

Universidade Federal de Minas Gerais
Instituto de Ciências Exatas
Departamento de Estatística

**Avaliação do Efeito da Topologia,
da Taxa de Chegada e do Número
de Nós no Tempo de Processamento
de Um Algoritmo para Otimização
em Redes Estocásticas**

F. G. Amorim & F. R. B. Cruz

Relatório Técnico
RTP-02/03

Relatório Técnico
Série Pesquisa

AVALIAÇÃO DO EFEITO DA TOPOLOGIA, DA TAXA DE CHEGADA E DO NÚMERO DE NÓS NO TEMPO DE PROCESSAMENTO DE UM ALGORITMO PARA OTIMIZAÇÃO EM REDES ESTOCÁSTICAS

F. G. Amorim & F. R. B. Cruz

Departamento de Estatística - ICEX - UFMG

31270-901 - Belo Horizonte - MG

E-mail: {famorim,fcruz}@ufmg.br

Resumo: Apresentamos resultados de experimentos computacionais realizados para testar o efeito da topologia, da taxa de chegada e do número de nós da rede no tempo de processamento de um algoritmo para determinação de alocação ótima de capacidades em redes de filas do tipo $M/G/C/C$ dependentes do estado, muito utilizadas para modelar o fenômeno de congestionamento. Para a análise de desempenho do algoritmo considerado, foi utilizado como ferramenta estatística o modelo fatorial com três fatores. Com base nos resultados, confirmamos que o algoritmo de fato recebe influência destes fatores, mas não apresenta um tempo de processamento explosivo, com o tamanho da rede, que pudesse inviabilizar sua aplicabilidade a problemas reais.

Palavras-chave: Teoria de filas; sistemas estocásticos; redes de filas dependentes do estado.

Abstract: In this paper, we present computational experiments designed to evaluate the effects that the topology, the arrival rate, and the number of nodes of a network may have in the processing time of a service and capacity allocation algorithm for state-dependent $M/G/C/C$ queueing networks. In order to evaluate its performance, we test the algorithm making use of well known and well established statistical tools. The results indicate that the algorithm is affected by the above factors but it does not have an explosive processing time on the size of the network that could compromise its applicability to real world concern problems.

Keywords: Queue theory, stochastic systems, state-dependent queueing networks.

1. Introdução

Há um grande interesse em se investigar o comportamento de um conjunto particular de redes de filas (ver Figura 1) do tipo $M/G/C/C$ dependentes do estado, apropriado para avaliar o congestionamento em sistemas de tráfego de veículos e pedestres, assim como outros sistemas de serviço e manufatura (Yuhaski & Macgregor Smith, 1989; Cheah & Macgregor Smith, 1994). Adotando-se a notação de Kendall, M indica que a chegada é um processo Markoviano, G representa um serviço com distribuição geral e dependente do estado, C se refere ao número de servidores e, por fim, assume-se que a capacidade do sistema está restrita a C usuários. A principal característica do modelo de redes de filas aqui tratado é o decaimento da taxa de serviço com o aumento do número de usuários no sistema, conforme ilustrado na Figura 2, para tráfego de pedestres.

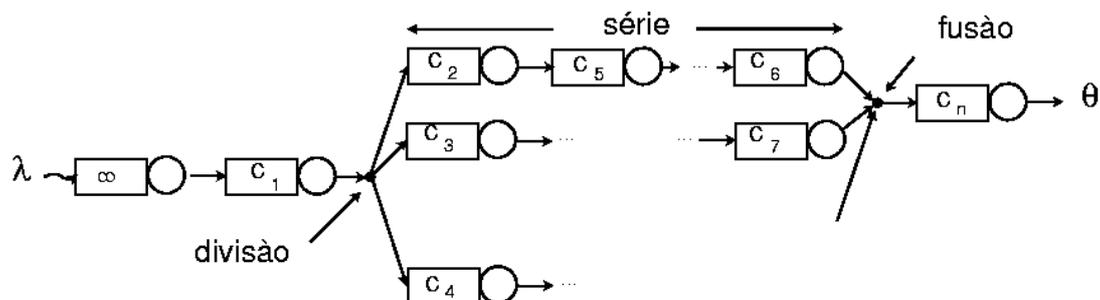


Figura 1: Rede de filas em topologia genérica.

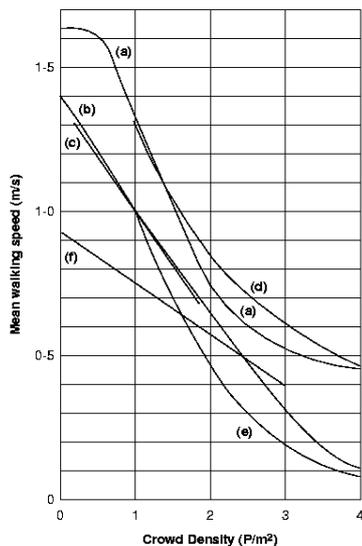


Figura 2: Curvas empíricas de Tregenza (1976).

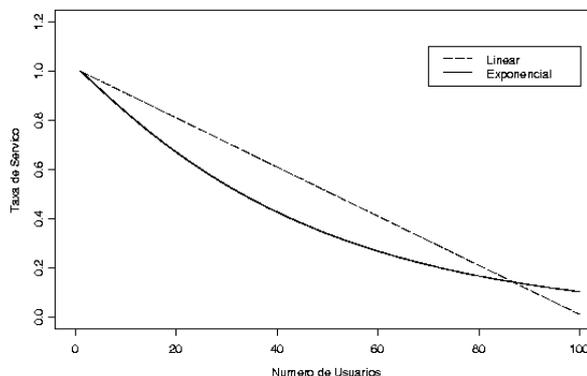


Figura 3: Taxa de serviço $f(n)$ versus número de usuários n para um corredor de $8 \times 2,5$ m.

Embora existam outras aplicações, estaremos considerando apenas o tráfego de pedestres, sobre o qual existem vários estudos a respeito da dependência que a velocidade de deslocamento tem com o incremento do número de usuários servidos. Um bom exemplo de tais estudos, aplicado a interiores de prédios, diz respeito à definição das rotas para as saídas de incêndio (MacGregor Smith, 1991).

