



Pós-Graduação em Ciência da Computação

**“Avaliação e Modelagem de Desempenho
para Planejamento de Capacidade do
Sistema de Transferência Eletrônica de
Fundos utilizando Tráfego em Rajada”**

Por

Carlos Julian Menezes Araújo

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE, AGOSTO/2009



Universidade Federal de Pernambuco

CENTRO DE INFORMÁTICA

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Carlos Julian Menezes Araújo

***“Avaliação e Modelagem de Desempenho
para Planejamento de Capacidade do
Sistema de Transferência Eletrônica de
Fundos utilizando Tráfego em Rajada”***

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestrado em Ciência da Computação.

ORIENTADOR: Prof. Dr. Paulo Romero Martins Maciel

RECIFE, AGOSTO/2009

Araújo, Carlos Julian Menezes

Avaliação e modelagem de desempenho para planejamento de capacidade de sistema de transferência eletrônica de fundos utilizando tráfego em rajada / Carlos Julian Menezes Araújo. - Recife: O Autor, 2009.

xxiii, 96 folhas: il., fig., tab.

Dissertação (mestrado) – Universidade Federal de Pernambuco. Cln. Ciência da Computação, 2009.

Inclui bibliografia.

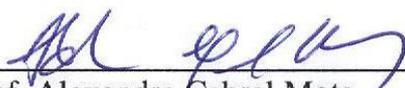
1. Avaliação de desempenho. 2. Redes de Petri – Modelagem de sistema. I. Título.

004.029

CDD (22. ed.)

MEI2009- 120

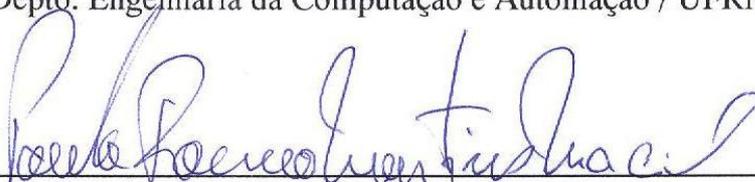
Dissertação de Mestrado apresentada por **Carlos Julian Menezes Araújo** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título “**Avaliação e Modelagem de Desempenho para Planejamento de Capacidade do Sistema de Transferência Eletrônica de Fundos utilizando Tráfego em Rajada**”, orientada pelo **Prof. Paulo Romero Martins Maciel** e aprovada pela Banca Examinadora formada pelos professores:



Prof. Alexandre Cabral Mota
Centro de Informática / UFPE

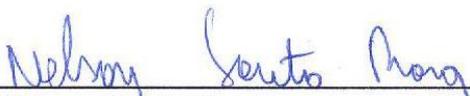


Prof. Luiz Affonso Henderson Guedes de Oliveira
Depto. Engenharia da Computação e Automação / UFRN



Prof. Paulo Romero Martins Maciel
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 25 de agosto de 2009.



Prof. Nelson Souto Rosa

Vice-Coordenador da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

Dedico a Deus, a minha família e aos amigos

AGRADECIMENTOS

Gostaria de registrar meus sinceros agradecimentos como reconhecimento da dedicação de muitas pessoas que, direta ou indiretamente, contribuíram para a realização deste trabalho.

Ao Laboratório de análise de performance CIn-Itautec pelo suporte dado ao desenvolvimento deste trabalho, fornecendo insumos suficientes para a realização prática do trabalho.

Ao meu orientador Prof. Paulo Maciel, pelo apoio, disponibilidade, comentários imprescindíveis, dedicação ao desenvolvimento desta pesquisa e por ter acreditado acima de tudo em mim e no meu trabalho.

Aos professores Alexandre Mota, e Luiz Affonso, pela participação da banca examinadora, contribuindo para a melhoria da qualidade deste trabalho.

Ao meus amigos do IFCE-Cariri, Tadeu Pereira, Jonh Hebert, Haroldo Muniz, Ewer-ton Rodrigues, Woshington Leite, Derig Almeida e ao Prof. Reuber Saraiva que sempre acreditaram no meu potencial.

