MÉTODO PARA OTIMIZAÇÃO DE UM SISTEMA DE FILAS M/PH/c/N COM INFORMAÇÃO PARCIAL

Solon Venâncio de Carvalho
Rita de Cássia Meneses Rodrigues
Laboratório Associado de Computação e Matemática Aplicada (LAC)
Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515, CEP 12245-970, São José dos Campos (SP), Brasil
E-mail: {solon, rita}@lac.inpe.br

Resumo: Neste trabalho propõem-se um modelo markoviano de decisão para um sistema de filas com tempos entre chegadas exponencialmente distribuídos, tempos de serviço seguindo uma distribuição PH com configuração de Cox e até c servidores trabalhando no sistema. Considera-se que o sistema tem capacidade limitada N e, desta forma, clientes que chegam quando a capacidade está preenchida são rejeitados. Deseja-se obter uma política admissível que dinamicamente prescreve o número de servidores ativos, baseados na observação parcial do estado do sistema, a fim de minimizar o custo esperado médio do sistema a longo prazo. O custo do sistema é composto de um custo de espera, um custo de serviço, um custo de ativação do servidor, um custo de rejeição e um custo de manter o servidor ativo.

Palavras-chave: *sistema de filas, processo markoviano de decisão, distribuição do tipo fase*

Abstract: In this paper we propose a Markov decision model with partial information for a queueing system with inter-arrival times following an exponential distribution, the service times following a PH distribution with Coxian configuration, and up to c servers working in the system. We assume that the system has finite capacity N, and customers that arrive when the system is full are rejected. We want to obtain an admissible policy that dynamically prescribes the number of active servers, based on the partially observed state of the system, in order to minimize the long run average cost. The cost of the system is composed of a holding cost, a service cost, an activation cost, a rejection cost, and a cost to keep the server activated.

Key words: *queueing system, Markov decision process, Phase-type distribution*

1 – Introdução

Neste artigo analisa-se um sistema de filas com c servidores, tempos entre chegadas exponencialmente distribuídos e tempos de serviço que seguem distribuições PH com configuração de Cox. Este sistema é modelado por um processo markoviano de decisão com informação parcial, no qual supõe-se que as fases de serviço não são observáveis.

Uma distribuição PH de ordem M com configuração de Cox é definida pelos parâmetros $\{M; \mu_1, \mu_2,...,\mu_M; p_1, p_2,..., p_{M-1}\}$, onde M é o número de estados transitórios, μ_k, $k = 1, 2,..., M$, são as taxas de transições das fases e p_k é a probabilidade do processo ir da fase k para a próxima fase $(k+1)$. Com probabilidade $1 - p_k$, $k = 1, 2,..., M-1$, o processo vai diretamente da fase k para o estado absorvente Ab. Na Figura 1 representa-se uma distribuição PH de ordem M com configuração de Cox.

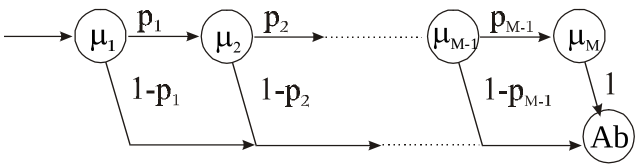

Figura 1 – Distribuição PH com configuração de Cox

Clientes que chegam no sistema entram em um fila de espera com disciplina FIFO (*first-come, first-served*), embora esta hipótese não seja necessária para o modelo proposto. Os tempos entre chegadas são variáveis aleatórias independentes e identicamente distribuídas segundo uma distribuição exponencial com taxa λ. Existem até c servidores iguais para atender os cliente; somente um servidor atende cada cliente. O tempos de serviço são independentes e identicamente distribuídos com distribuição PH com configuração de Cox e parâmetros $\{M; \mu_1, \mu_2,...,\mu_M; p_1, p_2,..., p_{M-1}\}$. Assume-se que o sistema tem capacidade finita N e que clientes que chegam quando o sistema está com sua capacidade preenchida são rejeitados.