Em áreas de circulação de pedestres, vários fatores podem ser apontados como responsáveis por diferentes velocidades de deslocamento. Por exemplo, adultos tendem a andar mais rápido do que crianças, e analogias semelhantes podem ser encontradas em diferentes grupos. Quando a densidade do tráfego aumenta, estes fatores têm um efeito grande sobre a velocidade de caminhada de pedestres individuais. Portanto, dentro de um corredor, o movimento mais rápido de pedestres tende a tornar-se mais lento à medida em que o espaço do corredor é ocupado por mais e mais pedestres, resultando em um fluxo com características dependentes do estado.

Usando-se redes $M/G/C/C$ dependentes do estados, algoritmos foram desenvolvidos para alocação de serviço e capacidades em áreas de circulação de pedestres, baseados em modelos de programação matemática inteira (Cruz et al., 2002). O objetivo desse artigo é apresentar resultados da avaliação de desempenho de um algoritmo para determinação de alocação ótima de capacidades nestas redes, analisando os efeitos da topologia, da taxa de chegada, do número de nós, bem como de suas interações, no tempo de processamento.

Na Seção 2, apresentamos a formalização do problema de alocação de serviços e capacidades, através de um modelo de programação matemática inteira, bem como os modelos estocásticos associados, incluindo a modelagem de uma única fila $M/G/C/C$ (corredor), os modelos de congestionamento, o algoritmo para estimação de medidas de desempenho e, finalmente, o algoritmo de alocação de capacidades cujo desempenho estamos interessados em conhecer. A Seção 3 apresenta o planejamento do experimento e os resultados obtidos para cada topologia, taxa de chegada e número de nós. Já na Seção 4, apresentamos uma análise descritiva do efeito exercido pelos fatores considerados sobre o tempo de processamento. Na Seção 5, mostramos os resultados alcançados através da análise de variância e das comparações múltiplas. A seção 6, com algumas observações finais, encerra o artigo.

2. Modelo de otimização

2.1. Formulação matemática

Assumimos que a topologia de rede seja conhecida de antemão, definida em um grafo $G(N,A)$, no qual N é o conjunto de nós (corredores) e A é o conjunto de arcos (conexões entre pares de nós). O problema de alocação de serviços e capacidades (SCA) se preocupa em determinar a mínima capacidade c_i que deveria ser alocada à i -ésima fila da rede, $\forall i \in N$, para que a probabilidade de bloqueio nesta fila esteja abaixo de um limiar ε previamente especificado, isto é, $P(C_i=c_i)=p(c_i) \leq \varepsilon$, com C_i representando o número (aleatório) de usuários na fila.

Uma possível formulação matemática para o problema SCA é a seguinte (Cruz et al., 2002):

$$z = \min \left[g(\mathbf{c}) = \sum_{\forall i \in N} f_i c_i \right], \quad (1)$$

s.a.:

$$p(c_i) \leq \varepsilon, \quad \forall i \in N, \quad (2)$$

$$c_i \in \{0, 1, \dots\}, \quad \forall i \in N, \quad (3)$$

que minimiza, em uma topologia qualquer, o custo total de alocação $\sum_i f_i c_i$, restrito ao provimento de uma probabilidade de bloqueio máxima ε , em todos os nós do grafo.

2.2. Modelo analítico de um único corredor

Para um único corredor, o tempo de ocupação é igual ao tempo gasto para que um pedestre o atravesse. A taxa de serviço $f(n)$ depende do número de ocupantes n e segue uma distribuição geral G dependente do estado. Considera-se, ainda, que a densidade de aglomeração de pedestres dentro do sistema está aproximadamente distribuída de maneira uniforme e que essa densidade determina a velocidade média de um indivíduo. Assim, se existirem n pedestres dentro de um corredor, esses terão uma taxa de serviço $f(n)$, enquanto que numa determinada chegada ou saída, essa taxa de serviço será alterada para $f(n+1)$ ou $f(n-1)$, respectivamente.

Sob esse ponto de vista, assumindo o modelo de filas $M/G/C/C$ dependentes do estado, pode-se expressar a distribuição de probabilidades do número de pedestres por (Cheah & MacGregor Smith, 1994):

$$p(n) = P(C = n) = \left[\frac{[\lambda(E(T_s))]^n}{n! f(n) \dots f(2) f(1)} \right] \times p(0), n = 1, 2, \dots, c, \quad (4)$$

sendo

$$p(0) = P(C = 0) = 1 / \left\{ 1 + \sum_{i=1}^c \left[\frac{[\lambda(E(T_s))]^i}{i! f(i) \dots f(2) f(1)} \right] \right\} \quad (5)$$

em que $p(0)$ é a probabilidade de não se ter pedestre no sistema, λ é a taxa de chegada, $E(T_s)$ é o tempo esperado de atendimento considerando-se apenas um único usuários no sistema e $f(n)$ é a taxa de serviço para n usuários simultâneos no sistema.

Utilizando-se a Equação (4), é possível calcular medidas de desempenho. Essas medidas incluem, entre outras, a probabilidade de bloqueio, a taxa de atendimento, o número médio de usuários e o tempo médio no sistema. O bloqueio ocorre quando o número de usuários C for igual à capacidade máxima c do sistema.

A taxa de atendimento pode ser denotada por:

$$\theta = \lambda[1 - p(c)]. \quad (6)$$

O número médio de usuários no sistema é:

$$E(Q) = \sum_{n=1}^c np(n). \quad (7)$$

O tempo médio no sistema pode ser dado por:

$$E(T_w) = \frac{E(Q)}{\theta}. \quad (8)$$

2.3. Modelos de congestionamento para corredores

Precisamos agora definir as taxas de serviço $f(n)$. Estaremos utilizando as seguintes definições:

- $V_n \rightarrow$ velocidade média para uma ocupação de n pedestres;
- $V_1 \rightarrow$ velocidade média de um pedestre sozinho, que corresponde a 1,5 m/s;
- $V_a \rightarrow$ velocidade média quando a densidade de aglomeração é $a=2$ ped/m²;
- $V_b \rightarrow$ velocidade média quando a densidade de aglomeração é de $b=4$ ped/m²;
- γ e $\beta \rightarrow$ parâmetros de forma e escala para o modelo exponencial;
- $a=2 \times l \times w$;
- $b=4 \times l \times w$.

