ABORDAGEM MULTIOBJETIVO PARA OTIMIZAÇÃO DE REDES DE FILAS FINITAS

F. R. B. Cruz
Departamento de Estatística, Universidade Federal de Minas Gerais
31270-901 – Belo Horizonte – MG
fcruz@est.ufmg.br

J. H. Ferreira
Departamento de Engenharia Elétrica, Centro Federal de Educação Tecnológica de Minas Gerais, 30510-000 – Belo Horizonte – MG
jhissa@des.cefetmg.br

Resumo

Neste artigo é apresentada uma discussão sobre os resultados obtidos por meio de um algoritmo multiobjetivo recentemente desenvolvido para otimizar, simultaneamente, o tamanho total das áreas de espera, a taxa total de serviço e a taxa de atendimento global de uma rede de filas finitas com serviço geral. Como tais objetivos são conflitantes, utiliza-se uma versão multiobjetivo de um algoritmo genético projetado para encontrar soluções ótimas para mais de um objetivo. São obtidas algumas propriedades que podem auxiliar na análise e projeto desses importantes sistemas estocásticos.
Palavras-Chaves: Redes de filas; algoritmos genéticos; otimização multiobjetivo.

Abstract

In this paper a discussion is presented about results obtained from a multi-objective algorithm recently developed to simultaneously optimize the total number of buffers, the overall service rate, and the throughput of a general-service finite queueing network. These conflicting objectives are optimized by means of a multi-objective genetic algorithm, designed to produce solutions for more than one objective. Some properties are identified that may help the analysis and design of these important stochastic systems.
Keywords: Network of queues; genetic algorithms; multiobjective optimization.

1. INTRODUÇÃO

Enfocam-se neste trabalho as filas de um único servidor com tempos entre chegadas exponencialmente distribuídos e tempos de serviço com distribuição geral, configuradas em redes em uma topologia acíclica arbitrária. Mais especificamente, o foco está nas redes de filas *M/G/1/K*, que, na notação de Kendall (1953), possui chegadas *M*arkovianas, tempos de serviço *G*erais, *um* único servidor e capacidade total de *K* itens, incluindo um item em serviço. Um exemplo deste tipo de rede é mostrado na Figura 1.

O objetivo é alcançar, simultaneamente, uma taxa de atendimento máxima (Θ), utilizando-se o menor número possível de áreas de espera ($\mathbf{K} = K_1, K_2, \ldots, K_n$) e com as menores taxas de serviço possíveis ($\boldsymbol{\mu} = \mu_1, \mu_2, \ldots, \mu_n$), conhecidas a topologia da rede e as taxas de chegada externas ($\Lambda = \Lambda_1, \Lambda_2, \ldots, \Lambda_n$). Os cientistas da computação e engenheiros estão entre os potenciais usuários destes modelos de filas. Na verdade, esses modelos podem ajudar a entender e a melhorar vários sistemas encontrados da vida real, como sistemas de manufatura (Youssef e Elmaraghy, 2008), de produção (Andriansyah *et al.*, 2010) e de saúde (Osorio e Bierlaire, 2009), incluindo, ainda, sistemas de tráfego urbano e de pedestres (Cruz *et al.*, 2010), de computação e de comunicação (Gontijo *et al.*, 2011) e sistemas baseados na *web* (Chaudhuri et al., 2007).

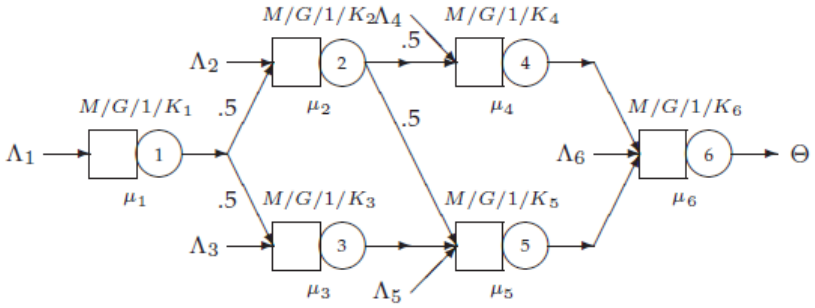


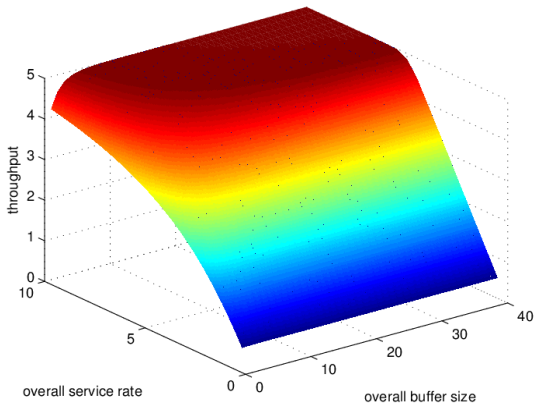
Figura 1 – Uma rede de filas *M/G/1/K*

Há um compromisso entre o tamanho total das áreas de espera, as taxas de serviço e a taxa de atendimento resultante. Devido ao elevado custo representado pelas áreas de espera e serviços, o tamanho global da área de espera e a capacidade total de serviço alocado devem ser restritos. Do ponto de vista da rede, por outro lado, deve-se alcançar a maior taxa de atendimento possível. Infelizmente, a taxa de atendimento é diretamente afetada pelas áreas de espera e pelas taxas de serviço, tornando tais objetivos conflitantes. De fato, se a área de espera e a capacidade de serviço são reduzidos, pode ocorrer uma redução indesejável na taxa de atendimento, como pode ser observado na Figura 2, que ilustra uma taxa de atendimento Θ para uma fila única *M/G/1/K* com $cv^2 = 1{,}5$ (quadrado do coeficiente de variação do tempo de serviço) e $\Lambda = 5$ usuários por unidade de tempo (taxa de chegada externa), representados como uma função do tamanho da área de espera, *K*, e da taxa de serviço, μ (ver Equações 4 e 10, mais à frente). As respectivas curvas de nível também são mostradas.