O problema tratado é como alocar dinamicamente o número de servidores ativos no sistema. Baseado no estado do sistema, pode-se decidir por ativar um servidor quando um cliente chega e pode-se decidir por desativar um servidor quando um serviço termina.

O custo do sistema é composto de um custo de espera, um custo de serviço, um custo de ativação, um custo de rejeição e um custo de manter o servidor ativado.

O objetivo deste trabalho é obter uma política que dinamicamente prescreve o número de cliente ativados, baseado no estado observado do sistema, de maneira a minimizar o custo esperado a longo prazo do sistema, admitindo que, por hipótese, as fases dos serviços em curso não podem ser observadas. Para isso, o sistema é modelado por um Processo Markoviano de Decisão Parcialmente Observável.

Para a construção do modelo, primeiramente a restrição de não-observalidade das fases de serviço é relaxada e constrói-se um modelo markoviano de decisão (com informação total). Este modelo é apresentado na Seção 2. Em seguida, a restrição relaxada é considerada e, a partir do modelo com informação total, constrói-se o modelo com informação parcial proposto neste trabalho. Na seção 3 são apresentados este modelo, o algoritmo proposto em Hordijk & Loeve (1994), denominado de Algoritmo HL, que é usado para a obtenção de uma política de controle periódica e, finalmente, uma heurística que obtém uma política estacionária a partir dos resultados do Algoritmo HL. Nas Seções 4 e 5, são apresentados, respectivamente, alguns resultados numéricos fornecidos pelo modelo e os comentários finais.

2 – Modelo markoviano de decisão

O estado do sistema é definido pelo conjunto de valores $(n, s, ev, n_1, n_2,...,n_M)$ onde n é o número de clientes no sistema (incluindo os clientes que estão sendo atendidos), s é o número de servidores ativos, ev é o último evento, n_k é o número de clientes no sistema na fase de serviço k, k = 1,2,...,M, onde M é o número máximo de fases de serviço. O último evento (ev) pode ser uma chegada (Ar), o final de um serviço (Ser) ou o final de uma fase de serviço (Ph). A informação sobre o último evento foi introduzida no espaço de estados para definir o conjunto de possíveis ações em cada estado. Cada estado representa a configuração do sistema logo após a ocorrência de um evento e antes que uma decisão seja tomada.

O espaço de estados é dado por:

$E = \{ (n, s, ev, n_1, n_2,...,n_M) /$ $n = 0, 1,..., N;\ s = 0, 1,..., c;\ ev \in \{Ser, Ar, Ph\};$
$n_k = 0, 1,..., N;\ k = 1,...,M;$
$se\ \ ev = Ar\ então\ n \neq 0;$
$se\ (ev = Ar\ \ e\ \ s \geq n)\ então\ n_1 \neq 0;$
$se\ ((ev{=}Ar\ ou\ ev{=}Ph)\ e\ n \geq s)\ então\ n_1{+}n_2{+}...{+}n_M = s;$
$se\ ((ev{=}Ar\ ou\ ev = Ph)\ e\ n < s)\ então\ n_1{+}n_2{+}...{+}n_M = n;$
$se\ ev = Ser\ então\ s \neq 0\ e\ n < N;$
$se\ (ev{=}Ser\ e\ n < s\text{-}1)\ então\ n_1{+}n_2{+}...{+}n_M = n;$
$se\ (ev{=}Ser\ e\ n \geq s\text{-}1)\ então\ n_1{+}n_2{+}...{+}n_M = s\text{-}1 \}.$

Note que as restrições introduzidas no espaço de estados implicam que:

- logo após uma chegada ($ev = Ar$)
  - deve existir pelo menos um cliente no sistema ($n \neq 0$);
  - se existe pelo menos um servidor ocioso no sistema $(s \geq n)$, este deve começar a atender o novo cliente imediatamente, então deve existir pelo menos um cliente na primeira fase de serviço $(n_1 \neq 0)$;
  - se o número de clientes no sistema é maior ou igual ao número de servidores no sistema $(n \geq s)$, o número de serviço em curso ($n_1+n_2+...+n_M$) deve ser igual ao número de servidores ativados s;
  - se o número de clientes no sistema é menor que o ao número de servidores no sistema $(n < s)$, o número de serviço em curso ($n_1+n_2+...+n_M$) deve ser igual ao número de clientes no sistema;
- logo após o término de um serviço $(ev = Ser)$
  - pelo menos um servidor deve estar ativo ($s \neq 0$) (o servidor que executou o serviço) e a capacidade do sistema não pode estar preenchida ($n < N$) visto que um cliente acabou de deixar o sistema;
  - considerando que um cliente deixa o sistema imediatamente após o final de um serviço e se existem ou não clientes esperando a disponibilização de um servidor, restrições sobre o número de clientes e de serviços em curso, similares àquelas impostas nos instantes de chegada, devem ser consideradas.

Os instantes de decisão são os instantes de chegada, os instantes de final de serviço e os instantes de final de fase de serviço. Nos instantes de chegada, pode-se manter o mesmo número de servidores ativos (nenhuma ação) ou ativar mais um servidor, se existem servidores desativados. Nos instantes de final de serviço pode-se decidir por manter o mesmo número de servidores ativos ou desativar um servidor (isto é sempre possível, pois, nestes instantes, pelo menos um servidor está ativo – o servidor que terminou o serviço). Nos instantes de final de fase de serviço, a única ação possível é manter o mesmo número de servidores trabalhando no sistema (nenhuma ação). Considera-se observar o sistema no final de uma fase de serviço, mesmo quando em alguns sistemas isto não é possível. No modelo proposto neste trabalho, não se toma realmente qualquer ação nestes instantes, mas, por hipótese, o sistema deve ser observado nestes instantes para manter a propriedade markoviana, visto que estas fases são exponenciais. Então, para cada estado do sistema $i = (n, s, ev, n_1, n_2,..., n_M) \in E$, o espaço de possíveis ações é:

$$A(i) = \begin{cases} \{NA, Inc\} & se\ \ ev = Ar\ \ \ e\ \ \ s < c \\ \{NA, Dec\} & se\ \ ev = Ser \\ \{NA\} & caso\ contrário \end{cases}$$

onde a ação $a = NA$ significa manter o mesmo número de servidores ativos; $a = Inc$ significa ativar um servidor; e $a = Dec$ significa desativar um servidor.

Como o tempo entre chegadas é exponencialmente distribuído e os tempos de serviço seguem distribuições PH, que são construídas usando-se variáveis aleatórias exponenciais, pode-se modelar o sistema em estudo como um processo markoviano de decisão a tempo contínuo.

Para estes processos, dado que em um instante de decisão o sistema está no estado $i \in E$ e a ação $a \in A(i)$ é escolhida, definem-se:

$\tau_i(a)$ = o tempo esperado até o próximo instante de decisão;
$p_{ij}(a)$ = a probabilidade que no próximo instante de decisão o estado seja j $\in E$;
$C_i(a)$ = o custo esperado incorrido até o próximo instante de decisão.

A partir das taxas de transição $\Lambda_{ij}(a)$, obtém-se facilmente a taxa total de saída de cada estado dada por $\Lambda_i(a) = \sum_{j \neq i} \Lambda_{ij}(a)$. Note que $\Lambda_i(a)$ é o parâmetro da distribuição exponencial negativa que descreve o tempo de permanência no estado i quando a ação a é escolhida. Então, as probabilidades de transição são dadas por $p_{ij}(a) = \Lambda_{ij}(a)/\Lambda_i(a)$ e o tempo esperado entre transições é dado por $\tau_i(a) = 1/\Lambda_i(a)$.