Utilizando-se o modelo de filas $M/G/C/C$ dependentes do estado, um corredor pode ser considerado como servidor para seus ocupantes. O número de servidores em paralelo é igual à capacidade do corredor, que também representa o número total de usuários permitidos simultaneamente no sistema, e é dado por:

$$c = \lfloor 5 \times l \times w \rfloor, \quad (9)$$

na qual l é o comprimento do corredor, w é a largura e c , sua capacidade, pois, de acordo com estudos realizados por Tregenza (1976), pôde-se constatar que 5 ped/m² representa a máxima densidade de aglomeração admissível em uma área de circulação. Desta forma, o tempo esperado de atendimento considerando-se apenas um único usuário no sistema, utilizado nas Eq. (1) e (2), pode ser dado por $E(T_s) = l/V_1$.

No modelo de congestionamento, o fluxo de tráfego uni-direcional através de corredores segue uma velocidade média de caminhar de um pedestre V_n , como uma função do número de indivíduos n que ocupam o corredor e sua capacidade, c . Baseado nessa e em outras observações empíricas, modelos analíticos linear e exponencial foram desenvolvidos para essa velocidade (uni-direcional) em função do número de pedestres em um corredor (Yuhaski & MacGregor Smith, 1989).

De acordo com Tregenza (1976), a velocidade de um único ocupante no corredor, V_1 , é cerca de 1,5 m/s. Assumindo que uma população de $n=c+1$ não é possível, V_n é zero para todo $n \geq c+1$. Uma relação linear que satisfaz a essas condições é como segue:

$$V_n = \frac{V_1}{c}(c+1-n). \quad (10)$$

Um modelo exponencial parece fornecer uma aproximação mais exata para a média da velocidade de um pedestre *versus* a densidade de aglomeração, conforme o formato das curvas apresentadas por Tregenza (1976), Figura 2. O modelo exponencial de fluxo uni-direcional, proposto por Yuhaski e Smith (1989) é o seguinte:

$$V_n = V_1 \exp \left[- \left(\frac{n-1}{\beta} \right)^\gamma \right], \quad (11)$$

no qual:

$$\gamma = \ln \left[\frac{\ln(V_a/V_1)}{\ln(V_b/V_1)} \right] / \ln \left(\frac{a-1}{b-1} \right), \quad (12)$$

e

$$\beta = \frac{a-1}{[\ln(V_1/V_a)]^{\frac{1}{\gamma}}} = \frac{b-1}{[\ln(V_1/V_b)]^{\frac{1}{\gamma}}}. \quad (13)$$

A Figura 3 mostra o comportamento da taxa de serviço para pedestres, $f(n)=V_n/V_1$, com o aumento da densidade, para fluxos unidirecionais, considerando-se os modelos linear e exponencial.

2.4. Análise de desempenho em redes de corredores

O problema de determinação das medidas de desempenho em redes de filas fica consideravelmente mais complexo. Note que na topologia apresentada na Figura 1, a probabilidade de bloqueio do i -ésimo nó depende de todo o fluxo anterior e também das probabilidades de bloqueio dos nós seguintes. O Método da Expansão Generalizado (GEM) é um algoritmo que permite lidar com tais redes. É uma combinação de métodos de aproximações sucessivas e decomposição nó-a-nó. Este método tem como característica chave a colocação de um nó artificial antes de cada fila finita no sistema, a fim de registrar os pedestres bloqueados que tentam entrar num corredor finito quando este está em seu limite. Desta forma, a rede de filas é “expandida” em uma equivalente rede Jackson, na qual cada nó pode ser decomposto e analisado separadamente. O método de expansão é composto por três estágios, a saber:

- Estágio I - reconfiguração da rede: antes de cada um dos nós com capacidade finita, é adicionado um nó artificial. Os usuários são redirecionados para esse compartimento, quando o próximo nó estiver na sua capacidade.
- Estágio II - estimação dos parâmetros: essa fase é dedicada ao cálculo das estimativas dos parâmetros envolvidos na análise.
- Estágio III - eliminação do retorno: nesse momento, é realizada uma reconfiguração do nó artificial, de maneira que as dependências nos processos de chegada causadas por visitas repetidas a esses ambientes sejam retiradas.

O GEM pode ser estendido para qualquer topologia, como por exemplo, para a análise de topologias complexas como a apresentada na Figura 1. Adaptações do GEM a topologias genéricas de redes $M/G/C/C$ podem ser encontradas no trabalho de Cruz e MacGregor Smith (2002).

2.5. Algoritmo de resolução

O algoritmo para resolução do problema SCA é apresentado na Figura 4 (Cruz et al. 2002). O algoritmo principal implementa uma variação do método de busca por coordenadas. Assim o algoritmo lê as configurações iniciais e então é alocada uma capacidade inicial que satisfaça as restrições, ou seja, que dê uma probabilidade de bloqueio bem inferior ao limiar dado. Por conveniência, a capacidade inicial é dada na forma 2^M , na qual M é um número "grande".

Na busca local, o algoritmo escolhe um nó i (corredor) e para ele define dois limites para a sua capacidade, que, para facilitar a busca, é sempre potência de 2. O primeiro valor é denominado limite inferior c_i^{inf} , uma capacidade que com certeza *não* atende às restrições de as probabilidades de bloqueio serem menores que o limiar dado. Já c_i^{sup} é um limite superior, isto é, um valor que garante pelo menos a viabilidade da solução. Daí o algoritmo começa a redução do intervalo tirando a média dos dois limites, determinando $c_i^{\text{can}} = (c_i^{\text{sup}} + c_i^{\text{inf}})/2$. Se a probabilidade de bloqueio considerando-se o valor c_i^{can} for menor que o limiar ε , então isto significa que o limite superior c_i^{sup} está superestimado e pode ser atualizado para o valor c_i^{can} . Caso contrário, o limite inferior c_i^{inf} está subestimado, podendo portanto ser atualizado para o valor c_i^{can} . E assim é feito até que a diferença entre estes dois limites seja menor ou igual à unidade. Então, o valor dado para o limite superior c_i^{sup} é a menor capacidade para a qual a probabilidade de bloqueio é igual ou inferior ao limiar ε .

```

algorithm
  leia  $G(N,A)$ 
  leia as probabilidades de roteamento  $p_{ij}, \forall (i,j) \in A$ 
  leia as taxas de chegada  $\lambda_i, \forall i \in N$ 
  /* encontre um vetor inicial viável */
  for  $\forall i \in N$  do
     $c_i^{\text{opt}} \leftarrow c_i^{\text{sup}} \leftarrow 2^M$ 
  end for
  /* encontre uma solução ótima */
  iter  $\leftarrow 0$ 
  repeat
    iter  $\leftarrow$  iter + 1
    /* otimize a  $i$ -ésima fila */
    for  $\forall i \in N$  do
      OptQueue( $i, c^{\text{sup}}$ )
    end for
    /* atualize para a melhor solução */
    if  $g(c^{\text{sup}}) < g(c^{\text{opt}})$  then
       $c^{\text{opt}} \leftarrow c^{\text{sup}}$ 
      tire rótulo de todos os nós
    else
      exit
    end if
  end repeat
  write  $c^{\text{opt}}$ 
end algorithm