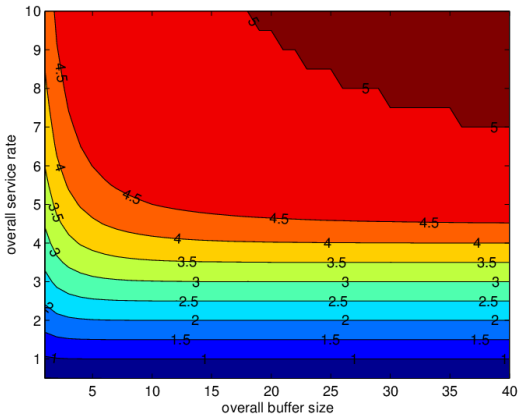
Comportamento semelhante também será observado para a taxa de atendimento de filas configuradas em redes. A suavidade da superfície do gráfico mostrado na Figura 2 parece sugerir uma função convexa. Resultados semelhantes foram relatados para redes de filas simples (Meester e Shanthikumar, 1990). No entanto, a característica plana na parte superior da superfície se apresenta como um problema para os métodos tradicionais de otimização. Em Smith e Cruz (2005) é proposto um algoritmo de otimização que combina, com sucesso, o método clássico de Powell com múltiplos pontos iniciais, evitando-se uma convergência prematura para soluções ótimas locais.

Neste artigo, é apresentada uma abordagem multiobjetivo capaz de otimizar, simultaneamente, o tamanho total das áreas de espera, a taxa global de serviço alocada e a taxa de atendimento para redes de filas *M/G/1/K*. O método proposto produz um conjunto de

soluções eficientes, denominadas conjunto de Pareto ótimo, para mais de um objetivo (Chankong e Haimes, 1983). Com a abordagem proposta, o decisor tem condições de avaliar o resultado final devido a cada solução escolhida. Além disso, a abordagem multiobjetivo também permite ao usuário aumentar um dos objetivos (por exemplo, a taxa de atendimento), enquanto reduz simultaneamente os outros objetivos (por exemplo, as áreas de espera e as taxas de serviço alocadas).



(a) taxa de atendimento Θ versus taxa de serviço μ e área de espera K



(b) curvas de nível

Figura 2 – Resultado para uma única fila M/G/1/K, com taxa de chegada Λ = 5.0

Este artigo está organizado da seguinte forma. A Seção 2 trata da apresentação de um algoritmo evolutivo multiobjetivo especificamente desenvolvido, juntamente com o método de expansão generalizado, que é uma ferramenta de avaliação de desempenho aqui utilizada para estimar a taxa de atendimento Θ. Na Seção 3, são discutidos os resultados dos experimentos computacionais realizados com o algoritmo. Finalmente, a Seção 4 conclui o artigo com considerações finais e sugestões para futuras pesquisas na área.

2. ALGORITMOS

2.1. FORMULAÇÃO DE PROGRAMAÇÃO MATEMÁTICA

Do ponto de vista da modelagem, o problema de maximização da taxa de atendimento pode ser definido por uma formulação de programação matemática inteira-mista, em que os custos da área de espera total e a taxa global de serviço são minimizados, enquanto a taxa de atendimento é maximizada, sujeitos à atribuição inteira de áreas de espera e taxas de serviço não-negativas. Definindo-se uma rede de filas como um dígrafo $G(N, A)$, onde N é um conjunto finito de nós (filas), e A é um conjunto finito de arcos (par de filas conectadas), uma formulação possível é (Cruz *et al.*, 2012):

$$\text{minimize } \mathbf{F}(\mathbf{K}, \boldsymbol{\mu}), \tag{1}$$

sujeito a:

$$K_i \in \{1,2,\cdots\}, \forall i \in N, \tag{2}$$

$$\mu_i \geq 0, \forall i \in N, \tag{3}$$

em que as variáveis de decisão Ki e μi indicam a área de espera e a taxa de atendimento para a i-ésima fila M/G/1/K, respectivamente. As funções objetivo, $\mathbf{F}(\mathbf{K}, \boldsymbol{\mu}) = [f_1(\mathbf{K}), f_2(\boldsymbol{\mu}), -f_3(\mathbf{K}, \boldsymbol{\mu})]$, representam a área de espera total alocada, $f_1(\mathbf{K}) = \sum_{\forall i \in N} K_i$, a taxa global de atendimento, $f_2(\boldsymbol{\mu}) = \sum_{\forall i \in N} \mu_i$, e a taxa de atendimento, $f_3(\mathbf{K}, \boldsymbol{\mu}) = \Theta(\mathbf{K}, \boldsymbol{\mu})$.

