,k)}$ cost to adjust the process (included for states when $k = 2$).

For the state (w_0, k_1) , the following costs have to be considered: (i) the cost related to produce undetected non-conforming items (the probability to manufacture a non-conforming item at this state is p_1); (ii) the cost related to the inspected item (it is known that the examined item is declared conforming and it goes to later stages, but it can be a non-conforming item due to errors, so the possibility of wrong classification must be considered); and (iii) the process is not adjusted. These costs are, respectively,

$$\xi_{(0,1)} = c_{nc}(m-1)p_1 \quad (15a)$$

$$\theta_{(0,1)} = c_{nc} \frac{p_1\beta^*}{(1-p_1)(1-\alpha^*) + p_1\beta^*} \quad (15b)$$

$$\varphi_{(0,1)} = 0 \quad (15c)$$

For the state (w_0, k_2) , the composition of the costs is similar; however, the examined item is declared non-conforming, it is scrapped (due to errors, that item may actually be conforming or non-conforming), and the process is adjusted. The costs are given by

$$\xi_{(0,2)} = c_{nc}(m-1)p_1 \quad (16a)$$

$$\theta_{(0,2)} = c_{s-nc} \frac{p_1(1-\beta^*)}{(1-p_1)\alpha^* + p_1(1-\beta^*)} + c_{s-c} \frac{(1-p_1)\alpha^*}{(1-p_1)\alpha^* + p_1(1-\beta^*)} \quad (16b)$$

$$\varphi_{(0,2)} = c_a \quad (16c)$$

The state (w_1, k_1) denotes the first inspection after a shift to State II and the inspected item is declared conforming. It means that at least one item (at least the inspected one) was manufactured at State II, so sets of items are produced at State I and State II. As the defective probability at State II depends on the number of items produced, the index i denotes the change point when the first item was produced at State II. As the inspected item is declared conforming, no adjustment takes place. The components of the cost are

$$\xi_{(1,1)} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \quad (17a)$$

$$\theta_{(1,1)} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)\beta^*}{\bar{\psi}(m-i+1)(1-\alpha^*) + \psi(m-i)\beta^*} \right] \quad (17b)$$

$$\varphi_{(1,1)} = 0 \quad (17c)$$

Similarly, the state (w_1, k_2) indicates a first cycle at State II and the examined item declared non-conforming. Adjustment and the scrapping of the inspected item must be taken into consideration. The components of the costs are

$$\xi_{(1,2)} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[(i-1)p_1 + \sum_{l=1}^{m-i} \psi(l) \right] \quad (18a)$$

$$\begin{aligned} \theta_{(1,2)} = c_{s_nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(m-i+1)(1-\beta^*)}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] \\ + c_{s_c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(m-i+1)\alpha^*}{\bar{\psi}(m-i+1)\alpha^* + \psi(m-i+1)(1-\beta^*)} \right] \end{aligned} \quad (18b)$$

$$\varphi_{(1,2)} = c_a \quad (18c)$$

For the states $(w_j, k_1)|_{j=2\dots n}$, there is no adjustment to the process and the costs are

$$\xi_{(j,1)|_{j=2\dots n}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1) \quad (19a)$$

$$\theta_{(j,1)|_{j=2\dots n}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm - i + 1)\beta^*}{\bar{\psi}(jm - i + 1)(1-\alpha^*) + \psi(jm - i + 1)\beta^*} \right] \quad (19b)$$

$$\varphi_{(j,1)|_{j=2\dots n}} = 0 \quad (19c)$$

For the states $(w_j, k_2)|_{j=2\dots n}$, adjustments are made and the costs related to these states are

$$\xi_{(j,2)|_{j=2\dots n}} = c_{nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \sum_{l=1}^{m-1} \psi((j-1)m - i + l + 1) \quad (20a)$$

