Tese de Doutorado

Otimização Multiobjetivo em Redes de Filas

por

Nilson Luiz Castelucio Brito

Orientador:

Frederico R. B. Cruz

Coorientador:

Anderson Ribeiro Duarte

Março de 2013

Nilson Luiz Castelucio Brito

Otimização Multiobjetivo em Redes de Filas

Tese de Doutorado apresentada ao Departamento de Estatística do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais, como requisito parcial à obtenção do título de Doutor em Estatística.

Orientador: Frederico R. B. Cruz
Co-orientador: Anderson Ribeiro Duarte

Universidade Federal de Minas Gerais
Belo Horizonte, março de 2013

Universidade Federal de Minas Gerais
Instituto de Ciências Exatas
Departamento de Estatística
Programa de Pós-Graduação
Caixa Postal 702
31270-901 Belo Horizonte- MG – Brasil

Telefone (31) 3409-5923
Fax (31) 3409-5924
E-mail: pgest@ufmg.br
WEB: http://www.est.ufmg.br/posgrad/

ATA DA DEFESA DE TESE DO ALUNO NILSON LUIZ CASTELUCIO BRITO

Realizou-se, no dia 04 de Março de dois mil e treze, às 09:00 horas na sala 2025 do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais, à 19ª defesa de Tese de Doutorado em Estatística. A tese foi intitulada **"Otimização Multiobjetivo em Redes de Filas"** e apresentada por **Nilson Luiz Castelucio Brito**, como requisito parcial para a obtenção do grau de Doutor em Estatística, à seguinte Comissão Examinadora: Prof. Frederico Rodrigues Borges da Cruz - orientador (Departamento de Estatística - UFMG), Prof. Anderson Ribeiro Duarte - coorientador (Departamento de Estatística - UFOP), Prof. Luiz Henrique Duczmal (Departamento de Estatística - UFMG), Prof. Roberto da Costa Quinino (Departamento de Estatística - UFMG), Prof. Fernando Luiz Pereira de Oliveira (Departamento de Estatística - UFOP), Prof. Lupércio França Bessegato (Departamento de Estatística - UFJF) e Prof. Marcone Jamilson Freitas Souza (Departamento de Computação - UFOP).

A Comissão considerou a tese:

(X) Aprovada
( ) Aprovada condicionalmente, sujeita a alterações, conforme folha de modificações, anexa
( ) Reprovada, conforme folha de Justificativas, anexa

Finalizados os trabalhos, lavrei a presente ata que, lida e aprovada, vai assinada por mim e pelos membros da Comissão.

Belo Horizonte, 04 de Março de 2013.

Prof. Frederico Rodrigues Borges da Cruz
Orientador / Departamento de Estatística - UFMG

Prof. Anderson Ribeiro Duarte
Coorientador / Departamento de Estatística - UFOP

Prof. Luiz Henrique Duczmal
Departamento de Estatística - UFMG

Prof. Roberto da Costa Quinino
Departamento de Estatística - UFMG

Prof. Fernando Luiz Pereira de Oliveira
Departamento de Estatística - UFOP

Prof. Lupércio França Bessegato
Departamento de Estatística - UFJF

Prof. Marcone Jamilson Freitas Souza
Departamento de Computação – UFOP

Universidade Federal de Minas Gerais
Instituto de Ciências Exatas
Departamento de Estatística
Programa de Pós-Graduação
Caixa Postal 702
31270-901 Belo Horizonte- MG – Brasil

Telefone (31) 3409-5923
Fax (31) 3409-5924
E-mail: pgest@est.ufmg.br
WEB: http://www.est.ufmg.br/posgrad/

FOLHA DE APROVAÇÃO

Otimização Multiobjetivo em Redes de Filas

Nilson Luiz Castelucio Brito

Tese defendida e aprovada pela banca examinadora constituída pelos professores:

Prof. Frederico Rodrigues Borges da Cruz
Orientador / Departamento de Estatística - UFMG

Prof. Anderson Ribeiro Duarte
Coorientador / Departamento de Estatística - UFOP

Prof. Luiz Henrique Duczmal
Departamento de Estatística - UFMG

Prof. Roberto da Costa Quinino
Departamento de Estatística - UFMG

Prof. Fernando Luiz Pereira de Oliveira
Departamento de Estatística - UFOP

Prof. Lupércio França Bessegato
Departamento de Estatística - UFJF

Prof. Marcone Jamilson Freitas Souza
Departamento de Computação - UFOP

Belo Horizonte, 04 de Março de 2013

Agradecimentos

Gostaria de deixar expressos aqui os meu sinceros agradecimentos a todos aqueles que de certa forma contribuíram para a realização desta tese.

Agradeço aos meus orientadores, Anderson e Frederico, por toda a ajuda;

À Fapemig, pela bolsa de doutorado referente ao PCRH, Plano de Capacitação de Recursos Humanos;

À Unimontes, pela minha liberação para cursar o doutorado;

À minha esposa Elza Aparecida de Freitas Brito, pela paciência;

Aos membros da comissão avaliadora, pelas excelentes sugestões de melhoria.

Gostaria de agradecer também a todos os meus amigos do Curso de Pós-Graduação em Estatística, pela força durante toda esta jornada;

A todos os professores e funcionários do Departamento de Estatística, agradeço pela excelente recepção.

Resumo

Em um dos mais desafiadores problemas de otimização de redes de filas finitas, a área total de espera (do inglês, *buffer*), f_1, e a taxa total de serviço, f_2, devem ser as menores possíveis, enquanto que a taxa de saída (do inglês, *throughput*), f_3, deve ser a maior possível. Para satisfazer a esses três objetivos conflitantes ($\min f_1$, $\min f_2$ e $\max f_3$), um algoritmo do tipo genético multiobjetivo foi desenvolvido, especialmente para redes de filas finitas, com tempos de serviço com distribuição geral e configuradas em redes acíclicas. Assim, o método proposto produziu um conjunto de soluções eficientes, para os três objetivos f_1, f_2 e f_3. Um conjunto completo de experimentos computacionais foi conduzido, para determinar a eficácia da abordagem proposta. As conclusões apresentadas, obtidas através da análise de várias redes, podem auxiliar aos profissionais da área no planejamento de redes de filas gerais.

Palavras-chaves: Manufatura; redes de filas; alocação de áreas de espera; alocação de serviços; algoritmos genéticos.

Abstract

In one of the most challenging finite queueing network optimization problems, the number of buffers (f_1) and the overall service rate (f_2) must be reduced while the throughput (f_3) must be maximized. In order to meet these three conflicting objectives ($\min f_1$, $\min f_2$, and $\max f_3$), a multi-objective genetic algorithm was developed specially for acyclic general-service queueing networks. The proposed method is shown to produce a set of efficient solutions for the three objectives f_1, f_2, and f_3. In order to determine the efficacy of the proposed approach, a comprehensive set of computational experiments was conducted and analyzed. The insights obtained from the analysis of some queueing networks may be helpful to practitioners and scientists in the complex task of analyzing and planning general-service time queueing acyclic networks.

Keywords: Manufacturing; queueing networks; buffer allocation; service allocation; genetic algorithms.

Índice

Glossário

Neste glossário encontram-se definições de símbolos e abreviações usadas frequentemente e consistentemente ao longo do texto. Símbolos que são usados apenas ocasionalmente em seções isoladas podem não estar incluídos. Os símbolos estão listados em ordem alfabética, com as letras gregas inseridas de acordo com seu nome em português.

$A/B/X/Y/Z$	Notação para descrição dos modelos de filas, em que A indica o padrão de chegada, B indica o padrão do serviço, X indica o número de canais, Y indica o limite de capacidade do sistema (inclusive os itens em serviço) e Z indica a disciplina da fila;
A	Conjunto (finito) de arcos (conexões entre filas);
$A(t)$	Função de distribuição acumulada do tempo entre chegadas;
$a(t)$	Função de densidade de probabilidade do tempo entre chegadas;
$B(t)$	Função de distribuição acumulada do tempo de serviço;
$b(t)$	Função de densidade de probabilidade do tempo de serviço;
$\mathbf{B}$	Vetor de áreas de espera das filas ($\equiv (B_1, B_2, \ldots, B_n)^{\mathrm{T}}$);
B_i	Área de espera da i-ésima fila da rede de filas ($\equiv K_i - 1$, em filas $M/G/1/K$);

c	Número de servidores;
CV^2	Quadrado do coeficiente de variação da variável aleatória S, tempo de serviço ($\equiv \mathrm{Var}(S)/\mathbb{E}(S)^2$);
$\mathbf{e}$	Vetor coluna com todos os elementos unitários ($\equiv (1, 1, \ldots, 1)^{\mathrm{T}}$);
D	Tempo determinístico entre chegadas;
E_k	Distribuição Erlang tipo-k de tempos entre chegadas;
$\mathbb{E}[.]$	Valor esperado;
FCFS	Disciplina de fila 'primeiro a chegar, primeiro a ser atendido' (do inglês *First-Come, First-Served*);
$f_i(\mathbf{x})$	Funções objetivo (com $i = 1, 2, \ldots, I$);
$\phi_j(\mathbf{x}) = 0$	Restrições de igualdade (com $j = 1, 2, \ldots, J$);
G	Distribuição geral do tempo entre chegadas e/ou do tempo de serviço;
$G(t)$	Função de distribuição acumulada do período ocupado para os modelos $M/G/1$ e $G/M/1$;
$G(N, A)$	Grafo direcionado (dígrafo);
I	Número de objetivos;
J	Número de restrições de igualdade;
K	(1) Número de restrições de desigualdade; (2) Limite da capacidade do sistema;
$\mathbf{K}$	Vetor dos limites de capacidade das filas ($\equiv (K_1, K_2, \ldots, K_n)^{\mathrm{T}}$);

K_i	Capacidade total da i-ésima fila da rede de filas ($\equiv B_i + 1$, em filas $M/G/1/K$);
L	Tamanho esperado do sistema;
LCFS	Disciplina de fila 'último a chegar, primeiro a ser atendido' (do inglês *Last-Come, First-Served*);
L_q	Tamanho esperado da fila;
$\boldsymbol{\lambda}$	Vetor de taxas de chegada externas das filas ($\equiv (\lambda_1, \lambda_2, \ldots, \lambda_n)^{\mathrm{T}}$);
λ_{efe}	Taxa de chegada efetiva (descontados os usuários perdidos);
λ_i	Taxa de chegada externa da i-ésima fila da rede de filas;
M	Processo de chegada e/ou de serviço Poisson (equivalentemente, tempo entre chegadas e/ou de serviço exponencial);
$\boldsymbol{\mu}$	Vetor de taxas de serviço das filas ($\equiv (\mu_1, \mu_2, \ldots, \mu_n)^{\mathrm{T}}$);
μ_i	Taxa de serviço da i-ésima fila da rede de filas;
N	Conjunto (finito) de nós (filas finitas);
$o(\Delta t)$	Ordem Δt, ou seja, $\lim_{\Delta t \downarrow 0} \dfrac{o(\Delta t)}{\Delta t} = 0$;
$\mathbf{P}$	Matriz de transição de um passo de uma cadeia de Markov de tempo discreto;
$\mathbf{P}^{(m)}$	Matriz de transição de m passos de uma cadeia de Markov de tempo discreto;
$\mathbf{p}$	Vetor de probabilidade estacionária de uma cadeia de Markov de tempo contínuo;

p_n	(1) Probabilidade estacionária de n usuários no sistema de filas; (2) Probabilidade estacionária que uma cadeia de Markov de tempo contínuo esteja no estado n;
p_{ij}	Probabilidade de transição do estado i ao estado j, em um passo;
$p_{ij}^{(m)}$	Probabilidade de transição do estado i ao estado j, em m passos;
p_K	Probabilidade de bloqueio (i.e., existir K usuários no sistema de filas);
$\boldsymbol{\pi}$	Vetor de probabilidades estacionárias de uma cadeia de Markov de tempo discreto;
π_n	(1) Probabilidade estacionária de n usuários no sistema de filas no instante de partida; (2) Probabilidade estacionária de que uma cadeia de Markov de tempo discreto esteja no estado n;
$\pi_n^{(t)}$	Probabilidade de que uma cadeia de Markov de tempo discreto esteja no estado n no tempo t independente do ponto de partida da cadeia de Markov;
$\psi_k(\mathbf{x}) \leq 0$	Restrições de desigualdade (com $k = 1, 2, \ldots, K$);
$\mathcal{R}^n$	Espaço de busca (espaço das variáveis de decisão);
r	Razão entre a taxa de entrada e a taxa de atendimento de uma fila ($\equiv \lambda/\mu$);
ρ	Intensidade de tráfego ($\equiv \lambda/\mu = r$, para servidor único, e $\equiv \lambda/c\mu$, para múltiplos servidores);

S_{OC}	Evento em que o cliente que chega em uma fila encontra o servidor ocupado;
θ	Taxa de saída do sistema (do inglês *throughput*);
θ_{lmr}	Taxa de saída limiar;
T_{SR}	Variável aleatória que representa o tempo de serviço residual para algum cliente já em atendimento pelo servidor de uma fila;
Var[.]	Variância;
W	Tempo esperado no sistema;
W_q	Tempo esperado na fila;
$X(t)$	Processo estocástico com espaço de estados X e parâmetro t;
$\mathbf{x}$	Vetor de decisão ou de projeto ($\equiv (x_1, x_2, \ldots, x_n)^{\text{T}}$);
x_i	Componentes do vetor de decisão ou de projeto.

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Conforme ressaltado por Yang [65], no seu recente livro, *Engineering Optimization: An Introduction with Metaheuristic Applications*, embora os problemas de otimização estejam presente em todo lugar, da engenharia à ciência da computação e do sequenciamento de tarefas à economia, constatar tal fato não torna a resolução de tais problemas mais fácil. Na verdade, problemas de descrição bastante simples podem ser muito difíceis de resolver. Tome-se por exemplo o problema do caixeiro viajante, no qual um vendedor precisa visitar, digamos, 50 cidades, exatamente uma única vez, em uma sequência tal que a distância total percorrida seja minimizada. A despeito da facilidade de definição e compreensão deste problema e da simplicidade do objetivo a ser minimizado, é de certa forma surpreendente que *não* se conheça ainda um algoritmo *eficiente* para ele. Os desenvolvimentos mais recentes que foram criados ao longo nas últimas duas décadas para o problema do caixeiro viajante tendem a usar algoritmos metaheurísticos. Na verdade, as mais modernas técnicas de otimização de maneira geral são usualmente heurísticas ou metaheurísticas, tais como o recosimento simulado, a otimização por enxame de partículas, a busca harmônica e os algoritmos genéticos. Estes algoritmos vêm se tornando bastante poderosos na resolução de difíceis problemas de

otimização, em todas as principais áreas da ciência e da engenharia. É sobre um destes problemas difíceis e um destes algoritmos metaheurísticos que trata esta tese.

1.1 Motivação

Sempre que há incerteza sobre o fluxo de produtos, usuários, mensagens, e assim por diante, com uma taxa de chegada λ, e incerteza quanto ao seu processamento, com uma taxa de serviço μ, tem-se como resultado um sistema de filas. O seu arranjo em uma configuração em rede é uma generalização bastante natural e relevante, pelos diversos sistemas reais que pode modelar.

Este trabalho trata da maximização do número de usuários atendidos por unidade de tempo (do inglês *throughput*), θ, em uma rede de filas acíclica (rede de filas em que os usuários que passaram por uma determinada estação de atendimento nunca retornam a mesma estação, ou seja não existe nenhuma possibilidade de o usuário retornar para alguma estação anteriormente visitada na rede de filas), com servidor único e tempos de serviço geral (por exemplo, veja Fig. 1.1). Em outras palavras, trata-se de redes de filas $M/G/1/K$, em que, conforme a notação de Kendall [38], M representa tempos entre chegadas independentes e com distribuição exponencial (markoviana), G representa tempos de serviço com distribuição geral (não especificada), "1" indica que há um único servidor para realizar os atendimentos e K é a capacidade total do sistema, *incluindo* o item em serviço (isto é, a área de espera em fila e o servidor).

O máximo θ é procurado simultaneamente com as menores capacidades, $\mathbf{K} = (K_1, K_2, \ldots, K_n)^{\mathrm{T}}$, e as menores taxas de serviço, $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)^{\mathrm{T}}$, que precisam ser alocadas à rede de filas, para uma topologia dada (para um exemplo, ver Fig. 1.1) e para um vetor de taxas de chegada externas

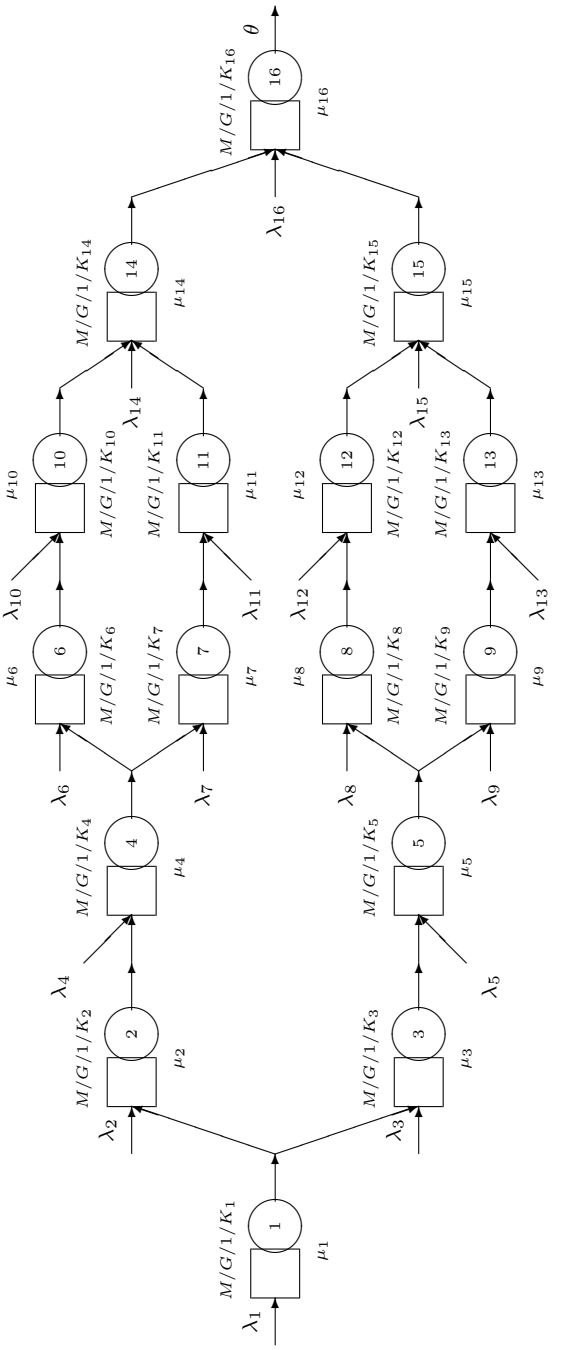


Figura 1.1: Uma rede complexa (adaptada de Smith & Cruz [59])

especificado, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_n)^{\mathrm{T}}$. Potenciais usuários destes modelos de otimização baseados em redes de filas finitas gerais incluem cientistas da computação e engenheiros de produção. De fato, tais modelos podem auxiliar na compreensão e na melhoria de vários sistemas reais, incluindo sistemas de manufatura [45, 36, 66, 28, 2], de produção [35, 3, 60] e de saúde [43, 23, 54], sistemas de tráfego de veículos e de pedestres [20, 22], sistemas de computação e de comunicação [1, 12, 64, 33], aplicações baseadas na *web* [10], e métodos para garantia de qualidade de serviço, medida em termos de tempo de resposta, taxa de atendimento, disponibilidade de serviço e segurança [51].

De fato, há um compromisso (*trade-off*) crítico entre a capacidade total, as taxas de serviço e a taxa de saída. Devido ao alto custo, a capacidade total e as taxas de serviço devem ser as menores possíveis. Infelizmente, a taxa de serviço é diretamente afetada pela capacidade, ou seja, um aumento da capacidade do sistema implica em uma taxa de atendimento em geral maior. De forma similar, a taxa de serviço afeta a taxa de atendimento, isto é, afeta-a diretamente.

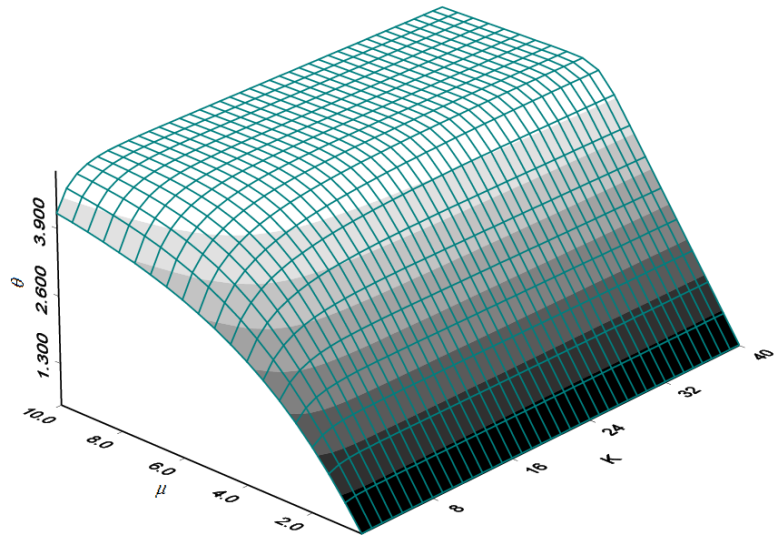
Um caso mais simples de estudo pode ser obtido através da análise de filas $M/M/1/K$, em que tanto os tempos entre chegadas seguem uma distribuição exponencial, sendo independentes entre si, quanto os tempos de serviço no único servidor da fila também são exponenciais e independentes. Entretanto em muitas situações de interesse, os tempos entre chegadas preservam a distribuição exponencial, mas os tempos de serviço não, ou seja, seguem uma distribuição geral. Tal situação pode ser exemplificada pelos sistemas denominados *hipoexponenciais* e *hiperexponenciais*. Em sistemas hipoexponenciais, o quadrado do coeficiente de variação (razão entre a variância e o quadrado da média) é menor que a unidade. Já os sistemas hiperexponenciais apresentam o quadrado do coeficiente de variação maior que a unidade. Vale

ressaltar que para os sistemas markovianos (tempos de serviço independentes com distribuição exponencial) o quadrado do coeficiente de variação é igual a 1. Os casos hipoexponenciais e hiperexponenciais são utilizados nesta tese como casos particulares de filas $M/G/1/K$.

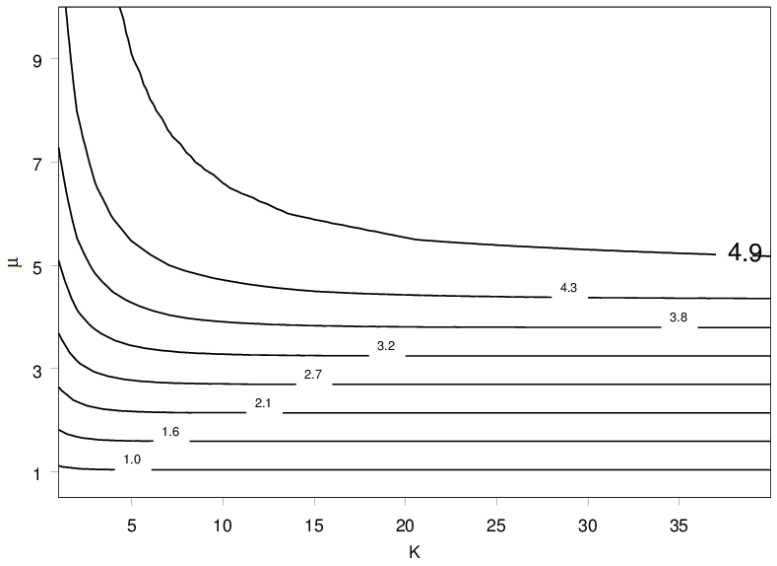
1.2 Escopo e objetivos da tese

A Figura 1.2-a mostra a taxa de saída, θ, em função da capacidade do sistema, K, e da taxa de serviço, μ (ver Equações 4.1 e 4.3, na pág. 69), para uma única fila $M/G/1/K$, com quadrado do coeficiente de variação para a variável aleatória tempo de serviço (representado por CV^2) igual a 1,5 (variabilidade maior, portanto, que aquela apresentada por uma distribuição exponencial), e taxa de chegada externa λ igual a 5 usuários por unidade de tempo. As respetivas curvas de nível (contorno) podem ser vistas na Figura 1.2-b. Um comportamento similar é também observado em redes de filas, conforme será mostrado mais adiante. O gráfico de curvas de nível mostrado na Fig. 1.2-b é suave e tem uma concavidade aparente, semelhante aos resultados observados em redes de filas [56, 50]. No entanto, a parte superior da superfície 3-d (Figura 1.2-a) é visivelmente plana, o que dificulta bastante a utilização de métodos de otimização tradicionais. Por exemplo, Smith & Cruz [59] foram bem sucedidos com o emprego do método Powell, mas precisaram de usar múltiplos pontos de partida, para evitar a convergência prematura para um ótimo local e obter um algoritmo de otimização bem sucedido.

O tipo de problema de interesse é o de desenvolvimento de algoritmos para otimização da taxa de saída de uma rede de filas finitas gerais, simultaneamente com a alocação ótima de capacidades e taxas de serviço. Busca-se a alocação ótima e o planejamento da topologia ótima, incluindo a determinação da ordem dos servidores, bem como a iteração destes dois fatores.



(a) taxa de saída (θ) *versus* taxa de serviço (μ) e área de espera (K)



(b) curvas de nível

Figura 1.2: Resultados para fila $M/G/1/K$ única com $\lambda = 5$

A questão colocada nesta tese é de como é possível modelar sistemas reais envolvendo incerteza no fluxo de produtos e no seu processamento, predizer com precisão medidas de desempenho dos modelos e planejar adequadamente tais sistemas para que tenham desempenho ótimo.

Nesta tese procura-se caracterizar e otimizar a topologia de um sistema de redes de filas finitas gerais, via formulação multiobjetivo. Procura-se por propriedades que permitam modelar e construir algoritmos para otimizá-los. Trabalhos anteriores são estendidos, nos quais redes de filas finitas com um único servidor foram consideradas apenas na abordagem mono-objetivo. Assim, para a formulação multiobjetivo, precisa-se compreender como as taxas de serviço influenciam as capacidades ótimas e como as várias topologias e os valores dos quadrados dos coeficientes de variação do tempo de serviço podem influenciar a configuração ótima do sistema de filas finitas gerais configuradas em redes.

1.3 Contribuições

As principais contribuições da tese consistem nos seguintes aspectos.

- Uma extensiva e atualizada revisão bibliográfica sobre os avanços na área de otimização de redes de filas gerais é apresentada.

- Uma formulação multiobjetivo alternativa para o problema de minimização de áreas de espera (*buffers*) é estudada em detalhes. Essa formulação substitui com vantagens algumas formulações existentes, por permitir a obtenção de novas informações a respeito das redes de filas.

- Algoritmos para estimação das medidas de desempenho são descritos em detalhes. Em particular é detalhado o método da expansão generalizada (GEM, do inglês *Generalized Expansion Method*), que permite a

determinação acurada de medidas de desempenho da rede de filas, via combinação de uma decomposição nó-a-nó e tentativas repetidas.

- Um algoritmo multiobjetivo é proposto para a determinação de uma configuração sub-ótima de menor custo para as áreas de espera e as taxas de serviço, assegurando simultaneamente a maximização da taxa de saída global da rede. É um algoritmo do tipo evolucionário, bastante eficaz e eficiente na resolução de problemas similares.

- É feito um extensivo estudo empírico sobre a sintonia da configuração de inicialização do algoritmo de otimização multiobjetivo, em que é também atestada a robustez deste algoritmo. De fato, o desempenho do algoritmo, em termos de tempo de processamento até a convergência, apresenta-se pouco dependente da instância do problema que é resolvida.

- É mostrada a similaridade entre o comportamento de uma rede de filas finitas, com o de uma fila finita simples.

- Resultados são obtidos com a metodologia multiobjetivo para uma rede extensa (com 16 filas), em configuração mista, com divisão e fusão de fluxos. Os tempos de processamento apresentaram-se razoáveis.

- É demonstrado como os resultados obtidos podem auxiliar a análise e o planejamento de redes de filas finitas gerais.

- Apresentam-se e discutem-se possíveis formas de melhoria dos resultados obtidos, com a sugestão de possíveis direções para a continuidade da pesquisa nesta área.

1.4 Organização

Este texto encontra-se organizado como se segue. No Capítulo 2 apresentam-se conceitos fundamentais relevantes para a melhor compreensão dos problemas de alocação de área de espera em filas finitas configuradas em redes. No Capítulo 3 é descrita a formalização deste problema, como um problema de programação matemática inteira estocástica e também a motivação para a escolha de tal formulação. Os algoritmos de resolução são apresentados detalhadamente no Capítulo 4. No Capítulo 5 são apresentados e discutidos os resultados dos experimentos computacionais realizados. Várias das redes têm seus resultados analisados de forma detalhada. Finalmente, no Capítulo 6 as principais conclusões são apresentadas, juntamente com a sugestão de tópicos para trabalhos futuros nesta área, que demonstra ter carências.