É importante ressaltar que na literatura a taxa de atendimento é frequentemente modelada como uma restrição que deve ser maior do que um valor limiar, $\Theta\tau$ (ver, por exemplo, Andriansyah *et al.*, 2010), em vez de ser um objetivo que deve ser maximizado, conforme considerado neste artigo. O problema é que, para se resolver a versão mono-objetivo do problema, a restrição da taxa de atendimento deve ser relaxada e um valor arbitrário deve ser estabelecido para $\Theta\tau$, o que não é uma tarefa trivial. Além disso, muitas vezes um pequeno decréscimo no valor limiar produz uma redução significativa na atribuição de áreas de espera e de taxas de serviço. Este compromisso entre a taxa de atendimento, o tamanho da área de espera e as taxa de serviço, infelizmente, não ficará visível em uma formulação mono-objetivo equivalente (que geralmente combina os objetivos múltiplos e um único objetivo por meio de um vetor de pesos, $\boldsymbol{\omega}$). Além disso, a determinação do vetor $\boldsymbol{\omega}$ é difícil e, frequentemente, conduz a formulações mono-objetivo arbitrárias.

Neste artigo, um algoritmo multiobjetivo evolucionário (em ingês, MOEA) é usado, em combinação com o método da expansão generalizado (em inglês, GEM), que é uma ferramente bem conhecida e eficaz na obtenção de aproximações precisas para o desempenho de filas de espera configuradas em redes (Kerbache e Smith, 1987). MOEAs são particularmente adequados para problemas com vários objetivos simultâneos, devido ao bom desempenho demostrado em problemas multiobjetivos semelhantes de otimização em redes (por exemplo, ver Carrano *et al.*, 2006 e referências). Os algoritmos serão apresentados em duas partes. Inicialmente, o algoritmo de avaliação de desempenho será descrito. Em seguida, será detalhado o algoritmo proposto para otimizar o problema.

2.2. AVALIAÇÃO DE DESEMPENHO EM FILAS FINITAS INDIVIDUAIS

Nas filas únicas (não exatamente o caso de interesse aqui), a taxa de atendimento, $\Theta(K,\mu)$, é dada por:

$$\theta(K,\mu) = \Lambda(1 - p_k), \tag{4}$$

em que Λ é a taxa de chega externa e p_K é a probabilidade de bloqueio, que é a probabilidade de se encontrar o sistema cheio (isto é, com um número de itens igual à capacidade total K). Assim, o problema de determinar $\Theta(K,\mu)$ reduz-se a determinar p_K.

Para o caso especial de sistemas markovianos puros (*i.e.*, filas $M/M/1/K$), a expressão da probabilidade de bloqueio pode ser facilmente deduzida, conforme mostrado na literatura de teoria de filas (*e.g.*, Gross *et al.*, 2009):

$$p_K = \frac{(1-\rho)\rho^K}{1-\rho^{K+1}}, \tag{5}$$

válida para $\rho < 1$, em que $\rho \equiv \lambda/\mu$ é a utilização do sistema. Relaxando-se a restrição de integralidade de K é possível expressar, em forma fechada, a alocação ótima de áreas de espera para filas $M/M/1/K$ em termos de ρ e p_K:

$$K_M = \left\lceil \frac{\ln\left(\dfrac{p_K}{1-\rho+p_K\rho}\right)}{\ln(\rho)} \right\rceil, \tag{6}$$

em que $\lceil x \rceil$ é o menor inteiro não inferior a x. Consequentemente, é possível mostrar que a alocação ótima de área de espera (excluídos os itens em serviço) para filas $M/M/1/K$ é:

$$x_M = K_M - 1. \tag{7}$$

Para filas gerais $M/G/c/K$, a probabilidade de bloqueio pode ser determinada apenas por técnicas de aproximação. Em particular, Smith e Cruz (2005) usaram uma aproximação a dois momentos, baseada na expressão markoviana, Eq. (7), que é bem efetiva:

$$x_\varepsilon\left(cv^2\right) = x_M + \text{INT}\left[\frac{\left(cv^2 - 1\right)\sqrt{\rho}}{2} x_M\right], \tag{8}$$

em que INT[x] é a parte inteira de x. Em particular, para filas gerais com servidor único, *M/G/1/K*, conhecido ρ e cv^2, a alocação ótima de áreas de espera pode ser escrita como:

$$x_\varepsilon\left(cv^2\right) = \frac{\left[\ln\left(\frac{p_K}{1-\rho+p_K\rho}\right) + \ln(\rho)\right]\left(2+\sqrt{\rho}cv^2 - \sqrt{\rho}\right)}{2\ln(\rho)}. \tag{9}$$

Finalmente, pode-se isolar p_K, para se determinar uma expressão fechada para a probabilidade de bloqueio em filas *M*/*G*/1/*K*, em função de *K* (note que, para filas *M*/*G*/1/*K*, $K = 1 + x_\varepsilon$):

$$p_K = \frac{(1-\rho)\rho^{\left(\frac{2+\sqrt{\rho}cv^2-\sqrt{\rho}+2(K-1)}{2+\sqrt{\rho}cv^2-\sqrt{\rho}}\right)}}{1-\rho^{\left(2\frac{2+\sqrt{\rho}cv^2-\sqrt{\rho}+(K-1)}{2+\sqrt{\rho}s^2-\sqrt{\rho}}\right)}}. \tag{10}$$

Como última observação, note-se a similaridade existente entre as expressões para p_K, para filas *M*/*M*/1/*K* e *M*/*G*/1/*K*, respectivamente Equações (5) e (10).

2.3. AVALIAÇÃO DE DESEMPENHO EM FILAS FINITAS CONFIGURADAS EM REDE

Para as redes de filas, a estimativa da taxa de atendimento é feita por meio do método de expansão generalizado (em inglês, GEM), utilizado com sucesso para estimar o desempenho de redes acíclicas, de filas finitas, arbitrariamente configuradas (Kerbache e Smith, 1987). O GEM é uma combinação de decomposição nó a nó e tentativas repetidos, em que cada fila é analisada separadamente e são realizadas modificações que representam os efeitos entre as filas da rede.