$$\begin{aligned} \theta_{(j,2)|_{j=2\dots n}} = c_{s_nc} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\psi(jm - i + 1)(1-\beta^*)}{\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i + 1)(1-\beta^*)} \right] \\ + c_{s_c} \sum_{i=1}^m \frac{(1-\pi)^{i-1}\pi}{1-(1-\pi)^m} \left[\frac{\bar{\psi}(jm - i + 1)\alpha^*}{\bar{\psi}(jm - i + 1)\alpha^* + \psi(jm - i)(1-\beta^*)} \right] \end{aligned} \quad (20b)$$

$$\varphi_{(j,2)|_{j=2\dots n}} = c_a \quad (20c)$$

For a sufficiently large number of inspections, each element of the vector $\mathbf{y}$ expresses the probability of each state of the chain $\mathbf{P}$, and at each cycle of inspection m items are manufactured. So, the average cost per manufactured item, C , is given by

$$C(m, r, n) = \frac{1}{m} \left(rc_i + \sum_{i=0}^n \sum_{j=1}^2 \mathbf{y}_{[2i+j]} (\xi_{(w_i, k_j)} + \theta_{(w_i, k_j)} + \varphi_{(w_i, k_j)}) \right) \quad (21)$$

where $\mathbf{y}_{[2i+j]}$ denotes the stationary probability of the state (w_i, k_j) . The determination of the values m and r such that minimize (21) subjected to the statistical restriction is the problem to be solved. This can be expressed by

$$(m^0, r^0) = \arg \min_{(m, r)} C \quad \text{subject to } \phi < \phi_{\max}$$

where ϕ is the expected proportion of non-conforming items produced in the process and $\phi_{\max}$ its maximum value allowed. Note that probability ϕ can be derived from Equation (21): just fixing cost components related to a non-conforming item equals to 1, and all others equal to 0. This result can be written as

$$\phi(m, r, n) = C(m, r, n)|_{c_i=0; c_a=0; c_{nc}=1; c_{s_c}=0; c_{s_nc}=1}$$

4. NUMERICAL EXAMPLE AND DISCUSSIONS

To illustrate the proposed procedure, consider the example adapted from Taguchi *et al.* [1, 17]. A manufacturer of high-volume-integrated circuits is facing difficulties in the soldering process. Soldered connections may present problems such as insufficient or excess solder, lead projection, or incorrect positioning of a device or lead. Historical data allow adopting $p_1 = 0.01$ as defective probability in a welding process (State I: in control), and the shifts to out-of-control state (State II) can be described by a geometric distribution (i.e. $\pi = 0.0001$). Once in State II, conformance probability deterioration follows the exponential model $p_2 = 1 - 0.9e^{-0.001(z-1)}$, where z is the number of integrated circuits manufactured after a shift. The automated X-ray inspection system installed in the production line is imperfect and inspection errors $\alpha = \beta = 0.01$ are assumed. The cost components are estimated as $c_i = \$0.25$; $c_{nc} = \$20$; $c_a = \$100$; $c_{s_nc} = \$2.0$; $c_{s_c} = \$2.5$ and the maximum proportion of non-conforming integrated circuits allowed is 3% (i.e. $\phi_{\max} = 0.03$). These values are summarized in Table I.

4.1. Truncation criteria

Suppose the current control procedure consists of inspecting a single item at every 25 items produced ($m = 25$) and that each item is classified 3 times ($r = 3$). Considering these values and those in Table I, the only unassigned variable is the chain size n . According to the previous description of the variable w in the chain states ($w = 0$, all items produced at State I in the current inspection cycle; $w = 1$, cycle when the shift occurs and at last one item is produced at State II; $w = 2$, second inspection after the shift and the first one when necessarily all items are produced at State II) it is natural to suppose that the minimum chain size to represent adequately the process (State I $\rightarrow$ shift $\rightarrow$ State II) is $n = 2$.