Capítulo 2

Introdução à Teoria de Filas

Este capítulo tem por objetivo apresentar conceitos associados à teoria de filas. Entretanto, para uma facilidade de leitura, é necessária a revisão de conceitos mais elementares da área de processos estocásticos, o que será feito a seguir.

2.1 Processos estocásticos e cadeias de Markov

Um processo estocástico é uma abstração matemática de um processo empírico cujo desenvolvimento é governado por leis probabilísticas. Do ponto de vista da teoria matemática de probabilidade, um processo estocástico é melhor definido como uma família de variáveis aleatórias $\{X(t), t \in T\}$ definidas sobre um conjunto de índices ou espaço paramétrico T. O conjunto T é algumas vezes chamado de intervalo de tempo e $X(t)$ denota o estado do processo no tempo t. Dependendo da natureza de T, o processo é classificado como de parâmetro discreto ou de parâmetro contínuo.

- Se T é uma sequência enumerável, por exemplo, $T = \{0, \pm 1, \pm 2 \dots\}$ ou $T = \{0, 1, 2, \dots\}$, então o processo estocástico $\{X(t), t \in T\}$ é chamado de processo de parâmetro discreto definido no conjunto de índices T.

- Se T é um intervalo ou uma combinação algébrica de intervalos, por exemplo, $T = \{t : -\infty < t < +\infty\}$ ou $T = \{t : 0 < t < +\infty\}$, então o processo estocástico $\{X(t), t \in T\}$ é chamado de processo de parâmetro contínuo definido no conjunto de índices T.

2.1.1 Processo de Markov

Um processo estocástico de parâmetro discreto $\{X(t), t = 0, 1, 2, \dots\}$ ou de parâmetro contínuo $\{X(t), t > 0\}$ é chamado de processo de Markov se, para qualquer conjunto de n pontos $t_1 < t_2 < \cdots < t_n$ no conjunto de índices ou intervalo de índices do processo, a distribuição condicional de $X(t_n)$, dados os valores de $X(t_1), X(t_2), X(t_3), \dots, X(t_{n-1})$, depende apenas do valor imediatamente precedente, $X(t_{n-1})$; mais precisamente, para quaisquer números reais $x_1, x_2, \dots, x_n$,

$$P\{X(t_n) \leq x_n | X(t_1) = x_1, \dots, X(t_{n-1}) = x_{n-1}\} =$$
$$P\{X(t_n) \leq x_n | X(t_{n-1}) = x_{n-1}\}.$$

Em linguagem não matemática isso significa que, dada a condição "presente" do processo, o "futuro" é independente do "passado" e o processo tem, então, perda de memória.

Os processos de Markov são classificados conforme:

1. a natureza do conjunto de índices do processo (se o conjunto de índices é discreto ou contínuo);

2. a natureza do estado de espaços do processo (se o espaço de estados é discreto ou contínuo).

Diz-se que um número real x é algum dos estados de um processo estocástico $\{X(t), t \in T\}$ se existe um ponto no tempo t tal que a probabilidade

$P\{x - h < X(t) < x + h\}$ é não nula para todo $h > 0$. O conjunto de todos os estados possíveis constitui o espaço de estados do processo. Se o espaço de estados é *discreto*, o processo de Markov é geralmente chamado de *cadeia de Markov*, apesar de alguns autores reservarem o termo "cadeia" somente para aqueles processos com espaço de estados e espaço de parâmetro, ambos discretos. Neste texto, diremos que um processo de Markov de parâmetro discreto com espaço de estados discreto é uma cadeia de Markov plana e que um processo de Markov de parâmetro contínuo com espaço de estados discreto é uma cadeia de Markov de tempo contínuo. (Extensões multivariadas podem ser formuladas para vetores de estado $\mathbf{x}$.)

Uma cadeia de Markov é finita se o espaço de estados é finito; caso contrário, ela é infinita. Uma vez que um processo de parâmetro discreto é observado em uma quantidade enumerável de pontos no tempo, sejam as sucessivas observações denotadas por $X_0, X_1, X_2, \ldots, X_n$, em que X_n é a variável aleatória cujos valores representam o estado do sistema no n-ésimo ponto do tempo. Uma sequência arbitrária de variáveis aleatórias $\{X_n\}$ é uma cadeia de Markov se cada variável aleatória X_n é discreta e vale a seguinte propriedade:

Para qualquer valor inteiro $m > 2$ e qualquer conjunto de m pontos $n_1 < n_2 < \cdots < n_m$, a distribuição condicional de X_{n_m}, dados valores de $X_{n_1}, X_{n_2}, \ldots, X_{n_{m-1}}$, depende apenas de $X_{n_{m-1}}$, o valor imediatamente precedente, isto é,

$$P\{X_{n_m} = x_{n_m} | X_1 = x_{n_1}, \ldots, X_{n_{m-1}} = x_{n_{m-1}}\} = P\{X_{n_m} = x_{n_m} | X_{n_{m-1}} = x_{n_{m-1}}\}.$$

Uma importante generalização da cadeia de Markov, que é muito usada em teoria de filas, é o semi-processo de Markov (SMP, do inglês *semi-Markov*

process) ou processo de renovação de Markov (MRP, do inglês *Markov renewal process*). As transições de estado em um SMP formam uma cadeia de Markov discreta, mas os tempos entre sucessivas transições são variáveis aleatórias. Se estas variáveis aleatórias têm distribuição exponencial, no caso de parâmetro contínuo, ou distribuição geométrica, no caso discreto, com média dependente apenas do estado atual, o SMP reduz-se a um processo de Markov devido à falta de memória dessas variáveis aleatórias.

2.1.2 Cadeia de Markov de parâmetro discreto

Considere uma sequência $\{X_n; n = 0, 1, 2, \dots\}$ de variáveis aleatórias com $X_n \in \{0, 1, 2, \dots\}$, formando uma cadeia de Markov com espaço de parâmetros discreto, isto é, para todo n,

$$P\{X_n = j | X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}\} = P\{X_n = j | X_{n-1} = i_{n-1}\}.$$

Se o valor da variável aleatória X_n é j, então diz-se que o sistema está no estado j após n passos de transições. Nesse caso, as probabilidades condicionais $P\{X_n = j | X_{n-1} = i\}$ são chamadas *probabilidades de transição de um passo do estado i para o estado j* ou simplesmente *probabilidades de transição de i para j*. Se estas probabilidades são independentes de n, então diz-se que a cadeia é *homogênea* e as probabilidades $P\{X_n = j | X_{n-1} = i\}$ podem ser escritas como p_{ij}. A matriz definida através dos elementos p_{ij} na posição (i, j) é conhecida como *matriz de transição*, sendo denotada por $\mathbf{P}$. Para cadeias homogêneas, as probabilidades de transição de m passos $P\{X_{n+m} = j | X_n = i\} = p_{ij}^{(m)}$ também são independentes de n. A probabilidade não condicional do estado j na n-ésima tentativa será escrita como $P\{X_n = j\} = \pi_j^{(n)}$, tal que a distribuição inicial é dada por $P\{X_0 = j\} = \pi_j^{(0)}$.

Das leis básicas de probabilidade, podemos mostrar que a matriz $\mathbf{P}^{(m)}$, formada pelos elementos $\left\{p_{ij}^{(m)}\right\}$ pode ser obtida multiplicando $\mathbf{P}^{(m-k)}$ por $\mathbf{P}^{(k)}$ para qualquer valor de k, com $0 < k < m$. Esta matriz é equivalente a obtida através das clássicas equações de Chapman-Kolmogorov para este processo de Markov, qual seja, $p_{ij}^{(m)} = \sum_r p_{ir}^{(m-k)} p_{rj}^{(k)}, (0 < k < m)$, ou em notação matricial,

$$\mathbf{P}^{(m)} = \mathbf{P}^{(m-k)}\mathbf{P}^{(k)}. \tag{2.1}$$

Tomando $k = m - 1$ na Equação (2.1), temos:

$$\mathbf{P}^{(m)} = \mathbf{P}.\mathbf{P}^{(m-1)}, \tag{2.2}$$

que, recursivamente, fornece $\mathbf{P}^{(m)} = \mathbf{P}.\mathbf{P}\ldots\mathbf{P} = \mathbf{P}^m$. Por esta razão $\mathbf{P}^{(m)}$ pode ser obtida multiplicando a matriz $\mathbf{P}$ por ela mesma m vezes.

Frequentemente, estamos interessados nas probabilidades de a cadeia estar no estado j depois de m transições, independentemente do estado inicial. Se o vetor $\boldsymbol{\pi}^{(m)}$ têm coordenadas sendo as probabilidades $\left\{\pi_j^{(m)}\right\}$, então temos:

$$\boldsymbol{\pi}^{(m)} = \boldsymbol{\pi}^{(m-1)}\mathbf{P}, \tag{2.3}$$

que, recursivamente, fornece

$$\boldsymbol{\pi}^{(m)} = \boldsymbol{\pi}^{(0)}\mathbf{P}^m, \tag{2.4}$$

para o vetor $\boldsymbol{\pi}^{(0)}$ do estado inicial.

2.1.3 Cadeia de Markov de parâmetro contínuo

Consideremos agora uma cadeia de Markov de parâmetro contínuo $\{X(t), t \in T\}$, para $T = \{t : 0 \leq t < \infty\}$. Considere quaisquer tempos u, t e s com

$0 \leq u < t < s$ e os estados i e j, então:

$$p_{ij}(u, s) = \sum_r p_{ir}(u, t)p_{rj}(t, s), \tag{2.5}$$

em que $p_{ij}(u, s)$ é a probabilidade de ir do estado i para o estado j no tempo começando em u e terminando em s, e o somatório é sobre todos os estados da cadeia. Esta é a equação de Chapman-Kolmogorov para o processo contínuo (análoga à Equação (2.1) para o processo discreto).

Em notação matricial, a Equação (2.5) pode ser escrita da seguinte forma:

$$P(u, s) = \mathbf{P}(u, t)\mathbf{P}(t, s).$$

Fazendo $u = 0$ e $s = t + \Delta t$ na Equação (2.5) vem:

$$p_{ij}(0, t + \Delta t) = \sum_r p_{ir}(0, t)p_{rj}(t, t + \Delta t).$$

Definindo $p_i(0)$ como a probabilidade de a cadeia iniciar no estado i no tempo 0 e $p_j(t)$ como a probabilidade incondicional de a cadeia estar no estado j no tempo t, indiferentemente do estado de início, podemos multiplicar a equação acima por $p_i(0)$ e, da soma sobre todos os estados, obter:

$$\sum_i p_i(0)p_{ij}(0, t + \Delta t) = \sum_r \sum_i p_{ir}(0, t)p_i(0)p_{rj}(t, t + \Delta t),$$

ou

$$p_j(t + \Delta t) = \sum_r p_r(t)p_{rj}(t, t + \Delta t). \tag{2.6}$$

Para o processo de Poisson tratado anteriormente, temos:

$$p_{rj}(t, t+\Delta t) = \begin{cases} \lambda\Delta t + o(\Delta t), & \text{para } r = j-1 \text{ e } j \geq 1, \\ 1 - \lambda\Delta t + o(\Delta t), & \text{para } r = j, \\ o(\Delta t), & \text{caso contrário.} \end{cases}$$

Substituindo esta última expressão na Equação (2.6), vem que $p_j(t + \Delta t) = [\lambda\Delta t + o(\Delta t)]p_{j-1}(t) + [1 - \lambda\Delta t + o(\Delta t)]p_j(t) + o(\Delta t),$ para $j \geq 1$.

Se as funções de transição de probabilidade $p_{ij}(u, s)$ da cadeia tem a propriedade da existência de funções contínuas $q_i(t)$ e $q_{ij}(t)$ tais que:

$$\begin{aligned} &P\{X(t+\Delta t) - X(t) \neq 0\} = 1 - p_{ii}(t, t+\Delta t) = q_i(t)\Delta t + o(\Delta t) \text{ e} \\ &p_{ij}(t, t+\Delta t) = q_{ij}(t)\Delta t + o(\Delta t). \end{aligned} \tag{2.7}$$

Para uma matriz $\mathbf{Q}(t)$ satisfazendo $\mathbf{Q}(t)\Delta t = \mathbf{P}(t, t+\Delta t) - I$ em que I é a matriz identidade, teríamos, a menos de $o(\Delta t)$:

$$\mathbf{Q}(t) = \begin{bmatrix} -q_0(t) & q_{01}(t) & q_{02}(t) & q_{03}(t) & \dots \\ q_{10}(t) & -q_1(t) & q_{12}(t) & q_{13}(t) & \dots \\ q_{20}(t) & q_{21}(t) & -q_2(t) & q_{23}(t) & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Se a matriz $\mathbf{Q}$ satisfaz a condição de $\mathbf{Q}(u) = \mathbf{Q}(t)\mathbf{P}(u, t)$, ou em outros termos, $q_{ij}(u) = \sum_{r \neq j} p_{ir}(u, t)q_{rj}(t) - p_{ij}(u, t)q_j(t)$, tem-se que

$$\frac{\partial}{\partial t}p_{ij}(u, t) = \lim_{\Delta t \downarrow 0} \frac{p_{ij}(u, t+\Delta t) - p_{ij}(u, t)}{\Delta t} = \lim_{\Delta t \downarrow 0} \frac{q_{ij}(u)\Delta t - q_{ij}(u) \times 0}{\Delta t} = q_{ij}(u),$$

e, portanto

$$\frac{\partial}{\partial t}p_{ij}(u, t) = -q_j(t)p_{ij}(u, t) + \sum_{r \neq j} p_{ir}(u, t)q_{rj}(t). \tag{2.8}$$

Considere a Equação (2.8), seja $u = 0$ e assuma um processo homogêneo, tal que $q_i(t) = q_i$ e $q_{ij}(t) = q_{ij}, \forall t$. Então,

$$\frac{dp_{ij}(0,t)}{dt} = -q_j p_{ij}(0,t) + \sum_{r \neq j} p_{ir}(0,t) q_{rj}.$$

Multiplicando ambos os lados da equação anterior por $p_i(0)$ e somando sobre todos os valores de i, vem:

$$\frac{dp_j(t)}{dt} = -q_j p_j(t) + \sum_{r \neq j} p_r(t) q_{rj},$$

que, em notação matricial é

$$\mathbf{p}'(t) = \mathbf{p}(t)\mathbf{Q}, \tag{2.9}$$

em que $\mathbf{p}(t)$ é o vetor $(p_0(t), p_1(t), \dots)$, $\mathbf{p}'(t)$ é a derivada desse vetor e $\mathbf{Q}$ independe de t.

Note que, da Equação (2.7), $q_i = \sum_{j \neq i} q_{ij}$, pois $\sum_j p_{ij}(t, t + \Delta t) = 1$, o que implica que $1 - q_i \Delta t + o(\Delta t) + \sum_{j \neq i} [q_{ij} \Delta t + o(\Delta t)] = 1$, ou ainda que $-q_i \Delta t + o(\Delta t) = -\sum_{j \neq i} [q_{ij} \Delta t + o(\Delta t)]$, tal que $q_i = \sum_{j \neq i} q_{ij}$.

Referindo-nos novamente ao processo de Poisson, podemos usar a Equação (2.9), observando que $q_j = \lambda$, $q_{rj} = \lambda$, para $r = j - 1, j \geq 1$, e $q_{ij} = 0$, caso contrário. Para obter diretamente, $\frac{dp_j(t)}{dt} = \lambda p_j(t) + \lambda p_{j-1}(t)$.

Como o espaço de estados é composto de inteiros não-negativos (representando o número de clientes presentes), uma grande percentagem de problemas de filas pode ser categorizado como cadeias de Markov de parâmetro de tempo contínuo. Muitos desses modelos têm a propriedade adicional de nascimento e morte tal que um estado só pode dar um passo para a frente, um passo para trás ou permanecer no estado em que se encontra, assim:

$$P\{X(t+\Delta t) - X(t) = 1 | X(t) = n\} = \lambda_n \Delta t + o(\Delta t), \text{ para } n \geq 0,$$
$$P\{X(t+\Delta t) - X(t) = -1 | X(t) = n\} = \mu_n \Delta t + o(\Delta t), \text{ para } n \geq 1,$$
$$P\{X(t+\Delta t) - X(t) = 0 | X(t) = n\} = 1 - (\lambda_n + \mu_n)\Delta t + o(\Delta t), \text{ para } n \geq 1.$$

Este resultado leva a $q_{n,n+1} = \lambda_n$, também a $q_{n,n-1} = \mu_n$ considerando $\mu_n \neq 0$, também a $q_n = \lambda_n + \mu_n$ para $q_0 = \lambda_0$ e ainda $q_{rj} = 0$ caso contrário.

Substituindo por q_i e q_{ij}, a matriz $\mathbf{Q}(t)$ $\forall t$ passa a ser

$$\mathbf{Q} = \begin{bmatrix} -\lambda_0 & \lambda_0 & 0 & 0 & \dots \\ \mu_1 & -(\lambda_1 + \mu_1) & \lambda_1 & 0 & \dots \\ 0 & \mu_2 & -(\lambda_2 + \mu_2) & \lambda_2 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

2.1.4 Comportamento de longo prazo do processo de Markov

Na maioria das vezes estamos interessados no comportamento de um processo de Markov após um longo período de tempo, particularmente se esse comportamento se "estabiliza" probabilisticamente. Serão discutidos três conceitos relacionados ao comportamento de longo prazo, quais sejam: *distribuições limite*, *distribuições estacionárias* e *ergodicidade*.

Considere uma cadeia de Markov de parâmetro discreto e suponha existir π_j sendo a probabilidade de o processo estar no estado j quando o comprimento do prazo de execução do processo tende a infinito. Para tal situação temos então $\lim_{m\to\infty} p_{ij}^{(m)} = \pi_j$, para todo i. Isto é, após um longo tempo, a probabilidade de que o processo esteja no estado j, dado que ele começou no estado i, é independente do estado inicial i. Isto significa que $\mathbf{P}^m$ se aproxima de um limite, à medida que m tende a infinito, a saber, que todas as linhas de $\mathbf{P}^m$ tornam-se iguais. Neste caso, as probabilidades $\{\pi_j\}$ são chamadas

de *probabilidades de estado estacionário* ou *probabilidades limite* da cadeia de Markov.

Considere agora as probabilidades incondicionais de estado após m passos dadas por $\boldsymbol{\pi}^{(m)} = \boldsymbol{\pi}^{(0)}\mathbf{P}^m$, Equação (2.4), isto é, $\pi_j^{(m)} = \sum_i \pi_i^{(0)} p_{ij}^{(m)}$, então:

$$\lim_{m\to\infty} \pi_j^{(m)} =$$
$$\lim_{m\to\infty} \sum_i \pi_i^{(0)} p_{ij}^{(m)} = \sum_i \pi_i^{(0)} \lim_{m\to\infty} p_{ij}^{(m)} = \sum_i \pi_i^{(0)} \pi_j = \pi_j \sum_i \pi_i^{(0)} = \pi_j.$$

Daí $\pi_j^{(m)}$ converge para o mesmo limite $p_{ij}^{(m)}$ e é independente das probabilidades do estado inicial e do parâmetro de tempo m. Quando estas probabilidades incondicionais limitantes existem, elas podem ser obtidas como segue.

Da Equação (2.3), vem $\lim_{m\to\infty} \boldsymbol{\pi}^{(m)} = \lim_{m\to\infty} \boldsymbol{\pi}^{(m-1)}\mathbf{P}$. Assim, fazendo $\boldsymbol{\pi} = (\pi_0, \pi_1, \dots)$ representar o vetor limite, temos $\lim_{m\to\infty} \boldsymbol{\pi}^{(m)} = \lim_{m\to\infty} \boldsymbol{\pi}^{(m-1)} = \boldsymbol{\pi}$, tal que:

$$\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{P}. \tag{2.10}$$

Disto, junto com a condição de fronteira, $\sum_j \pi_j = 1$, podemos obter $\{\pi_j\}$. Estas condições, bem conhecidas, são chamadas *equações estacionárias* da cadeia de Markov e sua solução é chamada *distribuição estacionária*. Note que a condição de fronteira pode ser escrita em notação vetorial como $\boldsymbol{\pi}\mathbf{e} = 1$, sendo $\mathbf{e}$ um vetor coluna com todos os elementos iguais a 1.

2.1.5 Ergodicidade

Intimamente associado com os conceitos de distribuições limite e estacionária está a ideia de *ergodicidade*. A ergodicidade é importante na medida em que lida com problemas de determinação das medidas de um processo estocástico

$X(t)$ de uma única realização, como é frequentemente feito em análise de saída de simulação. Um processo estocástico $X(t)$ é ergódico, no senso mais geral, se, com probabilidade 1, todas as suas "medidas" podem ser determinadas ou bem aproximadas por uma única realização, $x_0(t)$, do processo. Um processo estocástico é ergódico se o valor esperado é calculado "através da média do conjunto".

Em geral, não é fácil mostrar ergodicidade através de métodos diretos. Antes de enunciar os teoremas fundamentais que nos permitirão determinar quando uma solução para a equação estacionária existir, se uma distribuição limite existe e quando o processo é ergódico ou não, apresentaremos algumas definições necessárias para caracterizar cadeias de Markov de parâmetro discreto.

Dois estados i e j se *comunicam* entre si ($i \leftrightarrow j$) se i pode ser acessado de j ($j \rightarrow i$) e j pode ser acessado de i ($i \rightarrow j$), ou seja, existem n e m inteiros positivos, tais que a probabilidade de ir do estado i para o estado j em n passos é não nula, assim como, a probabilidade de ir do estado j para o estado i em m passos é não nula.

Uma cadeia é chamada *irredutível* se todos os seus estados se comunicam, isto é, se existe algum n_{ij} inteiro positivo, tal que $p_{ij}^{(n_{ij})} > 0$, para todos os pares (i, j).

O período de retorno a um estado k de uma cadeia é definido como o maior divisor comum (MDC) do conjunto de inteiros $\{n\}$ para os quais $p_{kk}^{(n)} > 0$. Um estado é chamado aperiódico se o MDC do conjunto de valores n definido anteriormente é igual a 1, isto é, se seu período é igual a 1. Diz-se que uma cadeia é aperiódica se cada um de seus estados é aperiódico.

Defina $f_{jj}^{(n)}$ como a probabilidade que uma cadeia, começando no estado j retornar para j pela primeira vez em n transições. Por isso, a probabilidade

de a cadeia retornar sempre a j é $f_{jj} = \sum_{n=1}^{\infty} f_{jj}^{(n)}$.

Se $f_{jj} = 1$, então diz-se que j é um estado *recorrente*. Por sua vez, se $f_{jj} < 1$, diz-se que j é um estado *transiente*. Quando $f_{jj} = 1$, o tempo médio de recorrência, denotado por m_{jj} é dado por $m_{jj} = \sum_{n=1}^{\infty} n f_{jj}^{(n)}$.

Se $m_{jj} < \infty$, então j é conhecido como *estado recorrente positivo*. Se $m_{jj} = \infty$, então j é um *estado recorrente nulo*.

Defina $f_{ij}^{(n)}$, $i \neq j$, como a probabilidade de a primeira passagem do estado i para o estado j ocorrer em n passos. Então, a probabilidade de o estado j ter vindo de i é $f_{ij} = \sum_{n=1}^{\infty} f_{ij}^{(n)}$.

Existe uma extensa lista de teoremas na literatura que permitem determinar a presença de recorrência em uma cadeia de Markov e calcular o tempo médio de recorrência. Os teoremas a seguir, cujas provas são omitidas, relacionam os conceitos de ergodicidade, probabilidades limite e probabilidades estacionárias para cadeias de estado discreto.

Teorema 2.1

1. *Em uma cadeia de Markov de parâmetro discreto irredutível e recorrente positiva, uma solução não-degenerada para as equações estacionárias* $\boldsymbol{\pi} = \boldsymbol{\pi P}$, $\boldsymbol{\pi e} = 1$ *sempre existe, quando o vetor* $\boldsymbol{\pi} = \{\pi_j\}$ *é tal que* $\pi_j = 1/m_{jj}$.

2. *Se o vetor de probabilidades de partida* $\boldsymbol{\pi}^{(0)}$ *é um conjunto igual ao vetor de probabilidades estacionárias* $\boldsymbol{\pi}$, *a cadeia acima torna-se um processo estocástico estacionário e, por isso, ergódico.*

3. *Se a cadeia é aperiódica, bem como irredutível e recorrente positiva, então o processo é ergódico e possui uma distribuiçao de probabilidade limite igual à distribuição estacionária.*

Note que a existência de uma distribuição limite é a condição mais forte, ergodicidade é um pouco mais fraca e a solução não-degenerada para as equações estacionárias é a mais fraca das três condições.

Teorema 2.2 *Uma cadeia irredutível, aperiódica é recorrente positiva se existe uma solução não-negativa do sistema* $\sum_{j=0}^{\infty} p_{ij}x_j \leq x_i - 1$, *para* $i \neq 0$, *tal que* $\sum_{j=0}^{\infty} p_{0j}x_j < \infty$.

2.2 O Processo de Poisson e a distribuição exponencial

São bastante comuns os casos de estudo de processos estocáticos em que os tempos entre chegadas e os tempos de serviço obedecem a uma distribuiçao exponencial ou, equivalentemente, que o número de chegadas no sistema o número de atendimento por unidade de tempo seguem uma distribuição de Poisson. Nesta seção será derivada a distribuição de Poisson e mostrado que assumir que o número de ocorrências em algum intervalo de tempo é uma variável aleatória de Poisson é equivalente a assumir que o tempo entre ocorrências sucessivas é uma variável aleatória com distribuição exponencial.

Seja $\{N(t), t \geq 0\}$ a variável aleatória que contabiliza o número de chegadas até o instante de tempo t, com $N(0) = 0$, que satisfaz as seguintes hipóteses:

1. A probabilidade de uma chegada ocorrer entre os instantes t e $t + \Delta t$ é igual a $\lambda\Delta t + o(\Delta t)$, em que λ é uma constante independente de $N(t)$, Δt é um elemento incremental e $o(\Delta t)$ denota a quantidade que pode ser negligenciada na comparação com Δt, quando $\Delta t \downarrow 0$; isto é, $\lim_{\Delta t \downarrow 0} \frac{o(\Delta t)}{\Delta t} = 0$.
2. A probabilidade de ocorrer mais que uma chegada entre os instantes t e $t + \Delta t$ é igual a $o(\Delta t)$.

3. Os números de chegadas em intervalos não sobrepostos são estatisticamente independentes, isto é, o processo tem incrementos independentes.

Busca-se calcular a probabilidade $p_n(t)$ de ocorrerem n chegadas em um intervalo de tempo de comprimento t, sendo $n \geq 0$ inteiro. Para $n \geq 1$ temos:

$$p_n(t + \Delta t) = \sum_{i=0}^{n} P\{N(t + \Delta t) - N(T) = i | N(t) = n - i\}$$

Usando as hipóteses 1, 2 e 3 acima, vem:

$$p_n(t + \Delta t) = p_n(t)[1 - \lambda\Delta t - o(\Delta t)] + p_{n-1}(t)[\lambda\Delta t + o(\Delta t)] + o(\Delta t), \quad (2.11)$$

em que o último termo, $o(\Delta t)$, representa a soma dos termos do somatório de probabilidades $P\{N(t + \Delta t) - N(T) = i | N(t) = n - i\}$, para $2 \leq i \leq n$.

Para o caso $n = 0$, temos:

$$p_0(t + \Delta t) = p_0(t)[1 - \lambda\Delta t - o(\Delta t)]. \quad (2.12)$$

Reescrevendo as Equações (2.11) e (2.12) e combinando todos os valores $o(\Delta t)$, temos:

$$p_0(t + \Delta t) - p_0(t) = -\lambda\Delta t p_0(t) + o(\Delta t) \quad (2.13)$$

e

$$p_n(t + \Delta t) - p_n(t) = -\lambda\Delta t p_n(t) + \lambda\Delta t p_{n-1}(t) + o(\Delta t), \text{ para } n \geq 1. \quad (2.14)$$

Dividindo as Equações (2.13) e (2.14) por Δt e tomando o limite quando $\Delta t \downarrow 0$, temos:

$$\begin{aligned}
\lim_{\Delta t \downarrow 0} \left[\frac{p_0(t + \Delta t) - p_0(t)}{\Delta t} \right] &= -\lambda p_0(t) + \lim_{\Delta t \downarrow 0} \frac{o(\Delta t)}{\Delta t}, \\
\lim_{\Delta t \downarrow 0} \left[\frac{p_n(t + \Delta t) - p_n(t)}{\Delta t} \right] &= -\lambda p_n(t) + \lambda p_{n-1}(t) + \lim_{\Delta t \downarrow 0} \frac{o(\Delta t)}{\Delta t}, \text{ para } n \geq 1,
\end{aligned}$$

reduzindo-se portanto a:

$$\frac{dp_0(t)}{dt} = -\lambda p_0(t), \text{ e} \tag{2.15}$$

$$\frac{dp_n(t)}{dt} = -\lambda p_n(t) + \lambda p_{n-1}(t), \text{ para } n \geq 1. \tag{2.16}$$

A equação diferencial ordinária de primeira ordem, Equação (2.16), tem solução geral $p_0(t) = Ce^{-\lambda t}$, em que a constante C é igual a 1, pois $p_0(0) = 1$. Agora, seja $n = 1$ na Equação (2.16). Então:

$$\frac{dp_1(t)}{dt} = -\lambda p_1(t) + \lambda p_0(t) \Longrightarrow \frac{dp_1(t)}{dt} + \lambda p_1(t) = \lambda e^{-\lambda t}.$$

A solução desta equação é $p_1(t) = Ce^{-\lambda t} + \lambda t e^{-\lambda t}$. Usando a condição $p_1(0) = 0$, temos $C = 0$, então $p_1(t) = \lambda t e^{-\lambda t}$.