Na Figura 2 apresenta-se o algoritmo gerador das transições do modelo. Neste algoritmo, se o estado do sistema é $i = (n, s, ev, n_1, n_2,...,n_M) \in E$, denomina-se $i.n$ a primeira coordenada do estado i, isto é, o número de clientes no sistema quando o sistema está no estado i; $i.s$ a segunda coordenada do estado i, isto é, o número de servidores ativos quando o sistema está no estado i; e assim sucessivamente. Denominação análoga é considerada para os estados $j, r \in E$.

```
Para cada estado i ∈ E e ação a ∈ A(i)
  início
    // Reagir a ação
    r ← i;
    se i.ev = Ar  e  a = Inc
      início
        r.s ← i.s + 1;
        se i.n_1+ i.n_2+...+ i.n_M < i.n
          r. n_1 ← i. n_1 + 1;
      fim
    se  i.ev = Ser
      início
        se  a = Dec
          r.s ← i.s - 1;
        senão  se i.n_1+ i.n_2+...+ i.n_M < i.n
          r. n_1 ← i. n_1 + 1;
      fim

    se  r.c < N  então
    //um cliente pode chegar e ser aceito
      início
        j ← r;
        j.n ← r.n+1;
        j.ev ← Ar;
        se  r.s>r.n  então  j.n_1← j.n_1+1;
        Transição de i para j com taxaλ;
      fim
    senão
      //um cliente pode chegar e ser rejeitado
      início
        j ← r;
        j.ev ← Ar;
        se  r.s>r.n  então  j.n_1← r.n_1+1;
        Transição de i para j com taxa λ;
      fim

  para k = 0 até M
    início
      se  r.n_k > 0  e  1-ps_k > 0 )
        início
          //um serviço pode terminar
          j ← r;
          j.n ← r.n-1;
          j.ev ← Ser;
          j.n_k ← r.n_k -1;
          Transição de i para j com taxa
                  r.n_k ×  (1-p_k)× μ_k;
        fim
      se  p_k > 0  então
        início
          //um serviço pode ir para a próxima
                  fase
          j ← r;
          j.ev ← Ph
          j.n_k ← r.n_k -1;
          j.n_k+1 ← r.n_k+1 +1;
          Transição de  i to j com taxa
                  r.n_k ×  p_k × μ_k;
        fim
    fim
  fim
```

Figura 2 Algoritmo gerador de transições

Para ilustrar o raciocínio usado no algoritmo gerador de transições, suponha que, em um instante de decisão, o estado do sistema é $i = (4,3,Ser,1,1,0,...,0)$ e a ação escolhida é $a = NA$. Neste caso, um dos cinco eventos pode ocorrer: (1) um cliente chega no sistema e o estado do sistema passa a ser $j_1 = (5,3,Ar,2,1,0,...,0)$ com taxa de transição $\Lambda_{ij}(a) = \lambda$; (2) um cliente vai para da primeira para a segunda fase de serviço e o estado passa a ser $j_2 = (4,3,Ph,1,2,0,...,0)$ com $\Lambda_{ij}(a) = 2p_1\mu_1$; (3) um cliente vai da segunda para a terceira fase de serviço e o estado passa a ser $j_3 = 4,3,Ph,2,0,1,...,0)$ com $\Lambda_{ij}(a) = p_2\mu_2$; (4) um servidor termina o serviço na primeira fase e o estado passa a ser $j_4 = (3,3,Ser,1,1,0,...,0)$ com $\Lambda_{ij}(a) = 2(1-p_1)\mu_1$; (5) um servidor termina o serviço na segunda fase e o estado do sistema passa a ser $j_5 = (3,3,Ser,2,0,0,...,0)$ com $\Lambda_{ij}(a) = (1-p_2)\mu_2$.

O custo esperado entre decisões $C_i(a)$ é composto por todos os custos incorridos ao sistema. Assim,