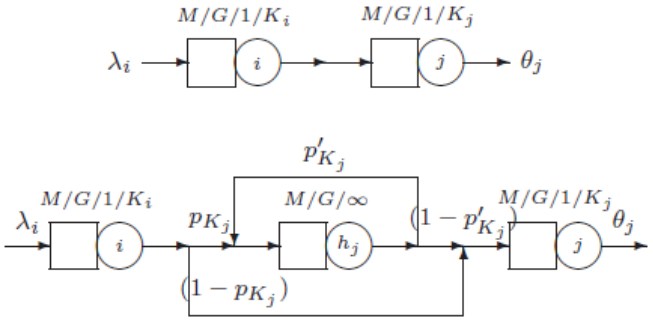


Figura 3 – Método da expansão generalizado

Como descrito em detalhes por Kerbache e Smith (1987), o GEM cria para cada fila finita *j* uma fila auxiliar, h_j, que é modelada como uma fila *M/G/*∞, conforme ilustrado na Figura 3. Cada entidade que dirige à fila *j* pode ser bloqueada (com probabilidade p_{Kj}), ou não (com probabilidade $1–p_{Kj}$). Quando o bloqueio ocorre, a entidade é encaminhadas à fila h_j, onde sofre um atraso para dar tempo que um espaço seja liberado na fila *j*. Assim, o papel da fila auxiliar h_j é registrar o tempo que uma entidade tem de esperar antes de entrar na fila *j*. O objetivo final do GEM é prover um procedimento aproximado para atualizar as taxas de serviço de cada uma das filas *i* que são seguidas por filas finitas:

$$\tilde{\mu}_i^{-1} = \mu_i^{-1} + p_{k_j}\left(\mu_h'\right)^{-1}, \tag{11}$$

e assim possibilitar a obtenção de uma aproximação acurada para a taxa de saída da rede Θ_j.

Observe que o processo de avaliação de desempenho deve ser conduzido em uma ordem específica. A avaliação do desempenho da rede em estudo, definido como dígrafo çG(N,A), é apresentada na Figura 4. O algoritmo calcula os bloqueios nos nós de serviços a montante, resultando em taxas de serviço eficazes que são reduzidas de acordo com a

Equação (11). Note-se que o algoritmo de avaliação de desempenho é uma variante do algoritmo de Dijkstra (1959), para a determinação de caminhos mínimos. Por exemplo, na rede ilustrada na Figura 1, uma sequência de avaliação válida é 1 → 2 → 4→3 → 5 → 6. Especificamente, a sequência deve certificar que um nó somente será acessado após todos os seus antecessores. Assumindo que os circuitos não estão presentes em $G(N,A)$, o GEM tem uma complexidade de tempo de execução de $O(N^2)$, que está de acordo com o algoritmo de Dijkstra.

```
algorithm
    read graph, G(N, A)
    read routing probabilities, p_[ij], ∀ (i, j) ∈ A
    read external arrival rates and service rates, Λ_i, μ_i, ∀ i ∈ N
    initialize set of labeled nodes, P ← ∅
    while P ≠ V
        choose j such that (j ∈ N) and (j ∉ P)
        if {i| (i, j) ∈ A} ⊆ P then
            /* compute performance measures */
            compute p_Kj θ_j
            /* forward information to successors */
            for ∀ k ∈ {k'| (j, k') ∈ A} then
                λ_k ← λ_k + θ_j p_[jk]
            end for
            /* label node as pre-evaluated */
            P ← P ∪ {j}
        end if
    end while
end algorithm
```

Figura 4 – Algoritmo para a análise de desempenho

2.4. ALGORITMO DE OTIMIZAÇÃO

Para a rede considerada neste artigo, o MOEA parece ser uma escolha adequada, para a maximização multiobjetivo da taxa de atendimento. Os MOEAs são algoritmos de otimização que realizam uma busca global aproximada, baseada em informações obtidas a partir da avaliação de vários indivíduos do espaço de busca (Deb, 2001). A população de indivíduos que convergem para um valor ótimo é obtida através da aplicação dos operadores genéticos de *mutação*, *cruzamento*, *seleção* e *elitismo*.

```
algorithm
    read graph, arrival, service rates, G(N, A), Λ_i ∀ i ∈ N
    P_1 ← GenerateInitialPopulation(popSize)
    for i = 1 until numGen do
        /* generate offspring by crossover and mutation */
        Q_i ← MakeNewPop(P_i)
        /* combine parent and offspring */
        R_i ← P_i ∪ Q_i
        /* find non-dominated fronts F = (F_1, F_2, ...) */
        F ← FastNonDominatedSort(R_i)
        /* find new population by */
        /* the crowding-distance-assignment */
        P_{i+1} ← GenerateNewPopulation(R_i)
    end for
    P_{numGen+1} ← ExtractParetoSet(P_numGen)
    write P_{numGen+1}
end algorithm
```

Figura 5 – Algoritmo genético multiobjetivo com elitismo NSGA-II (Deb *et al.*, 2002)

Cada um destes operadores define um tipo de MOEA, que pode ser implementado de várias maneiras diferentes. Além disso, a convergência do MOEA é garantida pela atribuição de um valor de aptidão para cada membro da população, preservando-se a diversidade. De fato, aplicações recentes, bem sucedidas de algoritmos genéticos (AGs) foram relatadas em estudos mono-objetivos (Lin, 2008) e multiobjetivos (Carrano *et al.*, 2006). O exemplo do

MOEA utilizado neste estudo se baseia no algoritmo genético com elitismo e ordenação não-dominada (NSGA-II) desenvolvido por Deb *et al.* (2002), mostrado na Figura 5. Na aplicação de AGs para otimização multiobjetivo, os operadores de seleção e elitismo devem ser especificamente ajustados para identificar corretamente as condições ótimas, como será mostrado a seguir.