The procedure to determine n starts by calculating the cost (21) for $\mathbf{P}_{n=2}$ and denoting it as $C_{n=2}$. Now the cost $C_{n=3}$ should be calculated. Let δ_n be the relative difference of cost in two successive chain sizes defined by

$$\delta_n = \frac{|C_n - C_{n-1}|}{C_n} \quad (22)$$

Let $\Delta_{\max}$ be the desired convergence tolerance. The chain size $n = 3$ will be adequate if $\delta_3 < \Delta_{\max}$. Otherwise, the cost $C_{n=4}$ should be calculated and so on. Hence, the truncation criterion can be written as

$$\min\{n: \delta_n < \Delta_{\max}\} \quad (23)$$

In contrast with some possible criteria presented in (14a), (23) measures directly the performance of the chain $\mathbf{P}_n$ in terms of the target variable: the cost. Moreover, in some way those criteria are included in (23), once cost depends on $\mathbf{P}_n$ and $p_2 = \psi(z)$.

Table I. Base parameters.

$p_1 = 0.01$	$\pi = 0.0001$	$\phi_{\max} = 0.03$	$c_a = 100$
$p_2 = 1 - 0.9e^{-0.001(z-1)}$	$\alpha = 0.01$	$c_i = 0.25$	$c_{s_nc} = 2.00$
$\Delta_{\max} = 1 \times 10^{-6}$	$\beta = 0.01$	$c_{nc} = 20$	$c_{s_c} = 2.50$

Table II. Cost as function of chain size n for three distinct deterioration functions.

n	Exponential $p_2 = 1 - 0.9e^{-0.001(z-1)}$		Linear $p_2 = \min \left\{ 0.1 + \frac{0.9(z-1)}{400}; 1 \right\}$		Constant $p_2 = 0.6$	
	C_n	δ_n	C_n	δ_n	C_n	δ_n
2	0.310763	—	0.308729	—	0.306953	—
3	0.311810	3.36×10^{-03}	0.310587	5.98×10^{-03}	0.306953	*0
4	0.312467	2.10×10^{-03}	0.311466	2.82×10^{-03}	0.306953	0
5	0.312885	1.34×10^{-03}	0.311888	1.35×10^{-03}	0.306953	0
6	0.313152	8.53×10^{-04}	0.312086	6.35×10^{-04}	0.306953	0
7	0.313322	5.44×10^{-04}	0.312175	2.85×10^{-04}	0.306953	0
8	0.313430	3.45×10^{-04}	0.312212	1.20×10^{-04}	0.306953	0
9	0.313498	2.16×10^{-04}	0.312226	4.61×10^{-05}	0.306953	0
10	0.313540	1.34×10^{-04}	0.312231	1.61×10^{-05}	0.306953	0
11	0.313566	8.23×10^{-05}	0.312233	4.93×10^{-06}	0.306953	0
12	0.313582	4.97×10^{-05}	0.312233	1.30×10^{-06}	0.306953	0
13	0.313591	2.95×10^{-05}	0.312233	* 2.84×10^{-07}	0.306953	0
14	0.313596	1.72×10^{-05}	0.312233	4.90×10^{-08}	0.306953	0
15	0.313599	9.85×10^{-06}	0.312233	6.10×10^{-09}	0.306953	0
16	0.313601	5.54×10^{-06}	0.312233	4.64×10^{-10}	0.306953	0
17	0.313602	3.05×10^{-06}	0.312233	9.82×10^{-11}	0.306953	0
18	0.313603	1.65×10^{-06}	0.312233	1.40×10^{-14}	0.306953	0
19	0.313603	* 8.71×10^{-07}	0.312233	1.78×10^{-16}	0.306953	0
20	0.313603	4.51×10^{-07}	0.312233	0	0.306953	0
100	0.313603	0	0.312233	0	0.306953	0

To illustrate the application of (23) under different degradation conditions, three functions of p_2 were considered: the exponential, the linear (from Nandi and Sreehari [13]), and the constant. Table II exhibits the values of C_n and δ_n for different chain sizes. The parameters used were those in Table I and the control procedure ($m = 25, r = 3$).

Considering $\Delta_{\max} = 1 \times 10^{-6}$, the symbol * in Table II indicates the minimum chain size that meets the criterion for the exponential ($n = 19$), the linear ($n = 13$), and the constant models ($n = 3$). To verify the convergence, a chain size $n = 100$ was used, and the resulting costs confirm that the convergence occurs.