$$C_i(a) = C_H(i,a) + C_S(i,a) + C_{At}(i,a) + C_R(i,a) + C_{ON}(i,a),$$

onde $C_H(i,a)$, $C_S(i,a)$, $C_{ON}(i,a)$, $C_{At}(i,a)$, $C_R(i,a)$ e são, respectivamente, o custo de espera no sistema que é dado pelo produto da taxa $h > 0$ e o número de clientes no sistema (incluindo os clientes que estão sendo atendidos); o custo de serviço dado pelo produto da taxa $c_s > 0$ e o número de servidores atendendo os clientes; o custo de manter o servidor ativado dado pelo produto da taxa $c_{on} > 0$ e o número de servidores ativos; o custo constante de ativação incorrido quando um servidor é ativado e o custo de rejeição por cliente rejeitado incorridos até o próximo instante de decisão, dado que o estado $i = (n, s, ev, n_1, n_2,..., n_M) \in E$ e a ação $a \in A(i)$ foi adotada. Note que o custo de manter o servidor ativado é incorrido mesmo quando o servidor está ativado, mas ocioso. Definindo o número de servidores executando um serviço no estado i, imediatamente antes a uma tomada de ação, como $s_w(i)$, a expressão para estes custos são dadas por:

$$C_H(i,a) = \begin{cases} h.n.\tau_i(a) & se\ n > 0 \\ 0 & se\ n = 0 \end{cases} \qquad C_{At}(i,a) = \begin{cases} c_t & se\ a = Inc \\ 0 & caso\ contrário \end{cases}$$

$$C_R(i,a) = \begin{cases} c_R.\lambda.\tau_i(a), & se\ n = N \\ 0, & caso\ contrário \end{cases} \qquad C_{ON}(i,a) = \begin{cases} c_{on}.(s+1).\tau_i(a) & se\ a = Inc \\ c_{on}.s.\tau_i(a) & se\ a = NA \\ c_{on}.(s-1).\tau_i(a) & se\ a = Dec \end{cases}$$

$$C_S(i,a) = \begin{cases} c_s.[s_w(i)+1]\tau_i(a) & se\ ev = Ar\ \ e\ \ a = Inc\ \ e\ \ s_w(i) < n\ \ ou \\ & se\ ev = Ser\ \ e\ \ a = NA\ \ e\ \ s_w(i) < n \\ c_s.s_w(i).\tau_i(a) & caso\ contrário \end{cases}$$

Com os valores dos parâmetros $\tau_i(a)$, $p_{ij}(a)$ e $C_i(a)$, usando a teoria dos processos markovianos de decisão, pode-se obter uma política estacionária ótima para o sistema. Uma política estacionária R, definida pela regra de decisão $f{:}E\to A$, prescreve a ação $f(i)\in A(i)$ toda vez que o sistema é observado no estado $i \in E$. Em geral, as ações prescritas pela política estacionária pode depender do número de clientes em cada fase de serviço. Por hipótese, neste trabalho, estas informações não são observáveis e, desta forma, a decisão a ser tomada não pode ser baseada nelas. A fim de se obter uma política admissível, isto é, uma política que prescreve ações sem se basear nas informações sobre as fases, propõe-se um modelo com informação parcial.

Deve-se notar que, tanto a caracterização como a obtenção da política de controle ótima para o problema com informação parcial é um problema não resolvido para o tempo contínuo. Assim, propõe-se a utilização do custo da política estacionária obtida pelo modelo com informação total como um limitante inferior para o custo ótimo com informação parcial.

3 – Modelo com informação parcial

No modelo proposto neste trabalho, como não é possível observar as fases das distribuições PH, utilizam-se processos markovianos de decisão com informação parcial. Nestes processos, o espaço de estados E é particionado em subconjuntos $\{E_1, E_2,..., E_B\}$, onde $\bigcup_{b=1}^{B} E_b = E$ e $E_b \cap E_z = \emptyset$ se $b \neq z$, tal que no instante de decisão a única informação disponível é o subconjunto E_b no qual o estado está contido.

Então, o espaço de estados E é particionado em subconjuntos $E_{(n,s,ev)}$ onde $n = 0,1,...,N$; $s = 1,..., c$; $ev \in \{Ser, Ar, Ph\}$, tal que

$$E_{(n,s,ev)} = \{ (n', s', ev', n_1', n_2',..., n_M') \in E \,/ n' = n, s' = s, ev' = ev \}.$$

Para controlar o sistema em estudo, consideram-se políticas admissíveis. Sob estas políticas, observa-se o subconjunto E_b, $b = 1,2,...,B$, e escolhe-se uma ação, ou seja, para todos os estados pertencentes ao subconjunto E_b observado a mesma decisão deve ser tomada. Isto segue para todos os subconjuntos de estados pertencentes à partição.