Elitismo se baseia no conceito de dominância. Um ponto $\mathbf{x}_i = (x_{i1}, x_{i2}, ..., x_{in})$ domina um ponto $\mathbf{x}_j = (x_{j1}, x_{j2}, ..., x_{jn})$ se $\mathbf{x}_i$ é *melhor* que $\mathbf{x}_j$ em um dos objetivos k (*p.e.*, $f_k(\mathbf{x}_i) < f_k(\mathbf{x}_j)$, para minimização) e não é *pior* em nenhum outro objetivo l (*p.e.*, $f_l(\mathbf{x}_i) \not> f_l(\mathbf{x}_j)$, para minimização). Para incluir o operador de elitismo, utilizou-se um procedimento conhecido como algoritmo rápido de ordenação não-dominante (Deb *et al.*, 2002). Este algoritmo isola os indivíduos na população em várias camadas (fronteiras) F_i, de modo que as soluções desta camada são não dominadas e toda solução em uma dada camada F_i, $i > 1$, é dominada por pelo menos uma solução na camada F_{i-1} e por nenhuma solução em F_j, $j \geq i$. Como mostrado em Deb *et al.* (2002), isto pode ser realizado com complexidade de tempo da ordem de $O(n\log n)$.

A seleção é realizada a partir da escolha sequencial de indivíduos de cada fronteira não dominada (F_1, F_2, ...), até se obter o número máximo de indivíduos que foi estabelecido para seguir à próxima geração. Algumas decisões devem ser tomadas, caso esse número máximo seja excedido, após a adição dos indivíduos da fronteira F_i. Uma possibilidade é calcular uma medida de diversidade (como, por exemplo, a distância entre os aglomerados da população), de forma a assegurar a maior diversidade da população, como definido em Deb *et al.* (2002). Dessa forma, apenas os indivíduos com a maior distância entre si nos aglomerados são mantidos para as próximas gerações (iterações).

É conhecido que os operadores de cruzamento e mutação são dependentes da aplicação. Para o problema aqui tratado, foi escolhido um mecanismo de cruzamento uniforme (Bäck *et al.*, 1997), muito utilizado em codificações multivariáveis, devido à sua eficiência em identificar, herdar e proteger os genes comuns, enquanto garante a recombinação dos genes não triviais (Hu e Di Paolo, 2007). Neste mecanismo, o cruzamento é realizado para cada variável com uma probabilidade **rateCro**, de acordo com o operador de cruzamento. O operador de cruzamento utilizado no algoritmo é o *operador de cruzamento binário simulado* (em inglês, SBX). O SBX (Deb, 2001) é bastante conveniente para AGs com variáveis reais, em razão da sua capacidade de simular os *operadores de cruzamento binário* e evitar a recodificação das variáveis. Os indivíduos filhos são calculados a partir de seus pais, de acordo com as seguintes equações:

$$x_{i,(1,t+1)} = 0.5\left[\left(1+\beta\right)x_{i,(1,t)} + \left(1-\beta\right)x_{i,(2,t)}\right], \tag{12}$$

$$x_{i,(2,t+1)} = 0.5\left[\left(1-\beta\right)x_{i,(1,t)} + \left(1+\beta\right)x_{i,(2,t)}\right], \tag{13}$$

em que β é uma variável aleatória obtida da seguinte função de distribuição de probabilidade:

$$f(\beta) = \begin{cases} 0.5(\eta+1)\beta^{\eta}, & \text{se} \quad \beta \leq 1, \\ 0.5(\eta+1)\dfrac{1}{\beta^{\eta+2}}, & \text{caso contrário.} \end{cases} \tag{14}$$

Note que Equações (12) e (13) são projetadas para gerar soluções-filho que possuem uma grande aptidão de busca, similar à de um cruzamento codificado em binário para os AGs (Deb, 2001). Pelo ajuste de η, podem ser gerados diferentes pesos β capazes de gerar filhos que sejam mais semelhantes (η pequeno) ou menos semelhantes (η elevado) aos seus pais.

Para cada gene individual (cada uma das variáveis de decisão K_i ou μ_i), o operador de mutação atua com uma probabilidade específica **rateMut**. Como sugerido por Deb (2001), perturbações gaussianas foram adicionados às variáveis de decisão, $K_i + \varepsilon_i$ e $\mu_i + \varepsilon_{N+i}$, para todo $i \in N$, com $\varepsilon_i \sim Normal(0, 1)$, $i \in \{1,2, ..., 2N\}$.

Finalmente, para garantir a factibilidade das restrições (2) e (3), após a aplicação dos

operadores de cruzamento e mutação, os valores das variáveis inteiras devem ser arredondados adequadamente e todas as variáveis são reajustadas pela aplicação dos seguintes operadores de reflexão:

$$K_{rfl_i} = K_{\text{lowlim}} + \left|K_i - K_{\text{lowlim}}\right|, \tag{15}$$

$$\mu_{rfl_i} = \mu_{\text{lowlim}_i} + \left|\mu_i - \mu_{\text{lowlim}_i}\right|, \tag{16}$$

em que K_{lowlim} é o limite inferior de alocação da área de espera (ou seja, $K_{\text{lowlim}} = 1$) e μ_{lowlim} é o limite inferior do serviço de alocação, de modo a assegurar que $\rho < 1$. Note que K_i e μ_i são os valores resultantes após o cruzamento e a mutação e que K_{rfli} e μ_{rfli} são os valores obtidos após a operação de reflexão. O esquema proposto garantidamente gera soluções viáveis, sem evitar ou favorecer qualquer solução em particular.