Assumindo $p_{n-1}(t) = \dfrac{(\lambda t)^{n-1}}{(n-1)!}e^{-\lambda t}$, pode-se obter através do princípio da indução matemática uma expressão para $p_n(t)$. A equação diferencial ordinária de primeira ordem, Equação (2.16), torna-se:

$$\frac{dp_n(t)}{dt} + \lambda p_n(t) = \lambda \frac{(\lambda t)^{n-1}}{(n-1)!}e^{-\lambda t}, \text{ para } n \geq 1,$$

com solução geral $p_n(t) = Ce^{-\lambda t} + \dfrac{(\lambda t)^n}{(n)!}e^{-\lambda t}$.

Usando a condição $p_n(0) = 0, \forall n > 0$, temos $C = 0$, então:

$$p_n(t) = \frac{(\lambda t)^n}{n!}e^{-\lambda t}. \tag{2.17}$$

O processo de Poisson possui algumas propriedades interessantes. Uma das mais importantes é que os números de ocorrências em intervalos de largura igual são identicamente distribuídos (incrementos estacionários). Em particular, para $t > s$, a diferença $N(t) - N(s)$ é identicamente distribuída a $N(t + h) - N(s + h)\ \forall h$, com função de probabilidade:

$$p_n(t - s) = \frac{[\lambda(t - s)]^n}{n!}e^{-\lambda(t-s)}.$$

Veremos agora que, se o processo é de Poisson, então uma variável aleatória associada definida como o tempo entre sucessivas chegadas (tempo entre chegadas) segue uma distribuição exponencial. Seja T a variável aleatória "tempo entre chegadas sucessivas". Então

$$P\{T \geq t\} = P\{N(t) = 0\} = p_0(t) = e^{-\lambda t}.$$

Portanto, a função de distribuição de T pode ser escrita como

$$A(t) = P\{T \leq t\} = 1 - e^{-\lambda t},$$

que corresponde à função densidade

$$a(t) = \frac{dA(t)}{dt} = \lambda e^{-\lambda t}.$$

Assim, T tem distribuição exponencial com média $1/\lambda$. Intuitivamente, é esperado que o tempo médio entre chegadas seja $1/\lambda$, se a taxa média de chegada é λ. A seguir será demostrado que se os tempos entre chegadas são independentes e possuem a mesma distribuição exponencial, então a taxa de chegada segue a distribuição de Poisson.

Inicialmente, seja a função de distribuição do processo de contagem de chegadas até o instante t, $P\{N(t) \leq n\}$ denotada por $P_n(t)$. Então,

$$p_n(t) = P\{N(t) = n\} = P_n(t) - P_{n-1}(t).$$

Mas $P_n(t)$ é também a probabilidade da soma de $n + 1$ tempos entre chegadas ser superior à t. Como a soma de variáveis aleatórias independentes e identicamente distribuídas tem distribuição Erlang, então

$$P_n(t) = \int_t^\infty \frac{\lambda(\lambda x)^n}{n!} e^{-\lambda x} dx. \tag{2.18}$$

Usando a transformação de variáveis $u = x - t$, temos:

$$P_n(t) = \int_0^\infty \frac{\lambda^{n+1}(u+t)^n}{n!} e^{-\lambda t} e^{-\lambda u} du = \int_0^\infty \frac{\lambda^{n+1} e^{-\lambda t} e^{-\lambda u}}{n!} \left(\sum_{i=0}^{n} u^{n-i} t^i \frac{n!}{(n-i)!i!} \right) du,$$

em que a última igualdade procede do teorema binomial. Trocando a ordem entre a soma e a integral, temos:

$$P_n(t) = \sum_{i=0}^{n} \frac{\lambda^{n+1} e^{-\lambda t} t^i}{(n-i)!i!} \int_0^\infty e^{-\lambda u} u^{n-i} du.$$

Entretanto, a integral acima, a menos de uma adequada substituição, é a função gama e vale $(n-i)!/\lambda^{n-i+1}$. Então:

$$P_n(t) = \sum_{i=0}^{n} \frac{(\lambda t)^i e^{-\lambda t}}{i!},$$

que é a função de distribuição do processo de Poisson.

Uma importante consequência da propriedade uniforme do processo de Poisson é que os resultados de observações aleatórias de um processo estocástico $X(t)$ têm as mesmas probabilidades como se os exames fossem realizados em pontos selecionados. Quando $X(t)$ é uma fila, esta propriedade é chamada *PASTA*, do inglês "*Poisson arrivals see times averages*".

2.2.1 Propriedade markoviana da distribuição exponencial

Nesta seção provaremos a propriedade markoviana (ou falta de memória) da distribuição exponencial. Para explicar esta propriedade em palavras, suponha que os tempos de serviço são exponencialmente distribuídos. Esta propriedade estabelece que a probabilidade de um cliente atualmente em

serviço ter t unidades de serviço remanescentes é independente de quanto tempo já se passou em durante seu atendimento. Assim, desejamos provar que

$$P\{T \leq t_1 | T \geq t_0\} = \mathbf{Pr}\{0 \leq T \leq t_1 - t_0\} \text{ para } t_1 > t_0.$$

Da definição de probabilidade condicional, temos:

$$\begin{aligned} P\{T \leq t_1 | T \geq t_0\} &= \frac{P\{(T \leq t_1) \cap (T \geq t_0)\}}{P\{T \geq t_0\}} = \frac{e^{-\lambda t_0} - e^{-\lambda t_1}}{e^{-\lambda t_0}} \\ &= 1 - e^{-\lambda(t_1 - t_0)} = P\{0 \leq T \leq t_1 - t_0\}. \end{aligned}$$

2.2.2 O processo nascimento e morte

Um processo nascimento e morte é um tipo específico de cadeia de Markov de tempo contínuo, que consiste em um conjunto de estados $\{0, 1, 2, \dots\}$, denotando a "população" de algum sistema. Transições de estado ocorrem com saltos de uma unidade para frente ou para trás, em relação ao estado atual. Mais especificamente, quando o sistema está no estado $n \geq 0$, o tempo até a próxima chegada ("nascimento") é uma variável aleatória exponencial com taxa λ_n e move-se do estado n para o estado $n + 1$. Quando o sistema está no estado $n \geq 1$, o tempo até a próxima partida ("morte") é uma variável aleatória exponencial com taxa μ_n e move-se do estado n para o estado $n-1$. Isto é uma cadeia de Markov de tempo contínuo.

Defina p_n como a fração de tempo do sistema no estado n em regime estacionário. Então, a solução $\{p_n\}$ existe e pode ser determinada através da equação matricial $\mathbf{0} = \mathbf{pQ}$ (com a matriz $\mathbf{Q}$ obtida anteriormente para as cadeias de Markov em tempo contínuo), sujeita a certas condições sobre λ_n e μ_n. Para o processo nascimento e morte, a equação matricial $\mathbf{0} = \mathbf{pQ}$ pode ser reescrita como:

$$0 = -(\lambda_n + \mu_n)p_n + \lambda_{n-1}p_{n-1} + \mu_{n+1}p_{n+1} \ (n \geq 1),$$
$$0 = -\lambda_0 p_0 + \mu_1 p_1,$$

ou

$$\begin{aligned}(\lambda_n + \mu_n)p_n &= \lambda_{n-1}p_{n-1} + \mu_{n+1}p_{n+1} \ (n \geq 1),\\ \lambda_0 p_0 &= \mu_1 p_1.\end{aligned} \tag{2.19}$$

Estas equações também podem ser obtidas usando o conceito de *equilíbrio de fluxo*, cuja ideia básica é que, em estado estacionário, a taxa de transições para fora de um dado estado precisa ser igual à taxa de transições para dentro daquele estado. Assim, a Equação (2.19) é simplesmente o balanço da taxa de transições de entrada e saída do estado n, como será explicado mais detalhadamente no parágrafo a seguir.

Quando o sistema está no estado n, a taxa média de chegadas (ou "nascimentos") é λ_n chegadas por unidade de tempo. Uma vez que o sistema encontra-se no estado n a fração p_n do tempo, $\lambda_n p_n$ é a taxa de transições em regime permanente de n para $n+1$. Por outro lado, quando o sistema está no estado n, a taxa de partida (ou "morte") é μ_n partidas por unidade de tempo. Assim, $\mu_n p_n$ é a taxa de transições em regime estacionário de n para $n-1$. Como uma transição para fora do estado n pode ser para frente ou para trás, $(\lambda_n + \mu_n)p_n$ é a taxa de saída do estado n em regime estacionário.

De forma similar, como transições para o estado n podem ocorrer do estado $(n-1)$ ou do estado $(n+1)$, a taxa de transições para dentro do estado n é $\lambda_{n-1}p_{n-1} + \mu_{n+1}p_{n+1}$. Assim, a Equação (2.19) faz o balanço de transições para fora e para dentro do estado $n \geq 1$.

A segunda equação, $\lambda_0 p_0 = \mu_1 p_1$, representa o balanço para o estado de fronteira 0, que é diferente dos demais estados pois não podem ocorrer transições do estado 0 para o estado -1 ou vice-versa. Os próximos passos

serão dados para obter a solução para a Equação (2.19), que pode ser reescrita como:

$$p_{n+1} = \frac{\lambda_n + \mu_n}{\mu_{n+1}} p_n - \frac{\lambda_{n-1}}{\mu_{n+1}} p_{n-1} \text{ para } n \geq 1 \text{ e } p_1 = \frac{\lambda_0}{\mu_1} p_0. \tag{2.20}$$

Fazendo $n = 1$ na Equação 2.20, vem:

$$p_2 = \frac{\lambda_1 + \mu_1}{\mu_2} p_1 - \frac{\lambda_0}{\mu_2} p_0 = \frac{\lambda_1 + \mu_1}{\mu_2} \frac{\lambda_0}{\mu_1} p_0 - \frac{\lambda_0}{\mu_2} p_0 = \frac{\lambda_1 \lambda_0}{\mu_2 \mu_1} p_0.$$

Similarmente, podemos obter $p_3, p_4, \ldots$, de modo que o padrão é:

$$p_n = \frac{\lambda_{n-1} \lambda_{n-2} \ldots \lambda_0}{\mu_n \mu_{n-1} \ldots \mu_1} p_0 = p_0 \prod_{i=1}^{n} \frac{\lambda_{i-1}}{\mu_i}, \text{ para } n \geq 1. \tag{2.21}$$

Para verificar que esta, de fato, é a fórmula correta para todo $n \geq 0$, aplica-se o princípio da indução matemática à Equação (2.21). Primeiramente, note que a Equação (2.21) é satisfeita para $n = 0$, pois $\prod_{i=1}^{n}(.)$ é assumida por definição igual a 1 quando $n = 0$. É possível também mostrar que a Equação (2.21) é satisfeita para $n = 1, 2, 3$. A partir daí, demonstra-se que se ela está satisfeita para $n = k \geq 0$, então também estará satisfeita para $n = k + 1$ (Para a prova completa, ver Gross et al. [34]).

Considerando que as probabilidades precisam somar 1, segue que:

$$p_0 = \left(1 + \sum_{n=1}^{\infty} \prod_{i=1}^{n} \frac{\lambda_{i-1}}{\mu_i}\right)^{-1}. \tag{2.22}$$

2.3 Teoria de filas

Nenhum de nós consegue escapar de uma fila. Cada um de nós provavelmente tem uma situação relativa a uma fila que marcou. Infelizmente, este fenômeno é cada vez mais comum nas sociedades desenvolvidas. Esperamos em

uma fila dentro dos carros em congestionamentos de trânsito ou em praças de pedágio. Esperamos em fila para sermos atendidos em supermercados, bancos, restaurantes, etc. Assim como os clientes não gostam de filas, os gerentes dos serviços que as geram também não, pois elas representam um custo para os negócios. Então, por quê existe a fila de espera?

A resposta é simples. É porque existe maior demanda pelo serviço do que recursos para realizá-lo. Isto ocorre por diversas razões. Uma delas é a escassez de servidores disponíveis. Outra razão que podemos citar é que pode ser economicamente inviável para um negócio disponibilizar o nível de serviço necessário para prevenir filas de espera. Pode, ainda, haver limitação de espaço no volume de serviço que pode ser disponibilizado. De um modo geral, essas limitações podem ser resolvidas com gasto de capital e pelo conhecimento do quanto de serviço poderia ser diponibilizado, desde que conheçamos a resposta a questões do tipo "Qual o custo de um usuário ficar esperando?" ou "Quantas pessoas podem ficar em uma fila?" A teoria das filas tenta — e em muitos casos, com sucesso — responder a essas questões através de uma análise matemática detalhada.

2.3.1 Notação usual

A caracterização de uma fila, em geral, é feita utilizando a notação de Kendall [38], cuja forma é $A/B/X/Y/Z$, em que A descreve a distribuição do tempo entre chegadas, B, a distribuição do tempo de serviço, X, o número de estações de serviço em paralelo, Y, a capacidade do sistema, ou seja, número de usuários na sala de espera mais o número de usuários que estão sendo atendidos, e Z, a disciplina de atendimento. Alguns exemplos de escolhas para A e B:

M**:** distribuição exponencial (markoviana);

E_k**:** distribuição Erlang tipo-k (distribuição gama com parâmetro de forma inteiro);

G**:** distribuição geral (não especificada);

D**:** determinística.

Por exemplo, a notação $M/D/2/\infty$/FCFS indica um processo de fila com tempo entre chegadas exponencial, tempo de serviço determinístico, dois servidores em paralelo, sem restrição no tamanho máximo da capacidade do sistema e disciplina de fila 'primeiro a chegar, primeiro a ser atendido' (FCFS, do inglês *First-Come, First-Served*).

Quando são omitidos Y e Z na notação de Kendall, entende-se que a fila tem capacidade infinita e disciplina FCFS. Por exemplo, a fila $M/G/1$ tem chegadas exponenciais, serviço com distribuição geral, um único servidor, não há limite na capacidade do sistema e o atendimento é por ordem de chegada.

Pode parecer estranho utilizar o símbolo M para a distribuição exponencial, em vez do símbolo usual E. A razão para isso é evitar confusão com E_k, símbolo utilizado para a distribuição Erlang do tipo k. Da propriedade da falta de memória (do inglês *memoryless*) da distribuição exponencial, bem como em associação à propriedade markoviana, surge a ideia do símbolo M.

As características de um sistema de filas são (1) modelo de chegada dos usuários, (2) modelo de serviço dos servidores, (3) disciplina de atendimento da fila, (4) capacidade do sistema, (5) número de canais de serviço e (6) número de estágios de serviço, que são detalhadas a seguir.

Modelo de chegada dos usuários

Em situações usuais de fila, o processo de chegada é *estocástico*, sendo, então, necessário conhecer a distribuição de probabilidade que descreve os tempos

entre chegadas sucessivas (tempos entre chegadas). Também se faz necessário saber se as chegadas são unitárias ou em blocos, como é o caso da chegada de passageiros de um vôo em um aeroporto. Neste último caso, precisa-se conhecer a distribuição de probabilidade que descreve o tamanho do bloco. Outro fator importante a ser considerado no que diz respeito ao processo de chegada é a maneira como ele muda com o tempo. Um processo de chegada que não muda com o tempo, isto é, aquele cuja distribuição da probabilidade é independente do tempo, é chamado um processo de chegada *estacionário*.

Modelo de serviço

Muito da discussão anterior a respeito de modelo de chegada também é adequado ao serviço. Uma distribuição de probabilidade é necessária para descrever a sequência dos tempos de serviço dos usuários. O serviço também pode ser unitário ou em blocos. Geralmente, pensamos em um usuário sendo servido a cada instante por um dado servidor, mas há casos em que vários usuários podem ser servidos simultaneamente pelo mesmo servidor, como em um computador com processamento em paralelo ou pessoas embarcando em um trem.

O processo de serviço pode depender do número de usuários esperando pelo serviço. A situação em que o serviço depende do número de usuários em espera é denominado serviço *dependente do estado*. O serviço, assim como a chegada, pode ser *estacionário* ou *não-estacionário*, com relação ao tempo. A dependência no tempo não deve ser confundida com dependência no estado. O primeiro caso não depende do número de clientes no sistema, mas de quanto tempo o sistema está em operação. A dependência no estado não depende de quanto tempo o sistema está em operação, mas somente do estado do sistema em um dado tempo, isto é, de quantos clientes estão atualmente no

sistema. Note que um sistema de filas pode ser *não-estacionário* e *dependente do estado*.

Mesmo com uma taxa de serviço alta, é provável que alguns usuários sejam atrasados pela espera na fila. Em geral, usuários chegam e saem em intervalos irregulares. Portanto, o comprimento da fila não assumirá um padrão determinístico, mas sim um padrão aleatório. Segue-se, então, que a distribuição de probabilidade para comprimentos de fila será o resultado de dois processos separados — chegadas e serviços — que são, geralmente, mas não universalmente, assumidos mutuamente independentes.

Disciplina de atendimento da fila

A disciplina de atendimento refere-se a maneira pela qual os clientes são selecionados para o serviço quando a fila está formada. A disciplina mais comum que tem sido observada é "primeiro a chegar, primeiro a ser servido" (FCFS, do inglês *first come, first served*). Outra disciplina comum de atendimento é "último a chegar, primeiro a ser servido" (LCFS, do inglês *last come, first served*), que se aplica a muitos sistemas de estoque, quando não há obsolescência de unidades armazenadas, uma vez que é mais fácil atingir os itens mais próximos, no caso o último que chegou.

Capacidade do sistema

Em alguns processos de fila, existe uma limitação física no tamanho da área de espera. Então, quando o comprimento da fila atinge certo tamanho, nenhum cliente adicional será admitido na área de espera até que haja espaço disponível devido a conclusão de um serviço. Este tipo de situação é conhecido como *fila finita*. Isto é, existe um limite finito máximo para o tamanho do sistema.

Número de canais de serviço

O número de canais de serviço refere-se ao número de estações de serviço em paralelo que podem atender usuários simultaneamente. Os sistemas podem ser de um único canal ou multicanal. Estes dois sistemas diferem porque o primeiro possui uma fila única e o segundo admite uma fila para cada canal. Um salão de cabeleireiro com muitas cadeiras é um exemplo do primeiro tipo de sistema multicanal (assumindo que nenhum cliente espera por algum cabeleireiro particular). Um restaurante *fast-food* ou um supermercado são exemplos do segundo tipo de sistema multicanal. De um modo geral é assumido que os mecanismos de serviço de canais em paralelo operam independentemente um do outro.

Número de estágios de serviço

Um sistema de filas pode ter apenas um estágio de serviço, como em um salão de cabeleireiro, ou muitos estágios, como no caso do procedimento de um exame físico, em que cada paciente precisa passar por vários estágios, tais como, recepção, exame de ouvido, nariz e garganta, exame de sangue, eletrocardiograma, exame oftalmológico, e assim por diante. Em alguns processos de fila multiestágio pode ocorrer reciclagem, muito comum em processos de fabricação, em que inspeções de controle de qualidade são feitas após determinados estágios e as partes que não se encontram dentro dos padrões de qualidade são enviadas de volta para reprocessamento. De forma similar, uma rede de telecomunicação pode processar mensagens através de uma sequência de nós selecionados aleatoriamente, com a possibilidade de que algumas mensagens irão requerer reencaminhamento de vez em quando através do mesmo estágio.

Observações finais

As seis características de um sistema de filas citadas nesta seção, em geral são suficientes para descrever completamente um processo sob estudo. Claramente, uma grande variedade de sistemas de filas pode ser encontrada. Entretanto, antes de realizar qualquer análise matemática, é necessário descrever adequadamente o processo que está sendo modelado e o conhecimento das seis características básicas é essencial nesta tarefa.

No procedimento de seleção do modelo para o processo de fila é extremamente importante utilizar o modelo correto ou pelo menos um modelo que melhor descreve a real situação que está sendo estudada. Por exemplo, reconsiderando o caso do supermercado citado anteriormente, suponha que existem c estações de serviço. Se os clientes escolhem uma estação de serviço de maneira puramente aleatória e nunca mudam de fila (não há disputa), então há verdadeiramente c modelos independentes de filas simples. Por outro lado, se existe uma fila única de espera e, quando uma estação de serviço fica ociosa o cliente que está na frente da fila entra no serviço, diz-se que o modelo é de c canais. Este não é o caso da maioria dos supermercados, embora alguns adotem parte do serviço com este modelo (caixas para até n pacotes). Usualmente, o que ocorre é que são formadas filas na frente de cada estação de serviço e novos clientes entram na menor delas. A questão neste ponto é escolher qual modelo é mais adequado — c filas simples independentes ou uma única fila com c canais.

2.3.2 Resultados gerais e relações para filas

Denotando por λ a taxa média de entrada de usuários no sistema e μ a taxa média de serviço, pode-se definir a *intensidade de tráfego* $\rho = \dfrac{\lambda}{c\mu}$, uma medida do congestionamento de tráfego para um sistema com c servidores.

Quando $\rho > 1$, isto é, $\lambda > c\mu$, a taxa de chegada excede a máxima taxa de serviço do sistema, é esperado, com o passar do tempo, que a fila se torne cada vez maior, a menos que novos usuários não sejam mais autorizados a entrar no sistema. Se o interesse reside nas condições de estado de equilíbrio (o estado do sistema após estar em operação durante um longo tempo), quando $\rho > 1$, o tamanho da fila nunca se estabiliza e não há estado de equilíbrio. Verifica-se que para os resultados de estado de equilíbrio existirem, ρ precisa ser estritamente menor do que 1. Quando $\rho = 1$, a menos que chegadas e serviço sejam determinísticos e perfeitamente programados, não existe estado de equilíbrio, pois a aleatoriedade vai evitar que a fila se esvazie e que os servidores continuem buscando clientes, causando assim um crescimento sem limite na fila. No entanto, conhecendo-se a taxa média de chegada e a taxa média de serviço, pode-se calcular o número mínimo de servidores em paralelo para garantir uma solução de estado de equilíbrio encontrando o menor valor de c tal que $\dfrac{\lambda}{c\mu} < 1$.

Na maioria das vezes, resolver modelos de filas significa encontrar a distribuição de probabilidade do número total de usuários no sistema no tempo $t, N(t)$, que é composto por aqueles que esperam na fila, $N_q(t)$, como também os do serviço, $N_s(t)$. Sejam $p_n(t) = P\{N(t) = n\}$ e $p_n = P\{N = n\}$, no estado de equilíbrio. Considerando uma fila com c servidores em estado de equilíbrio, duas medidas de grande interesse são:

$$L = \mathbb{E}[N] = \sum_{n=0}^{\infty} np_n,$$

$$L_q = \mathbb{E}[N_q] = \sum_{n=c+1}^{\infty} (n-c)p_n,$$

respectivamente, o número médio de clientes no sistema e o número médio de clientes na fila.

Fórmulas de Little

Uma das mais importantes relações em teoria de filas foi desenvolvida por John D. C. Little no começo dos anos 1960. Little relacionou o tamanho médio do sistema ao tempo médio de espera do cliente em estado de equilíbrio. Seja T_q a variável aleatória que descreve o tempo que um cliente gasta esperando na fila para entrar no sistema e T a variável aleatória que descreve o tempo total que o cliente gasta no sistema, ou seja, $T = T_q + S$, em que S é a variável aleatória que descreve o tempo de serviço. Duas medidas de desempenho do sistema frequentemente usadas, com respeito ao cliente, são $W_q = \mathbb{E}[T_q]$ e $W = \mathbb{E}[T]$, respectivamente, o tempo médio de espera na fila e o tempo médio de espera no sistema. As fórmulas de Little são:

$$L = \lambda W \tag{2.23}$$

e

$$L_q = \lambda W_q \tag{2.24}$$

Então, é necessário encontrar apenas um dos quatro valores esperados, em vista das fórmulas de Little e do fato que $\mathbb{E}[T] = \mathbb{E}[T_q] + \mathbb{E}[S]$ ou, equivalentemente, $W = W_q + \frac{1}{\mu}$, em que μ, como definido anteriormente, é a taxa média de serviço.

Apesar de o procedimento a seguir não constituir uma prova, ele serve para ilustrar o conceito das fórmulas de Little, pois analisa um caminho aleatório de um período ocupado (tempo desde quando um cliente entra em um sistema vazio até quando ele esvazia novamente).

Considere a Figura 2.1, em que o número de clientes N_c que chega durante o período de tempo $(0, T)$ é 4. Os valores de L e W podem ser calculados pelas equações abaixo:

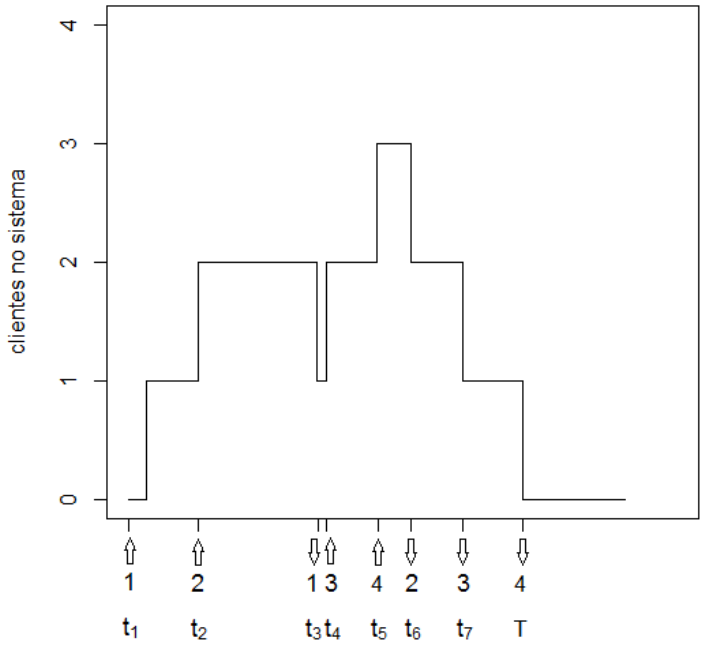


Figura 2.1: Ilustração para um processo de chegada e atendimento de clientes

$$\begin{aligned} L = & \ [1(t_2 - t_1) + 2(t_3 - t_2) + 1(t_4 - t_3) + 2(t_5 - t_4) \\ & +3(t_6 - t_5) + 2(t_7 - t_6) + 1(T - t_7)]/T \\ = & \ \text{(área sob a curva)} \ /T \\ = & \ (T + t_7 + t_6 - t_5 - t_4 + t_3 - t_2 - t_1)/T. \end{aligned} \tag{2.25}$$

e

$$\begin{aligned} W = & \ [(t_3 - t_1) + (t_6 - t_2) + (t_7 - t_4) + (T - t_5)]/4 \\ = & \ (T + t_7 + t_6 - t_5 - t_4 + t_3 - t_2 - t_1)/4 \\ = & \ \text{(área sob a curva)} \ /N_c. \end{aligned} \tag{2.26}$$

Das Equações (2.25) e (2.26) temos que a área sob a curva é $LT = WN_c$, o que implica $L = WN_c/T$. A fração N_c/T é o número de clientes chegando durante o tempo T e, para este período, a taxa de chegada é λ, logo, $L = \lambda W$.

Uma argumentação similar produz para, um número de clientes na fila N_q no período $(0, T)$, $L_q = \lambda W_q$.

Um resultado interessante que pode ser obtido das fórmulas de Little e das relações entre W e W_q é

$$L - L_q = \lambda(W - W_q) = \lambda(1/\mu). \tag{2.27}$$

Mas, $L - L_q = \mathbb{E}[N] - \mathbb{E}[N_q] = \mathbb{E}[N - N_q] = \mathbb{E}[N_s]$. Então, o número esperado de clientes em serviço no estado de equilíbrio é λ/μ, que será denotado por r. Note que, para um sistema com um único servidor, temos $r = \rho$ e, então, segue que

$$L - L_q = \sum_{n=1}^{\infty} np_n - \sum_{n=1}^{\infty}(n-1)p_n = \sum_{n=1}^{\infty} p_n = 1 - p_0.$$

Disto, podemos derivar a probabilidade de algum servidor estar ocupado em um sistema multiservidor em estado de equilíbrio. Esta probabilidade será denotada por p_{ocup}. Como já mostrado, o número esperado de clientes em serviço em qualquer instante no estado de equilíbrio, r, segue, da simetria dos c servidores, que o número esperado de clientes em um servidor é r/c. Então, $r/c = \rho = 0 \times (1 - p_{\text{ocup}}) + 1 \times p_{\text{ocup}}$.