A todos os colegas da FJN. Dentre eles: Prof^a Aline Dantas, Prof. Dalton Alencar, Prof. Sidney Pinto, Prof. José Marcondes, Prof^a Maria Oderlânia, Isabel Cristina, Djean Carle, Andréia Silva, Rosângela Santos, Elisângela Marques, Gil, Márcia, Erika, Yunara, Disnê e Patrícia Cruz por todo apoio.

Aos meus grandes amigos Nayara Monteiro, Thiago Bandeira, Beth Monteiro, Isaac Pinto, Jorge Luiz, Juliana Gonçalves, Tiago Alencar e Luísa Araújo, que para mim muito representam.

A todos os colegas do CIn. Dentre eles: Ermeson Carneiro, Patrícia Endo, Rafael Amorim, Gustavo Callou, Erica Sousa, Carlos Romerio (*in memoriam*), Gabriel Alves, Eduardo Tavares, Bruno Silva, Rafael Antonelo, Kalil Bispo, Fabrício Cabral, Jalmaratan Macêdo, Sérgio Galdino, Angelo Ribeiro e Fábio Chicout. Meu agradecimento especial a Ermeson, Gustavo, Gabriel, Jalmaratan e a Fabrício pelos momentos de descontração e por todo apoio e ajuda. À Suilan por todo carinho e assistência no inesquecível laboratório de pesquisa C1.

Agradecimento especial a Polyanne Fernandes, Geraldo Neri, Elizabeth Pinheiro, Breno Neri, Tiana e Betânia, por todo o apoio na minha chegada em Recife.

Aos meus pais Vanvan e Corrinha, bem como as minhas irmãs Rafaella e Karyne e ao meu sobrinho Daniel, pois mesmo distante, me deram todo apoio e suporte necessário.

Especialmente a Vanina, pelo companheirismo, carinho, atenção, paciência e amor.

Dividir para conquistar.

—AUTOR DESCONHECIDO

RESUMO

Ao longo dos últimos anos, a confiabilidade e a disponibilidade tornaram-se fatores cruciais para projetar sistemas computacionais, especialmente aqueles tolerantes a falhas. Em tais sistemas, componentes redundantes são adicionados para que um sistema continue a funcionar ininterruptamente, apesar da ocorrência de falhas. Algumas técnicas conhecidas operam com módulos reservas, são eles: *hot standby*, *cold standby* e TMR (*Triple Modular Redundancy*). A tolerância a falhas é alcançada a partir da junção dessas técnicas por meio de um mecanismo de detecção de falha. Geralmente, falhas podem reduzir o desempenho dos sistemas, prolongar o tempo de parada ou levar o sistema a executar uma medida defensiva. Um importante exemplo de sistemas que necessitam de alta confiabilidade, e que ainda não têm atraído muita atenção na literatura, são os sistemas de transferência eletrônica de fundos (TEF). Os sistemas TEF, permitem aos clientes efetuar transações eletrônicas através de cartões de crédito, de débito ou ordem de pagamento. Organizações que provém TEF são pressionadas pelos clientes a oferecer serviços confiáveis, com alta disponibilidade e segurança a preços acessíveis, isso porque, a indisponibilidade de um serviço contratado pode levar a sérios prejuízos econômicos. O crescimento dos pagamentos eletrônicos aumentou a importância para alcançar bons níveis de desempenho nos servidores TEF. Com o aumento dos serviços de pagamento, cresce também o mercado TEF, exigindo cada vez mais que empresas, por exemplo, lojas, bancos ou restaurantes, ofereçam serviços confiáveis, de altas disponibilidade e segurança a custos acessíveis. Dessa forma, este trabalho propõe um modelo em redes de Petri estocástica para avaliar o impacto da disponibilidade e da confiabilidade dos recursos computacionais. Além disso, uma metodologia de avaliação de desempenho é proposta, com o intuito de auxiliar os processos de modelagem e de avaliação. Por fim, estudos de caso serão apresentados mostrando a aplicabilidade do trabalho em comento.

Palavras-chave: Sistema TEF, análise de desempenho e confiabilidade, planejamento de capacidade, modelagem, modelo de falha, GSPN.