2.5. PROBLEMAS DE CONVERGÊNCIA

Recentemente, o critério de parada dos algoritmos de otimização multiobjetivo evolutivos foi analisado em detalhe. Evidentemente, o número máximo de gerações **numGen** desempenha um papel importante na qualidade das soluções. No entanto, aumentar o número de gerações pode não ser ideal, porque o tempo computacional é desperdiçado com muitas iterações que não levam a uma melhora significativa do resultado alcançado. Assim, Rudenko (2004) sugeriu que um melhor critério de parada é obtido quando um número fixo de iterações é realizado sem nenhuma melhora. Para demonstrar a complexidade do tema, Rudenko (2004) realizou um estudo abrangente de experimentos computacionais. Os resultados revelaram que um critério de parada óbvio, que é quando toda a população está na fronteira F_i, não é adequado. Rudenko (2004) propôs, então, um critério de parada local que calcula uma medida da estabilidade da solução não dominada após cada iteração, com base na convergência da distância máxima do aglomerado, d_l, medida ao longo de L gerações e calculada pelo seguinte desvio padrão:

$$\sigma_L = \sqrt{\frac{1}{L}\sum_{l=1}^{L}\left(d_l - \bar{d}_L\right)^2}, \tag{17}$$

em que $\bar{d}_L$ é a média de d_l ao longo de L gerações e o critério $\sigma_L < \delta_{\text{lim}}$ deve indicar quando se deve encerrar a execução do MOEA. Rudenko (2004) também sugere que L e δ_{lim} devem ser ajustados para 40 e 0,02, respectivamente, o que leva ao critério de parada $\sigma_{40} < 0{,}02$.

3. RESULTADOS E DISCUSSÃO

Para manter compatibilidade com uma implementação do GEM baseada na biblioteca IMSL, o algoritmo de otimização foi codificado em FORTRAN (Smith e Cruz, 2005). O código está disponível, mediante solicitação aos autores, para fins de ensino e pesquisa. Em primeiro lugar, os experimentos computacionais foram realizados para se descobrir um conjunto subótimo de parâmetros que garantam uma rápida convergência. Finalmente, uma análise detalhada de uma rede de filas foi realizada.

3.1. AJUSTE DOS PARÂMETROS

Tal como indicado por estudos anteriores sobre o AG, o conjunto subótimo de parâmetros que asseguram uma rápida convergência, com uma quantidade mínima de esforço computacional, deve ser determinado por tentativa e erro. Diferentes topologias de redes acíclicas de tamanhos variados foram testadas e os resultados são semelhantes.

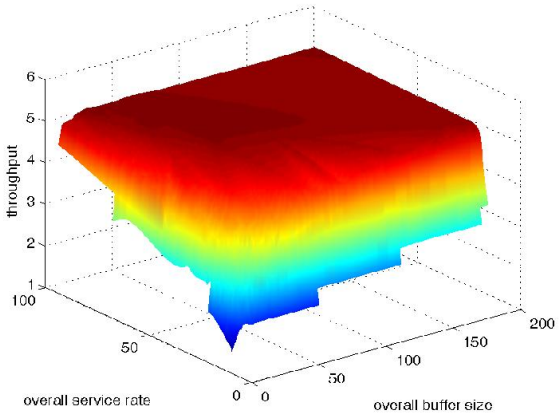
O melhor grupo de parâmetros para o algoritmo é a seguinte combinação: (*i*) a utilização combinada do SBX e da mutação, com (*ii*) uma taxa de mutação inferior a 2%, (*iii*) embora quanto maior, melhor, uma população de 400 indivíduos parece ser suficiente, e (*iv*) o parâmetro de dispersão, η, não deve ser maior do que 8. Para assegurar um tempo de computação finito, o número máximo de gerações, **numGen**, foi ajustado para 4000.

Felizmente, o algoritmo MOEA é suficientemente robusto e apresenta um bom desempenho para uma ampla variedade de problemas, como confirmado pela ampla variedade de experimentos realizados (não mostrados).

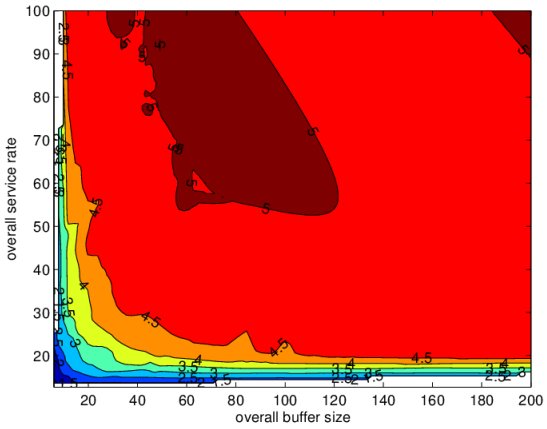
3.2. ANÁLISE DE UMA REDE DE FILAS

A rede mostrada na Figura 1 foi analisada com o método proposto. Dois diferentes coeficientes de variação quadrática foram analisados, cv^2 = 0,5 e cv^2 = 1,5, com taxa de chegada (Λ_1 = 5,0). Em primeiro lugar, a velocidade de convergência e a robustez do algoritmo genético foram confirmadas para este tipo de problema. O ajuste do conjunto experimental foi idêntico ao da análise anterior. No entanto, os resultados indicaram que a convergência era estável a 2.000 iterações.