Para uma fila de um único servidor $G/G/1$, a probabilidade de um sistema estar ocioso é a mesma probabilidade de um servidor estar ocioso. Assim, $p_0 = 1 - p_{\text{ocup}}$ neste caso, e $p_0 = 1 - \rho = 1 - r = 1 - \lambda/\mu$.

2.4 Modelos de filas markovianas simples

O propósito desta seção é desenvolver uma ampla classe de modelos de filas utilizando a teoria dos processos de nascimento e morte. São exemplos de filas que podem ser modeladas como processos de nascimento e morte: $M/M/1$, $M/M/c$, $M/M/c/K$, $M/M/c/c$, $M/M/\infty$ e as variações dessas filas com independência entre taxas de chegada e taxas de serviço.

2.4.1 Filas $M/M/1$

Para uma fila com um único servidor $M/M/1$ em estado estacionário, os tempos entre chegadas e os tempos de serviço têm distribuiçao exponencial com funções de densidade, respectivamente, $a(t) = \lambda e^{-\lambda t}$ e $b(t) = \mu e^{-\mu t}$. Seja n o número de clientes no sistema, as chegadas podem ser consideradas como "nascimentos" e as saídas, como "mortes". A taxa de chegadas λ é fixa, assim como a taxa de serviço μ, independentemente do número de clientes no sistema. Assim, a fila $M/M/1$ é um processo de nascimento e morte com $\lambda_n = \lambda, \forall n \geq 0$ e $\mu_n = \mu, \forall n \geq 1$. As equações de balanço de fluxo para este sistema são:

$$\begin{aligned} (\lambda + \mu)p_n &= \mu p_{n+1} + \lambda p_{n-1}, \text{ para } n \geq 1 \\ \lambda p_0 &= \mu p_1. \end{aligned} \tag{2.28}$$

De forma alternativa, elas podem ser escritas como:

$$\begin{aligned} p_{n+1} &= \frac{\lambda + \mu}{\mu} p_n - \frac{\lambda}{\mu} p_{n-1}, \text{ para } n \geq 1 \\ p_1 &= \frac{\lambda}{\mu} p_0. \end{aligned} \tag{2.29}$$

Três métodos podem ser utilizados para resolver as Euações (2.28). Desenvolveremos apenas o método que envolve um procedimento iterativo. Para os demais possibilidades de solução, ver Gross et al. [34].

Obtenção de $\{p_n\}$ por método iterativo

Nesta seção usaremos iterativamente as equações de balanço dadas pela Equação (2.28) e (2.29), para obter uma sequência de probabilidades de estado, $p_1, p_2, p_3, \ldots$, em termos de p_0.

Como o sistema $M/M/1$ é um processo nascimento e morte com taxas de nascimento e morte constantes, podemos aplicar diretamente a Equa-

ção (2.21), com $\lambda_n = \lambda$ e $\mu_n = \mu$, para todo n. Segue-se, então, que:

$$p_n = p_0 \prod_{i=1}^{n} \left(\frac{\lambda}{\mu}\right) = p_0 \left(\frac{\lambda}{\mu}\right)^n, \text{ para } n \geq 1.$$

Para obter p_0, usa-se o fato que as probabilidades $\{p_n\}$ deverão somar 1:

$$1 = \sum_{n=0}^{\infty} p_n = \sum_{n=0}^{\infty} p_0 \left(\frac{\lambda}{\mu}\right)^n = p_0 \sum_{n=0}^{\infty} \rho^n.$$

No último passo foi utilizado o fato que $\rho = \frac{\lambda}{\mu}$, para filas de um único servidor, em que ρ é a intensidade de tráfego ou utilização.

Então, $p_0 = \dfrac{1}{\sum_{n=0}^{\infty} \rho^n}$, que é uma série geométrica que converge se, e somente se, $\rho < 1$. Usando o resultado conhecido para séries geométricas, temos

$$\sum_{n=0}^{\infty} \rho^n = 1 + \rho + \rho^2 + \cdots = \frac{1}{1-\rho}, \text{ para } \rho < 1,$$

que implica em $p_0 = 1 - \rho$, com $\rho = \lambda/\mu < 1$.

Em resumo, a solução completa de estado permanente para o sistema $M/M/1$ é a função de probabilidade geométrica

$$p_n = (1-\rho)\rho^n, \tag{2.30}$$

para $\rho = \lambda/\mu < 1$.

É bom ressaltar que a existência de uma solução de estado permanente depende da condição $\rho < 1$ ou, de forma equivalente, $\lambda < \mu$.

Medidas de desempenho

Duas medidas de interesse imediato são o valor esperado do número de usuários no sistema e do número de usuários na fila. Sejam N a variável aleatória

"número de usuários no sistema em regime estacionário" e L seu valor esperado, temos:

$$L = \mathbb{E}[N] = \sum_{n=0}^{\infty} np_n = (1-\rho)\sum_{n=0}^{\infty} n\rho^n. \tag{2.31}$$

Considere a soma

$$\sum_{n=0}^{\infty} n\rho^n = \rho + 2\rho^2 + 3\rho^3 + \cdots = \rho(1 + 2\rho + 3\rho^2 + \dots) = \rho\sum_{n=1}^{\infty} n\rho^{n-1}.$$

Podemos observar que $\sum_{n=1}^{\infty} n\rho^{n-1}$ é a derivada de $\sum_{n=0}^{\infty} \rho^n$ com respeito a ρ e, como as operações de somatório e diferenciação podem ser intercambiadas.

Como $\rho < 1$ e $\sum_{n=0}^{\infty} \rho^n = \dfrac{1}{1-\rho}$, então $\sum_{n=1}^{\infty} n\rho^{n-1} = \dfrac{d[1/(1-\rho)]}{d\rho} = \dfrac{1}{(1-\rho)^2}$.

Logo, o número de usuários esperado no sistema no regime estacionário é $L = \dfrac{\rho(1-\rho)}{(1-\rho)^2}$, ou simplesmente,

$$L = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda}. \tag{2.32}$$

Se a variável aleatória "número de usuários na fila em regime estacionário" é denotada por N_q e seu valor esperado, por L_q, então:

$$L_q = \mathbb{E}(N_q) = \sum_{n=1}^{\infty}(n-1)p_n = \sum_{n=1}^{\infty} np_n - \sum_{n=1}^{\infty} p_n = L - (1-p_0) = \frac{\rho}{1-\rho} - \rho = \frac{\rho^2}{1-\rho}.$$

Equivalentemente, podemos escrever:

$$L_q = \frac{\rho^2}{1-\rho} = \frac{\lambda^2}{\mu(\mu-\lambda)}. \tag{2.33}$$

Finalmente, os valores esperados do tempo de espera no sistema, W, e na fila, W_q, em regime estacionário, podem ser obtidos utilizando as fórmulas de Little:

$$W = \frac{L}{\lambda} = \frac{\rho}{\lambda(1-\rho)} = \frac{1}{\mu - \lambda}. \tag{2.34}$$

$$W_q = \frac{L_q}{\lambda} = \frac{\rho}{\mu(1-\rho)} = \frac{\rho}{\mu - \lambda}. \tag{2.35}$$

2.4.2 Filas multiservidores $M/M/c$

Analogamente à fila $M/M/1$, a fila multiservidor $M/M/c$ pode ser modelada por um processo nascimento e morte. A taxa de chegada é constante (nascimento) $\lambda_n = \lambda$ para todo n e a taxa de serviço (morte) é dada por:

$$\mu_n = \begin{cases} n\mu, & \text{para } 1 \leq n < c, \\ c\mu, & \text{para } n \geq c. \end{cases} \tag{2.36}$$

Usando a teoria desenvolvida anteriormente para o processo nascimento e morte podemos inserir os valores de λ_n e μ_n na Equação (2.21), para obter

$$p_n = \begin{cases} \dfrac{\lambda^n}{n!\mu^n} p_0, & \text{se } 0 \leq n < c, \\ \dfrac{\lambda^n}{(c^{n-c})\, c!\mu^n} p_0, & \text{se } n \geq c, \end{cases}$$

Assim,

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} + \sum_{n=c}^{\infty} \frac{\lambda^n}{c^{n-c} c!\mu^n} \right)^{-1}.$$

Defina $r = \dfrac{\lambda}{\mu}$, logo $\rho = \dfrac{r}{c} = \dfrac{\lambda}{c\mu}$, então obtemos:

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \sum_{n=c}^{\infty} \frac{r^n}{c^{n-c} c!} \right)^{-1}.$$

Entretanto, o segundo somatório da equação acima pode ser reescrito como uma série geométrica, qual seja:

$$\frac{r^c}{c!} \sum_{n=c}^{\infty} \left(\frac{r}{c}\right)^{n-c} = \frac{r^c}{c!} \sum_{m=0}^{\infty} \left(\frac{r}{c}\right)^{m} = \frac{r^c}{c!} \frac{1}{1 - \frac{r}{c}} = \frac{r^c}{c!(1-\rho)}.$$

Logo,

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!(1-\rho)} \right)^{-1} \text{ com } \frac{r}{c} = \rho < 1. \tag{2.37}$$

É importante notar que, quando $c = 1$, temos $p_0 = 1 - \rho$ que é o resultado para fila $M/M/1$.

Medidas de desempenho para uma fila $M/M/c$

As medidas de desempenho da fila $M/M/c$ serão determinadas a partir das equações de estado estacionário para p_n. Começaremos por L_q, por apresentar maior facilidade algébrica em sua manipulação.

$$L_q = \mathbb{E}(N_q) = \sum_{n=c+1}^{\infty} (n-c)p_n = \sum_{n=c+1}^{\infty} (n-c)\frac{r^n}{c^{n-c}c!}p_0 = \frac{r^c p_0}{c!} \sum_{n=c+1}^{\infty} (n-c)\rho^{n-c}.$$

Fazendo uma mudança de variável podemos escrever:

$$\frac{r^c p_0}{c!} \sum_{m=1}^{\infty} m\rho^m = \frac{r^c \rho p_0}{c!} \sum_{m=1}^{\infty} m\rho^{m-1} = \frac{r^c \rho p_0}{c!} \frac{d}{d\rho} \sum_{m=1}^{\infty} \rho^m.$$

Como o último somatório é uma série geométrica, podemos substituí-lo pelo resultado de sua convergência, qual seja,

$$\frac{r^c \rho p_0}{c!} \frac{d}{d\rho} \left(\frac{1}{1-\rho} - 1 \right) = \frac{r^c \rho p_0}{c!(1-\rho)^2},$$

e assim obter:

$$L_q = \left(\frac{r^c \rho}{c!(1-\rho)^2} \right) p_0. \tag{2.38}$$

Usando as equações das medidas de desempenho e considerando o fato que $W = W_q + \frac{1}{\mu}$, obtemos:

$$W_q = \frac{L_q}{\lambda} = \left(\frac{r^c \rho p_0}{\lambda c!(1-\rho)^2} \right) = \left(\frac{r^c p_0}{c!(c\mu)(1-\rho)^2} \right), \tag{2.39}$$

$$W = \frac{1}{\mu} + \left(\frac{r^c p_0}{c!(c\mu)(1-\rho)^2} \right) = \left(\frac{c!(c)(1-\rho)^2 + r^c p_0}{c!(c\mu)(1-\rho)^2} \right), \tag{2.40}$$

$$L = \lambda W = \left(\frac{c!(c)\lambda(1-\rho)^2 + \lambda r^c p_0}{c!(c\mu)(1-\rho)^2} \right) = \left(\frac{c!(c)\rho(1-\rho)^2 + \rho r^c p_0}{c!(1-\rho)^2} \right). \tag{2.41}$$

2.4.3 Filas finitas $M/M/c/\mathbf{K}$

As taxas de serviço são idênticas às do modelo $M/M/c$. Porém as taxas de chegada se modificarão, uma vez que o sistema não aceita mais de K clientes no sistema. Ou seja, a capacidade da fila agora é limitada. Teremos, então:

$$\mu_n = \begin{cases} n\mu, & \text{para } 1 \leq n < c, \\ c\mu, & \text{para } n \geq c, \end{cases}$$

$$\lambda_n = \begin{cases} \lambda, & \text{para } 0 \leq n < K, \\ 0, & \text{para } n \geq K, \end{cases}$$

$$p_n = \begin{cases} \dfrac{\lambda^n}{n!\mu^n} p_0, & \text{se } 0 \leq n < c, \\ \dfrac{\lambda^n}{(c^{n-c})c!\mu^n} p_0, & \text{se } c \leq n \leq K. \end{cases} \tag{2.42}$$

É importante notar que o espaço de estados está restrito a $\{0, 1, 2, \ldots, K\}$. Assim,

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{\lambda^n}{n!\mu^n} + \sum_{n=c}^{K} \frac{\lambda^n}{c^{n-c}c!\mu^n} \right)^{-1}.$$

Fazendo novamente, $r = \dfrac{\lambda}{\mu}$ e $\rho = \dfrac{r}{c} = \dfrac{\lambda}{c\mu}$, obtemos para a segunda soma:

$$\sum_{n=c}^{K} \frac{r^n}{c^{n-c}c!} = \frac{r^c}{c!}\sum_{n=c}^{K}\rho^{n-c} = \begin{cases} \dfrac{r^c}{c!}\left(\dfrac{1-\rho^{K-c+1}}{1-\rho}\right), & \text{se } \rho \neq 1, \\ \\ \dfrac{r^c}{c!}(K-c+1), & \text{se } \rho = 1, \end{cases}$$

assim,

$$p_0 = \begin{cases} \left[\dfrac{r^c}{c!}\left(\dfrac{1-\rho^{K-c+1}}{1-\rho}\right) + \displaystyle\sum_{n=0}^{c-1}\frac{r^n}{n!}\right]^{-1} & \text{, se } \rho \neq 1, \\ \\ \left[\dfrac{r^c}{c!}(K-c+1) + \displaystyle\sum_{n=0}^{c-1}\frac{r^n}{n!}\right]^{-1} & \text{, se } \rho = 1. \end{cases} \tag{2.43}$$

Podemos obter o comprimento esperado da fila L_q, para $\rho \neq 1$:

$$\begin{aligned} L_q = \mathbb{E}(N_q) = \sum_{n=c+1}^{K}(n-c)p_n = p_0\sum_{n=c+1}^{K}(n-c)\frac{\lambda^n}{c^{n-c}c!\mu^n} = \\ \frac{p_0 r^c}{c!}\sum_{n=c+1}^{K}(n-c)\frac{r^{n-c}}{c^{n-c}} = \frac{p_0 r^c \rho}{c!}\sum_{n=c+1}^{K}(n-c)\rho^{n-c-1} = \\ \frac{p_0 r^c \rho}{c!}\sum_{i=1}^{K-c} i\rho^{i-1} = \frac{p_0 r^c \rho}{c!}\frac{d}{d\rho}\left(\sum_{i=0}^{K-c}\rho^i\right) = \\ \frac{p_0 r^c \rho}{c!}\frac{d}{d\rho}\left(\frac{1-\rho^{K-c+1}}{1-\rho}\right), \end{aligned}$$

logo,

$$L_q = \mathbb{E}(N_q) = \frac{p_0 r^c \rho}{c!(1-\rho)^2}\left(1-\rho^{K-c+1}-(1-\rho)(K-c+1)\rho^{K-c}\right). \tag{2.44}$$

Para obter o tamanho esperado do sistema, vamos lembrar do modelo $M/M/c$, em que $L = L_q + r$. No entanto, no caso da fila finita $M/M/c/K$ precisamos fazer um ajuste na taxa de entrada. É importante notar que

algumas chegadas podem ser "perdidas" pelo sistema, por não haver mais espaço na área de espera — *overflow* do sistema (que se deve à capacidade finita K do sistema).

A probabilidade de o sistema permitir entradas é $1 - p_K$ e, assim, a taxa efetiva de entrada λ_{efe} será $\lambda_{\text{efe}} = \lambda(1 - p_K)$.

Assim,

$$L = L_q + \frac{\lambda_{\text{efe}}}{\mu} = L_q + \frac{\lambda(1 - p_K)}{\mu} = L_q + r(1 - p_K). \tag{2.45}$$

Note que a quantidade $r(1 - p_K)$ deve ser menor que c, uma vez que o número médio de usuários em serviço deve ser menor do que o número total de servidores disponíveis.

Temos, então:

$$W = \frac{L}{\lambda_{\text{efe}}} = \frac{L}{\lambda(1 - p_K)}, \tag{2.46}$$

$$W_q = W - \frac{1}{\mu} = \frac{L_q}{\lambda_{\text{efe}}} = \frac{L_q}{\lambda(1 - p_K)}. \tag{2.47}$$

2.5 Filas gerais $M/G/1$

Vamos tratar de filas com distribuição geral do serviço, isto é, filas que não possuem propriedade especial alguma, a não ser a independência entre sucessivos serviços e entre serviços e chegadas.

De acordo com notação de Kendal [38] a fila $M/G/1$ tem chegadas seguindo um processo de Poisson com taxa λ, sala de espera de capacidade ilimitada, disciplina de atendimento FCFS e um único servidor com tempo de serviço denotado por S, que segue uma distribuição geral G.

Sendo λ a taxa de chegada e S o tempo de serviço que segue distribuição geral, seja $\mu = \dfrac{1}{\mathbb{E}(S)}$ a taxa de serviço. Assume-se que $\rho = \dfrac{\lambda}{\mu} < 1$ e que a fila esteja em estado permanente.

Inicialmente será deduzida uma coleção de resultados para os valores esperados de medidas de desempenho, W_q, W, L_q e L. Estes resultados para medidas de desempenho são conhecidas como fórmulas de *Pollaczek-Khintchine* (PK). A estratégia será obter uma dessas medidas e deduzir as outras utilizando a fórmula de Little e/ou $W = W_q + \mathbb{E}(S)$.

O tempo de espera para uma nova chegada é exponencial com parâmetro λ. Entretanto, o tempo para completar o serviço não pode ser determinado, pois sua distribuição não tem necessariamente a propriedade de falta de memória. Dessa forma, a informação disponível para um estado não é suficiente para determinar as probabilidades de transição e, consequentemente, $N(t)$, o número de clientes no sistema no instante t, não é um processo markoviano. A ideia é encontrar alguns pontos de tempo especiais, tais que, a partir deles, se possa obter o comportamento do sistema. Isto pode ser tratado de duas maneiras: a primeira obtem resultados considerando o sistema nos instantes em que os clientes chegam e a segunda obtem resultados considerando o sistema nos momentos de partida dos clientes. Neste trabalho serão deduzidos apenas os resultados utilizando os tempos de chegada. A outra abordagem pode ser vista em Gross et al. [34].

Considere um cliente chegando na fila, seu tempo de espera é determinado pelos usuários que já estão no sistema quando esse cliente chega. Mais especificamente, podem haver usuários na fila e pode haver um usuário em serviço. Primeiramente considere os usuários na fila no instante em que o cliente chega. Cada usuário que está posicionado à sua frente na fila contribui, em média, $\mathbb{E}(S)$ para seu tempo de espera. Há, em média, L_q usuários na fila quando o cliente chega. Então seu tempo de espera médio devido a esses usuários é $L_q\mathbb{E}(S)$ (esta lógica exige a hipótese de chegadas seguindo um processo de Poisson e a propriedade *PASTA* implicando que o número

médio de usuários na fila, observado por um cliente que chega, é o mesmo tempo médio do número de usuários na fila, ou seja, L_q).

O usuário que está em serviço quando um cliente chega contribui com um valor diferente para seu tempo de espera. Se esse usuário já completou alguma parte de seu serviço, então sua contribuição para o tempo de espera é seu tempo de serviço remanescente e não o tempo de serviço total. Em geral, esses tempos não tem esperanças iguais.

Considere S_{OC} o evento *servidor estar ocupado* e T_{SR} o *tempo de serviço residual*, resumindo, a fila média para o cliente que chega é dada por:

$$W_q = L_q\mathbb{E}(S) + P(S_{OC})\mathbb{E}(T_{SR}|S_{OC}).$$

Usando $L_q = \lambda W_q$ para eliminar L_q da expressão anterior e rearranjando termos, tem-se:

$$W_q = \frac{P(S_{OC})\mathbb{E}(T_{SR}|S_{OC})}{1-\rho}, \tag{2.48}$$

Entretanto, $P(S_{OC})$ é a probabilidade de o cliente que chega encontrar o servidor ocupado. Pela propriedade *PASTA* ela é a mesma que a fração de tempo em que o servidor está ocupado, ou seja, $P(S_{OC}) = \rho$.

Assim, só resta obter o valor esperado do tempo de serviço residual dado que o servidor está ocupado. Utilizando "informações" do tempo médio residual de um processo de renovação é possível mostrar que:

$$\mathbb{E}(T_{SR}|S_{OC}) = \frac{\mathbb{E}(S^2)}{2\mathbb{E}(S)} = \frac{1+\mathrm{CV}^2}{2}\mathbb{E}(S),$$

em que $\mathrm{CV}^2 = \dfrac{\mathrm{Var}(S)}{\mathbb{E}^2(S)}$ é o quadrado do coeficiente de variação da distribuição do serviço.

Este resultado está relacionado com o resultado padrão da teoria da renovação. A fórmula é denominada *tempo médio residual de um processo de renovação*. Intuitivamente, é o tempo médio até o fim de um ciclo de renovação, visto por um observador que chega ao processo em um instante aleatório.

Então, combinando os resultados precedentes, temos:

$$W_q = \frac{1 + \mathrm{CV}^2}{2} \frac{\rho}{1 - \rho} \mathbb{E}(S), \tag{2.49}$$

Note que este é um resultado poderoso. São necessários apenas três parâmetros para calcular W_q: a taxa de chegadas λ, a média da distribuição do serviço $\mathbb{E}(S) = \dfrac{1}{\mu}$ e o quadrado do coeficiente de variação CV^2 da distribuição do serviço.

Equivalentemente, o segundo momento $\mathbb{E}(S^2)$ ou a variância $\mathrm{Var}(S)$ da distribuição do serviço pode ser usado no lugar de CV^2, através das relações $\mathrm{CV}^2 = \dfrac{\mathrm{Var}(S)}{\mathbb{E}^2(S)}$ e $\mathrm{Var}(S) = \sigma_G^2 = \mathbb{E}(S^2) - \mathbb{E}^2(S)$.

Em um sistema real, as informações com respeito ao mecanismo de serviço são frequentemente disponíveis e os parâmetros podem facilmente estimados.

As outras medidas de desempenho podem ser obtidas pelas expressões:

$$\begin{aligned} L_q &= \lambda W_q, \\ W &= W_q + \frac{1}{\mu}, \text{ e} \\ L &= \lambda W = L_q + \rho. \end{aligned} \tag{2.50}$$

A Tabela 2.1 fornece diversas maneiras de expressar os resultados. A primeira parte utiliza CV^2; a segunda parte, o segundo momento da distribuição do serviço $\mathbb{E}(S^2)$, e a terceira, a variância da distribuição do serviço σ_G^2. Cada uma dessas fórmulas é conhecida como fórmula de Pollaczek-Khintchine, ou fórmulas PK.

Tabela 2.1: Expressões para as medidas de desempenho

	em função de		
Medida	CV^2	$\mathbb{E}(S^2)$	σ_B^2
L_q	$\frac{1+\mathrm{CV}^2}{2}\frac{\rho^2}{1-\rho}$	$\frac{\lambda^2\mathbb{E}(S^2)}{2(1-\rho)}$	$\frac{\rho^2+\lambda^2\sigma_B^2}{2(1-\rho)}$
W_q	$\frac{1+\mathrm{CV}^2}{2}\frac{\rho}{\mu-\lambda}$	$\frac{\lambda\mathbb{E}(S^2)}{2(1-\rho)}$	$\frac{\frac{\rho^2}{\lambda}+\lambda\sigma_B^2}{2(1-\rho)}$
W	$\frac{1+\mathrm{CV}^2}{2}\frac{\rho}{\mu-\lambda}+\frac{1}{\mu}$	$\frac{\lambda\mathbb{E}(S^2)}{2(1-\rho)}+\frac{1}{\mu}$	$\frac{\frac{\rho^2}{\lambda}+\lambda\sigma_B^2}{2(1-\rho)}+\frac{1}{\mu}$
L	$\frac{1+\mathrm{CV}^2}{2}\frac{\rho^2}{1-\rho}+\rho$	$\frac{\lambda^2\mathbb{E}(S^2)}{2(1-\rho)}+\rho$	$\frac{\rho^2+\lambda^2\sigma_B^2}{2(1-\rho)}+\rho$

2.5.1 Probabilidades do regime estacionário

Seja π_n a probabilidade de regime estacionário de haver n clientes no sistema em um ponto de saída (um instante imediatamente após um usuário ter completado o serviço) e seja p_n a probabilidade de estado permanente de n no sistema em um ponto arbitrário no tempo. Prova-se que no caso da fila $M/G/1$, $\{\pi_n\} = \{p_n\}$.

Mostra-se que a fila $M/G/1$, vista somente nos instantes de saída, resulta em uma cadeia de Markov de tempo discreto. Seja $t_1, t_2, t_3, ...$ uma sequência de instantes de partida da fila. Seja $X_n = X(t_n)$ o número de clientes no sistema imediatamente após a saída do usuário no instante t_n. Tem-se:

$$X_{n+1} = \begin{cases} X_n - 1 + A_{n+1}, & \text{se } X_n \geq 1, \\ A_{n+1}, & \text{se } X_n = 0, \end{cases} \tag{2.51}$$

em que X_n é o número de clientes remanescentes no sistema imediatamente após a saída do n-ésimo usuário (obviamente, o cliente que saiu não foi in-

cluído nesta contagem) e A_{n+1} é o número de clientes que chegaram durante o tempo de serviço do $(n+1)$-ésimo cliente.

Para mostrar que $X_1, X_2, \ldots$ é uma cadeia de Markov é preciso provar que os estados futuros da cadeia dependem apenas do estado atual, ou seja, é preciso provar que, dado o estado atual X_n, o estado futuro X_{n+1} é independente dos estados anteriores $X_{n-1}, X_{n-2}, \ldots$. Para isto, basta observar as Equações (2.51), nas quais X_{n+1} depende de X_n e de A_{n+1}. Como A_{n+1} é independente dos estados passados $X_{n-1}, X_{n-2}, \ldots$, então (X_n) é uma cadeia de Markov. Isto é verdade porque A_{n+1} depende do tamanho do tempo de serviço do $(n+1)$-ésimo cliente, mas não depende dos eventos que ocorreram anteriormente, ou seja, não depende do tamanho da fila nos pontos de partida anteriores $X_{n-1}, X_{n-2}, \ldots$. Assim, o processo de tempo discreto $X_1, X_2, \ldots$ é uma cadeia de Markov de tempo discreto.

As probabilidades de transição para esta cadeia de Markov são $p_{ij} = P(X_{n+1} = j | X_n = i)$, que dependem da distribuição do número de clientes que chegam durante um tempo de serviço. Como essa distribuição não depende do índice do cliente, pode-se abandonar o subscrito. Especificamente, denote por S a variável aleatória tempo de serviço, por A a variável aleatória número de clientes que chegaram durante esse tempo e $G(t)$ a função de distribuição para a variável aleatória S. Defina então:

$$k_i = \mathrm{P}(i \text{ chegadas durante um tempo de serviço}) = \mathrm{P}(A = i).$$

Tem-se que $P(A = i)$ pode ser calculado condicionando no comprimento do tempo de serviço, pela integral de Stieltjes:

$$k_i = P(A = i) = \int_0^\infty P(A = i | S = t) dG(t). \tag{2.52}$$

Nos casos em que a função densidade existe, $dG(t)$ pode ser substituído por $g(t)dt$ e então calcula-se através da integral de Riemann.

Entretanto, $\{A|S=t\}$ é uma variável aleatória de Poisson com média λt. Logo,

$$\begin{aligned} P(A=i|S=t) &= \frac{e^{-\lambda t}(\lambda t)^i}{i!}, \\ k_i &= \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^i}{i!} dG(t). \end{aligned} \tag{2.53}$$

Então, da Equação (2.51), tem-se:

$$P(X_{n+1}=j|X_n=i) = \begin{cases} P(A=j-i+1) & \text{, se } i \geq 1, \\ P(A=j) & \text{, se } i=0, \end{cases} \tag{2.54}$$

e a matriz de transição é:

$$\mathbf{P} = \{p_{ij}\} = \begin{bmatrix} k_0 & k_1 & k_2 & k_3 & \dots \\ k_0 & k_1 & k_2 & k_3 & \dots \\ 0 & k_0 & k_1 & k_2 & \dots \\ 0 & 0 & k_0 & k_1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Assumindo que o estado permanente é alcançado, o vetor de probabilidades de estado permanente $\boldsymbol{\pi} = \{\pi_n\}$ pode ser obtido usando a teoria de cadeia de Markov. Em particular, $\{\pi_n\}$ é a solução para a equação estacionária $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$ e, então:

$$\pi_i = \pi_0 k_i + \sum_{j=1}^{i+1} \pi_j k_{i-j+1}, \text{ para } i = 0, 1, \dots. \tag{2.55}$$

2.6 Filas gerais finitas $M/G/1/K$

A análise de uma fila com capacidade finita $M/G/1/K$ é feita de maneira similar ao caso sala de espera ilimitada. Serão examinados cada um dos principais resultados da fila $M/G/1/\infty$ e aplicados à fila $M/G/1/K$.