Para se obter "boas" políticas admissíveis, neste trabalho, utiliza-se o Algoritmo HL. Para utilizar este algoritmo, além das políticas estacionárias, deve-se considerar as políticas markovianas periódicas. A política R é denominada periódica se $f^u = f^{u+L}$, para L fixo e todo u, e denotada por $(f^1,f^2,...,f^L)^\infty$, onde L é denominado período da política, f^u é uma regra de decisão admissível e $u \in \{1,2,...,L\}$ é denominada passo do período L.

O Algoritmo HL, que é um algoritmo baseado no Algoritmo de Iteração de Valores, foi desenvolvido por Hordijk & Loeve (1994) para otimizar os processos markovianos de decisão com informação parcial. Ao contrário dos processos markovianos de decisão a tempo contínuo, não se pode garantir a existência de uma política markoviana de custo mínimo para um processo markoviano de decisão com informação parcial, pois, como estes processos são um caso particular dos processos markovianos de decisão parcialmente observáveis, a política ótima pode depender de todo o histórico do processo. Portanto, este algoritmo busca uma "boa" política dentro da classe das políticas admissíveis.

Denotando-se por $P(f) = [p_{ij}(f)]_{i,j\in E}$, a matriz de transição do processo e $c(f)$ o vetor do custo até a próxima transição quando a regra de decisão f é usada, e V o vetor do custo esperado total mínimo, o Algoritmo HL é dado por:

<u>Algoritmo HL</u>

Escolher uma regra de decisão determinística admissível f^1 e $\varepsilon > 0$;

Escolher o vetor das probabilidades iniciais x^1 e $V^1 \in \Re^N$;

Para y = 1, 2, 3,... fazer

iniciar

Para cada b, b = 1, 2,..., B, calcular

$$g^y(b,a) \leftarrow \sum_{i\in E_s} x_i^y \left\{ C_i(a) + \sum_{j=1}^{N} p_{ij}(a) V_j^y \right\}, a \in A_b.$$

Para cada b, b = 1, 2,..., B, e para $\forall i \in E_b$

se $f^y(i)$ minimiza $g^y(b,.)$

então $f^{y+1}(i) \leftarrow f^y(i)$;

senão $f^{y+1}(i) \leftarrow a$ para algum $a \in arg\,min_{q\in A_b}\, g^n(b,q)$;

$x^{y+1} \leftarrow x^y P(f^{y+1})$;

$V^{y+1} \leftarrow c(f^{y+1}) + P(f^{y+1}) V^y$.

Verificar se critério de parada é atingido.

terminar

Critério de Parada: o algoritmo pára se para algum L e $m = y - L$ é assegurado que:

- $f^{u+L} = f^{u}$;
- $\left\| x^{u+L} - x^{u} \right\| < \varepsilon$ e
- $span(V^{u+L} - V^{u}) < \varepsilon$;

onde $span(V) = max_i\, V_i - min_i\, V_i$, para um vetor V, e $||x||$ é a norma de x, tal que $||x|| < \varepsilon$ para $x \in \Re^N$ significa que $|x_i| < \varepsilon$, $\forall i$.

O Algoritmo HL foi desenvolvido para modelos markovianos de decisão a tempo discreto. Para utilizá-lo em modelos a tempo contínuo, aplicou-se o método de uniformização apresentado em Tijms (1994).

Rodrigues & Carvalho (2001) detectaram alguns problemas quando se consideram políticas periódicas em modelos a tempo contínuo e propuseram a seguinte heurística para obtenção de uma política estacionária a partir da política periódica obtida pelo Algoritmo HL.

Heurística

Se a política obtida pelo Algoritmo HL for periódica, ou seja, se $R = (f^1, f^2, ..., f^{L-1}, f^L)^{\infty}$:

- extrair todas as políticas estacionárias possíveis a partir das regras de decisão da política periódica obtida, ou seja, $f^1, f^2, ..., f^{L-1}, f^L$;
- calcular o custo de cada uma destas políticas estacionárias;
- comparar os custos e escolher a política estacionária caracterizada pela regra de decisão de menor custo. A política assim obtida é admissível e será adotada como uma solução sub-ótima para o problema.