```

a) otimização da rede

```

algorithm OptQueue( $i, c^{\text{sup}}$ )
  /* passo recursivo de rotulação */
  for  $\forall (j,i) \in A$  do
    if nó  $j$  está desrotulado then
      OptQueue( $j, c^{\text{sup}}$ )
      rotule  $no\ j$ 
    end if
  end for
  /* isole o ótimo */
   $j \leftarrow M_i$ 
   $c_i^{\text{inf}} \leftarrow c_i^{\text{sup}} \leftarrow 2^j$ 
  while  $p_i(c_i^{\text{sup}}) \geq \varepsilon, \forall i \in N$ 
     $j \leftarrow j + 1$ 
     $c_i^{\text{sup}} \leftarrow 2^j$ 
  end while
  /* reduzir o intervalo */
   $c^{\text{can}} \leftarrow c^{\text{sup}}$ 
  while  $(c_i^{\text{sup}} - c_i^{\text{inf}}) > 1$ 
     $c_i^{\text{can}} \leftarrow (c_i^{\text{inf}} + c_i^{\text{sup}})/2$ 
    if  $p_i(c_i^{\text{can}}) \leq \varepsilon, \forall i \in N$  then
       $c_i^{\text{sup}} \leftarrow c_i^{\text{can}}$ 
    else
       $c_i^{\text{inf}} \leftarrow c_i^{\text{can}}$ 
    end if
  end while
end algorithm

```

b) otimização de uma única fila

Figura 4: Algoritmo de resolução do problema SCA.

3. Planejamento do experimento

Os experimentos com o algoritmo de otimização foram realizados para as topologias básicas, série, divisão e fusão, com 3, 5 e 7 nós, considerando as taxas de chegada 1, 2, 4 e 8 pessoas/s. O comprimento dos corredores assumiu apenas o valor de 8,0 m. O linguagem C++ foi utilizada, devido principalmente à sua capacidade de abstração dos dados e à possibilidade de se produzir códigos eficientes e facilmente modificáveis.

Como ferramenta estatística, foi utilizado um modelo fatorial com três fatores fixos (maiores detalhes em Durmond et al., 1996). Todos os fatores foram cruzados entre si, num total de 36 tratamentos. Porém os dados não foram aleatorizados. Três fatores foram considerados. O fator A incluiu as

topologias séries, divisão e fusão. O fator B, as taxas de chegada de 1, 2, 4 e 8 pedestres/s. Finalmente, o fator C incluiu topologias com 3, 5 e 7 nós. A Figura 1 mostra as três topologias consideradas, para cinco nós. As tabelas que se seguem apresentam os resultados obtidos para cada topologia, cada taxa de chegada (λ) e número de nós.

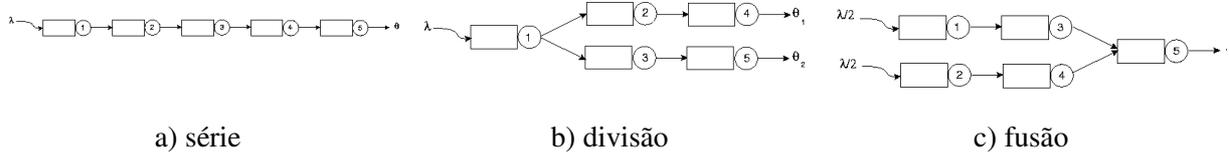


Figura 5: Topologias básicas consideradas, para cinco nós.

Através da Tabela 1 (topologia série) é possível perceber uma alocação crescente de capacidades ao longo da rede. Isto pode ser explicado pelo fato de que a probabilidade de bloqueio depende do fluxo dos corredores anteriores. Assim, para que o fluxo não tenha bloqueio, há a necessidade de aumentar as capacidades dos corredores do final da topologia.

Tabela 1: Resultados para a topologia série.

λ	c (ped)	w (m)	tempo(s)
1	(42;45;46)	(1,062;1,137;1,163)	0,43
	(42;45;46;47;48)	(1,062;1,137;1,163;1,188;1,213)	1,4
	(42;45;46;47;48;48;48)	(1,062;1,137;1,163;1,188;1,213;1,213;1,213)	2,9
2	(79;79;82)	(1,988;1,988;2,062)	0,61
	(79;79;82;83;83)	(1,988;1,988;2,062;2,087;2,087)	2,0
	(79;79;82;83;83;84;87)	(1,988;1,988;2,062;2,087;2,087;2,113;2,188)	4,1
4	(151;152;155)	(3,788;3,812;3,888)	1,0
	(151;152;155;157;158)	(3,788;3,812;3,888;3,938;3,963)	3,2
	(151;152;155;157;158;160;161)	(3,788;3,812;3,888;3,938;3,963;4,013;4,038)	6,3
8	(296;297;302)	(7,413;7,438;7,562)	1,7
	(296;297;302;302;305)	(7,413;7,438;7,562;7,562;7,638)	5,4
	(296;297;302;302;305;306;307)	(7,413;7,438;7,562;7,562;7,638;7,663;7,688)	11

Tabela 2: Resultados para a topologia divisão.