The constant function reveals an interesting behaviour: for any chain size $\delta_n = 0$ (see Table II). Figure 4(a) and (b) shows that for a non-constant function $p_2 = \psi(z)$, the larger the chain size (that is, $n \uparrow$), then $\delta_n \rightarrow 0$. The dashed line is the original function and the flat line corresponds

Figure 4. Representation of truncated chain considering non-constant and constant functions.

to the truncated chain. Figure 4(c) shows that when $p_2 = \text{constant}$, the flat line coincides with the dashed one. In this case, any chain size represents exactly the function $p_2 = \text{constant}$.

4.2. Optimum direct search

To obtain values for (m^0, r^0) , a program was implemented using the software Matlab (6.5 or later versions) and Microsoft Excel (97 or later versions), both are ready to use and available for download from <http://www.est.ufmg.br/~roberto>. Briefly, the direct search is presented in following steps:

- Step 1: Define $c_i, c_{nc}, c_a, c_{s_nc}, c_{s_c}, \alpha, \beta, \pi, p_1, p_2 = \psi(z), \phi_{\max}$.
- Step 2: Define $\Delta_{\max}, m_{\min}, m_{\max}, r_{\min}, r_{\max}$.
- Step 3: Set $m = m_{\min}, C^0 = \infty, m^0 = 0, r^0 = 0$.
- Step 4: Set $r = r_{\min}, C_{\text{new}} = \infty$.
- Step 5: Set $n = 2$, calculate $C_{n=2}$.
- Step 6: Set $n = n + 1$, calculate C_n and δ_n .
- Step 7: If $\delta_n < \Delta_{\max}$ go to Step 8. Otherwise, return to Step 6.
- Step 8: If $C_n < C_{\text{new}}$, set $C_{\text{new}} = C_n$ and $r_{\text{new}} = r$.
- Step 9: Set $r = r + 1$. If $r > r_{\max}$, go to Step 10. Otherwise, go to Step 5.
- Step 10: Calculate ϕ . If $\phi > \phi_{\max}$ go to Step 13.
- Step 11: If $C_{\text{new}} < C^0$, set $C^0 = C_{\text{new}}, r^0 = r_{\text{new}}$ and $m^0 = m$.
- Step 12: Set $m = m + 1$. If $m > m_{\max}$ go to Step 13. Otherwise, go to Step 4.
- Step 13: The optimum is C^0, m^0, r^0 .

4.3. The optimum policy for the numeric example

Figure 5 presents the average cost and the correspondent chain size considering the parameters from Table I for $1 \leq m \leq 10^6$. Running the program, the optimum strategy is achieved as $(m^0 = 32, r^0 = 3, C^0 = 0.310873)$, which corresponds to a non-conforming fraction of $\phi = 1.27\%$ (the restriction $\phi_{\max}$ is satisfied for $m < 300$). According to Figure 5(b), note that, as m increases ($m > 7$), the chain size n required to satisfy the convergence criteria $\delta_n < \Delta_{\max}$ decreases, as expected. For small values of m ($m \leq 7$), the opposite occurs: frequent inspections lead for an adjustment before a shift to State II; therefore, the presence of State II is only perceived with an increase of m and, in reaction to this, the chain size n increases.

Figure 5. Cost (C) and values of (n) for different inspection intervals (m).

Table III. Optimum policy under distinct deterioration schemes.

$p_2(z)$	$c_{nc} = 20$			$c_{nc} = 5$		
	m°	r°	C°	m°	r°	C°
$p_2 = 1 - 0.9e^{-0.001(z-1)}$	32	3	0.3109	69	3	0.1120
$p_2 = \min \left\{ 0.1 + \frac{0.9(z-1)}{400}; 1 \right\}$	33	3	0.3085	75	3	0.1097
$p_2 = 0.6$	34	3	0.3023	72	3	0.1085

For several numerical examples, the shapes of the cost function are similar to that presented in Figure 5(a): a fast decreasing average cost as m increases provides a minimum value, followed by a gradual increase in the average cost as m increases from the minimum value. The curve presents the asymptotical behaviour as $m \rightarrow \infty$, since all items are produced (and sent to the customer) under a defective probability $p_2 = \psi(\infty)$, and the average cost will be $\lim_{m \rightarrow \infty} C = \psi(\infty)c_{nc}$. Considering the current example, $\psi(\infty) = 1$ and $\lim_{m \rightarrow \infty} C = c_{nc} = 20$, as Figure 5(a) shows.