Ao aplicar a metodologia proposta no modelo em estudo, resultados numéricos mostram que, entre as políticas estacionárias admissíveis, uma classe de políticas, chamada política histerética multinível, parece ser muito adequada ao modelo, embora não seja provada sua otimalidade. De fato, em todos os testes realizados, a melhor política admissível encontrada pertencia a esta classe.

Uma política histerética multinível é caracterizada pelo conjunto de níveis $\{I_A(s), s=0,1,\ldots,c-1\} \cup \{I_S(s), s=1,2,\ldots,c\}$. Logo após uma chegada, a política prescreve ativar um servidor se existem s servidores ativos e o número de clientes no sistema excede $I_A(s)$, e, logo após o final de um serviço, a política prescreve desativar um servidor se existem s servidores ativos e o número de clientes no sistema é menor que $I_S(s)$.

Formalmente, para cada estado $i = (n, s, ev, , n_1, n_2, ..., n_M) \in E$, uma política histerética multinível $\{I_A(s), s=0,1,\ldots,c-1\} \cup \{I_S(s), s=1,2,\ldots,c\}$ prescreve a ação $f(i) \in A$, onde $f(i)$ é dada por:

$$f(i) = \begin{cases} Inc & se\ ev = Ar \quad e \quad n > I_A(s) \\ Dec & se\ ev = Ser \quad e \quad n < I_S(s) \\ NA & caso\ \ contrário \end{cases}$$

Em resumo, o método de solução do modelo proposto neste trabalho compõe-se do seguintes passos:

- resolver o modelo com informação total, apresentado na Seção 2, para obter um limitante inferior para o custo mínimo do modelo;
- utilizar o Algoritmo HL para a obtenção de uma política de controle admissível sub-ótima, possivelmente periódica;
- se o Algoritmo HL fornece uma política admissível periódica, utilizar a heurística proposta em Rodrigues & Carvalho (2001) para a obtenção de uma política admissível estacionária para o sistema;
- verificar se política obtida é histerética multinível.

4 – Exemplo numérico

A fim de analisar o modelo proposto, utilizaram-se os seguintes dados numéricos:
Número máximo de clientes no sistema: N =20
Número máximo de servidores: c = 3
Tempo médio entre chegadas: λ =1
Tempo de serviço de cada servidor: *média* = 1,5
h = 5 c_s = 20 c_t = 100
c_R = 100 c_{on} = 10
coeficiente de variação ao quadrado do tempo de serviço: c_s^2 = 0,125; 0,25; 0,5; 1; 2; 4 e 8

Considera-se para o tempo de serviço $c_s^2 \in \{0{,}125;\ 0{,}25;\ 0{,}5;\ 1;\ 2;\ 4 \text{ e } 8\}$. Para $c_s^2 < 1$, um método apresentado em Tijms (1994) foi usado para se obter os parâmetros das distribuições Erlang. Para $c_s^2 = 1$ uma distribuição exponencial (M) foi obtida. Para $c_s^2 > 1$ utilizou-se um método apresentado em Tijms (1994) para se obter as distribuições hiper-exponenciais com duas taxas, e utilizaram-se os resultados de Carvalho (1991) para convertê-las em distribuições PH com duas fases paralelas. Enfatiza-se que, em todas os sistemas de fila considerados, alterou-se o coeficiente de variação dos tempos de serviço sem alterar suas médias. Na Tabela 1 apresentam-se as distribuição PH obtidas.