ABSTRACT

Over the last years, reliability and availability have become a crucial factor in designing computer systems, especially fault-tolerant ones. In such systems, redundant components are added to the system that continues to function uninterruptedly despite the occurrence of failures. Some well-known techniques to operate spare modules include hot standby, cold standby and TMR (Triple Modular Redundancy). Fault tolerance is achieved as a result of joining these techniques with a fault detection mechanism. Failures may in general reduce the performance of systems, prolong downtime, or cause the system to perform countermeasures. An important example of highly reliable requirements, which have not yet been attracted much attention in the literature, are Electronic Funds Transfer (EFT) systems. EFT systems ensure that customers can carry out transactions through electronic payments via debit cards, credit cards, or checks. Organizations that provide EFT are obliged to offer reliable services such as high availability and security at affordable costs, because non-availability of a contracted service may lead to serious economical consequences. The increasing popularity of electronic payment on the market illustrates the importance of achieving good performance with EFT Servers. Payment services increase with the increment of the EFT market, thus demanding that companies (e.g., stores, banks, or restaurants) offer reliable services, high availability, and security at an affordable cost. This work proposes a stochastic Petri net model to evaluate the impact of availability and reliability issues on computational resources. Furthermore, a performance evaluation methodology is proposed, with the goal of helping the process of modeling and evaluation. Lastly, case studies will be presented showing the applicability of this work.

Keywords: EFT system, performance and reliability analysis, capacity planning, modeling, faults model, GSPN.

SUMÁRIO

Capítulo 1—Introdução	1
1.1 Contexto	1
1.2 Motivação	3
1.3 Trabalhos Relacionados	5
1.4 Objetivos	7
1.5 Estrutura da Dissertação	9
Capítulo 2—Fundamentos	11
2.1 Introdução	11
2.2 Processo Estocástico	12
2.3 Cadeias de Markov	13
2.3.1 Escala de Tempo	14
2.3.2 Propriedades	14
2.4 Medição de desempenho	15
2.5 Dependabilidade	17
2.5.1 Medidas de Funcionamento	18
2.6 Sistemas de Redundância	19
2.7 Redes de Petri	20
2.7.1 Rede de Petri Marcada	24
2.7.2 Transições: Habilitação e Disparo	24
2.7.3 Grafo de Alcançabilidade	25
2.7.4 Redes Elementares	25
2.7.4.1 Seqüência.	26
2.7.4.2 Distribuição.	26
2.7.4.3 Junção.	27
2.7.4.4 Escolha Não-Determinística.	28
2.7.4.5 Atribuição.	28

2.7.4.6	<i>Confusion.</i>	28
2.7.5	Propriedades das Redes de Petri	29
2.7.5.1	Propriedades Comportamentais.	29
2.7.5.2	Propriedades Estruturais	30
2.7.6	Métodos de Análise	31
2.7.6.1	Árvore de Cobertura.	31
2.7.6.2	Redução.	32
2.7.7	<i>Generalized Stochastic Petri Nets</i> (GSPN)	32
2.7.8	Aproximação por Fases	35
2.8	Sistema TEF	37
2.8.1	Pagamento Eletrônico	38
2.8.2	Benefícios do pagamento eletrônico	38
2.8.3	Transferência Eletrônica de Fundos	39
2.8.4	Componentes do Sistema TEF	40
2.9	Considerações Finais	41
Capítulo 3—Metodologia de Avaliação de Desempenho		43
3.1	Introdução	43
3.2	Conhecimento e Levantamento de Informações do Sistema	45
3.3	Seleção das Métricas de Desempenho	46
3.4	Criação do modelo de rede de Petri abstrato	47
3.5	Medição	48
3.6	Tratamento Estatístico dos Dados	48
3.7	Refinamento do Modelo de Redes de Petri	50
3.8	Validação Qualitativa e Quantitativa	51
3.9	Escolha do método de avaliação	51
3.10	Avaliação e Interpretação dos Resultados	52
3.11	Considerações finais	53
Capítulo 4—Modelos		55
4.1	Modelos	55
4.1.1	Loja	55
4.1.2	<i>Buffer</i>	56
4.1.3	Processador TEF	57
4.1.4	Disco TEF	58