4.4. The optimum policy for different deterioration schemes

Table III presents the optimum control policies obtained for the three deterioration functions used in Table II, considering two levels of component cost c_{nc} . All other parameters are those in Table I. The results show that distinct deterioration functions can produce different control policies and costs. The magnitude of this difference will depend on the specific combination of the parameters in Table I. So, the function that better describes the production process under study must be used to guarantee a real optimum policy. The programs in Matlab and Excel can be easily adapted to any specific function $p_2 = \psi(z)$ (only one line in each program must be changed).

4.5. The impact of classification errors

Suppose that the parameters presented in Table I are the underlying parameters of a current production process. To illustrate the effect of classification errors, consider the three following scenarios.

Scenario 1: The presence of classification errors is unknown. In this situation, the control policy is determined assuming $\alpha = 0\%$ and $\beta = 0\%$. The optimum values are ($m^\circ = 26$, $r^\circ = 1$) and the average cost in a system without classification errors is 0.292440. Applying this control policy to the current production process (with errors $\alpha = \beta = 1\%$) yields the real cost 0.329149.

Scenario 2: The errors are considered, but a single classification is made. The knowledge of errors $\alpha = \beta = 1\%$ makes them to be taken into account. Solving the proposed model restricted to a single classification ($r = 1$, as in previous works), produces the minimum cost $C = 0.322724$ at control policy ($m^\circ = 36$, $r^\circ = 1$). This value represents a reduction of 2.0% compared to Scenario 1.

Scenario 3: The errors and multiple classifications are considered. In this scenario, the errors are known and one considers the hypothesis to submit the item inspected to multiple classifications. Now the optimum cost is $C = 0.310873$ at control policy ($m^\circ = 32$, $r^\circ = 3$). It represents a reduction of 5.6 and 3.7% compared to Scenarios 1 and 2, respectively.

These scenarios show the impact on cost if errors and multiple classifications are not taken into account, although that impact will depend on the specific combination of the parameters in Table I.

Figure 6 illustrates the behaviour of the inspection interval and repetitive tests as diagnosis errors increase, considering the parameters in Table I. An increase in the repetitive test (r°) is required in order to minimize the effects of classification errors in the example presented. As α increases, one is more likely to make wrong decisions regarding conforming items, resulting in unnecessary adjustments. To oppose this increase, an increase in sampling interval (m°) is observed. As β increases, is more likely to declare non-conforming items as conforming and slow down shift detection, requiring an increase in inspection effort (i.e. a reduction in sampling interval m°). Note that m° as well as r° are more sensitive to changes in α . This is observed in all combinations of (α, β) (see Figure 6) and can be explained when the probability of the state w_0k_1 is analysed: over 97% of the inspections are performed at this state (i.e. $y_{[1]} > 0.97$), in which the fraction of conforming items ($1 - p_1$) is high. So, the occurrence of misclassification error α is more likely than error β , once one is more likely to inspect a conforming item.

Figure 6. Values of α and β versus m and r .

4.6. The cost components and constrained optimization

As stated by Dasgupta [16], estimation of the cost components is a real difficult task and can never be expected to be accurate. However, Montgomery [23] claims that cost components do not need to be estimated with high accuracy. Note that ratios of the different components of cost can also be used to optimize Equation (21) since they are easier to be obtained, which further strengthens our earlier argument. The fact is that to use economic model, the cost components must be known some way.