As fórmulas PK não se aplicam, pois o número esperado de chegadas durante um período de serviço precisa estar condicionado ao tamanho do sistema. A melhor forma de conseguir esse novo resultado é diretamente das probabilidades de estado estacionário, uma vez que agora há um número finito de estados.

A matriz de transição truncada em $K-1$ é:

$$\begin{bmatrix} k_0 & k_1 & k_2 & \dots & 1-\sum_{n=0}^{K-2} k_n \\ k_0 & k_1 & k_2 & \dots & 1-\sum_{n=0}^{K-2} k_n \\ 0 & k_0 & k_1 & \dots & 1-\sum_{n=0}^{K-3} k_n \\ 0 & 0 & k_0 & \dots & 1-\sum_{n=0}^{K-4} k_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1-k_0 \end{bmatrix},$$

cuja equação estacionária é:

$$\pi_i = \begin{cases} \pi_0 k_i + \sum_{j=1}^{i+1} \pi_j k_{i-j+1}, & \text{para } i = 0, 1, 2, \dots, K-2, \\ 1 - \sum_{j=0}^{K-2} \pi_j, & \text{para } i = K-1. \end{cases} \tag{2.56}$$

Estas K equações (consistentes) em K incógnitas podem então ser resolvidas para todas as probabilidades e o tamanho médio do sistema nos pontos de partida é dado por $L = \sum_{i=0}^{K-1} i\pi_i$. (Note que o maior estado da cadeia de Markov não é K, pois o sistema está sendo observado o sistema após uma partida. Assuma $K > 1$ pois, caso contrário, o modelo resultante $M/G/1/1$ é um caso particular do modelo $M/G/c/c$.)

A primeira parte da equação de regime estacionário é idêntica à do caso de uma fila $M/G/1$ de capacidade ilimitada. No entanto, as respectivas probabilidades estacionárias $\{\pi_i\}$ para a fila $M/G/1/K$ e $\{\pi_i^*\}$ para a fila $M/G/1/\infty$ devem ser, na pior das hipóteses, proporcionais para $i \leq K-1$, isto é, $\pi_i = C\pi_i^*$, $i = 0, 1, ..., K-1$. A condição usual de que a soma das probabilidades é igual a um implica que $C = \dfrac{1}{\sum_{i=0}^{K-1} \pi_i^*}$.

Além disso, a distribuição de probabilidade para o tamanho da fila encontrada por uma chegada será diferente de $\{\pi_i\}$, pois agora o espaço de estados precisa ser aumentado para incluir K. Seja q'_n a probabilidade de um cliente que chega encontrar n usuários. Na Seção 2.5.1 afirmou-se que $\pi_n = p_n$ para uma fila $M/G/1$. Na prova dessa igualdade, a distribuição do tamanho do sistema nos pontos de chegada $\{q_n\}$ é idêntica às probabilidades nos pontos de partida $\{\pi_n\}$ com as chegadas ocorrendo isoladamente e o serviço não sendo em blocos. Este também é o caso com q'_n, exceto que o espaço de estado é diferente.

Neste caso,

$$\begin{aligned} \pi_n &= P(\text{encontrar } n \text{ clientes}|\text{usuários não estão juntos}) = \\ &= q_n = \frac{q'_n}{1 - q'_K}, \end{aligned} \tag{2.57}$$

para $0 \leq n \leq K-1$.

Ou equivalentemente,

$$q'_n = (1 - q'_K)\pi_n, \text{ para } 0 \leq n \leq K-1.$$

Para obter q'_n usa-se uma aproximação, em que se iguala a taxa efetiva de chegada com a taxa efetiva de partida, isto é,

$$\lambda(1 - q'_K) = \mu(1 - p_0), \tag{2.58}$$

então,

$$\begin{cases} {q'}_n = \dfrac{(1-p_0)\pi_n}{\rho}, & \text{se } 0 \leq n \leq K-1, \\ {q'}_K = \dfrac{\rho - 1 + p_0}{\rho}. \end{cases} \tag{2.59}$$

Entretanto, o processo inicial de chegada é Poisson, com ${q'}_n = p_n$, para todo n. Assim,

$${q'}_0 = p_0 = \frac{(1-p_0)\pi_0}{\rho} \Longrightarrow p_0 = \frac{\pi_0}{\pi_0 + \rho}, \tag{2.60}$$

finalmente,

$${q'}_n = \frac{\pi_n}{\pi_0 + \rho}. \tag{2.61}$$

De posse deste leque de informações, associadas aos processos estocásticos e mais particularmente à teoria das filas, será possível voltar a discussão do problema de otimização multiobjetivo em redes de filas finitas, que é o foco desta tese.

Capítulo 3

Otimização em Engenharia

De acordo com Yang [65], otimização pode incluir uma variedade grande de problemas cujo objetivo é buscar por alguma otimalidade. Otimização está em todo lugar, da engenharia ao mercado financeiro, das nossas atividades diárias ao planejamento dos nossos feriados. Uma organização procura maximizar suas receitas, minimizar seus custos e maximizar seu desempenho. Mesmo quando planejamos nossos feriados, queremos maximizar nossa satisfação ao menor custo (ou idealmente de graça). De fato, estamos procurando sempre pelas soluções ótimas de todos os problemas que encontramos, embora não necessariamente sejamos capazes de encontrar tais soluções. Não é nenhum exagero dizer que encontrar a solução de problemas de otimização, intencionalmente ou não, é tão antigo quanto a história humana. Por exemplo, o princípio do menor esforço pode frequentemente explicar muitos dos comportamentos humanos. Sabemos que a menor distância entre dois pontos diferentes quaisquer em um plano é uma linha reta, mesmo que seja necessário uma matemática complexa, tal como o cálculo das variações, para provar formalmente que um segmento de linha reta entre os dois pontos é realmente o mais curto. Sobre os problemas de otimização, há várias formas diferentes de nomeá-los e classificá-los (vide Figura 3.1) e tipicamente as téc-

nicas de otimização podem variar significativamente de problema a problema. Uma abordagem unificada é impossível e a complexidade de um problema de otimização depende largamente da forma da sua função objetivo e de suas restrições.

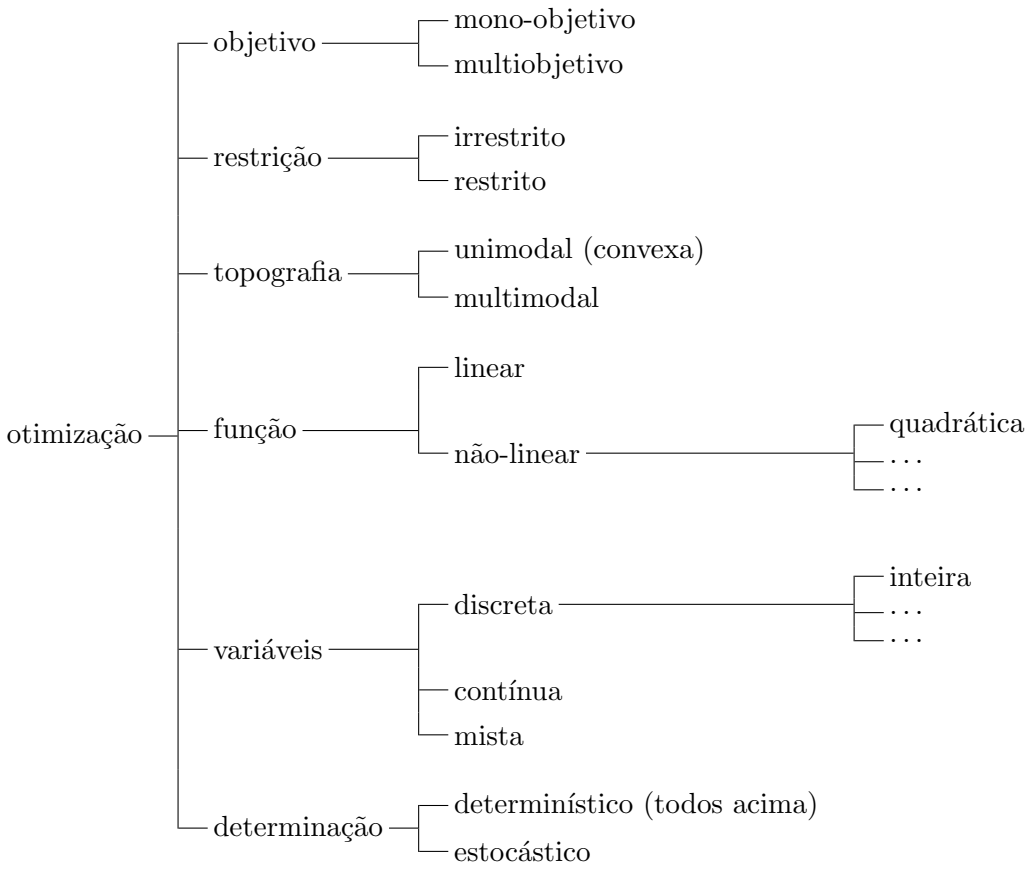


Figura 3.1: Classificação dos problemas de otimização segundo Yang [65]

Assim, nesta tese o foco de interesse recai sobre um problema *multiobjetivo*, *irrestrito*, *unimodal*, *não-linear geral*, *misto* e *estocástico*, como será definido a seguir.

3.1 Preliminares

Otimização pode significar muitas coisas diferentes. Entretanto, matematicamente falando, conforme Yang [65], é possível escrever um problema de otimização na seguinte forma genérica:

$$\underset{\mathbf{x}\in\mathcal{R}^n}{\text{minimize}}\, f_i(\mathbf{x}), \qquad (i = 1, 2, \ldots, I), \tag{3.1}$$

sujeito a:

$$\phi_j(\mathbf{x}) = 0, \qquad (j = 1, 2, \ldots, J), \tag{3.2}$$

$$\psi_k(\mathbf{x}) \leq 0, \qquad (k = 1, 2, \ldots, K), \tag{3.3}$$

em que $f_i(\mathbf{x})$, $\phi_j(\mathbf{x})$ e $\psi_k(\mathbf{x})$ são funções do vetor de decisão ou de projeto $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$. As componentes x_i de $\mathbf{x}$ são denominadas variáveis de decisão ou de projeto e podem ser contínuas reais, discretas ou uma mistura das duas. As funções $f_i(\mathbf{x})$, em que $i = 1, 2, \ldots, I$, são denominadas funções objetivo e, no caso de $M = 1$, há somente um objetivo. Na literatura, a função objetivo é algumas vezes chamada de função custo ou função energia. O espaço das variáveis de decisão é denominado de espaço de busca $\mathcal{R}^n$, enquanto que o espaço formado pelos valores das funções objetivo é denominado de espaço das soluções.

As funções objetivo podem ser lineares ou não-lineares. As igualdades para ϕ_j e desigualdades para ψ_k são denominadas restrições. É importante ressaltar que é possível escrever as inequações da forma $\geq$ e que se podem formular os objetivos como um problema de maximização. Isto porque a maximização de $f_i(\mathbf{x})$ é equivalente à minimização de $-f_i(\mathbf{x})$, e qualquer inequação $\psi(\mathbf{x}) \leq 0$ é equivalente a $-\psi(\mathbf{x}) \geq 0$. Para as restrições, o caso

mais simples para a variável de decisão x_i é $x_{i,\min} \leq x_i \leq x_{i,\max}$, chamados limites.

3.2 Formulações matemáticas tradicionais

Existem algumas possibilidades de formulação para o problema de otimização de taxa de saída, objeto desta tese, já tratadas na literatura. Pode-se optar por uma abordagem mono-objetivo, em que a função objetivo seria expressa em termos da taxa de atendimento alcançada pelo sistema de filas e por meio de um conjunto de restrições associadas às áreas de circulação nas filas do sistema e às taxas de serviço em cada servidor na rede filas.

Uma outra possível opção remete a uma abordagem multiobjetivo em que a função objetivo teria imagem em um espaço de dimensão maior que um. Para tanto podem ser incluídos objetivos associados às áreas de circulação, às taxas de serviço ou outras variáveis de decisão associadas ao problema.

O Capítulo 1 já remete a uma abordagem multiobjetivo, mas uma questão que surge de forma bastante natural é sobre o motivo da utilização da abordagem multiobjetivo para o problema em estudo. Será discutido a seguir, de forma resumida, um conjunto de possibilidades para a formulação matemática do problema.

Uma formulação mono-objetivo, que se encontra na literatura e em muitos problemas aplicados, propõe a maximização da taxa de atendimento θ sujeita à restrição de uma soma máxima pré-fixada K, definindo-se a rede de filas como um grafo direcionado (dígrafo), $G(N, A)$, em que N é o conjunto (finito) de nós e A é o conjunto (também finito) de arcos. Esta formulação mono-objetivo é apresenta a seguir.

$$\text{maximize}\, \theta(\mathbf{K}), \tag{3.4}$$

sujeito a

$$K_i \in \{1, 2, \ldots\},\ \forall i \in N, \tag{3.5}$$

$$\sum_{i \in N} K_i \leq K. \tag{3.6}$$

As taxas de serviço μ_i não são mencionadas nesta formulação, por serem consideradas constante reais positivas já fixadas de antemão. Para esta formulação temos um problema combinatório bastante similar ao problema do caixeiro viajante citado no Capítulo 1. Este problema pode ser resolvido através de uma busca exaustiva, para problemas com uma quantidade pequena de filas e valores pequenos para a capacidade K. Por outro lado, para situações maiores não se conhece uma estratégia completamente eficiente para solucionar o problema.

Aliado ao problema anteriormente mencionado, é fácil observar que a soma total da capacidade do sistema de filas, $\sum_{i \in N} K_i$, pode ser considerada como um objetivo e não como uma restrição. Para tanto, basta considerar cenários em que uma maior capacidade é atribuída ao sistema (piora neste objetivo específico), com a vantagem de haver um aumento na taxa de saída (melhora neste objetivo). Desta forma estar-se-ia substituindo a formulação mono-objetivo por uma nova formulação bi-objetivo, que é descrita a seguir.

$$\text{minimize } F(\mathbf{K}) \equiv \Big(\sum_{i \in N} K_i,\ -\theta(\mathbf{K}) \Big)^{\mathrm{T}}, \tag{3.7}$$

sujeito a

$$K_i \in \{1, 2, \ldots\},\ \forall i \in N. \tag{3.8}$$

Nessa formulação, a soma total das capacidades, $\sum_{i \in N} K_i$, é minimizada e a taxa de atendimento do sistema, $\theta(\mathbf{K})$, é maximizada de forma simultânea. As taxas de serviço μ_i novamente não são mencionadas nesta formulação, sendo também constante reais positivas já fixadas de antemão.

Andriansyah et al. [3] discutem essa abordagem para redes de filas bastante semelhantes às redes mencionadas nesta tese. Uma melhor compreensão sobre esta abordagem requer uma estratégia de ordenação de pares ordenados (elementos pertencentes ao $\mathcal{R}^2$). Estes assuntos serão discutidos em detalhes no Capítulo 4.

3.3 Uma nova formulação tri-objetivo

Do ponto de vista de modelagem, a otimização da taxa de atendimento pode ser definida por um modelo de programação matemática inteira mista, em que esta taxa de atendimento é maximizada, enquanto que são minimizados os custos das capacidades alocadas e das taxas de serviço, sujeito a um número inteiro de capacidade alocada e taxas de serviço não negativas. A seguinte formulação é proposta.

$$\text{minimize } F(\mathbf{K}, \boldsymbol{\mu}), \tag{3.9}$$

sujeito a

$$K_i \in \{1, 2, \ldots\}, \ \forall i \in N, \tag{3.10}$$

$$\mu_i \geq 0, \ \forall i \in N, \tag{3.11}$$

em que as variáveis de decisão, K_i e μ_i, indicam a capacidade do sistema e a taxa de serviço, respectivamente, para a i-ésima fila. A função multiobjetivo

$F(\mathbf{K}, \boldsymbol{\mu}) \equiv \Big(f_1(\mathbf{K}), f_2(\boldsymbol{\mu}), -f_3(\mathbf{K}, \boldsymbol{\mu})\Big)^{\mathrm{T}}$ inclui a minimização da capacidade alocada global, $f_1(\mathbf{K}) = \sum_{\forall i \in N} K_i$, a taxa de serviço total alocada, $f_2(\boldsymbol{\mu}) = \sum_{\forall i \in N} \mu_i$, e a taxa de atendimento, $f_3(\mathbf{K}, \boldsymbol{\mu}) = \theta(\mathbf{K}, \boldsymbol{\mu})$.

A nova proposta de formulação matemática, Equações (3.9)–(3.11), acrescenta uma terceira função objetivo de otimização, que é associada às taxas de serviço em cada uma das filas do sistema. Um questionamento natural seria sobre a possibilidade real de variação nos valores μ_i, ou se os mesmos deveriam ser considerados fixos como nas abordagens anteriores. Um pensamento simplista pode levar à crença de que uma vez estabelecido o servidor, encontra-se automaticamente definida a taxa de serviço do mesmo. Entretanto, tal bijeção pode não ser verificada para uma grande gama de problemas. Neste caso, discutem-se situações em que o servidor pode se tornar mais eficiente, seja através de uma substituição de equipamentos, ou qualquer possível estratégia de treinamento voltada para os servidores. Obviamente o aumento da eficiência do servidor acarreta algum tipo de custo, que pode ser justificável ou não, de acordo com o ganho gerado na taxa de atendimento final do sistema.

A literatura que trata os problemas de otimização em redes de filas já inclui estudos que consideram as taxas de serviço, μ_i, como componentes da função objetivo. Dong et al. [29] consideram o problema de alocação de tempos de manutenção de equipamentos. O interesse está em obter a taxa de serviço (para a manutenção dos equipamentos) que seja adequada para um limiar de operacionalidade e para minimizar um problema de geração de resíduos decorrente da manutenção. Um estudo de caso é utilizado para demonstrar a forma como o modelo pode ser aplicado na prática.

Um processo de produção em linhas de multi-estágio de montagem é dis-

cutido por Manitz [47]. Esses sistemas de produção compreendem processamento simples em estações de montagem. São consideradas filas de capacidades finitas e tempos de processamento com distribuição geral, constituindo um sistema de filas $G/G/1/K$. É descrito um procedimento de aproximação para a determinação da taxa de atendimento na linha de montagem através de uma abordagem heurística em que as taxas de serviços, e os quadrados dos coeficientes de variação (caracterizando distribuição geral) são determinados.

Os problemas de otimização em redes de filas são abordados também na indústria automotiva. Um exemplo destes sistemas pode ser encontrado em Spieckermann et al. [62], em que uma rede de filas executa rotinas de acabamento na produção de veículos. O problema-alvo é encontrar um *layout* eficiente, caracterizado pelo tamanho das áreas de circulação e pela otimização dos *tempos* de serviço, sem deixar de cumprir a taxa de produção desejada. Adicionalmente, em Spieckermann et al. [62], algumas vezes, a otimização de algum servidor é feita manualmente, possivelmente apoiada por um modelo de simulação para analisar o impacto de diferentes tempos de serviço e tamanhos de área de circulação. É apresentada uma formulação matemática e um sistema automatizado de otimização para este problema de planejamento. Os módulos de otimização, que têm uma interface direta com o modelo de simulação subjacente, são baseados em metaheurísticas, tais como algoritmos genéticos e recozimento simulado. Um estudo de caso é apresentado para um fabricante de automóveis alemão.

3.4 Observações finais

Neste capítulo, foi descrito um novo modelo matemático para o problema de alocação de áreas de espera e de taxas de serviço, bem como para a maximização da taxa de saída. De maneira diferente da usual, na formulação

multiobjetivo aqui proposta, definida pelas Equações (3.9)–(3.11), a taxa de atendimento passa a ser considerada como um objetivo a ser maximizado.

Sobre o problema de alocação de áreas de espera é importante reforçar que se trata de um problema mais frequentemente descrito por uma formulação mono-objetivo [59, 17], em que a taxa de atendimento é modelada pela restrição de assumir um valor não inferior a uma certa taxa limiar (isto é, $\theta(\mathbf{K}, \boldsymbol{\mu}) \geq \theta_{\text{lmr}}$). Para resolver a formulação mono-objetivo com sucesso, a restrição da taxa de atendimento precisa ser relaxada. Assim, valores tais como a taxa de atendimento limiar θ_{lmr} e o peso da relaxação precisam ser determinados de antemão. E estabelecer um valor limiar apropriado não é uma tarefa trivial. Além disso, é possível que um pequeno decréscimo no valor da taxa de atendimento limiar resulte em uma redução significativa na área de espera ótima. Este compromisso (*trade-off*) entre a taxa de atendimento e a área de espera não fica evidente na formulação mono-objetivo. Também o vetor de pesos para a composição de uma função mono-objetivo relaxada têm efeitos significativos nos objetivos e nos parâmetros, incluindo os erros nas estimativas de medidas de desempenho e na taxa de atendimento limiar θ_{lmr}. Desse modo, a determinação dos pesos é difícil e os resultados obtidos por meio de técnicas de otimização mono-objetivo podem ser arbitrários.

No Capítulo 4, será descrito um método de otimização multiobjetivo para determinar o conjunto de Pareto ótimo para o modelo proposto. Assim, o método deverá produzir um conjunto de soluções eficientes para três objetivos simultâneos. Com o método multiobjetivo proposto, o efeito da substituição de soluções poderá ser avaliado por quem for tomar uma decisão. Além do mais, o tratamento multiobjetivo também permite ao usuário promover um aumento em um objetivo (por exemplo, a taxa de atendimento) enquanto reduz simultaneamente outro objetivo (por exemplo, a alocação das áreas

de espera e das taxas de serviço). Maiores detalhes sobre características e vantagens dos métodos de otimização multiobjetivo pode ser vistos em Chankong & Haimes [9].

Os algoritmos desenvolvidos combinam um algoritmo evolucionário multiobjetivo (MOEA) associado ao método de expansão generalizada (GEM), sendo este um método bem sucedido para a obtenção de boas aproximações de medidas de desempenho em redes de filas finitas [39, 40, 41]. MOEAs são particularmente adequados para problemas multiobjetivos e têm-se mostrado com bom desempenho em problemas similares de otimização multiobjetivo em redes (por exemplo, veja Carrano et al. [8], e referências citadas).

Capítulo 4

Algoritmos Propostos

Para diferentes problemas de otimização, frequentemente é necessário usar técnicas de otimização diferentes, uma vez que alguns algoritmos, entre as existentes (vide Figura 4.1), tal como o método Newton-Raphson, são mais adequados para certos tipos de otimização que outros. Conforme bem colocado por Yang [65], a busca pela solução ótima pode ser vista como uma busca por um tesouro escondido. Imagine que se tenta achar o tesouro em uma topografia montanhosa, durante um tempo limitado. Em um extremo, suponha que se está com os olhos vendados, sem nenhum guia. Assim, o processo de busca é essencialmente uma busca aleatória pura, a qual é usualmente muito menos eficiente que se deseja. No outro extremo, se se sabe que o tesouro está no pico mais elevado, adota-se a estratégia de subir o morro mais elevado e tentar encontrar o tesouro. Este cenário é o da clássica subida na direção de máximo crescimento. Na maioria dos casos, a busca está entre esses dois extremos, uma vez que usualmente não se está de olhos vendados nem se sabe exatamente onde procurar. Nesta tese, será utilizada uma metaheurística baseada em população. Aqui, *meta* significa 'além' ou 'mais alto nível' e as metaheurísticas geralmente funcionam melhor que as heurísticas, em parte porque toda metaheurística usa certo compromisso entre

aleatorização e busca local. Passa-se a seguir a descreverem-se os algoritmos propostos.

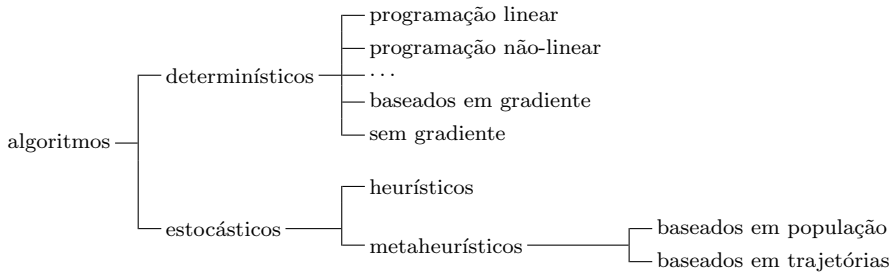


Figura 4.1: Classificação dos algoritmos de otimização segundo Yang [65]

4.1 Introdução

A explicação dos algoritmos propostos é apresentada em duas partes. Primeiramente, o algoritmo de análise de desempenho é descrito. Este algoritmo produz uma estimativa para a taxa de atendimento da rede de filas finitas gerais, $\theta(\mathbf{K}, \boldsymbol{\mu})$, conhecida a topologia da rede e dados os vetores de capacidades, $\mathbf{K}$, e das taxas de serviço, $\boldsymbol{\mu}$. Finalmente, o algoritmo de otimização proposto é detalhado. Este algoritmo permite a maximização simultânea da taxa de atendimento, enquanto são minimizadas a alocação de capacidades e as taxas de serviço.

4.2 Algoritmo para avaliação de desempenho

4.2.1 Filas simples

Para que seja possível a maximização da taxa de atendimento (*throughput*) $\theta(\mathbf{K}, \boldsymbol{\mu})$ é necessário algum método para estimá-la. Em uma *única fila* $M/G/1/K$, o procedimento de estimação pode ser executado através de

uma forma matemática fechada, computacionalmente eficiente, para a probabilidade p_K de ocorrência de bloqueio na fila. O método para obtenção da estimativa desta probabilidade apresentado a seguir, proposto por Smith [58], baseia-se na aproximação de dois momentos de Kimura [42] e é bastante eficaz:

$$p_K = \frac{\rho^{\left(\frac{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}+2(K-1)}{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}}\right)}(\rho-1)}{\rho^{\left(2\frac{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}+(K-1)}{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}}\right)}-1}, \tag{4.1}$$

em que $\rho = \lambda/\mu$ é a intensidade de tráfego (note que $\rho < 1$, pois, caso contrário, isto é, se $\lambda > \rho$, a fila "explode", conforme já mencionado anteriormente), CV^2 é o quadrado do coeficiente de variação da variável aleatória tempo de serviço, S, ou seja, $\mathrm{CV}^2 = \mathrm{Var}(S)/\mathbb{E}(S)^2$. Resultados empíricos indicam que esta aproximação para p_K é bastante acurada, para uma vasta gama de valores [59, 17, 57].

Para obter a taxa de atendimento para uma única fila $M/G/1/K$ é necessário o ajuste da taxa de chegada. Na verdade uma fração p_K dos recém-chegados não pode ingressar no sistema, porque eles vêm quando não há espaço deixado na área de espera. Assim, a taxa real de chegadas para ingressar no sistema deve ser ajustada convenientemente. De acordo com a propriedade PASTA (do inglês *Poisson Arrivals See Time Averages*), segue-se que a taxa de chegada efetiva vista pelos servidores é [34]:

$$\lambda_{\text{efe}} = \lambda(1 - p_K). \tag{4.2}$$

Assim, a taxa de atendimento efetiva pode ser dada pela expressão seguinte:

$$\theta = \lambda_{\text{efe}} = \lambda(1 - p_K). \tag{4.3}$$

4.2.2 Redes de filas

Para uma *rede de filas* a estimação da taxa de atendimento é consideravelmente mais complicada. O método de expansão generalizada (GEM) é um algoritmo que tem sido utilizado com sucesso na estimação do desempenho de redes acíclicas arbitrariamente configuradas de filas finitas [41]. O GEM é uma combinação de decomposição nó-a-nó e tentativas repetidas, na qual cada fila é analisada separadamente e correções são feitas para contabilizar os efeitos de inter-relacionamentos entre as filas finitas da rede. O GEM considera que os bloqueios ocorrem se, após o serviço ser concluído em uma fila, a fila seguinte estiver completamente cheia (isto é, um cliente está em serviço no único servidor da fila seguinte e todos os espaços de espera nela estão ocupados). Isto é frequentemente referido como "bloqueio após o serviço" (BAS, do inglês *Blocking After Service*).

Com o auxílio da Fig. 4.2 vamos descrever o GEM. Em primeiro lugar, lembramos que a distribuição exponencial é uma boa aproximação para os tempos entre partidas de entidades deixando um nó $M/G/1/K$. De fato, é possível mostrar a "quase-reversibilidade" de uma classe ampla de filas finitas, que é a classe das filas finitas gerais dependentes do estado, $M/G/c/c$ [11]. Quando os clientes perdidos pelo bloqueio são incluídos, o processo de saída é Poisson. Esta hipótese é confirmada por diversos resultados empíricos [3, 60, 22, 59, 17, 19]. Os três estágios descritos as seguir fazem parte do GEM: *reconfiguração de rede*, *estimação dos parâmetros*, e *eliminação da retroalimentação*.