4.1.5	Modelos de Redundância	59
4.1.5.1	Modelo Cold standby.	59
4.1.5.2	Modelo Hot standby.	61
4.1.5.3	Modelo TMR.	62
4.2	Considerações Finais	63
Capítulo 5—Estudo de Caso		65
5.1	Introdução	65
5.1.1	Conhecimento e levantamento de informações do sistema	65
5.1.2	Seleção das métricas de desempenho	66
5.2	Criação do modelo de redes de Petri abstrato	67
5.3	Medição	68
5.4	Tratamento Estatístico dos Dados	70
5.5	Refinamento e Análise do Modelo de Redes de Petri	70
5.6	Validação Qualitativa e Quantitativa	71
5.7	Escolha do método de avaliação e interpretação dos resultados	74
5.7.1	Cenário 1: Análise de Desempenho dos Recursos Processador e Disco do Sistema TEF	75
5.7.2	Cenário 2: Avaliação de Desempenho considerando Tráfego em Rajada do Sistema TEF	77
5.7.3	Cenário 3: Avaliação da Disponibilidade e Confiabilidade do Sistema TEF	79
5.8	Considerações Finais	84
Capítulo 6—Conclusão e Trabalhos Futuros		85
6.1	Contribuições, limitações e dificuldades	86
6.2	Trabalhos Futuros	87
Referências		96

LISTA DE FIGURAS

1.1	Número de cartões emitidos em cinco anos no Brasil.	3
2.1	Processo de modelagem simples.	12
2.2	Principais classificações de sistemas.	13
2.3	Curva da banheira.	18
2.4	Mecanismo de redundância modular tripla.	20
2.5	Mecanismo <i>Cold Standby</i>	20
2.6	Elementos de uma rede de Petri.	21
2.7	Exemplo de uma rede de Petri.	22
2.8	Exemplo de uma rede de Petri com peso nos arcos.	22
2.9	Exemplo de redes de Petri.	26
2.10	Exemplo de grafo de alcançabilidade.	26
2.11	Seqüência.	27
2.12	Distribuição.	27
2.13	Junção.	27
2.14	Escolha.	28
2.15	Atribuição.	28
2.16	<i>Confusion</i> simétrico e assimétrico.	29
2.17	Árvore de cobertura.	32
2.18	Redução e refinamento para as redes de Petri.	33
2.19	Elementos GSPN que estendem o comportamento das rede de Petri.	34
2.20	Sub-rede GSPN para representar distribuições polinômio-exponenciais utilizando <i>moment matching</i>	36
2.21	Fluxo dos processos do sistema TEF.	41
3.1	Fluxograma da metodologia de avaliação de desempenho.	44
3.2	Modelo redes de Petri abstrato para um sistema TEF.	47
3.3	Gráfico de caixa para os dados da utilização do processador.	49
3.4	Modelo GSPN refinado para o componente processador do sistema TEF.	50

4.1	Modelo GSPN para a loja.	56
4.2	Modelo GSPN para o <i>buffer</i>	57
4.3	Modelo GSPN para o processador TEF.	57
4.4	Modelo GSPN para o disco TEF.	58
4.5	Modelo GSPN <i>cold standby</i> para o sistema TEF.	60
4.6	Modelo GSPN <i>Hot standby</i> para o sistema TEF.	61
4.7	Modelo GSPN <i>TMR</i> para o sistema TEF.	63
5.1	Fluxo dos processos do sistema TEF.	66
5.2	Modelo redes de Petri abstrato do sistema TEF.	67
5.3	Ambiente de medição.	69
5.4	Modelo redes de Petri refinado do sistema TEF.	72
5.5	Utilização do Processador TEF.	73
5.6	<i>Throughput</i> do Processador TEF.	74
5.7	Utilização do Disco TEF.	74
5.8	<i>Throughput</i> do Disco TEF.	75
5.9	Cenário 1 (Utilização do Processador e Disco TEF).	76
5.10	Cenário 1 (<i>Throughput</i> do Processador e Disco TEF).	77
5.11	Cenário 2 (Utilização do Processador TEF).	78
5.12	Cenário 2 (Utilização do Disco TEF).	79
5.13	Modelo GSPN <i>cold standby</i> para representar o sistema TEF.	80
5.14	Disponibilidade do sistema TEF.	81
5.15	Confiabilidade do sistema TEF.	82
5.16	Resultados de <i>Performability</i> (Processador TEF).	83
5.17	Resultados de <i>Performability</i> (Disco TEF).	84