Suppose the costs presented in Table I were estimated with an error margin of $\pm 25\%$ and random samples of $(c_i, c_{nc}, c_a, c_{s_{nc}}, c_{s_c})$ are taken from a uniform distribution in the range (75–125%) of corresponding cost components in Table I (other parameters are kept fixed). For each sample, the resulting average costs considering control policies of Scenario 1 ($m^\circ = 26, r^\circ = 1$), Scenario 2 ($m^\circ = 36, r^\circ = 1$), and Scenario 3 ($m^\circ = 32, r^\circ = 3$) are calculated. This procedure was repeated 10 000 times and Figure 7 shows the comparative results. In all generated samples, the control policy of Scenario 3 produces a reduction of average cost when compared to Scenarios 1 and 2. This corroborates the results obtained in the previous subsection, despite an error of $\pm 25\%$.

To determine the effect of each cost component in the average cost, for example, a multiple linear regression may be used naively. In this case, the cost components sampled (actually, its values in the range 75–125%, denoted by the symbol $*$) are the independent variables and the average cost is a dependent one. Table IV shows the coefficient estimates; the highest coefficient is related to c_{nc} followed by c_a and c_i . This procedure may be used to identify which cost component must be estimated more accurately.

Figure 7. Cost reduction: scenario 3 versus scenarios 1 and 2.

Table IV. Estimates of the effects of cost components on the average cost.

Scenario	$C = b_0 + b_1 c_i^* + b_2 c_{nc}^* + b_3 c_a^* + b_4 c_{s_{nc}}^* + b_5 c_{s_c}^*$					
	b_0	b_1	b_2	b_3	b_4	b_5
Scenario 1	0.0000	0.0096	0.2327	0.0850	0.0009	0.0009
Scenario 2	0.0000	0.0069	0.2505	0.0639	0.0007	0.0007
Scenario 3	0.0000	0.0234	0.2451	0.0415	0.0008	0.0000

Figure 8 presents the distribution of inspection interval m° when the optimization search is run for each sampled cost component (in all cases, $r^\circ = 3$). The results show that the control policy based on the original values of cost components ($m^\circ = 32$, $r^\circ = 3$) is one of the most frequent sampling intervals and it represents a good choice to control the production process. To get a narrower distribution of m° it may be necessary to improve the estimate of the most important component costs.

Using the Monte Carlo simulation one may determine, for example, the control policy that minimizes the overall average cost (i.e. fixed m and r , calculate the average cost for each one of

Figure 8. Sampling distribution of the optimum sampling interval m° .

Table V. Results of sensibility analysis.

		c_i															
		0.25				0.50				0.75				1.00			
c_{nc}	c_a	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°	m°	r°	C°	ϕ°
5	50	57	1	0.0984	1.4%	63	1	0.1025	1.5%	68	1	0.1063	1.5%	73	1	0.1098	1.6%
	100	69	3	0.1120	1.6%	82	1	0.1206	1.6%	87	1	0.1236	1.7%	91	1	0.1264	1.7%
	150	79	3	0.1236	1.6%	92	3	0.1325	1.7%	102	1	0.1386	1.8%	106	1	0.1410	1.8%
	200	88	3	0.1345	1.7%	100	3	0.1425	1.8%	110	3	0.1497	1.8%	119	1	0.1543	1.9%
10	50	39	1	0.1653	1.3%	43	1	0.1714	1.3%	47	1	0.1770	1.4%	50	1	0.1821	1.4%
	100	47	3	0.1832	1.4%	57	1	0.1957	1.4%	60	1	0.2000	1.5%	63	1	0.2041	1.5%
	150	54	3	0.1981	1.4%	63	3	0.2109	1.5%	70	1	0.2198	1.5%	73	1	0.2233	1.6%
	200	60	3	0.2118	1.5%	68	3	0.2235	1.5%	76	3	0.2339	1.6%	82	1	0.2405	1.6%
15	50	31	1	0.2274	1.2%	34	1	0.2350	1.3%	37	1	0.2420	1.3%	40	1	0.2484	1.3%
	100	37	3	0.2486	1.3%	45	1	0.2643	1.4%	48	1	0.2697	1.4%	50	1	0.2748	1.4%
	150	43	3	0.2661	1.4%	50	3	0.2822	1.4%	57	1	0.2931	1.4%	59	1	0.2975	1.5%
	200	48	3	0.2820	1.4%	55	3	0.2966	1.4%	61	3	0.3096	1.5%	66	1	0.3179	1.5%
20	50	26	1	0.2867	1.2%	29	1	0.2957	1.2%	32	1	0.3039	1.3%	34	1	0.3115	1.3%
	100	32	3	0.3109	1.3%	39	1	0.3294	1.3%	41	1	0.3357	1.3%	43	1	0.3416	1.3%
	150	37	3	0.3306	1.3%	43	3	0.3495	1.4%	48	1	0.3624	1.4%	50	1	0.3674	1.4%
	200	41	3	0.3486	1.3%	47	3	0.3657	1.4%	52	3	0.3809	1.4%	56	1	0.3905	1.4%