Reconfiguração de rede

Este estágio envolve a reconfiguração de rede. Um nó auxiliar h_j é criado. Tal nó é modelado como uma fila $M/G/\infty$ com taxa de serviço μ_h e precede

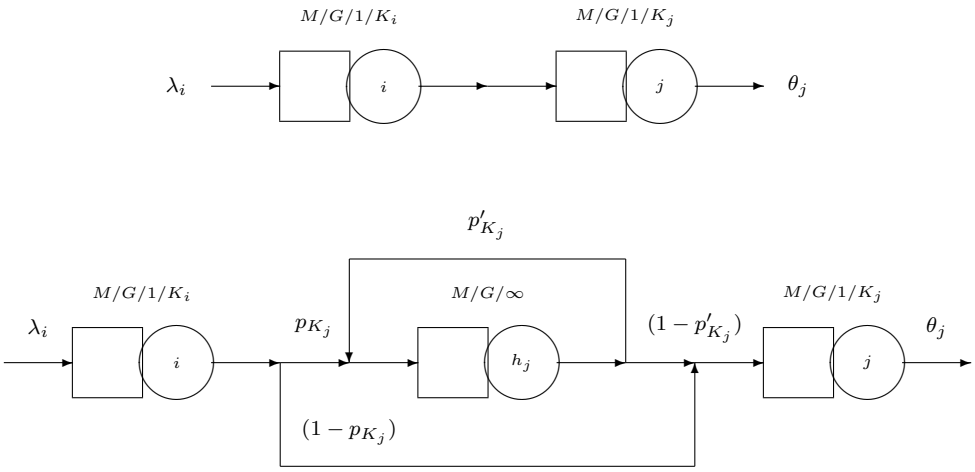


Figura 4.2: Método da expansão generalizada

cada fila finita j. Quando um cliente deixa o nó i, o nó seguinte, j, pode estar bloqueado (cheio) com probabilidade p_{K_j}, ou desbloqueado com probabilidade $(1 - p_{K_j})$. Sob bloqueio, os clientes são redirecionados para o nó h_j, para passarem por um período de tempo espera, enquanto o nó j estiver ocupado. Após essa espera, o cliente pode ser novamente bloqueado, com probabilidade p'_{K_j}, para um segundo período de espera. O nó h_j contabiliza o tempo que um cliente deve aguardar, antes de ser aceito no nó j. Contabiliza também a taxa de chegada efetiva (isto é, descontado os bloqueios) ao nó j.

Estimação de parâmetros

Neste estágio são estimados importantes parâmetros, quais sejam o p_K, o p'_K, e a taxa μ_h (para simplificar foi omitido o subscrito referente ao nó j).

1. A probabilidade de bloqueio, p_K, é obtida pela aproximação de dois momentos de Kimura, desenvolvida, para o caso $M/G/1/K$, por Smith [58], Eq. (4.1):

$$p_K = \frac{\rho^{\left(\frac{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}+2(K-1)}{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}}\right)}(\rho-1)}{\rho^{\left(2\frac{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}+(K-1)}{2+\sqrt{\rho}\mathrm{CV}^2-\sqrt{\rho}}\right)}-1}.$$

2. A probabilidade de um segundo bloqueio, p'_K, é obtida por uma aproximação via técnicas de difusão, desenvolvida por Labetoulle & Pujolle [44]:

$$p'_K = \left(\frac{\mu_j+\mu_h}{\mu_h} - \frac{\lambda\Big((r_2^K - r_1^K) - (r_2^{K-1} - r_1^{K-1})\Big)}{\mu_h\Big((r_2^{K+1} - r_1^{K+1}) - (r_2^K - r_1^K)\Big)}\right)^{-1}, \tag{4.4}$$

em que r_1 e r_2 são as raízes do polinômio

$$\lambda - (\lambda + \mu_h + \mu_j)x + \mu_h x^2 = 0, \tag{4.5}$$

com $\lambda = \lambda_j - \lambda_h(1 - p'_K)$, sendo λ_h a taxa de chegada real ao nó artificial criado e λ_j, a taxa de chegada real para o nó finito j, dadas pela expressão:

$$\lambda_j = \tilde{\lambda_i}(1 - p_K) = \tilde{\lambda_i} - \lambda_h. \tag{4.6}$$

em que $\tilde{\lambda_i}$ é a taxa de atendimento na fila antecessora.

3. Finalmente, a taxa μ_h é calculada como se segue, pela teoria da renovação:

$$\mu_h = \frac{2\mu_j}{1+\sigma_j^2\mu_j^2}, \tag{4.7}$$

em que σ_j^2 é a variância do tempo de serviço.

Eliminação da retroalimentação

As visitas repetidas ao nó artificial h_j (devidas à retroalimentação) produzem uma forte dependência no processo de chegada ao nó j. Portanto, devem ser eliminadas. Isto é conseguido por um acréscimo adequado ao tempo de serviço no nó i, durante sua primeira passagem através do nó de retenção. A taxa de serviço ajustada, para o nó h_j, μ'_h, é então:

$$\mu'_h = (1 - p'_K)\mu_h. \tag{4.8}$$

Resumo

Sumarizando, o objetivo do GEM é proporcionar um esquema algorítmico para aproximação das taxas de serviço dos nós i, que levem em conta o bloqueio após serviço, causados por possíveis bloqueios no nós que o seguem j:

$$\tilde{\mu}_i^{-1} = \mu_i^{-1} + p_K(\mu'_h)^{-1}. \tag{4.9}$$

Para cada nó (fila finita) j na rede, sucedendo o nó (fila finita) i, temos um conjunto de equações não lineares simultâneas para as variáveis p_K, p'_K, e μ_h, associadas com variáveis auxiliares, tais como λ e $\tilde{\lambda}_i$. Resolvendo tais equações simultaneamente de forma recursiva, pode-se calcular todas as medidas de desempenho da rede:

$$\lambda = \lambda_j - \lambda_h(1 - p'_K), \tag{4.10}$$

$$\lambda_j = \tilde{\lambda}_i(1 - p_K), \tag{4.11}$$

$$\lambda_j = \tilde{\lambda}_i - \lambda_h, \tag{4.12}$$

$$
\begin{aligned}
p'_K &= \left(\frac{\mu_j + \mu_h}{\mu_h} - \frac{\lambda\Big((r_2^K - r_1^K) - (r_2^{K-1} - r_1^{K-1})\Big)}{\mu_h\Big((r_2^{K+1} - r_1^{K+1}) - (r_2^K - r_1^K)\Big)} \right)^{-1}, && (4.13)\\
z &= (\lambda + 2\mu_h)^2 - 4\lambda\mu_h, && (4.14)\\
r_1 &= \frac{[(\lambda + 2\mu_h) - z^{\frac{1}{2}}]}{2\mu_h}, && (4.15)\\
r_2 &= \frac{[(\lambda + 2\mu_h) + z^{\frac{1}{2}}]}{2\mu_h}, && (4.16)\\
p_K &= \text{Eq. (4.1).}
\end{aligned}
$$

As Equações (4.10) a (4.13) são relativas às chegadas e à retroalimentação no nó artificial h_j. As Equações (4.14) a (4.16) são necessárias para resolver a Equação (4.13), sendo o z uma variável intermediária auxiliar, usada apenas por simplicidade. Finalmente, a Equação (4.1) fornece a probabilidade de bloqueio para a fila $M/G/1/K$. Assim, essencialmente temos cinco equações para resolver, *i.e.*, as Equações (4.10) a (4.13) e a Equação (4.1).

4.3 Algoritmo de otimização

Para o problema de otimização multiobjetivo em mãos, definido pelas Equações (3.9)–(3.11), os algoritmos multiobjetivos evolucionários (MOEAs, do inglês, *Multi-Objective Evolutionary Algorithms*), parecem ser uma escolha bastante apropriada. O MOEAs são algoritmos de otimização que realizam uma busca aproximada global baseada na informação obtida da estimativa de diversos pontos no espaço de busca [24, 14]. A população de pontos, que converge para um valor ótimo, é obtida através da aplicação dos operadores genéticos, quais sejam a *mutação*, o *cruzamento*, a *seleção*, e o *elitismo*. Cada um desse operadores caracteriza um tipo diferente de MOEA e pode ser implementado de diferentes maneiras. Além disso, a convergência do MOEA

é garantida pela atribuição de um valor de aptidão (do inglês, *fitness*) para cada membro da população, preservando a diversidade. De fato, aplicações bem-sucedidas recentes de algoritmos genéticos (GAs, do inglês *Genetic Algorithms*) em otimização mono-objetivo são relatadas por Lin [46] e Calvete et al. [7], enquanto que Carrano et al. [8] utilizam GAs em problemas de otimização multi-objetivo em redes. Adicionalmente, a eficiência dos GAs é bem estabelecida para problemas multiobjetivos [13, 32]. Muitas referências bibliográficas são fornecidas pelos autores mencionados.

4.3.1 Descrição

O tipo de MOEA usado nesta tese baseia-se no algoritmo NSGA-II (algoritmo genético de ordenação não-dominante elitista, do inglês *Elitist Non-dominated Sorting Genetic Algorithm*) de Deb et al. [27], mostrado na Figura 4.3. Na aplicação dos GAs a problemas de otimização multiobjetivo, os operadores de *seleção* e de *elitismo* precisam estar especificamente estruturados, para identificar corretamente as condições de otimalidade, conforme será mostrado em breve. O elitismo é baseado no conceito de dominância. Assim, um ponto $\mathbf{x}_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_n})$ é dito dominar um outro ponto $\mathbf{x}_j = (x_{j_1}, x_{j_2}, \ldots, x_{j_n})$ se $\mathbf{x}_i$ é superior a $\mathbf{x}_j$ em pelo menos um objetivo ($f_k(\mathbf{x}_i) < f_k(\mathbf{x}_j)$, para minimização) e é não inferior nos demais outros objetivos ($f_\ell(\mathbf{x}_i) \not> f_\ell(\mathbf{x}_j)$, para minimização).

Por exemplo, a Figura 4.4 mostra pontos (soluções) para um problema de maximização em duas dimensões (duas variáveis). Na Figura 4.4, o ponto V é dito dominado pelo ponto I, mas não pelos pontos II, III e IV. O ponto VI é dito dominado pelos pontos I, II e III, mas não pelo ponto IV. A melhor fronteira inclui os pontos de I a IV e é uma aproximação do conjunto de Pareto, que é o conjunto dos pontos que são superiores aos outros pontos. Para

algoritmo
leia grafo, taxas de chegada e de serviço, $G(N, A), \lambda_i \; \forall \; i \in N$
$P_1 \leftarrow$ **GeraPopulaçãoInicial**(`popSize`)
para $i = 1$ **até** `numGen` **faça**
/ gera filhos por cruzamento e mutação */*
$Q_i \leftarrow$ **FaçaPopulaçãoNova**(P_i)
/ combina pais e filhos */*
$R_i \leftarrow P_i \cup Q_i$
/ encontre fronteiras não-dominadas* $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2, \dots)$ **/*
$\mathcal{F} \leftarrow$ **OrdenaçãoNãoDominante**(R_i)
/ encontre nova população pela distância de aglomeração*/*
$P_{i+1} \leftarrow$ **GeraPopulaçãoNova**(R_i)
fim para
$P_{\texttt{numGen}+1} \leftarrow$ **ExtraiPareto**$(P_{\texttt{numGen}})$
escreva $P_{\texttt{numGen}+1}$
fim algoritmo

Figura 4.3: Algoritmo NSGA-II

executar o elitismo, foi empregado um algoritmo comumente referido como algoritmo rápido de ordenação não-dominante (detalhes podem ser encontrados em Deb et al. [27]). Este algoritmo separa os indivíduos da população em várias camadas (ou fronteiras) $\mathcal{F}_i$, tais que as soluções em $\mathcal{F}_1$ são não-dominantes e toda solução em uma dada camada $\mathcal{F}_1$ é dominada por ao menos uma solução em $\mathcal{F}_{i-1}$ e não dominada por nenhuma solução em $\mathcal{F}_j$, $j \geq i$. Isto pode ser conseguido de forma eficiente, por meio de um algoritmo com complexidade de tempo de execução de ordem $\mathcal{O}(n \log n)$ [27].

A seleção é realizada selecionando sequencialmente pontos de cada fronteira não-dominante $(\mathcal{F}_1, \mathcal{F}_2, \dots)$ até que o número de indivíduos necessários para a próxima iteração seja alcançado. O critério precisa ser aplicado se, após a adição de um grupo de indivíduos de $\mathcal{F}_i$, o máximo número de indivíduos, é excedido. O algoritmo calcula uma medida de diversidade, denomi-

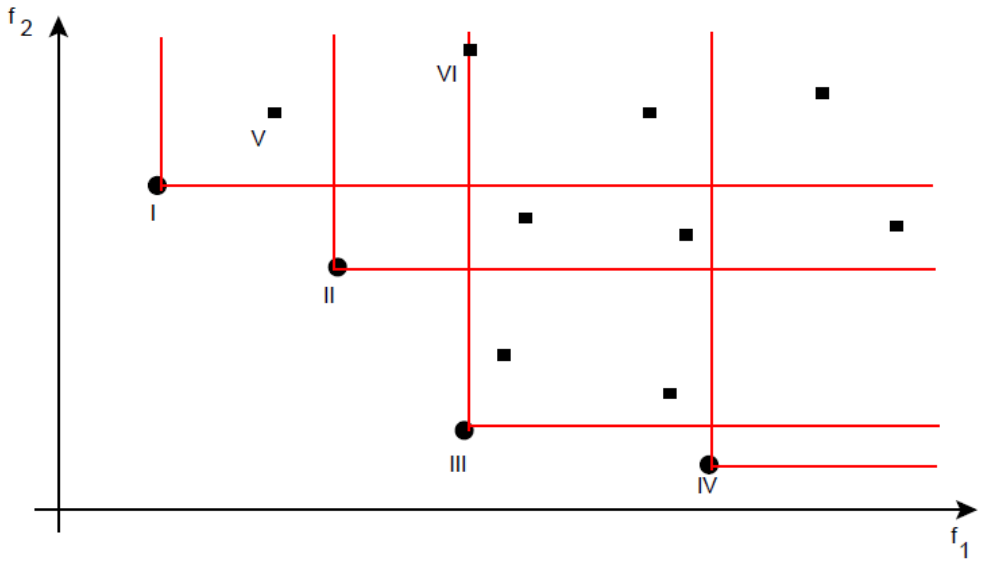


Figura 4.4: Pontos dominados (▪) e não-dominados (•)

nada *distância de aglomeração* (do inglês, *crowding distance*), como definido por Deb et al. [27], para assegurar a maior diversidade possível. A distância de aglomerção é uma métrica associada à área do retângulo definido na Figura 4.5, ou então, para dimensões maiores, associada ao hipervolume de um hiper-paralelogramo. Então, somente os pontos com a maior distância de aglomeração são mantidos para iterações futuras, como mostrado na Figura 4.5.

Os operadores de *cruzamento* e de *mutação* são um pouco independentes da natureza multiobjetivo do problema, mas são fortemente dependentes da aplicação. Para o problema em questão, o mecanismo de cruzamento uniforme (do inglês, *uniform crossover*) é selecionado [4]. O cruzamento uniforme é bem popular em codificações de variáveis múltiplas, devido à sua eficiência na identificação, herança e proteção de genes comuns, bem como na recombinação de genes incomuns [37, 63]. Neste mecanismo, os cruzamentos são realizados para cada variável com uma probabilidade (`rateCro`)

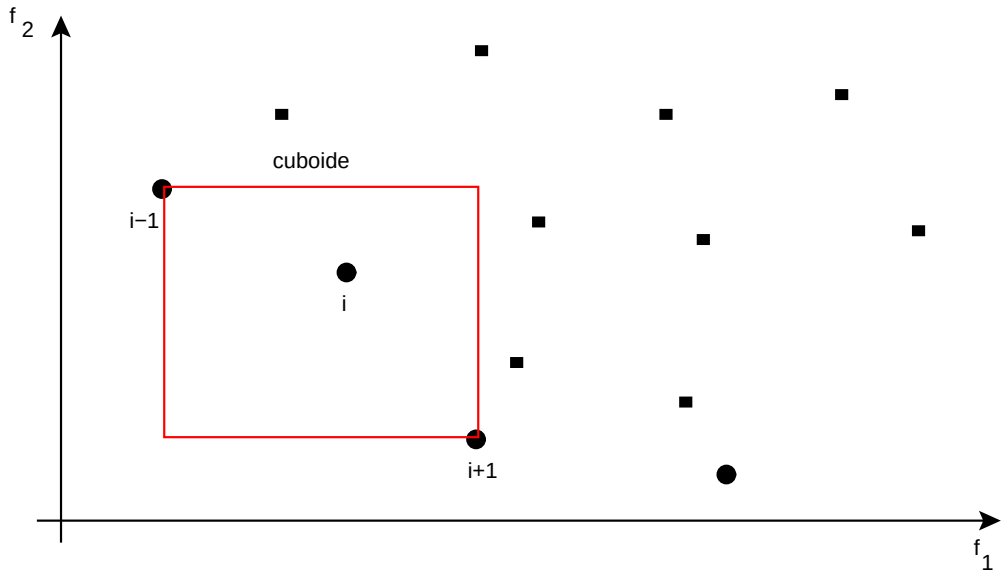


Figura 4.5: Ilustração da distância de aglomeração (*crowding distance*)

que está de acordo com o operador de cruzamento. O operador de cruzamento usado no algoritmo é o operador cruzamento binário simulado (SBX, do inglês *Simulated Binary Crossover*) [25, 26], como ilustrado na Figura 4.6.

O operador SBX é bastante conveniente para GAs com codificação das variáveis por números reais uma vez que é capaz de simular os operadores de cruzamento binários, *sem* a necessidade de recodificação das variáveis reais para números binários. Os filhos ($x_{i,(\bullet,t+1)}$) são obtidos dos pais ($x_{i,(\bullet,t)}$), conforme a seguinte equação.

$$\begin{aligned} x_{i,(1,t+1)} &= 0{,}5\Big[(1+\beta)x_{i,(1,t)} + (1-\beta)x_{i,(2,t)}\Big], \\ x_{i,(2,t+1)} &= 0{,}5\Big[(1-\beta)x_{i,(1,t)} + (1+\beta)x_{i,(2,t)}\Big], \end{aligned}$$

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

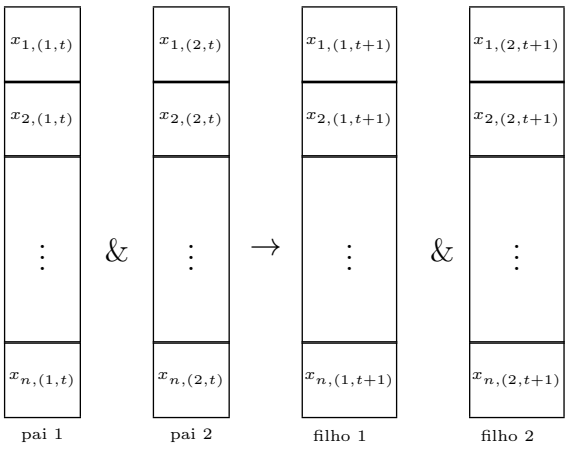


Figura 4.6: Representação dos cromossomos e o cruzamento binário simulado (SBX)

$$f(\beta) = \begin{cases} 0{,}5(\eta+1)\beta^{\eta}, & \text{se } \beta \leq 1; \\ 0{,}5(\eta+1)\frac{1}{\beta^{\eta+2}}, & \text{caso contrário.} \end{cases} \tag{4.17}$$

A função $f(\beta)$ foi projetada para produzir filhos (soluções) que possuam uma capacidade de busca semelhante à do cruzamento binário (mais comum) [25]. Pelo ajuste apropriado do parâmetro η, diferentes pesos (β) podem ser gerados, quer seja para produzir filhos mais parecidos aos seus pais (*i.e.*, valores elevados de η), quer seja para o contrário (*i.e.*, valores baixos de η). Diversas distribuições para a variável β são mostradas na Figura 4.7, em função do parâmetro η.

Para cada gene individual (isto é, para cada uma das variáveis de decisão K_i e μ_i), o esquema de mutação ocorre com uma probabilidade específica (`rateMut`). Como sugerido por Deb & Agrawal [25], perturbações gaussianas foram adicionadas às variáveis de decisão, isto é, $K_i + \varepsilon_i$ e $\mu_i + \varepsilon_{N+i}$, para todo $i \in N$, com $\varepsilon_i \sim \mathcal{N}(0,1)$, $i \in \{1, 2, \ldots, 2N\}$.

Após o cruzamento e a mutação, as restrições (3.10) e (3.11) podem ter sido violadas. Para garantir a sua viabilidade, os valores das variáveis inteiras

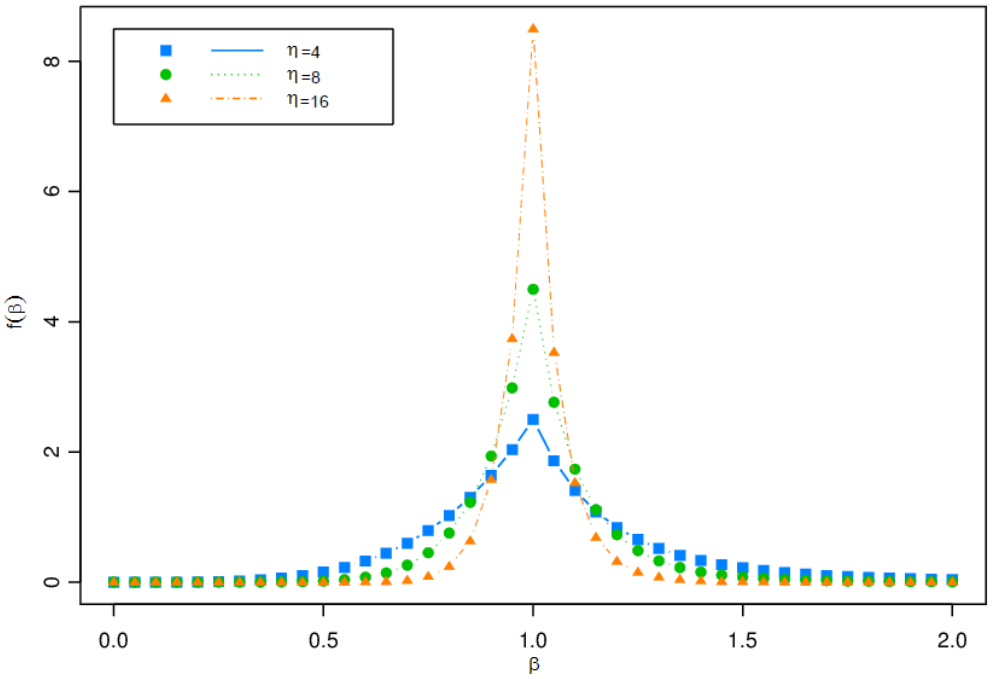


Figura 4.7: Função de densidade de probabilidade de β

são arredondados e depois, juntamente com as variáveis reais, são convenientemente ajustados pela aplicação dos seguintes operadores de reflexão:

$$K_{\text{refl}_i} = 1 + |K_i - 1|, \tag{4.18}$$

e

$$\mu_{\text{refl}_i} = \mu_{\text{linf}_i} + |\mu_i - \mu_{\text{linf}_i}|, \tag{4.19}$$

em que 1 é o limite inferior para a alocação das áreas de espera, μ_{linf_i} é o limite inferior da alocação do serviço (para assegurar que $\rho < 1$), K_i e μ_i são os valores resultantes após o cruzamento e a mutação, e K_{refl_i} e μ_{refl_i} são os valores que resultaram após a operação de reflexão. Este esquema garante sempre soluções viáveis, sem evitar ou favorecer qualquer solução particular.

4.3.2 Notas sobre convergência

Recentemente, o critério de parada de algoritmos evolucionários para otimização multiobjetivo tem sido analisado detalhadamente (veja, por exemplo, Rudenko & Schoenauer [55] e Martí et al. [48]). Evidentemente, o número máximo de gerações (`numGen`) desempenha um papel importante na qualidade das soluções. Entretanto, aumentar o número de gerações pode não ser ideal porque o tempo computacional é desperdiçado quando muitas iterações não conduzem a uma melhoria significativa. Rudenko & Schoenauer [55] sugeriram que um critério de parada eficiente é obtido monitorando-se o número de iterações realizado sem que tenha havido uma melhoria na solução. Para demonstrar a complexidade da questão, Rudenko & Schoenauer [55] conduziram um conjunto extensivo de experimentos computacionais. Seus resultados revelaram que um critério de parada óbvio, tal como a população inteira pertencer à fronteira $\mathcal{F}_1$, não indicava que a evolução havia sido concluída. Os autores propuseram um critério de parada local que calcula uma medida da estabilidade das soluções não-dominantes após cada iteração. Outro critério de parada global foi recentemente proposto por Martí et al. [48]. Este sofisticado método enxerga a evolução da população como um sistema dinâmico, no qual o seu estado pode ser estimado por um filtro de Kalman. Nesta tese, não é objetivo utilizar um critério acima de quaisquer outros. Por simplicidade, o critério de Rudenko & Schoenauer [55] será adotado. Este critério é baseado na estabilização da máxima distância de aglomeração, d_l, medida sobre L gerações, calculada pelo seguinte desvio padrão:

$$\sigma_L = \sqrt{\frac{1}{L}\sum_{l=1}^{L}(d_l - \bar{d}_L)^2}. \tag{4.20}$$

Como mostrado na Equação (4.20), $\bar{d}_L$ é a média de d_l sobre L gerações.

Além disso, a Equação (4.20) indica que o MOEA pode parar quando $\sigma_L < \delta_{\text{lim}}$. Rudenko & Schoenauer [55] sugerem que σ_L não depende dos valores atuais da função objetivo, uma vez que as distâncias de aglomeração são normalizadas. Além disso, eles também sugerem que L e δ_{lim} devem ser ajustados para 40 e 0,02, respectivamente, levando a um critério de parada que é $\sigma_{40} \leq 0,02$. Conforme evidências empíricas [55], esses valores são compatíveis com as redes em consideração nesta tese.

Capítulo 5

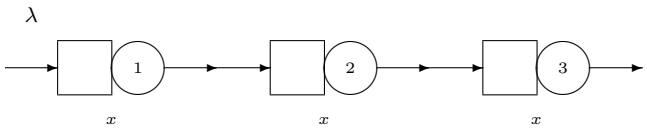
Resultados Computacionais e Discussão

Para aproveitar implementações prévias do GEM [58, 57], baseadas na biblioteca numérica IMSL (*International Mathematics and Statistics Library*), o algoritmo de otimização foi implementado na linguagem FORTRAN [18, 6, 5]. O código está disponível a pedido, diretamente com o autor, para para fins educacionais e de pesquisa. Experimentos computacionais iniciais foram conduzidos para determinar o conjunto adequado de parâmetros para garantir a rápida convergência do algoritmo de otimização multiobjetivo. Em seguida, a análise de uma rede maior e mais complexa é realizada com os algoritmos propostos.

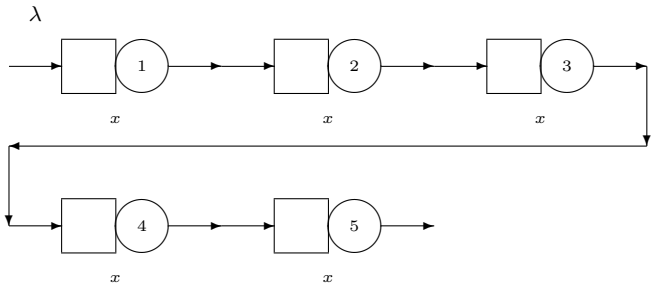
5.1 Configuração do algoritmo

Infelizmente, para assegurar uma rápida convergência com um mínimo de esforço computacional, a configuração adequada dos parâmetros que garantem a rápida convergência do algoritmo precisa ser determinada por tentativa e erro, como indicado por estudos prévios com os GAs. Assim, redes contendo 3, 5 e 10 filas $M/G/1/K$ foram experimentadas para definir esta configuração de parâmetros do MOEA (veja Figura 5.1). Por conveniência e concisão,

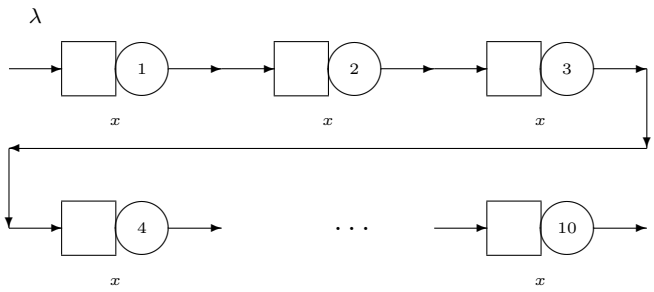
apenas os resultados obtidos com redes de 3 e 10 nós são apresentados (Figuras 5.2 a 5.5). Diferentes topologias de redes acíclicas, de diversos tamanhos, foram também testadas. Os resultados (não apresentados) foram similares.