λ	c (ped)	w (m)	tempo(s)
1	(42;26;26)	(1,062;0,663;0,663)	0,34
	(42;26;26;28;28)	(1,062;0,663;0,663;0,713;0,713)	1,1
	(42;26;26;28;28;29;29)	(1,062;0,663;0,663;0,713;0,713;0,738;0,738)	2,2
2	(79;42;44)	(1,988;1,062;1,113)	0,47
	(79;42;44;47;48)	(1,988;1,062;1,113;1,188;1,213)	1,5
	(79;42;44;47;48;48;49)	(1,988;1,062;1,113;1,188;1,213;1,213;1,238)	3,0
4	(151;79;81)	(3,788;1,988;2,038)	0,70
	(151;79;81;81;83)	(3,788;1,988;2,038;2,038;2,087)	2,1
	(151;79;81;81;83;87;87)	(3,788;1,988;2,038;2,038;2,087;2,188;2,188)	4,1
8	(296;152;152)	(7,413;3,812;3,812)	1,2
	(296;152;152;156;160)	(7,413;3,812;3,812;3,913;4,013)	3,5
	(296;152;152;156;160;160;161)	(7,413;3,812;3,812;3,913;4,013;4,013;4,038)	6,7

Observando a Tabela 2, notamos que, para as duas seqüências resultantes da divisão, a soma da capacidade dos corredores é maior que a capacidade do corredor a partir do qual se dá a divisão, o que não era o esperado. Em outras palavras, para $\lambda=1$ e $n=3$ nós, a um corredor de capacidade 42 se

seguem dois com capacidade 26, cuja soma, 52, é superior a 42. Após a divisão, essas seqüências apresentam um padrão de alocação crescente, similar ao de uma topologia série.

Tabela 3: Resultados para a topologia fusão.

λ	c (ped)	w (m)	tempo(s)
1	(24;24;44)	(0,613;0,613;1,113)	0,31
	(24;24;25;25;45)	(0,613;0,613;0,638;0,638;1,137)	1,0
	(24;24;25;25;27;27;46)	(0,613;0,613;0,638;0,638;0,688;0,688;1,163)	2,1
2	(42;42;81)	(1,062;1,062;2,038)	0,41
	(42;42;45;45;83)	(1,062;1,062;1,137;1,137;2,087)	1,4
	(42;42;45;45;46;46;83)	(1,062;1,062;1,137;1,137;1,163;1,163;2,087)	2,9
4	(79;79;152)	(1,988;1,988;3,812)	0,60
	(79;79;79;79;154)	(1,988;1,988;1,988;1,988;3,863)	2,0
	(79;79;79;79;82;82;155)	(1,988;1,988;1,988;1,988;2,062;2,062;3,888)	4,1
8	(151;151;297)	(3,788;3,788;7,438)	0,86
	(151;151;152;152;300)	(3,788;3,788;3,812;3,812;7,513)	3,0
	(151;151;152;152;155;155;302)	(3,788;3,788;3,812;3,812;3,888;3,888;7,562)	6,3

Pela Tabela 3, verificamos que as duas seqüências que antecedem a fusão também apresentam um padrão de alocação similar ao de filas em série. Quando ocorre a fusão, a capacidade do corredor que resulta do encontro das duas seqüências é inferior à soma das capacidades dos antecessores, evidenciando um surpreendente efeito de economia de escala.

4. Análise descritiva

Com o objetivo de obter indicações iniciais sobre a forma do efeito exercido pelos fatores sobre o tempo de processamento, foram construídos box-plots estratificados pela topologia, taxa de chegada e número de nós, bem como o gráfico de efeitos principais e suas interações.

A partir da análise da Figura 6 foi possível notar que o tempo de processamento parece não ser afetado pela topologia, o que não ocorre no gráfico de efeitos principais da topologia (Figura 9), no qual a topologia série parece se diferir das demais. Por outro lado, observando a Figura 7, a taxa de chegada parece influenciar a variável resposta (o tempo de processamento), sendo que quanto maior essa taxa, maior o tempo de processamento. O número de nós (Figura 8) também parece afetar a variável resposta, a qual possui um comportamento ascendente à medida que aumenta o número de nós. Na Figura 10, as interações topologia *versus* taxa de chegada e topologia *versus* número de nós parecem ser significativas. Não parece, entretanto, ser o caso da interação taxa de chegada *versus* número de nós, para o qual as linhas parecem ser paralelas.

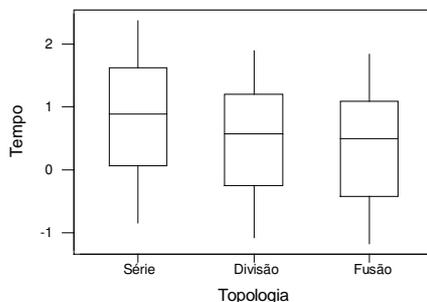


Figura 6: Box-plot para a topologia.

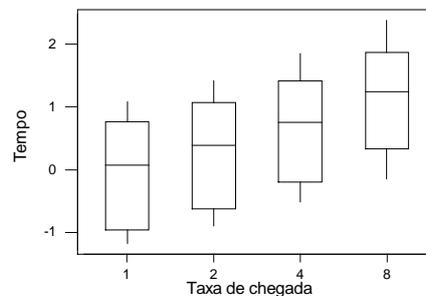


Figura 7: Box-plot para a taxa de chegada λ .

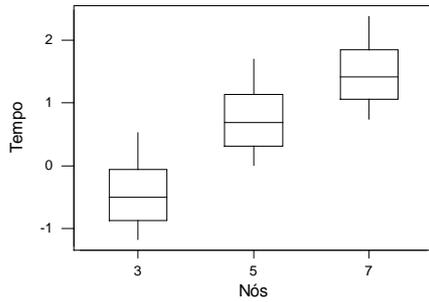


Figura 8: Box-plot para o número de nós.

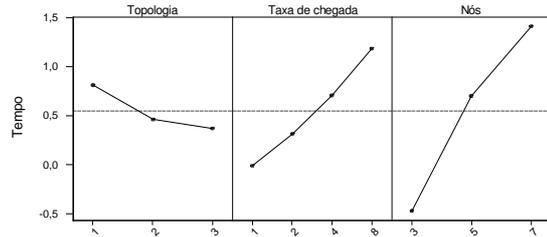


Figura 9: Gráfico de efeitos principais.