10 000 sampled cost components and use the average over all samples as the overall average cost). This procedure can be utilized to deal with the uncertainty of any parameter from Table I and the best control policy under that uncertainty is provided.

Table V presents the optimum policy (m° ; r°), average cost, and defective fraction at different levels of c_{nc} , c_a and c_i (with other parameters kept fixed as in Table I). An increase in c_a (adjustment cost) or c_i (inspection cost) reduces the frequency of inspections (i.e. increases m°), while an increase in c_{nc} reduces m° to speed up the shift detection and to avoid the production of defective items. The number of classifications r° is a strong function of c_i , as expected: an increase in c_i reduces to a single classification.

All the examples calculated are restricted to the criterion $\phi_{\max} = 3\%$; however, 3% is 3 times of the defective fraction when the process is in-control state (see Table I, $p_1 = 0.01$). So, if a more restrictive criterion is adopted, for example $\phi_{\max} = 1.5\%$, several rows of Table V have to be calculated again. To illustrate the effect of this more restrictive criterion, consider the cost components $c_{nc} = 5$, $c_a = 200$ and $c_i = 0.5$ from Table V, which yielded $\phi = 1.8\%$. Running the program with $\phi_{\max} = 1.5\%$, the result is $m^\circ = 63$, $r^\circ = 3$, $\phi = 1.4984\%$ and $C^\circ = 0.1503$ (an increase of 5.5% over that corresponding cost in Table V to meet the new statistical constraint).

5. CONCLUSIONS AND FINAL REMARKS

The proposed model is naturally more complex than those with which previous works were concerned since it utilizes the Markov chain with discrete state space. This can be justified by two new features included: the possibility of multiple classifications of an inspected item and the use of any degradation function $p_2 = \psi(z)$. Despite this complexity, the most sophisticated resource used is an algorithm to solve the system of linear equations to determine $\mathbf{y}$ (the stationary distribution of states). Therefore, any software that permits programming can be used to implement our model. Two versions (Matlab and Excel) are available to download from <http://www.est.ufmg.br/~roberto> and ready to use considering the exponential model in Table I. To any other degradation function, the user has to change only one line in those programs (this is just what was made to simulate the linear and constant models in Section 4).

Definitely the inspection of items in the monitoring procedure is not an error-free activity. Regarding this issue, procedures for attributes have shown that, if neglected, inspection errors can greatly affect the average cost. Results obtained in Section 4 indicate that under faulty inspection a single classification ($r = 1$) is often unlikely to be an optimum procedure. The repetitive tests proposed in this paper can be a viable option to reduce the average cost of the monitoring procedure, mainly when costs c_a or c_{nc} are very high when compared to c_i , or when α or β are not negligible. Note that even small misclassification errors can alter the expected cost as well as the optimum values, as shown in Scenarios 1–3.