(a) topologia série com três nós



(b) topologia série com cinco nós

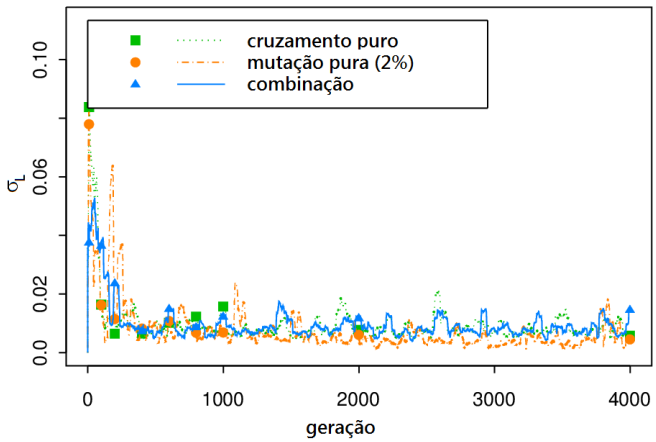


(c) topologia série com dez nós

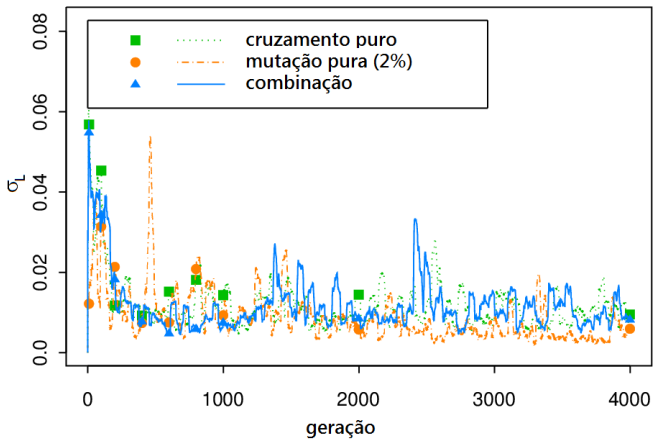
Figura 5.1: Topologias testadas

Neste estudo, cada fator foi analisado independentemente. Especificamente, cada fator foi variado individualmente, enquanto os outros parâme-

tros foram mantidos constantes. Montgomery [53] lembra-nos que a principal desvantagem de uma análise independente é que não se consegue explicar as interações entre as variáveis. Entretanto, experimentos recentes divulgados por Cruz et al. [21] indicaram que potenciais interações não são significativas. Desse modo, interações entre fatores não foram consideradas nesta tese.



(a) rede de três nós em topologia série

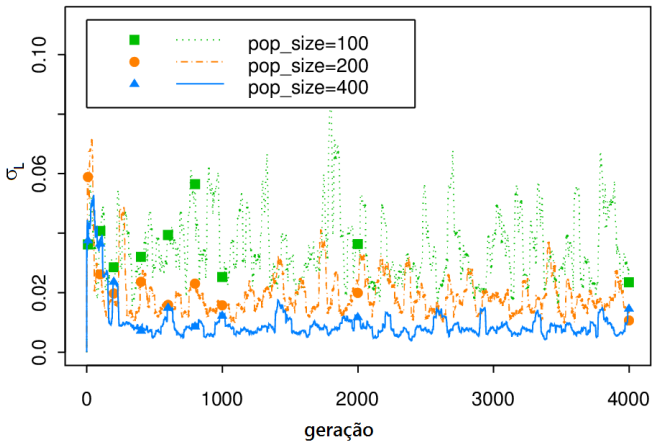


(b) rede de dez nós em topologia série

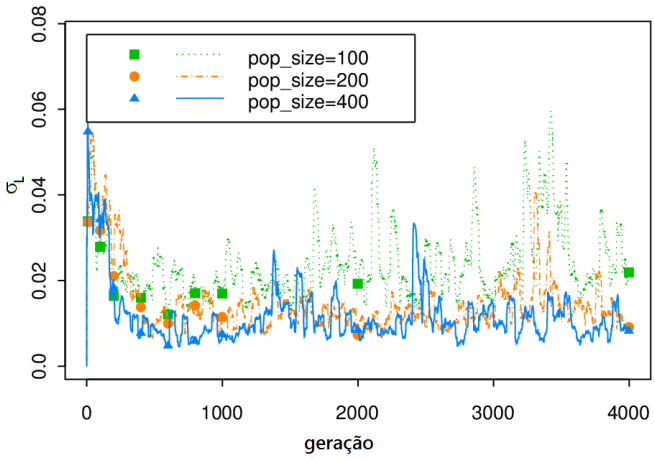
Figura 5.2: Efeito do cruzamento e da mutação

A Figura 5.2 apresenta a velocidade de convergência (em termos de σ_L) em função do número de gerações. É possível concluir que um algoritmo genético com mutação *pura* (isto é, sem a operação de cruzamento) teria sido

bem sucedido na descoberta das soluções sub-ótimas (de fato, algumas vezes a mutação pura resolve o problema, conforme visto, por exemplo, em Mathieu et al. [49]). Entretanto, o operador SBX foi também utilizado porque ele removeu irregularidades do perfil da convergência. A combinação de mutação pura e SBX proporcionou resultados satisfatórios, independentemente do número de filas na rede.



(a) rede de três nós em topologia série

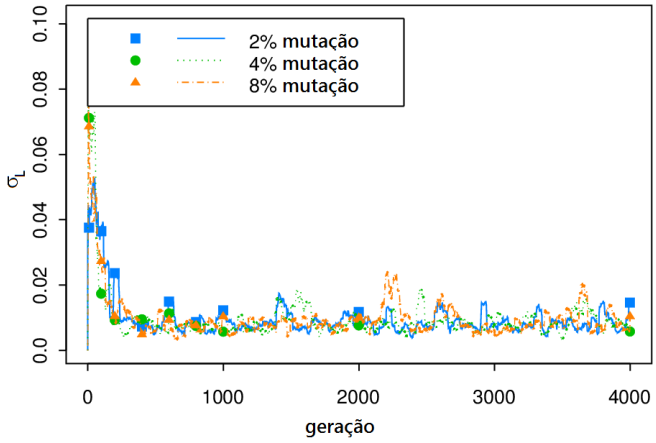


(b) rede de dez nós em topologia série

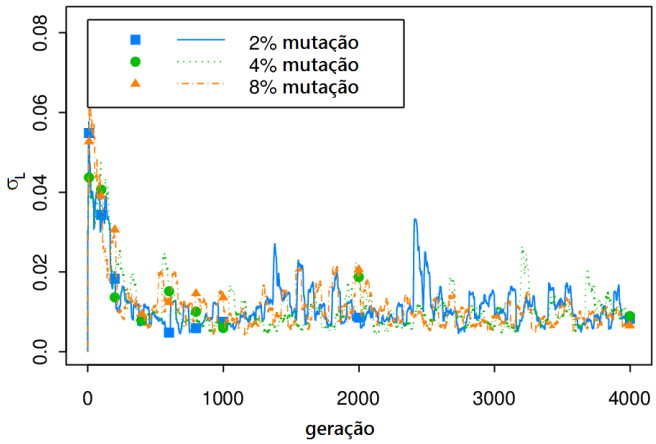
Figura 5.3: Efeito do tamanho da população

Os resultados apresentados na Figura 5.3 revelam que o tamanho da po-

pulação (`popSize`) teve um efeito significativo na convergência do algoritmo. Entretanto, o tamanho da população não pode crescer arbitrariamente porque o esforço computacional requerido inviabilizaria o estudo. Além disso, o desempenho do algoritmo não foi afetado pelo aumento no número de nós na rede.



(a) rede de três nós em topologia série

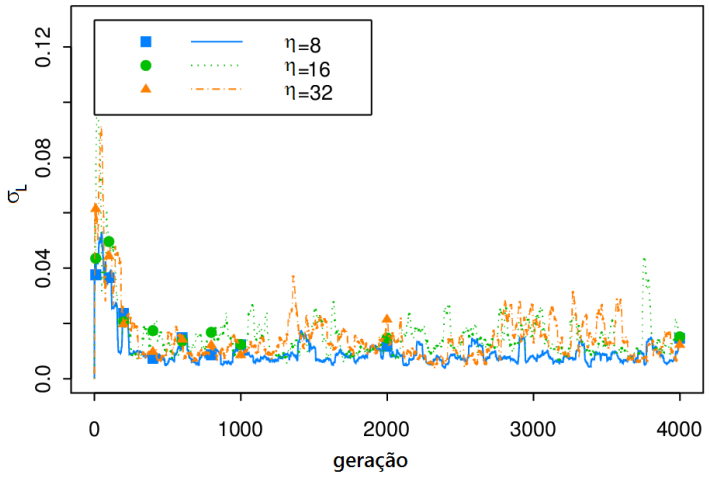


(b) rede de dez nós em topologia série

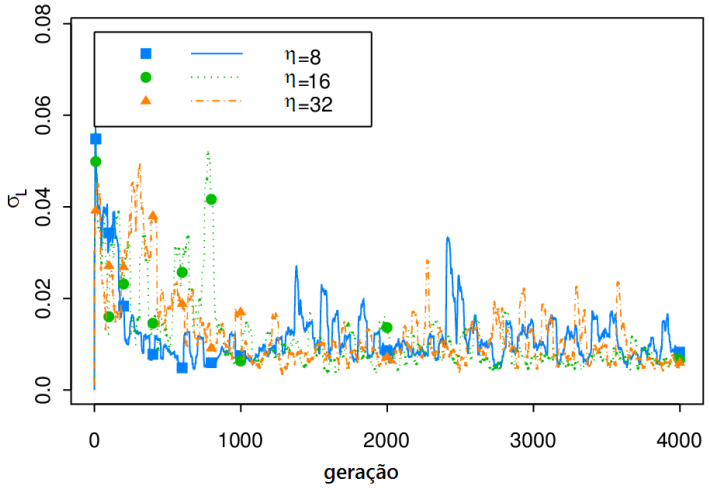
Figura 5.4: Efeito da taxa de mutação

A Figura 5.4 mostra a taxa de convergência como uma função da taxa de mutação `rateMut`. Os resultados mostraram que um aumento na taxa de mutação acelerou a convergência. No entanto, uma vez atingida uma taxa es-

pecífica, um aumento adicional não conduziu a um melhora da convergência. Sob as condições do experimento, taxas de mutação entre 1 e 2% conduziram a resultados superiores, independentemente do número de nós.



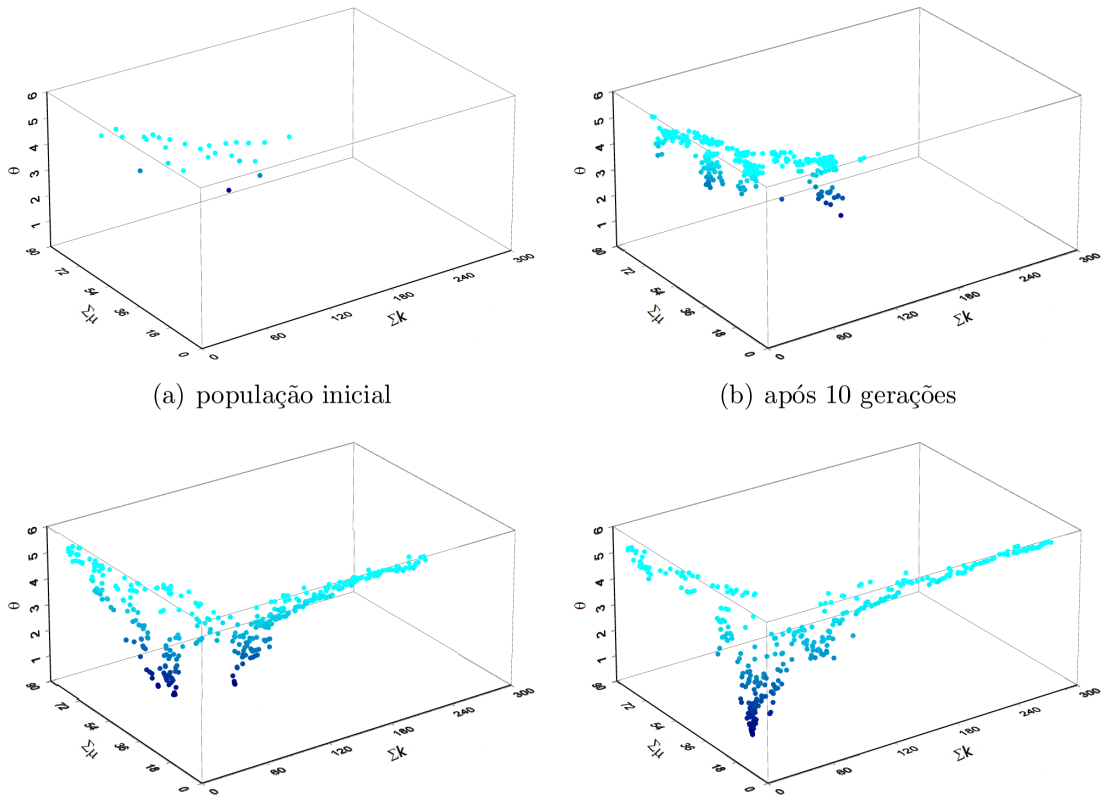
(a) rede de três nós em topologia série



(b) rede de dez nós em topologia série

Figura 5.5: Efeito do parâmetro η

A Figura 5.5 mostra a convergência como uma função de η, que controla o efeito de dispersão de β_q para o operador SBX, Equação (4.17). Uma melhoria adicional na velocidade de convergência não foi observada para valores de η



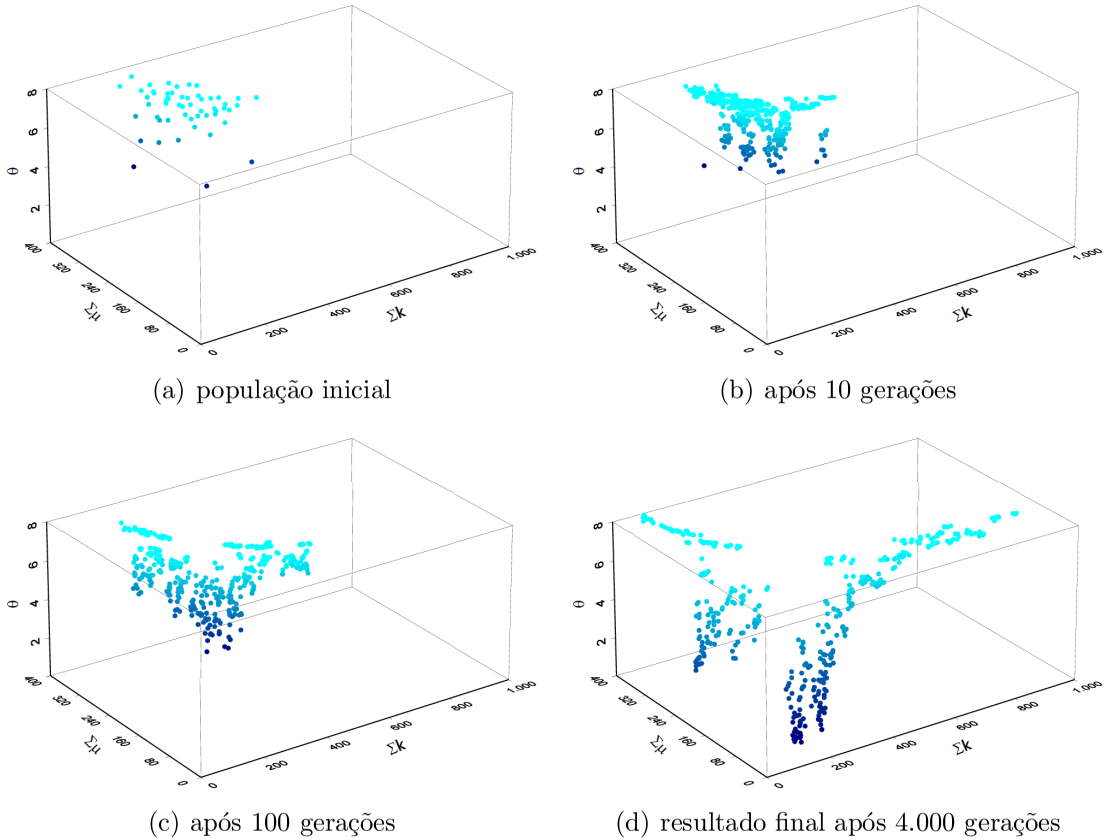
(a) população inicial

(b) após 10 gerações

(c) após 100 gerações

(d) resultado final após 4.000 gerações

Figura 5.6: Evolução da população para a rede de três nós



(a) população inicial

(b) após 10 gerações

(c) após 100 gerações

(d) resultado final após 4.000 gerações

Figura 5.7: Evolução da população para a rede de dez nós

maiores do que 8.

Finalmente, Figuras 5.6 e 5.7 ilustram a evolução da população, desde o ponto de partida até a geração final. Elas mostram a população se espalhando ao longo do tempo para cobrir uma proporção crescente do espaço objetivo.

Em resumo, todos os problemas testados puderam ser resolvidos com sucesso, pelo emprego da seguinte combinação:

1. uso combinado do operador de mutação e do SBX;
2. tamanho da população (`popSize`) igual a 400 indivíduos (400 soluções);
3. taxa de mutação de 2%;
4. parâmetro de dispersão (η) igual a 8;
5. para assegurar que o cálculo estará completo dentro de um tempo razoável, o número máximo de gerações (`numGen`) foi estabelecido em 4000 gerações.

Uma conclusão bastante importante diz respeito à robustez do algoritmo, que aparenta ter uma convergência aproximadamente independente da topologia (resultados não mostrados, para redes com divisão de fluxos, fusões e mistas), da taxa de chegada externa ($\boldsymbol{\lambda}$), do quadrado do coeficiente de variação do tempo de serviço (CV^2) e do número de nós da rede.

5.2 Análise do tempo de processamento

Considerada escolhida a configuração adequada de parâmetros para a utilização do NSGA-II, na resolução do problema proposto, as informações sobre o tempo de processamento do algoritmo são de relevância. Para tanto, foi

executado um conjunto de simulações para diferentes situações. Foram considerados os sistemas em série com 3, 5 e 10 nós, já mencionados anteriormente, para a calibração da configuração do algoritmo. Um sistema em série com 6 nós, além de configurações de topologia mais complexa (redes mistas com 6 e 16 nós), incluindo fusões e divisões entre as filas do sistema, também foram utilizados. As configurações mistas serão discutidas neste momento apenas no que diz respeito aos valores dos tempos de processamento. Entretanto, serão abordadas em mais detalhes, a seguir, com relação aos valores das funções objetivo.

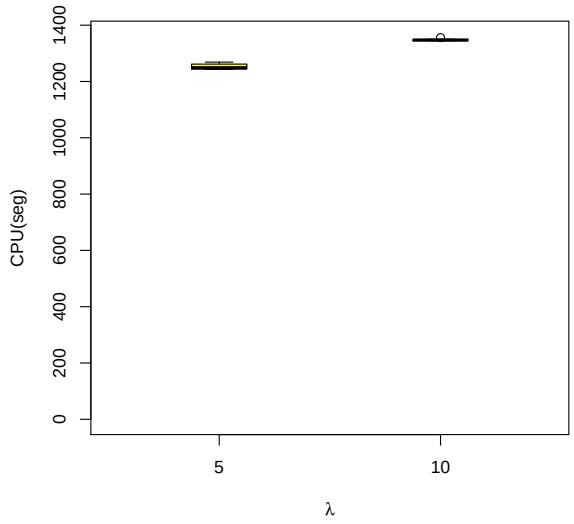


Figura 5.8: Tempos de processamento para rede mista com 6 nós e diferentes taxas de entrada λ

Para cada uma das configurações em análise, o experimento com a mesma configuração foi replicado por 20 vezes, para a produção dos dados desta análise. Os experimentos foram conduzidos em um equipamento com sistema operacional Windows 7 e um processador Intel Core i3 com 2.10 GHz.

A primeira análise realizada foi para verificar algum efeito no tempo com-

putacional decorrente da taxa de entrada no sistema (λ). Foi utilizada uma rede de topologia mista (ver Figura 5.16) composta por 6 nós. As taxas de entrada utilizadas foram $\lambda = 5$ e $\lambda = 10$.

A Figura 5.8 apresenta um aumento no tempo de processamento associado a um aumento na taxa de entrada. Entretanto, pode-se notar que este efeito não parece muito pronunciado. De fato, ocorre um aumento de aproximadamente 10% em tempo de processamento médio, para um aumento de 100% na taxa de entrada no sistema. Além disso, ocorreu uma redução do efeito de variabilidade entre os tempos medidos para o maior valor de λ.

Considerando ainda as configurações de sistemas com 6 filas, foi verificado o efeito da utilização de diferentes valores para o quadrado do coeficiente de variação. Nesta análise os valores de CV^2 utilizados foram 0,5, 1,0 e 1,5.

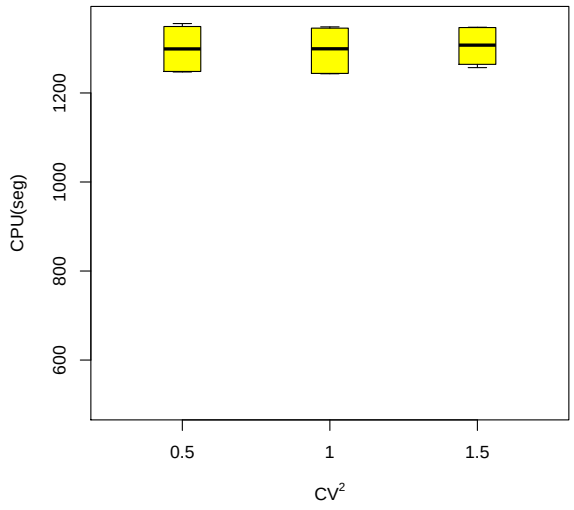


Figura 5.9: Tempos de processamento para redes com 6 nós e diferentes valores para CV^2

Pela Figura 5.9, nota-se que modificações nos valores para CV^2 não acarretam mudanças notáveis nos tempos de processamento médios.

Outro fator que poderia afetar o tempo de execução do algoritmo é a topologia da rede em análise. Para avaliação deste efeito, foram consideradas duas configurações de rede composta por 6 nós, uma delas com as filas do sistema dispostas em série e outra com a topologia incluindo fusão e divisão.

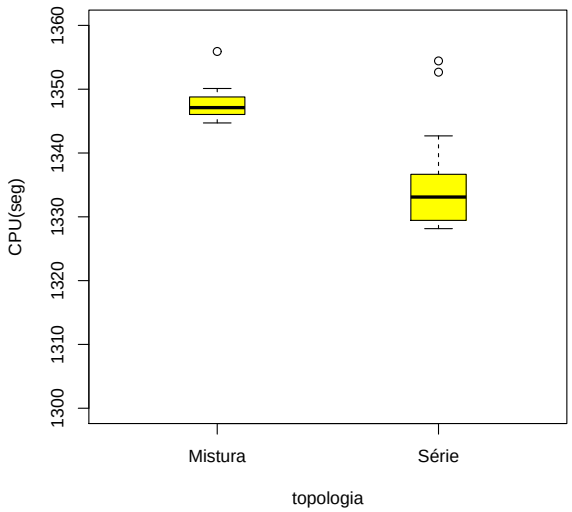


Figura 5.10: Tempos de processamento para redes em série e mista com 6 nós

Na Figura 5.10 observa-se um aumento no tempo de processamento em redes mistas, se comparado aos tempos para redes em série. Entretanto o acréscimo no tempo de processamento médio foi pequeno (aproximadamente 1,5%). Desta forma não existe um aumento de custo computacional significativo associado a topologia da rede em estudo.

O último efeito considerado foi a quantidade de filas no sistema. De antemão já era aguardado um aumento do tempo de processamento associado ao aumento do número de filas no sistema. Isso se deve ao aumento do número de possíveis configurações do sistema que deverão ser avaliadas pelo algoritmo.

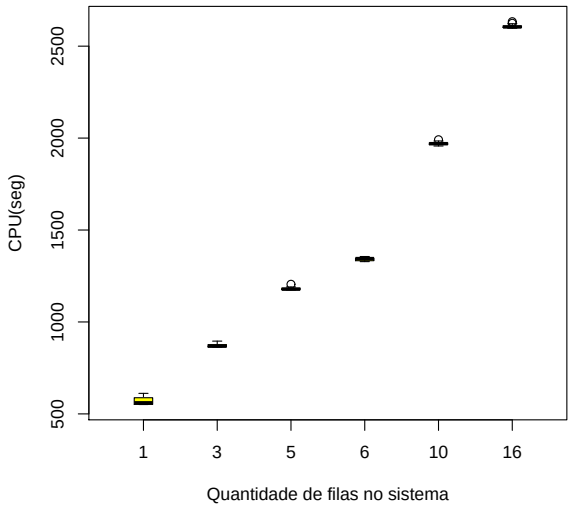


Figura 5.11: Tempos de processamento para diferentes quantidades de filas no sistema em estudo

O aumento esperado nos tempos de processamento é confirmado através da Figura 5.11. A expectativa era de que este aumento em tempo de execução não fosse "exponencial". Este resultado foi satisfeito e portanto não se observa restrições quanto ao custo computacional envolvido para diferente possibilidade de redes de filas em análise.

5.3 Analogia entre as formulações matemáticas

Andriansyah et al. [3] descrevem um grande volume de resultados baseados na formulação matemática bi-objetivo, definida pelas Equações (3.7)–(3.8), pág. 61. Em Andriansyah et al. [3], o algoritmo de otimização utilizado é também o NSGA-II. O estudo por eles realizado inicia-se com uma comparação entre uma solução de referência (o valor da taxa de atendimento para uma rede em que todas as áreas de circulação nas filas do sistema são de

mesmo tamanho, ou seja, sistema simétrico) com as soluções obtidas através da estratégia bi-objetivo. Nestes casos, a solução de referência é uma das soluções obtidas, ou existem soluções que fornecem uma taxa de atendimento suficientemente próxima da solução de referência para uma alocação total de áreas de circulação inferior (veja Figura 5.12).

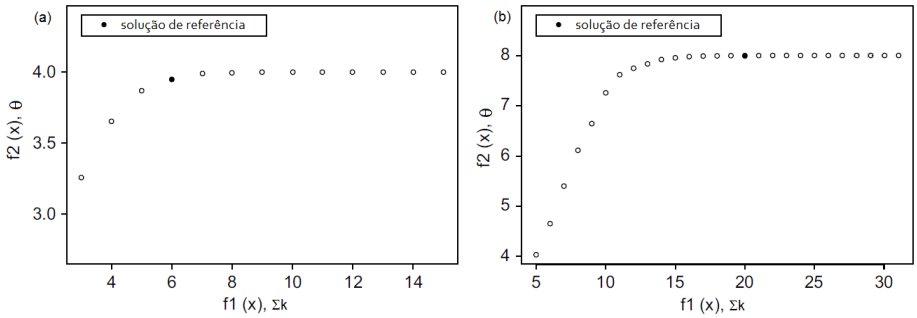


Figura 5.12: Pareto ótimo para formulação bi-objetivo com solução de referência para sistema de filas simétrico: (a) rede com 3 nós e $\lambda = 4$, (b) rede com 5 nós e $\lambda = 4$ (gráficos obtidos em Andriansyah et al. [3])

Andriansyah et al. [3] também avaliam casos em que as soluções de referência da literatura não possuem o mesmo tamanho de área de circulação (sistema assimétrico) em todas as filas do sistema, ou seja, possíveis soluções para a formulação matemática mono-objetivo, Equações (3.4)-(3.6), pág. 60. Em todos os casos avaliados as soluções de referência foram dominadas por soluções fornecidas pelo algoritmo genético para a formulação bi-objetivo (veja Figura 5.13).

Esses resultados atestam que a estratégia bi-ojetivo contempla os resultados para a avaliação mono-objetivo e aumentam o leque de opções disponível para o decisor ao apresentar uma família de soluções (conjunto Pareto) que podem ser escolhidas de acordo com necessidades mais específicas do problema em estudo. Parece que tais conclusões de Andriansyah et al. [3], obtidas com a abordagem bi-objetivo, poderiam ser generalizadas para a for-

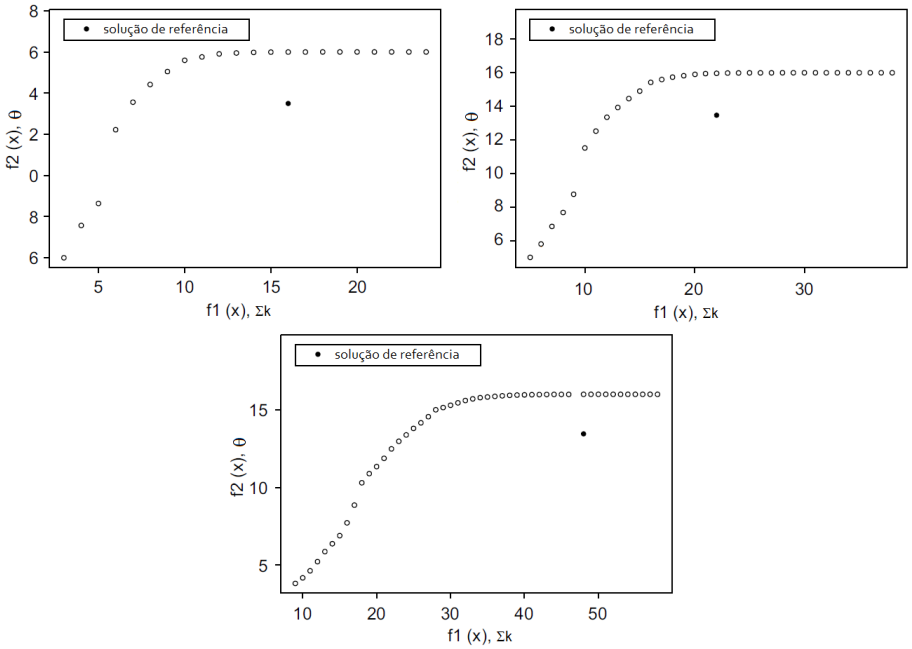


Figura 5.13: Pareto ótimo para formulação bi-objetivo com solução de referência para sistema de filas assimétrico: (a) rede com 3 nós e $\lambda = 4$, (b) rede com 5 nós e $\lambda = 4$, (c) rede com 9 nós e $\lambda = 16$ (gráficos obtidos em Andriansyah et al. [3])

mulação matemática tri-objetivo proposta nesta tese.