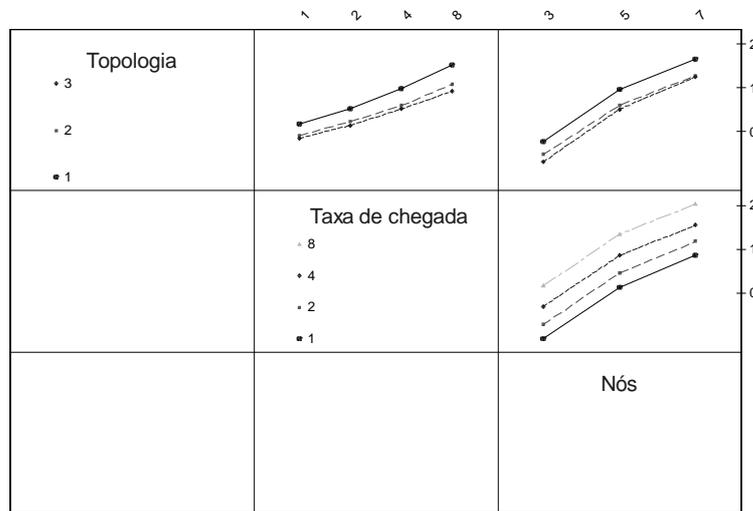


Figura 10: Gráfico de interação entre os fatores.

5. Análise de variância

Na análise de variância, os três fatores A, B e C são de interesse, bem como a possível interação entre eles. Interações de maior ordem foram desprezadas durante a análise pelo fato de o experimento não ter replicações, o que não causou grandes impactos. Por apresentar o quadrado médio (QM) inferior ao das demais interações, esta interação foi considerada insignificante. O tempo de processamento foi considerado como variável resposta de interesse. A transformação logarítmica foi utilizada para tornar válido o modelo, sendo uma das mais utilizadas em situações cuja resposta é o tempo até a ocorrência de um evento. Em outras palavras, a variável resposta considerada foi o logaritmo dos tempos de processamento. Feito isto, foi ajustado o seguinte modelo linear, adequado ao delineamento experimental anteriormente descrito:

$$Y_{ijkl} = \mu + \tau_i + \beta_j + \gamma_k + (\tau\beta)_{ij} + (\tau\gamma)_{ik} + (\beta\gamma)_{jk} + \varepsilon_{ijkl} \quad (14)$$

onde μ é o efeito geral da média, τ_i é o efeito do fator A, β_j é o efeito do fator B, γ_k é o efeito do fator C, $(\tau\beta)_{ij}$ é o efeito de interação entre os fatores A e B, $(\tau\gamma)_{ik}$ é o efeito de interação entre os fatores A e C, $(\beta\gamma)_{jk}$ é o efeito de interação entre os fatores B e C e ε_{ijkl} é o erro aleatório, independente, com distribuição $N(0, \sigma^2)$, para $i=1, \dots, 3; j=1, \dots, 4; k=1, \dots, 3; l=1$.

As hipóteses testadas foram:

$H_0: \tau_1 = \tau_2 = \tau_3 = 0$	v.s.	$H_1: \tau_i \neq 0$ para pelo menos um i
$H_0: \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$	v.s.	$H_1: \beta_j \neq 0$ para pelo menos um j
$H_0: \Upsilon_1 = \Upsilon_2 = \Upsilon_3 = 0$	v.s.	$H_1: \Upsilon_k \neq 0$ para pelo menos um k
H_0 : Não há efeito de interação entre A e B	v.s.	H_1 : Há efeito de interação entre A e B
H_0 : Não há efeito de interação entre A e C	v.s.	H_1 : Há efeito de interação entre A e C
H_0 : Não há efeito de interação entre B e C	v.s.	H_1 : Há efeito de interação entre B e C

As suposições de normalidade e homocedasticidade associadas ao modelo foram verificadas e confirmadas. A suposição de independência não foi verificada, pois não temos a ordem de coleta dos dados. Na Tabela 4, apresentamos o resultado do ajuste do modelo obtido utilizando o MINITAB® (Minitab Inc., 2000). Pela coluna de valores-p, notamos que tanto as topologias quanto as taxas de chegada e os nós têm um efeito significativo no tempo de processamento, assim como as interações entre eles, com exceção da interação entre a taxa de chegada e o número de nós, que não foi significativa (valor- $p > 0,05$).

Tabela 4: Análise de variâncias.

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Topologi	2	1,3025	1,3025	0,6513	1433,03	0,000
Taxa_che	3	7,1608	7,1608	2,3869	5252,14	0,000
Nós	2	21,6076	21,6076	10,8038	24000,00	0,000
Topologi*Taxa_che	6	0,0670	0,0670	0,0112	24,56	0,000
Topologi*Nós	4	0,0199	0,0199	0,0050	10,94	0,001
Taxa_che*Nós	6	0,0026	0,0026	0,0004	0,94	0,505
Error	12	0,0055	0,0055	0,0005		
Total	35	30,1657				

Prosseguindo, utilizamos o método de comparações múltiplas de Tukey (detalhes em Werkema & Aguiar, 1996), através de intervalos de 95% de confiança (vide Anexo) para verificar a natureza desses efeitos. Os resultados estão apresentados na Figura 11, através da qual foi possível concluir com 95% de confiança, que as médias do tempo de processamento nas topologias fusão e divisão, ao utilizar a taxa de chegada 2, e na topologia série, ao utilizar a taxa de chegada 1, não se diferem. O mesmo ocorre para a taxa de chegada 4 nas topologias fusão e divisão e para a taxa de chegada 2 na topologia série. As topologias fusão e série, ao utilizar as taxas de chegada 8 e 4, respectivamente, também não apresentam diferenças estatisticamente significativas. Para o conjunto de 7 nós, as médias do tempo de processamento das topologias divisão e fusão são equivalentes.