LISTA DE TABELAS

2.1	Interpretações para os lugares e transições.	23
2.2	Categorias de pagamento TEF.	40
4.1	Métricas do modelo processador TEF.	58
4.2	Métricas do modelo disco TEF.	59
4.3	Métricas do modelo <i>cold standby</i>	61
4.4	Métricas do modelo <i>hot standby</i>	62
4.5	Métricas do modelo <i>TMR</i>	63
5.1	Resumo do ambiente TEF.	66
5.2	Resumo das métricas para o modelo abstrato do sistema TEF.	68
5.3	Configuração do 2 ^o Experimento.	70
5.4	Média e Desvio Padrão.	71
5.5	Parâmetros da distribuição.	71
5.6	Experimento 2.	73
5.7	Resumo dos tempos para cada tpm.	78
5.8	Resumo dos parâmetros de Dependabilidade (horas).	80
5.9	Resultados de disponibilidade do sistema TEF.	81

LISTA DE ABREVIATURAS

ABECS	Associação Brasileira de Empresas de Cartões e Serviços., xxiii
ABRAS	Associação Brasileira de Supermercados., xxiii
CTMC	<i>Continuous Time Markov Chains.</i> , xxiii
DTMC	<i>Discrete Time Markov Chains.</i> , xxiii
GSPN	<i>Generalized Stochastic Petri nets</i> , xxiii
MTBF	<i>Mean Time Between Failure.</i> , xxiii
MTTF	<i>Mean Time to Failure.</i> , xxiii
MTTR	<i>Mean Time to Repair.</i> , xxiii
PDV	Ponto de Venda, xxiii
RBD	<i>Reliability Block Diagram.</i> , xxiii
RdP	Redes de Petri., xxiii
SLA	<i>Service Level Agreement.</i> , xxiii
SPN	<i>Stochastic Petri nets.</i> , xxiii
TEF	Transferência Eletrônica de Fundos, xxiii
TMR	<i>Triple Modular Redundancy.</i> , xxiii
TPM	Transações por minuto, xxiii

CAPÍTULO 1

INTRODUÇÃO

Este capítulo apresenta uma breve explanação acerca do surgimento dos cartões de crédito, destacando a sua evolução e importância no contexto comercial, como também, a necessidade da avaliação de desempenho de sistemas de transferência eletrônica de fundos. Em seguida, são apresentados a motivação, trabalhos relacionados e a proposta deste trabalho. Assim como, seus objetivos.

1.1 CONTEXTO

A moeda, como é conhecida nos tempos modernos, é o resultado de uma longa evolução. Nos primórdios, não havia moeda para a realização de negócios. Para tanto, as antigas civilizações praticavam o escambo ¹, que não possuía equivalência de valor nos negócios. Assim, quem produzisse alguma mercadoria além do necessário, poderia trocar este excesso por outra mercadoria. Esta foi a forma de comércio dominante no início das civilizações, e que pode ser encontrada, entre alguns povos de economia primitiva [Web06].

Com o passar do tempo, as mercadorias tornaram-se inconvenientes nas transações comerciais, devido à oscilação dos seus valores, pelo fato de não serem fracionáveis e também por serem facilmente perecíveis. Com a descoberta do metal, nasce a possibilidade do acúmulo de riquezas. O metal em barras, por exemplo, sob forma de objetos, ou jóias, passou a ser utilizado como meio de comércio. Mas, por falta de aferição de peso e da avaliação do grau de pureza dos objetos, o metal é substituído pelo uso da moeda. Durante muito tempo, a cunhagem de moedas garantiu o pagamento das negociações. Com isso, mais tarde, a moeda evolui e o papel passou a ser utilizado, tendo em consequência, a retratação da cultura do país emissor, em cada cédula confeccionada [GS99].