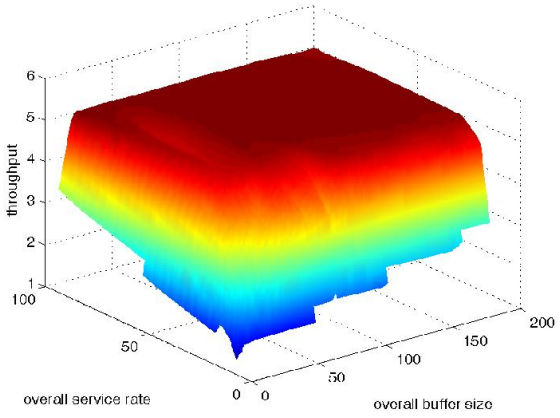
A Figura 6 mostra os resultados. É possível ver a população final e o respectivo traçado das curvas de nível. É notável a semelhança entre as curvas de nível e o gráfico exato para uma fila única, Figura 2(b). Os resultados sugerem que redes de filas parecem se comportar como uma fila única equivalente. Infelizmente, é desconhecido se seria ou não possível obter algum tipo de algoritmo capaz de prever os parâmetros equivalentes para uma fila única. Adicionalmente, nota-se que para cv^2 = 0,5 têm-se curvas de nível mais próximas da origem que para cv^2 = 1,5. Esse é um comportamento esperado, visto que um menor cv^2 indica menor variabilidade no tempo entre serviço. A metodologia mostra-se, portanto, consistente. Outro ponto interessante é que as curvas de nível ajudam a identificar os pontos a partir dos quais não compensa mais aumentar as áreas de espera (ou as taxas de serviço), por serem irrisórios os ganhos nas taxas de atendimento, Θ.



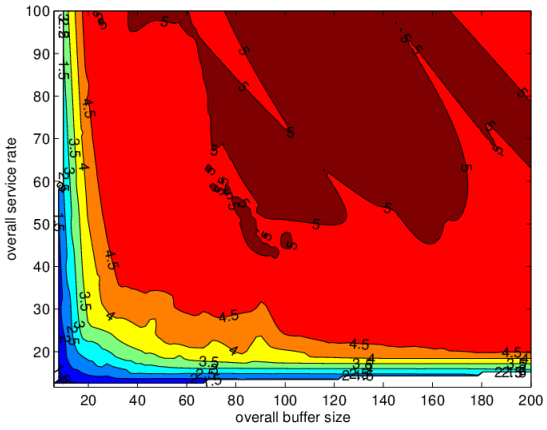
(a) superfície final para cv^2 = 0,5



(b) curvas de nível para cv^2 = 0,5



(c) superfície final para cv^2 = 1,5



(d) curvas de nível para cv^2 = 1,5

Figura 6 – Resultados finais para a rede da Figura 1

A Tabela 1 apresenta algumas soluções Pareto eficientes, para uma análise mais

detalhada. Nota-se que, com essa metodologia multiobjetivo, é possível identificar pontos a partir dos quais não mais interessa aumentar o gasto em recursos (áreas de espera e taxa de serviço), uma vez que o ganho obtido na taxa total de saída será muito pequeno. Por exemplo, para um cv^2 = 0,5, temos que, mantido fixo a taxa total de serviço, um incremento de 52% na área de espera total produziria um ganho de apenas 2% na taxa de saída. De maneira similar, pode haver um ponto na taxa de serviço onde isso também ocorre. De fato, pode-se notar que um aumento de 27% na taxa de serviço alocada pode produzir um aumento de 2%, o que pode ser considerado pouco significativo. Nota-se também que com um cv^2 = 1,5 tal fenômeno pode ocorrer de maneira ainda mais pronunciada. Observa-se que pode ser necessário um aumento de até 220% na área total alocada, para um aumento de apenas 6% na taxa de serviço. É, portanto, mais vantajoso manter um sistema com uma alocação que produza na saída 98% da entrada (4,914/5,000), do que gastar 11% a mais em (taxa total alocada de) serviço, para elevar a saída em 2% (isto é, elevá-la para 99,8% da taxa de entrada). Esses são apenas alguns exemplos de análises que podem ser feitas em filas finitas configuradas em redes, via metodologia multiobjetivo.

Tabela 1 - Soluções Pareto eficientes selecionadas a partir dos experimentos computacionais

cv^2	$\sum K$	Δ%	$\sum \mu$	Δ%	Θ	Δ%
0,5	316		20,3		4.850	
	480	52	20,3	0	4,935	2
	340		20,5		4,899	
	343	1	26,0	27	5,000	2
1,5	118		23,6		4,656	
	378	220	23,6	0	4,956	6
	520		20,5		4,914	
	520	0	22,7	11	4,991	2

4. CONCLUSÕES

Com o propósito de otimizar a taxa de atendimento, os tamanhos das áreas de espera e as taxas de serviço de redes de filas com um único servidor e distribuição geral de tempos de serviço, uma abordagem multiobjetivo foi apresentada. O método da expansão generalizada (GEM) foi acoplado a um algoritmo evolucionário multiobjetivo (MOEA), o que tornou possível melhorar a compreensão das redes de filas gerais finitas. De fato, curvas de Pareto (fronteira eficiente) foram obtidas, a partir das quais foi possível a identificação de pontos notáveis para a análise de *tradeoff* entre os objetivos otimizados, quais sejam a taxa de atendimento, a área de espera total alocada e a taxa de serviço global alocada.