c_s^2	Distribuição PH
0,125	$\mu_1 = \mu_2 = ... = \mu_8$ = 5,33333 $p_1 = p_2 = ... = p_7$ = 1
0,25	$\mu_1 = \mu_2 = \mu_3 = \mu_4$= 2,66667 $p_1 = p_2 = p_3$ = 1
0,5	$\mu_1 = \mu_2$ = 1,33333 p_1 = 1
1	μ_1 = 0,666667
2	μ_1 = 1,05157 μ_2 = 0,281766 p_1 = 0,154701
4	μ_1 = 1,18306 μ_2 = 0,150269 p_1 = 0,0983867
8	μ_1 = 1,25461 μ_2 = 0,0787219 p_1 = 0,0553368

Tabela 1 - Parâmetros das distribuições PH

Na Tabela 2 apresentam-se, para os dados considerados, os custos obtidos quando se resolveu o modelo com informação total. Para obtenção destes custos utilizou-se o Algoritmo de Iteração de Valores (Tijms, 1994). Nesta tabela, também se apresentam os custos e o período das políticas periódicas obtidas quando se utilizou o Algoritmo HL. As políticas periódicas com período 1 são, na realidade, políticas aperiódicas ou estacionárias.

Nas últimas duas colunas da Tabela 2, apresentam-se as políticas admissíveis estacionárias obtidas utilizando-se a heurística apresentada na seção anterior e os custos sob estas políticas. As políticas estacionárias apresentadas são políticas histeréticas multinível como definidas na seção anterior. Em todas as políticas estacionárias obtidas $I_s(1) = 0$ e $I_s(2) = 0$, isto significa que, logo após o final de um serviço, se o número de servidores ativos é menor ou igual a 2, a política nunca prescreve desativar um servidor.

5 – Comentários Finais

Neste trabalho trata-se do problema da otimização de sistemas de filas não markovianos. A metodologia proposta permite, ao se considerar distribuição PH para modelar os processos de decisão não markovianos, reduzir este problema a um processo markoviano de decisão com informação parcial. O modelo proposto neste trabalho pode ser visto como uma aproximação para sistemas M/G/c/N com número de servidores controlados.

Extensões futuras deste trabalho incluem a aplicação deste método a outros sistemas de filas, sistemas de computação e sistemas de produção e, também, refinar aspectos teóricos e computacionais do procedimento de otimização.

	Algoritmo de Iteração de Valores	Algoritmo HL		Heurística	
c_S^2	Custo Limitante inferior	Custo Política periódica	Período	Política estacionária histerética multinível s=0 s=1 s=2 s=3	Custo Política estacionária
0,125	62,3589	62,3762	18	I_A(s) 0 2 8 -- I_s(s) -- 0 0 3	62,3589
0,25	62,7166	62,7367	10	I_A(s) 0 2 7 -- I_s(s) -- 0 0 3	62,7391
0,5	63,3477	63,3703	38	I_A(s) 0 2 7 .. -- I_s(s) -- 0 0 3	63,3703
1	64,3505	64,3505	1	I_A(s) 0 2 7 -- I_s(s) -- 0 0 2	64,3505
2	65,2687	65,3354	1	I_A(s) 0 2 7 -- I_s(s) -- 0 0 2	65,3354
4	66,0335	66,2588	1	I_A(s) 0 3 6 -- I_s(s) -- 0 0 2	66,2588
8	66,8454	67,1901	1	I_A(s) 0 3 6 -- I_s(s) -- 0 0 1	67,1901

Tabela 2 – Resultados numéricos obtidos

6 – Referências

Carvalho, S.V., 1991. Modèles stochastiques appliqués a l´optimisation de la performance et de la sûreté de fonctionnement des systèmes de prodution. PhD thesis. Université Paul Sabatier, Toulouse, France.

Hordijk, A., Loeve, J.A., 1994. Undiscounted Markov decision chain with partial information: an algorithm for computing a locally optimal periodic policy. ZOR-Mathematical Methods of Operations Research 40, 163-181.

Rodrigues, R.C., Carvalho, S.V., 2001. Optimization of a PH/PH/1/N queue with controllable service time and partial information. International Transactions in Operational Research, 8(2): 235-248, March 2001. ISSN 0969-6016.

Tijms, H.C., (Ed.), 1994. Stochastic Modelling and Analysis: a Computational Approach. Wiley, New York.