No final da década de 1920, nos Estados Unidos, a empresa *Western Union* percebeu a necessidade de receber e realizar pagamentos sem utilizar papel moeda. Essa empresa, então, lançou um cartão e distribuiu para seus clientes preferenciais. Com essa nova

¹Troca de bens ou serviços sem intermediação do dinheiro [WeS98].

opção, tornou-se possível adquirir diversos serviços, bens ou benefícios especiais, como o atendimento diferenciado e maiores prazos para quitação das compras. Surgiu, dessa forma, a idéia da utilização de um cartão para a realização de pagamentos [Kla03].

Em 1949, Frank McNamara estava em um jantar de negócios e ao receber a conta percebeu que não tinha como pagá-la. Diante daquela situação, Frank negociou com o dono do restaurante o pagamento da conta para o dia seguinte, conquanto que colocasse sua assinatura na nota das despesas. A partir desse episódio, o executivo Frank McNamara lançou o primeiro cartão moderno denominado *Diners Club Card* [Clu]. Este era aceito em muitos restaurantes, hotéis e agências de locação de veículos nas mais variadas localizações, como, por exemplo, Nova York, Miami, Boston, Chicago, Los Angeles e San Francisco. Nos anos seguintes, outras instituições financeiras começaram a produzir seus próprios cartões [Kla03, C.08, Bid02].

Na América Latina, o Brasil foi um dos primeiros países a disseminar o uso de cartões de crédito. O interesse pelo uso dos cartões de crédito surgiu a partir dos anos 50 com o empresário Hanus Tauber. Este, por sua vez, foi até aos Estados Unidos para adquirir a franquia do *Diners Club* [ABE09]. No final dessa década, o *Diners Club* operava em 93 escritórios distribuídos por todo país, apresentava 185 mil sócios, 25 mil estabelecimentos associados e processava mais de 300 mil comprovantes de compras por mês. O faturamento das transações girava em torno de US\$ 35 milhões mensais.

Ao longo dos anos, por conseguinte, o uso do papel moeda tem sido substituído gradativamente pelo uso dos cartões. Diversos fatores motivam a mudança para o uso do cartão, como o avanço da tecnologia da informação, que combinado com a redução dos seus custos vêm tornando economicamente viável o surgimento e o crescimento dos meios eletrônicos de pagamento. Além disso, vários estabelecimentos comerciais já passaram a aderir esse meio de pagamento por ser prático e seguro para seus clientes.

Adicionalmente, com o intuito de aumentar o desempenho e a confiabilidade no processamento dos sistemas TEF (Transferência Eletrônica de Fundos), os sistemas de pagamento automatizaram a comunicação entre a operadora do cartão e a empresa que oferece a venda. A partir disso, houve um aumento na qualidade do serviço, pois o tempo de espera para o processamento da compra diminuiu de forma significativa. Antes da automatização, o processamento das transações de pagamento acontecia de forma manual, onde a operadora do estabelecimento comercial através de uma linha telefônica, entrava em contato com a operadora de cartão para aprovar ou não a venda.

1.2 MOTIVAÇÃO

Em 2006, segundo um diagnóstico do Sistema de Pagamentos de Varejo do Brasil realizado pelo BACEN (Banco Central do Brasil) mostrou que o dinheiro eletrônico ultrapassou o papel moeda em número de transações. O tipo de dinheiro eletrônico chegou a 1,7 bilhões de operações contra 1,6 bilhões de cheques compensados [BCB09]. Isso porque as transações realizadas com cheques ou papel moeda envolvem custos associados à produção, ao transporte e ao processamento físico dos documentos. Por outro lado, os pagamentos eletrônicos são transacionados por intermédio de instrumentos eletrônicos, fazendo com que os custos associados sejam menores.

De acordo com a ABECS (Associação Brasileira de Empresas de Cartões e Serviços), os negócios realizados com cartões de crédito e débito no Brasil ultrapassaram R\$ 375 bilhões de reais no ano de 2008 [ABE09]. O montante representa 24% a mais do que o atingido no ano de 2007, quando o setor movimentou cerca de R\$ 301 bilhões de reais em transações comerciais. Seguindo essa tendência, o número de cartões no Brasil, incluindo os cartões de crédito, débito e de loja, cresceu entre os anos de 2004 e 2008 a um valor médio de 18,4%. A Figura 1.1 ilustra o expressivo crescimento na emissão de cartões no Brasil.