Topologia X Taxa de chegada (λ)					
TOP _{fusão} λ_1	TOP _{divisão} λ_1	TOP _{fusão} λ_2	TOP _{série} λ_1	TOP _{divisão} λ_2	TOP _{fusão} λ_4

TOP _{série} λ_2	TOP _{divisão} λ_4	TOP _{fusão} λ_8	TOP _{série} λ_4	TOP _{divisão} λ_8	TOP _{série} λ_8

Topologia X Número de nós					
TOP _{fusão} Nó ₃	TOP _{divisão} Nó ₃	TOP _{série} Nó ₃	TOP _{fusão} Nó ₅	TOP _{divisão} Nó ₅	
TOP _{série} Nó ₅	TOP _{fusão} Nó ₇	TOP _{divisão} Nó ₇	TOP _{série} Nó ₇		

Figura 11: Comparações múltiplas de Tukey.

Vale destacar que tanto as taxas de chegada quanto os conjuntos de nós se diferem estatisticamente em todas as topologias. Portanto, o menor tempo de processamento pode ser obtido na topologia fusão, para 3 nós e taxa de chegada λ igual a 1 e o maior tempo é alcançado na topologia série, para 7 nós e taxa de chegada (λ) igual a 8.

6. Conclusões e observações finais

Apresentamos os resultados obtidos em experimentos computacionais com um algoritmos para alocação de serviços e capacidades em uma rede de filas $M/G/C/C$ dependentes do estado, úteis para modelar sistemas sujeitos a congestionamento. Os experimentos foram planejados e estudados por meio de conhecidas ferramentas estatísticas. Resultados inéditos foram obtidos para diversas topologias, levando-se em consideração diferentes taxas de chegada (λ) e números de nós da rede. Avaliamos o efeito de tais fatores sobre o tempo de processamento desse algoritmo através de um modelo fatorial com três fatores fixos. Os resultados evidenciaram a influência desses fatores e determinaram um padrão de alocação ótima inesperado para as capacidades em redes de filas $M/G/C/C$ dependentes do estado.

Agradecimentos

A pesquisa de Frederico R. B. Cruz é financiada com recursos do *Conselho Nacional de Desenvolvimento Científico e Tecnológico*, CNPq, processos 301809/96-8 e 201046/94-6, da *Fundação de Amparo à Pesquisa do Estado de Minas Gerais*, FAPEMIG, processos CEX-289/98 e CEX-855/98, e pela Pró-Reitoria de Pesquisa da UFMG, PRPq-UFMG. Flávia G. Amorim é bolsista de iniciação científica pela FAPEMIG.

Referências bibliográficas

- (1) Cheah, J. & Macgregor Smith, J. (1990). Generalized $M/G/C/C$ State Dependent Queueing Networks, *Queueing Systems*, 15:365-386.
- (2) Cruz, F.R.B. e MacGregor Smith, J. (2002). Analysis of generalized $M/G/C/C$ state dependent queueing networks, *manuscrito*. URL <ftp://ftp.est.ufmg.br/pub/fcruz/pub/ana.pdf>
- (3) Cruz, F.R.B; MacGregor Smith, J.; Queiroz, D.C e Oliveira, P.C. (2002). Service and Capacity Allocation in $M/G/C/C$ State Dependent Queueing Networks, *manuscrito*. URL <ftp://ftp.est.ufmg.br/pub/fcruz/pub/opt.pdf>
- (4) Drumond, F.B.; Werkema, M.C.C. e Aguiar, S. (1996) *Análise de Variância: Comparação de Várias Situações*, Editora Littera Maciel Ltda., Contagem, MG.
- (5) MacGregor Smith, J. (1991). State Dependent Queueing Models in Emergency Evacuation Networks, *Transportation Research Part B*, 25:373-389.
- (6) Minitab Inc. (2000). *MINITAB User's Guide 1 and 2*. url: <http://www.minitab.com>.
- (7) Tregenza, P.R. (1976). *The Design of Interior Circulation*, Van Nostrand Reinhold Company, New York, USA.
- (8) Werkema, M.C.C. e Aguiar, S. (1996). *Planejamento e Análise de Experimentos: Como Identificar as Principais Variáveis Influentes em um Processo*, Editora Littera Maciel Ltda., Contagem, MG.
- (9) Yuhaski, S.J. e MacGregor Smith, J. (1989). Modeling circulation systems in buildings using state dependent models. *Queueing Systems*, 4:319-338.

Anexo A

As Figuras 11 a 14 se referem às comparações múltiplas de Tukey, utilizadas para comparar todos os pares de médias significativas envolvidas na análise de variância. A Figura 15 e a Figura 16 confirmam a adequação do modelo de análise de variância.

Topologi = 1			
Nós = 3 subtracted from:			
Topologi*Nós	Lower	Center	Upper
1 5	1,1228	1,1789	1,2351
1 7	1,8323	1,8884	1,9446
2 3	-0,3579	-0,3017	-0,2455
2 5	0,7707	0,8269	0,8830
2 7	1,4463	1,5024	1,5586
3 3	-0,5263	-0,4701	-0,4139
3 5	0,6787	0,7349	0,7911
3 7	1,4175	1,4737	1,5299
Topologi = 1			
Nós = 5 subtracted from:			
Topologi*Nós	Lower	Center	Upper
1 7	0,653	0,709	0,766
2 3	-1,537	-1,481	-1,424
2 5	-0,408	-0,352	-0,296
2 7	0,267	0,323	0,380
3 3	-1,705	-1,649	-1,593
3 5	-0,500	-0,444	-0,388
3 7	0,239	0,295	0,351
Topologi = 1			
Nós = 7 subtracted from:			
Topologi*Nós	Lower	Center	Upper
2 3	-2,246	-2,190	-2,134
2 5	-1,118	-1,062	-1,005
2 7	-0,442	-0,386	-0,330
3 3	-2,415	-2,359	-2,302
3 5	-1,210	-1,154	-1,097
3 7	-0,471	-0,415	-0,359
Topologi = 2			
Nós = 3 subtracted from:			
Topologi*Nós	Lower	Center	Upper
2 5	1,0724	1,1286	1,1847
2 7	1,7480	1,8041	1,8603
3 3	-0,2246	-0,1684	-0,1122
3 5	0,9804	1,0366	1,0928
3 7	1,7192	1,7754	1,8315
Topologi = 2			
Nós = 5 subtracted from:			
Topologi*Nós	Lower	Center	Upper
2 7	0,619	0,676	0,732
3 3	-1,353	-1,297	-1,241
3 5	-0,148	-0,092	-0,036
3 7	0,591	0,647	0,703
Topologi = 2			
Nós = 7 subtracted from:			
Topologi*Nós	Lower	Center	Upper
3 3	-2,029	-1,973	-1,916
3 5	-0,824	-0,768	-0,711
3 7	-0,085	-0,029	0,027
Topologi = 3			
Nós = 3 subtracted from:			
Topologi*Nós	Lower	Center	Upper
3 5	1,149	1,205	1,261
3 7	1,888	1,944	2,000
Topologi = 3			
Nós = 5 subtracted from:			
Topologi*Nós	Lower	Center	Upper
3 7	0,6826	0,7388	0,7949

Figura 12: Comparações múltiplas de Tukey (continuação).