The precise estimate of cost components always will be a matter of discussion in the literature. Despite that, by associating the proposed model with the Monte Carlo simulation, the optimum procedure control can be obtained from inaccurate information as well as the most important cost components can be determined. Operationally there are no practical rules to suggest ideal values for m and r , since the average cost function depends on a large number of parameters. In this sense, it is advisable to run the program to make decisions.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the referees for their valuable comments and suggestions. This research was supported in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq); by Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) and by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

REFERENCES

1. Taguchi G, Elsayed EA, Hsiang T. *Quality Engineering in Production Systems*. McGraw-Hill: New York, 1989.
2. Adams BM, Woodall WH. An analysis of Taguchi's on-line process-control procedure under a random-walk model. *Technometrics* 1989; **31**(4):401–413.
3. Srivastava MS, Wu Y. A second order approximation to Taguchi *et al.*'s on-line control procedure. *Communications in Statistics A* 1991; **20**:2149–2168.
4. Srivastava MS, Wu Y. An improved version of Taguchi's on-line control procedure. *Journal of Statistical Planning and Inference* 1995; **43**:133–145.
5. Srivastava MS, Wu Y. On-line control procedures under the random-walk model with measurement error and attribute observations. *Canadian Journal of Statistics* 1994; **22**:377–386.
6. Srivastava MS, Wu Y. Economical quality control procedures based on symmetric random walk model. *Statistica Sinica* 1996; **6**:389–402.
7. Srivastava MS, Wu Y. On-line quality control procedures for a random walk model with measurement error. *Sequential Analysis* 1997; **16**:93–105.
8. Srivastava MS, Wu Y. Economical process adjustment with measurement error. *Communications in Statistics—Theory and Methods* 1999; **28**:989–1003.
9. Box GEP, Kramer T. Statistical process monitoring and feedback adjustment—a discussion. *Technometrics* 1992; **34**:251–285.
10. Box GEP, Luceno A. Selection of sampling interval and action limit for discrete feedback adjustment. *Technometrics* 1994; **36**:369–378.
11. Chou S, Wang MC. Taguchi's method and empirical procedure for on-line process control. *Communications in Statistics—Theory and Methods* 1996; **25**(2):301–320.
12. Nandi SN, Sreehari M. Economy based on-line quality control method for attributes. *The Indian Journal of Statistics* 1997; **59**:384–395.
13. Nandi SN, Sreehari M. Some improvements in Taguchi's economic method allowing continued quality deterioration in production process. *Communications in Statistics—Theory and Methods* 1999; **28**(5):1169–1181.
14. Wang MC, Yue J. Economic design of process adjustment for on-line control. *International Journal of Production Research* 2001; **29**(5):805–823.
15. Ishino N, Ogawa M. Study of on-line QE for manufacturing of the 'rectifiers' semi-conductor. *Quality Engineering* 1999; **7**:51–55.
16. Dasgupta T. An economic inspection interval for control of defective items in a hot rolling mill. *Journal of Applied Statistics* 2003; **30**:273–282.
17. Taguchi G, Chowdhury S, Wu Y. *Taguchi's Quality Engineering Handbook*. Wiley: New Jersey, 2004.
18. Nayeypour MR, Woodall WH. An analysis of Taguchi's on-line quality-monitoring procedures for attributes. *Technometrics* 1993; **35**(1):53–60.
19. Borges W, Ho LL, Turnes O. An analysis of Taguchi's on-line quality monitoring procedure for attributes with diagnosis errors. *Applied Stochastic Models in Business and Industry* 2001; **17**:261–276.
20. Johnson NL, Kotz S, Wu X. *Inspection Errors for Attributes in Quality Control*. Chapman & Hall: London, 1991.
21. Ranjan P, Xie M, Goh TN. Optimal control limits for CCC charts in the presence of inspection errors. *Quality and Reliability Engineering International* 2003; **19**:149–160.
22. Wang C-H. Economic off-line quality control strategy with two types of inspection errors. *European Journal of Operational Research* 2007; **179**:148–159.

23. Montgomery DC. *Introduction to Statistical Quality Control* (4th edn). Wiley: New York, 2001.
24. Greenberg BS, Stokes SL. Repetitive testing in the presence of inspection errors. *Technometrics* 1995; **37**(1): 102–111.
25. Johnson NL, Kotz S, Balakrishnan N. *Univariate Discrete Distributions* (2nd edn). Wiley: New York, 1992.
26. Ross SM. *Introduction to Probability Models* (8th edn). Academic Press: San Diego, CA, 2003.