Observando a Figura 5.13 pode-se notar que o conjunto Pareto ótimo fornecido pelo algoritmo genético é capaz de produzir soluções com o mesma área total de circulação, porém obtendo uma taxa de atendimento superior. Em outras palavras, os operadores genéticos do algoritmo asseguram a procura por algumas das possíveis soluções que preservam a mesma área total de circulação e fornecem uma maior taxa de atendimento. Assim, alguma recombinação na distribuição das áreas de circulação entre as filas do sistema é feita, mas não tanto quanto poderia ser se uma estratégia de pós-processamento (busca local) fosse utilizada. Esta é uma forma de melhorar o desempenho do algoritmo proposto, acelerando sua convergência [30, 61].

5.4 Análise de uma rede maior e mais complexa

A rede apresentada na Figura 1.1 foi retirada da literatura [59] e analisada com os algoritmos descritos. Diferentes valores para o quadrado do coeficiente de variação do tempo de serviço foram analisados (isto é, $CV^2 = \{0{,}5, 1{,}0, 1{,}5\}$), para uma taxa de chegada (λ_1) igual a 5. Primeiramente, a velocidade de convergência do algoritmo genético confirmou ser robusta para este tipo de problema. Além disso, os resultados indicaram que a convergência estabilizou-se em 2.000 iterações. Adicionalmente, como mostrado na Figura 5.14 a convergência parece ser independente do quadrado do coeficiente de variação do tempo de serviço.

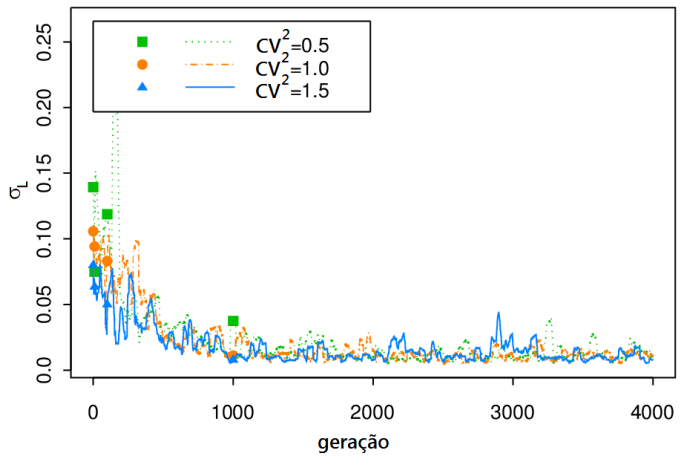
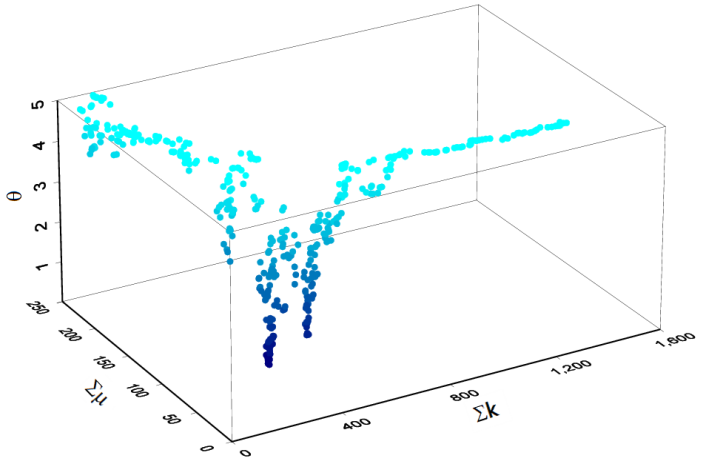
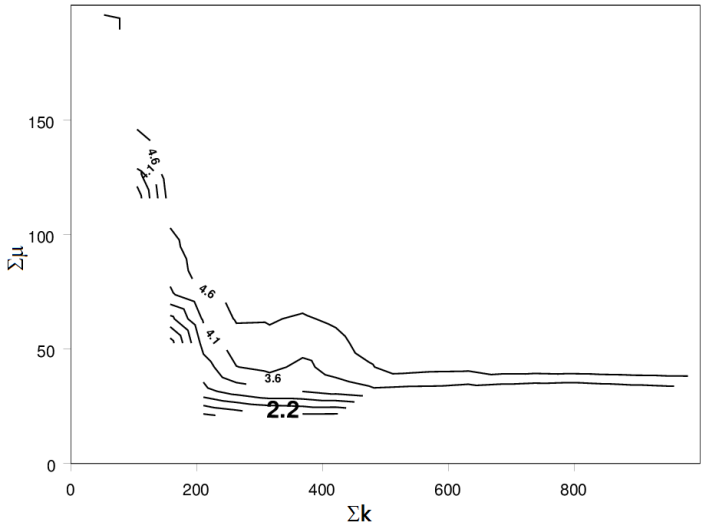


Figura 5.14: Convergência para a rede de dezesseis nós da figura 1.1

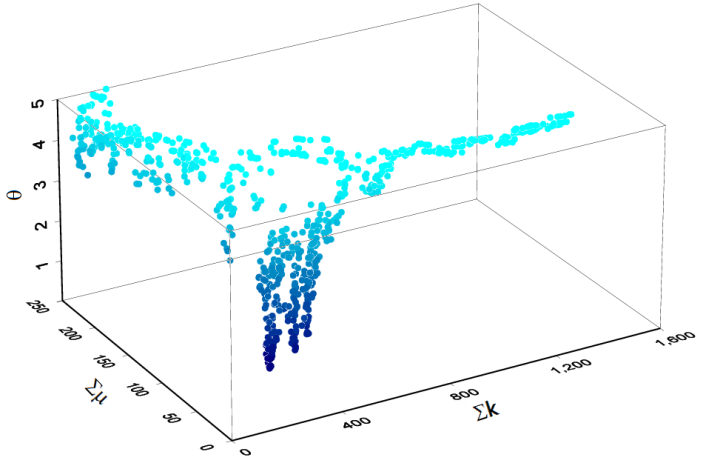
Os perfis correspondentes são mostrados na Figura 5.15, incluindo as curvas de nível e a superfície final após a convergência. Para comparação, curvas de nível exatas, para uma única fila geral $M/G/1/K$, são apresentadas na Figura 1.2-b (vide pág. 6). A similaridade entre as superfícies dos dois gráficos é encorajadora. Infelizmente, não parece que o comportamento de uma dada rede possa ser previsto, sem a utilização de uma abordagem algorítmica, tal



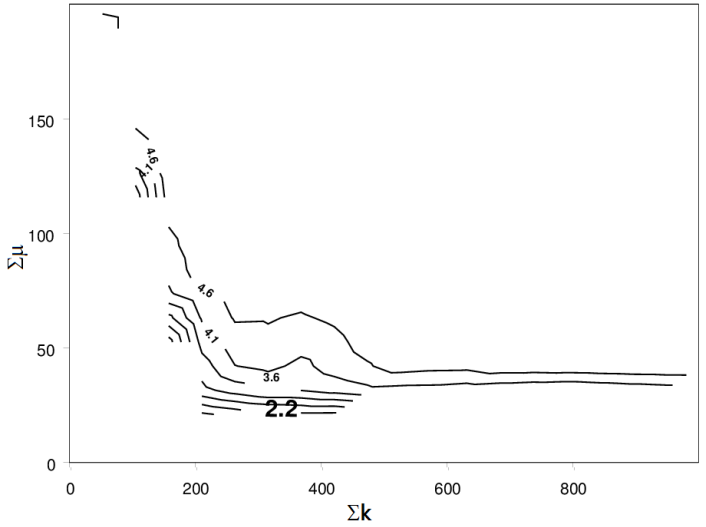
(a) superfície final para $\mathrm{CV}^2 = 0,5$



(b) contorno para $\mathrm{CV}^2 = 0,5$



(c) superfície final para $\mathrm{CV}^2 = 1,5$



(d) contorno para $\mathrm{CV}^2 = 1,5$

Figura 5.15: Resultado final para a rede de dezesseis nós da figura 1.1

como a abordagem aqui proposta.

Uma análise detalhada dos resultados na Figura 5.15 revela que muitas combinações diferentes para as áreas de espera e para as taxas de serviço alocadas podem ser selecionados, para assegurar uma certa taxa de atendimento. Adicionalmente, é possível estimar quais são os valores a partir dos quais um aumento no tamanho das áreas de espera e nas taxas de atendimento não mais têm efeito sobre a taxa de atendimento (isto é, quando as curvas de nível formam linhas paralelas ao eixo horizontal).

Com o algoritmo proposto, uma melhor compreensão é alcançada para estes sistemas de redes de filas. Por exemplo, os resultados na Figura 5.15-d sugerem que é mais fácil aumentar a taxa de atendimento de 2,6 a 3,1 (20% de acréscimo) do que de 4,1 a 4,5 (10% de acréscimo). Curvas de nível distantes entre si indicam que melhorias podem ser alcançadas apenas pelo aumento drástico das capacidades alocadas e das taxas de serviço.

5.5 Análise de uma rede em topologia mista

A rede apresentada na Figura 5.16 foi analisada para dois valores $CV^2 = \{0{,}5, 1{,}5\}$, com taxa de chegada $\lambda_1 = 1$. Primeiramente, a convergência do algoritmo foi confirmada, para o mesmo conjunto de parâmetros anteriormente utilizado. Entretanto, para este rede, que é menor, os resultados indicaram que a convergência chega mais cedo, ao final de 2.000 gerações (para maiores detalhes, veja Brito et al. [5]).

A Tabela 5.1 apresenta algumas soluções eficientes de Pareto, para uma análise mais detalhada. Note que a metodologia multiobjetivo permite a identificação de pontos a partir dos quais não há vantagem em aumentar o tamanho das áreas de espera devido ao fato de o ganho na taxa de atendimento ser pequeno. Por exemplo, para $CV^2 = 0{,}5$, mantendo-se as taxas

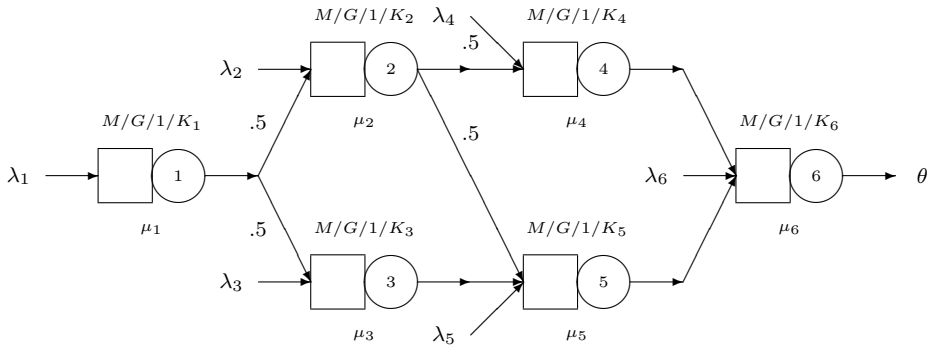


Figura 5.16: Rede com seis nós em topologia mista

de atendimento aproximadamente inalteradas, é necessário um aumento na área de espera de cerca de 22%, para o ganho de somente 0,001% na taxa de atendimento, o que pode ser considerado insignificante. Similarmente, pode haver pontos similares, com respeito às taxas de serviço. De fato, pode ser visto que, para um aumento de 12% na taxa de serviço, o ganho na taxa de atendimento é de cerca de 0,01%.

Tabela 5.1: Soluções eficientes de Pareto selecionadas

CV^2	$\sum_i K_i$	$\Delta\%$	$\sum_i \mu_i$	$\Delta\%$	θ	$\Delta\%$
0,5	18	-	51,0	-	0,9999	-
	22	22%	51,4	0,8%	1,0000	0,01%
	20	-	46,6	-	0,9999	-
	20	0%	52,1	12%	1,0000	0,01%
1,5	14	-	60,4	-	0,9944	-
	19	36%	61,1	1,1%	0,9999	0,6%
	19	-	61,1	-	0,9999	-
	19	0%	104,0	70%	1,0000	0,01%

Note que para $CV^2 = 1{,}5$ tal fenômeno pode ser ainda mais pronunciado. Observa-se que pode ser necessário um aumento de 36% na area espera total

para alcançar-se um aumento de apenas 0,6% na taxa de atendimento. Pode ser mais vantajoso um sistema de filas com uma alocação que produza uma saída correspondente a 99,99% da entrada (isto é, 0,9999/1,000) do que investir 70% a mais em taxa de serviço, para aumentar a saída em 0,01% (ou seja, aumentando-a para 100% da entrada). Estes são apenas alguns exemplos de análises que podem ser feitas em redes de filas finitas gerais por meio da metodologia multiobjetivo.

Capítulo 6

Conclusões e Observações Finais

6.1 Sumário

Nesta tese uma aproximação multiobjetivo é desenvolvida, para maximizar a taxa de atendimento de uma rede de filas geral com um único servidor, simultaneamente com a minimização das suas capacidades totais e taxas de serviço. Pela combinação de um conhecido método aproximado para obtenção das medidas de desempenho da rede de filas finitas, o método de expansão generalizada (GEM), e de um algoritmo evolucionário multiobjetivo (MOEA), o NSGA-II, um eficiente algoritmo foi proposto. Embora a determinação dos valores adequados dos parâmetros do algoritmo descrito, de forma a garantir a sua eficácia e eficiência, seja feita por um procedimento basicamente de tentativa e erro, o ajuste mostrou-se robusto a variações na instância do problema tratada.

Aproximações para curvas de Pareto foram obtidas para várias redes, em configurações e dimensões diversas. Por intermédio das curvas de Pareto obtidas, pode ser melhor identificado como se relacionam as grandezas otimizadas. Assim, foram identificados aqueles pontos a partir dos quais não

é mais compensador aumentar as áreas de espera, ou as taxas de serviço, pois o ganho proporcionado nas taxas de atendimento é insignificante. Estes resultados devem produzir impactos significativos em inúmeras situações de interesse prático.

6.2 Propostas de continuidade

Mais importante que melhorar a compreensão deste importantes sistemas estocásticos, representados pelos modelos de filas de espera configuradas em redes, o trabalho aqui descrito abre inúmeras questões que permanecem em aberto. Há, portanto, diversas direções que futuros esforços de pesquisa nesta área podem tomar. De fato, pesquisas futuras podem ser realizados para avaliar os algoritmos em situações da vida real. Isto inclui, por exemplo, a incerteza nas taxas de chegada $\boldsymbol{\lambda}$ ou o ajuste a dados reais, possivelmente pelo uso de núcleo-estimadores, uma técnica promissora na área de análise de filas, conforme resultados recentemente divulgados [52, 16, 15].

Tópicos para investigações futuras incluem também a extensão da metodologia aqui apresentada para outros tipos de redes de filas finitas. Por exemplo, seria este método extensível para a otimização das taxas de atendimento em redes de filas gerais com servidores *múltiplos*? Ou ainda, seria aplicável a redes de filas com *ciclos*, isto é, com laços de retorno, para representar o retrabalho causado por falhas ou outras causas?

Também interessante seria considerar outras medidas de desempenho diferentes para as redes, tais como o trabalho em processo (WIP, do inglês *work-in-process*), ou o tempo de permanência total no sistema (do inglês *sojourn time*, bastante crítico, por exemplo, em aplicações na área de saúde, [43]), entre outras importantes medidas de desempenho. Estes são apenas alguns tópicos, entre os muitos existentes, para futuras pesquisas nesta área.

Referências Bibliográficas

[1] Ahmed, N. U. & Ouyang, X. H. (2007). Suboptimal RED feedback control for buffered TCP flow dynamics in computer network, *Mathematical Problems in Engineering* **2007**(Article ID 54683): 17 p.

[2] Alves, F. S. Q., Yehia, H. C., Pedrosa, L. A. C., Cruz, F. R. B. & Kerbache, L. (2011). Upper bounds on performance measures of heterogeneous $M/M/c$ queues, *Mathematical Problems in Engineering* **2011**(Article ID 702834): 18 p.

[3] Andriansyah, R., van Woensel, T., Cruz, F. R. B. & Duczmal, L. (2010). Performance optimization of open zero-buffer multi-server queueing networks, *Computers & Operations Research* **37**(8): 1472–1487.

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

[5] Brito, N. L. C., Duarte, A. R. & Cruz, F. R. B. (2012). A multiobjective optimization approach for general finite queueing networks, *Methodology and Computing in Applied Probability* p. 14. (em revisão). **URL:** *http://www.est.ufmg.br/ftp/fcruz/publics/multiop.pdf*

[6] Brito, N. L. C., Duarte, A. R., Ferreira, J. H. & Cruz, F. R. B. (2012). Multiobjective optimization of finite queueing networks, *3rd Internatio-*

nal Conference on Engineering Optimization - EngOpt 2012 [CDROM], COPPE-UFRJ, Rio de Janeiro, RJ, Brazil, pp. 1–10.

[7] Calvete, H. I., Gale, C. & Mateo, P. M. (2008). A new approach for solving linear bilevel problems using genetic algorithms, *European Journal of Operational Research* **188**(1): 14–28.

[8] Carrano, E. G., Soares, L. A. E., Takahashi, R. H. C., Saldanha, R. R. & Neto, O. M. (2006). Electric distribution network multiobjective design using a problem-specific genetic algorithm, *IEEE Transactions on Power Delivery* **21**(2): 995–1005.

[9] Chankong, V. & Haimes, Y. Y. (1983). *Multiobjective Decision Making: Theory and Methodology*, Elsevier, Amsterdam, The Netherlands.

[10] Chaudhuri, K., Kothari, A., Pendavingh, R., Swaminathan, R., Tarjan, R. & Zhou, Y. (2007). Server allocation algorithms for tiered systems, *Algorithmica* **48**(2): 129–146.

[11] Cheah, J.-Y. & Smith, J. M. (1994). Generalized $M/G/C/C$ state dependent queueing models and pedestrian traffic flows, *Queueing Systems* **15**(1): 365–386.

[12] Chen, J., Hu, C. & Ji, Z. (2010). An improved ARED algorithm for congestion control of network transmission, *Mathematical Problems in Engineering* **2010**(Article ID 329035): 14 p.

[13] Coello Coello, C. A. (2000). An updated survey of GA-based multiobjective optimization techniques, *Proceedings of the ACM Computing Surveys*, Vol. 32, pp. 109–143.

[14] Coello Coello, C. A., van Veldhuizen, D. A. & Lamont, G. B. (2002). *Evolutionary Algorithms for Solving Multi-objective Problems*, Kluwer Academic Publishers, New York, NY.

[15] Cruz, F. R. B., Duarte, A. R. & Brito, N. L. C. (2011). Modelagem de chegadas em filas $GI^X/M/c/N$ via núcleo-estimadores, *III Encontro Fluminense de Engenharia de Produção - ENFEPro [Anais]*, ABEPRO, Rio de Janeiro, RJ, Brasil, pp. 1–6.

[16] Cruz, F. R. B., Duarte, A. R., Brito, N. L. C. & Gontijo, G. M. (2011). Arrival process modeling in $GI[X]/M/c/K$ queueing systems, *XLIII Simpósio Brasileiro de Pesquisa Operacional - XLIII SBPO [CDROM]*, SOBRAPO, Ubatuba, SP, Brasil, p. 2813.

[17] Cruz, F. R. B., Duarte, A. R. & van Woensel, T. (2008). Buffer allocation in general single-server queueing network, *Computers & Operations Research* **35**(11): 3581–3598.

[18] Cruz, F. R. B., Kendall, G., While, L., Duarte, A. R. & Brito, N. L. C. (2012). Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers, *Mathematical Problems in Engineering* **2012**(Article ID 348262): 19 p.

[19] Cruz, F. R. B., Oliveira, P. C. & Duczmal, L. (2010). State-dependent stochastic mobility model in mobile communication networks, *Simulation Modelling Practice and Theory* **18**(3): 348–365.

[20] Cruz, F. R. B., Smith, J. M. & Queiroz, D. C. (2005). Service and capacity allocation in $M/G/C/C$ state dependent queueing networks, *Computers & Operations Research* **32**(6): 1545–1563.

[21] Cruz, F. R. B., van Woensel, T. & Smith, J. M. (2010). Buffer and throughput trade-offs in $M/G/1/K$ queueing networks: A bi-criteria approach, *International Journal of Production Economics* **125**(2): 224–234.

[22] Cruz, F. R. B., van Woensel, T., Smith, J. M. & Lieckens, K. (2010). On the system optimum of traffic assignment in $M/G/c/c$ state-dependent queueing networks, *European Journal of Operational Research* **201**(1): 183–193.

[23] de Bruin, A. M., van Rossum, A. C., Visser, M. C. & Koole, G. M. (2007). Modeling the emergency cardiac in-patient flow: An application of queuing theory, *Health Care Management Science* **10**(2): 125–137.

[24] Deb, K. (2001). *Multi-objective Optimisation using Evolutionary Algorithms*, John Wiley & Sons, Inc., New York, NY.

[25] Deb, K. & Agrawal, R. B. (1995). Simulated binary crossover for continuous search space, *Complex Systems* **9**: 115–148.

[26] Deb, K. & Beyer, H.-G. (1999). Self-adaptive genetic algorithms with simulated binary crossover, *Technical report no. CI-61/99*, Department of Computer Science/XI, University of Dortmund, 44221 Dortmund, Germany.

[27] Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* **6**(2): 182–197.

[28] Dimitriou, I. & Langaris, C. (2010). A repairable queueing model with two-phase service, start-up times and retrial customers, *Computers & Operations Research* **37**(7): 1181–1190.

[29] Dong, Z., Chuan, L., Yongxiang, L. & Zhiqi, G. (2013). A system's mean time to repair allocation method based on the time factors, *Quality and Reliability Engineering International* (em impressão).

[30] Duczmal, L. H. (2013). Comunicação pessoal.

[31] Erlang, A. K. (1909). Sandsynlighetsregning og telefonsamtaler, *Nyt tidsskrift for Matematik B* **20**: 33–41.

[32] Fonseca, C. M. & Fleming, P. (1995). An overview of evolutionary algorithms in multiobjective optimization, *Evolutionary Computing* **3**(1): 1–16.

[33] Gontijo, G. M., Atuncar, G. S., Cruz, F. R. B. & Kerbache, L. (2011). Performance evaluation and dimensioning of $GIX/M/c/N$ systems through kernel estimation, *Mathematical Problems in Engineering* **2011**(Article ID 348262): 20 p.

[34] Gross, D., Shortle, J. F., Thompson, J. M. & Harris, C. M. (2009). *Fundamentals of Queueing Theory*, 4th edn, Wiley-Interscience, New York, NY, EUA.

[35] Harris, J. H. & Powell, S. G. (1999). An algorithm for optimal buffer placement in reliable serial lines, *IIE Transactions* **31**: 287–302.

[36] Hu, A. B. & Meerkov, S. M. (2006). Lean buffering in serial production lines with Bernoulli machines, *Mathematical Problems in Engineering* **2006**(Article ID 17105): 24 p.

[37] Hu, X.-B. & Di Paolo, E. (2007). An efficient genetic algorithm with uniform crossover for the multi-objective airport gate assignment problem,

IEEE Congress on Evolutionary Computation, CEC 2007, Singapore, pp. 55–62.

[38] Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of embedded Markov chains, *Annals Mathematical Statistics* **24**: 338–354.

[39] Kerbache, L. & Smith, J. M. (1987). The generalized expansion method for open finite queueing networks, *European Journal of Operational Research* **32**: 448–461.

[40] Kerbache, L. & Smith, J. M. (1988). Asymptotic behavior of the expansion method for open finite queueing networks, *Computers & Operations Research* **15**(2): 157–169.

[41] Kerbache, L. & Smith, J. M. (2000). Multi-objective routing within large scale facilities using open finite queueing networks, *European Journal of Operational Research* **121**(1): 105–123.

[42] Kimura, T. (1996). A transform-free approximation for the finite capacity $M/G/s$ queue, *Operations Research* **44**(6): 984–988.

[43] Koizumi, N., Kuno, E. & Smith, T. E. (2005). Modeling patient flows using a queuing network with blocking, *Health Care Management Science* **8**(1): 49–60.

[44] Labetoulle, J. & Pujolle, G. (1980). Isolation method in a network of queues, *IEEE Transactions on Software Engineering* **SE-6**(4): 373–381.

[45] Li, J., Enginarlar, E. & Meerkov, S. M. (2006). Conservation of filtering in manufacturing systems with unreliable machines and finished

goods buffers, *Mathematical Problems in Engineering* **2006**(Article ID 27328): 12 p.

[46] Lin, F.-T. (2008). Solving the knapsack problem with imprecise weight coefficients using genetic algorithms, *European Journal of Operational Research* **185**(1): 133–145.

[47] Manitz, M. (2008). Queueing-model based analysis of assembly lines with finite buffers and general service times, *Computers & Operations Research* **35**(8): 2520–2536.

[48] Martí, L., García, J., Berlanga, A. & Molina, J. M. (2007). A cumulative evidential stopping criterion for multiobjective optimization evolutionary algorithms, *GECCO '07: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, ACM, New York, NY, EUA, pp. 2835–2842.

[49] Mathieu, R., Pittard, L. & Anandalingam, G. (1994). Genetic algorithms based approach to bi-level linear programming, *Recherche Opérationnelle/Operations Research* **28**(1): 1–21.

[50] Meester, L. E. & Shanthikumar, J. G. (1990). Concavity of the throughput of tandem queueing systems with finite buffer storage space, *Advances in Applied Probability* **22**(3): 764–767.

[51] Menasce, D. A. (2002). QoS issues in web services, *IEEE Internet Computing* **06**(6): 72–75.

[52] Menezes, J. L., Gontijo, G. M., Brito, N. L. C., Duarte, A. R. & Cruz, F. R. B. (2011). Estimação do processo de chegada em filas $GI[X]/M/c/K$ via núleo-estimadores, *X Encontro Mineiro de Estatística - MGEST 2011 [CDROM]*, UFSJ, São João del Rei, MG, Brasil, p. 139.

[53] Montgomery, D. C. (2004). *Design and Analysis of Experiments*, 6th edn, John Wiley & Sons, Inc., New York, NY, EUA.

[54] Osorio, C. & Bierlaire, M. (2009). An analytic finite capacity queueing network model capturing the propagation of congestion and blocking, *European Journal of Operational Research* **196**(3): 996–1007.

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

[56] Shanthikumar, J. G. & Yao, D. D. (1987). Optimal server allocation in a system of multi-server stations, *Management Science* **33**(9): 1173–1180.

[57] Smith, J. M. (2003). $M/G/c/K$ blocking probability models and system performance, *Performance Evaluation* **52**(4): 237–267.

[58] Smith, J. M. (2004). Optimal design and performance modelling of $M/G/1/K$ queueing systems, *Mathematical and Computer Modelling* **39**(9-10): 1049–1081.

[59] Smith, J. M. & Cruz, F. R. B. (2005). The buffer allocation problem for general finite buffer queueing networks, *IIE Transactions* **37**(4): 343–365.

[60] Smith, J. M., Cruz, F. R. B. & van Woensel, T. (2010). Topological network design of general, finite, multi-server queueing networks, *European Journal of Operational Research* **201**(2): 427–441.

[61] Souza, M. J. F. (2013). Comunicação pessoal.

[62] Spieckermann, S., Gutenschwager, K., Heinze, H. & Voß, S. (2000). Simulation-based optimization in the automotive industry – A case study on body shop design, *Simulation* **74**(5): 276–286.

[63] Sywerda, G. (1989). Uniform crossover in genetic algorithms, *Proceedings of the Third International Conference on Genetic algorithms*, Morgan Kaufmann Publishers Inc., San Francisco, CA, EUA, pp. 2–9.

[64] Tang, L., sheng Xi, H., Zhu, J. & qun Yin, B. (2010). Modeling and optimization of $M/G/1$-type queueing networks: An efficient sensitivity analysis approach, *Mathematical Problems in Engineering* **2010**(Article ID 130319): 20 p.

[65] Yang, X.-S. (2010). *Engineering Optimization: An Introduction with Metaheuristic Applications*, Wiley, Hoboken, NJ, EUA.

[66] Youssef, A. M. & ElMaraghy, H. A. (2008). Performance analysis of manufacturing systems composed of modular machines using the universal generating function, *Journal of Manufacturing Systems* **27**(2): 55–69.