Tópicos para futuras investigações nesta área incluem extensões para redes de filas gerais multisservidoras, possivelmente por meio de núcleo-estimadores (Gontijo *et al.*, 2011). Também é interessante considerar diferentes medidas de desempenho, tais como o trabalho em processo (em inglês, WIP), o tempo de permanência e assim por diante. Estes são apenas alguns exemplos de possíveis temas para pesquisa.

AGRADECIMENTOS

Este trabalho foi parcialmente financiado pelo CNPq (projetos 201046/1994-6, 301809/1996-8, 307702/2004-9, 472066/2004-8, 304944/2007-6, 561259/2008-9, 553019/2009-0, 550207/2010-4, 501532/2010-2, 303388/2010-2), pela CAPES (projeto BEX-0522/07-4) e pela FAPEMIG (projetos CEX-289/98, CEX-855/98, TEC-875/07, CEX-PPM-00401/08, CEX-PPM-00390-10, CEX-PPM-00071-12).

5. REFERÊNCIAS BIBLIOGRÁFICAS

[1] Andriansyah, R., Van Woensel, T., Cruz, F. R. B. e Duczmal, L., Performance optimization of open zero-buffer multi-server queueing networks. Computers & Operations Research. Vol. 37, n. 8, p. 1472-1487, 2010.

[2] Bäck, T., Fogel, D., Michalewicz, Z. (Eds.), Handbook of Evolutionary Computation. Institute of Physics Publishing and Oxford University Press, 1997.

[3] Carrano, E. G., Soares, L. A. E., Takahashi, R. H. C., Saldanha, R. R. e Neto, O. M., Electric distribution network multiobjective design using a problem-specific genetic algorithm. IEEE Transactions on Power Delivery. Vol. 21, n. 2, p. 995-1005, 2006.

[4] Chankong, V. e Haimes, Y. Y., Multiobjective Decision Making: Theory and Methodology. Elsevier, Amsterdam, The Netherlands, 1983.

[5] Chaudhuri, K., Kothari, A., Pendavingh, R., Swaminathan, R., Tarjan, R. e Zhou, Y., Server allocation algorithms for tiered systems. Algorithmica. Vol. 48, n. 2, p. 129-146, 2007.

[6] Cruz, F. R. B., Kendall, G., While, L., Duarte, A. R. e Brito, N. L. C., Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers. Mathematical Problems in Engineering. Vol. 2012, n. Article ID 348262, 19 pages, 2012.

[7] Cruz, F. R. B., Van Woensel, T., Smith, J. M. e Lieckens, K., On the system optimum of traffic assignment in M/G/c/c state-dependent queueing networks. European Journal of Operational Research. Vol. 201, n. 1, p. 183-193, 2010.

[8] Deb, K., Multi-objective Optimisation using Evolutionary Algorithms. Wiley, 2001.

[9] Deb, K., Pratap, A., Agarwal, S. e Meyarivan, T., A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation. Vol. 6, n. 2, p. 182-197, 2002.

[10] Dijkstra, E. W., A note on two problems in connection with graphs. Numerical Mathematics. Vol. 1, p. 269-271, 1959.

[11] Gontijo, G. M., Atuncar, G. S., Cruz, F. R. B. e Kerbache, L., Performance evaluation and dimensioning of GIX/M/c/N systems through kernel estimation. Mathematical Problems in Engineering. Vol. 2011, (Article ID 348262), 20 pages, 2011.

[12] Gross, D., Shortle, J. F., Thompson, J. M. e Harris, C. M., Fundamentals of Queueing Theory, 4a Ed. Wiley-Interscience, New York, NY, USA, 2009.

[13] Hu, X.-B., Di Paolo, E., An efficient genetic algorithm with uniform crossover for the multiobjective airport gate assignment problem. IEEE Congress on Evolutionary Computation, CEC 2007. Singapore, pp. 55–62, 2007.

[14] Kendall, D. G., Stochastic processes occurring in the theory of queues and their analysis by the method of embedded Markov chains. Annals Mathematical Statistics. Vol. 24, p. 338-354, 1953.

[15] Kerbache, L. e Smith, J. M., The generalized expansion method for open finite queueing networks. European Journal of Operational Research, Vol. 32, p. 448-461, 1987.

[16] Lin, F.-T., Solving the knapsack problem with imprecise weight coefficients using genetic algorithms. European Journal of Operational Research. Vol. 185, n. 1, p. 133-145, 2008.

[17] Meester, L. E. e Shanthikumar, J. G., Concavity of the throughput of tandem queueing systems with finite buffer storage space. Advances in Applied Probability. Vol. 22, n. 3, p. 764-767, 1990.

[18] Osorio, C e Bierlaire, M., An analytic finite capacity queueing network model capturing the propagation of congestion and blocking. European Journal of Operational Research. vol. 196, n. 3, 996-1007, 2009.

[19] Rudenko, O., Schoenauer, M., A steady performance stopping criterion for Pareto-based evolutionary algorithms. Proceedings of the 6th International Multi-Objective Programming and Goal Programming Conference. Hammamet, Tunisia, 2004.

[20] Smith, J. M. e Cruz, F. R. B., The buffer allocation problem for general finite buffer queueing networks. IIE Transactions. Vol. 37, n. 4, p. 343-365, 2005.

[21] Youssef, A. M. e Elmaraghy, H. A., Performance analysis of manufacturing systems composed of modular machines using the universal generating function. Journal of Manufacturing Systems. Vol. 27, n. 2, p. 55-69, 2008.