Topologi = 1				
Taxa_che = 1 subtracted from:				
Topologi*Taxa_che				
1	2	0,2848	0,3539	0,4231
1	4	0,7325	0,8017	0,8709
1	8	1,2739	1,3431	1,4123
2	1	-0,3332	-0,2640	-0,1949
2	2	-0,0141	0,0551	0,1243
2	4	0,3475	0,4167	0,4859
2	8	0,8355	0,9046	0,9738
3	1	-0,3941	-0,3250	-0,2558
3	2	-0,1004	-0,0312	0,0380
3	4	0,2700	0,3392	0,4084
3	8	0,6747	0,7438	0,8130
Topologi = 1				
Taxa_che = 2 subtracted from:				
Topologi*Taxa_che				
1	4	0,3786	0,4478	0,5169
1	8	0,9200	0,9892	1,0584
2	1	-0,6871	-0,6180	-0,5488
2	2	-0,3680	-0,2988	-0,2297
2	4	-0,0064	0,0628	0,1319
2	8	0,4815	0,5507	0,6199
3	1	-0,7481	-0,6789	-0,6097
3	2	-0,4543	-0,3851	-0,3160
3	4	-0,0839	-0,0147	0,0545
3	8	0,3207	0,3899	0,4591
Topologi = 1				
Taxa_che = 4 subtracted from:				
Topologi*Taxa_che				
1	8	0,472	0,541	0,611
2	1	-1,135	-1,066	-0,997
2	2	-0,816	-0,747	-0,677
2	4	-0,454	-0,385	-0,316
2	8	0,034	0,103	0,172
3	1	-1,196	-1,127	-1,057
3	2	-0,902	-0,833	-0,764
3	4	-0,532	-0,462	-0,393
3	8	-0,127	-0,058	0,011
Topologi = 1				
Taxa_che = 8 subtracted from:				
Topologi*Taxa_che				
2	1	-1,676	-1,607	-1,538
2	2	-1,357	-1,288	-1,219
2	4	-0,996	-0,926	-0,857
2	8	-0,508	-0,438	-0,369
3	1	-1,737	-1,668	-1,599
3	2	-1,443	-1,374	-1,305
3	4	-1,073	-1,004	-0,935
3	8	-0,668	-0,599	-0,530

Figura 13: Comparações múltiplas de Tukey (continuação).

```

Topologi = 2
Taxa_che = 1 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
2          2          0,2499    0,31911    0,38828
2          4          0,6116    0,68072    0,74990
2          8          1,0995    1,16867    1,23784
3          1         -0,1301   -0,06093    0,00824
3          2          0,1636    0,23282    0,30199
3          4          0,5341    0,60324    0,67241
3          8          0,9387    1,00785    1,07702

Topologi = 2
Taxa_che = 2 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
2          4          0,2924    0,3616     0,4308
2          8          0,7804    0,8496     0,9187
3          1         -0,4492   -0,3800    -0,3109
3          2         -0,1555   -0,0863    -0,0171
3          4          0,2150    0,2841     0,3533
3          8          0,6196    0,6887     0,7579

Topologi = 2
Taxa_che = 4 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
2          8          0,4188    0,4879     0,5571
3          1         -0,8108   -0,7417    -0,6725
3          2         -0,5171   -0,4479    -0,3787
3          4         -0,1467   -0,0775    -0,0083
3          8          0,2580    0,3271     0,3963

Topologi = 2
Taxa_che = 8 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
3          1         -1,299    -1,230     -1,160
3          2         -1,005    -0,936     -0,867
3          4         -0,635    -0,565     -0,496
3          8         -0,230    -0,161     -0,092

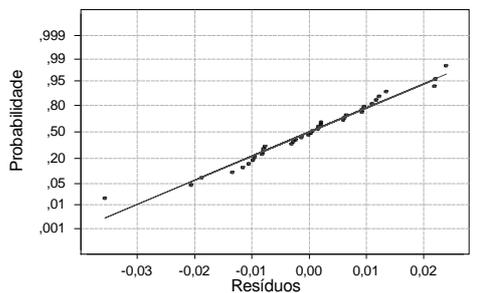
Topologi = 3
Taxa_che = 1 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
3          2          0,2246    0,2938     0,3629
3          4          0,5950    0,6642     0,7333
3          8          0,9996    1,0688     1,1380

Topologi = 3
Taxa_che = 2 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
3          4          0,3012    0,3704     0,4396
3          8          0,7059    0,7750     0,8442

Topologi = 3
Taxa_che = 4 subtracted from:
Topologi*Taxa_che      Lower      Center      Upper
3          8          0,3354    0,4046     0,4738

```

Figura 14: Comparações múltiplas de Tukey (parte final).



Average: -0,0000000
 StDev: 0,0124827
 N: 36
 Anderson-Darling Normality Test
 A-Squared: 0,263
 P-Value: 0,683

Figura 15: Gráfico de probabilidade normal para os resíduos.

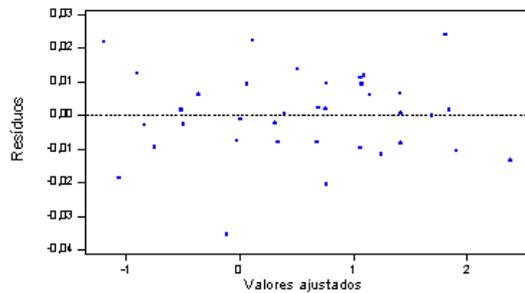


Figura 16: Gráfico dos resíduos versus valores ajustados..