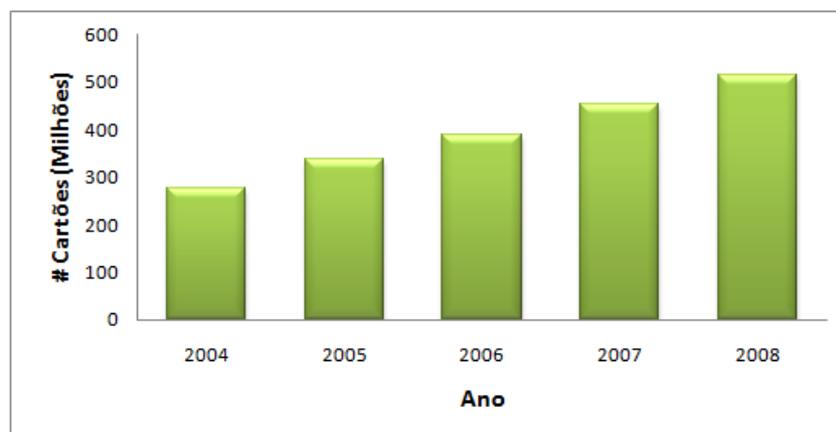


Figura 1.1 Número de cartões emitidos em cinco anos no Brasil.

Em 1999, o número total de estabelecimentos comerciais que aceitavam cartões era pouco superior a 550 mil, e hoje supera o número de 1,4 milhão. O número de pequenos e médios lojistas que passaram a trabalhar com cartões mais que dobrou, e atualmente correspondem a 87% do total de estabelecimentos cadastrados [ABE09]. Tendo em vista este mercado, bancos e administradoras de cartão estão adotando todas as ações possíveis e viáveis no sentido de fortalecer o setor de cartões de crédito e, em escala de permanente

crescimento, atingir os maiores níveis de aquisição e utilização por parte dos usuários consumidores.

Segundo a ABRAS (Associação Brasileira de Supermercados) [ABR09], nesse setor a participação dos pagamentos realizados com o uso de cartões (crédito e débito) cresceu de 47,3%, em 2006, para 54,1%, em 2007. Já a utilização do papel moeda deu um salto de 28,2%, em 2006, para 35%, em 2007. Porém, o crescimento desse recurso foi reflexo do aumento da renda dos brasileiros, visto que desde 2003 o uso do papel moeda vem perdendo espaço para o mercado dos cartões. Apesar dessa expansão, aumentar a utilização por parte dos usuários dos cartões em circulação é o grande desafio para as empresas do setor, uma vez que o índice de ativação dos cartões ainda é baixo.

A utilização exacerbada de cartões também está associado à evolução do volume de financiamento ao consumidor concedido pelos bancos e financeiras. Em 1996, o volume de crédito concedido a pessoas físicas estava na ordem dos R\$ 18 bilhões de reais e representavam 2,9% do PIB (Produto Interno Bruto). Nos últimos anos, esse indicador cresceu substancialmente atingindo os R\$ 155,2 bilhões de reais ao final de 2006, o equivalente a 9,7% do PIB. Em novembro de 2007, o volume de crédito concedido a pessoas físicas alcançou R\$ 239,5 bilhões de reais, o que demonstra que o crescimento do crédito cresce mais do que o crescimento do PIB [BCB09].

Com o crescimento do mercado de cartões aliado ao aumento da complexidade no processamento das transações eletrônicas, surge a necessidade de prover serviços com disponibilidade, confiabilidade, segurança e escalabilidade para alcançar níveis de qualidade. De sorte que, em sistemas TEF (Transferência Eletrônica de Fundos), esses parâmetros são determinantes para o sucesso da empresa diante do atual mercado tão competitivo. Assim, faz-se necessário que se obtenham indicadores de desempenho para o acompanhamento e a análise destes sistemas. Diversas métricas podem ser utilizadas como indicadores de desempenho, como por exemplo, a utilização dos recursos, os tempos de resposta e de fila, o *throughput*, entre outros. Através dessas métricas, é possível compreender o comportamento do sistema e identificar gargalos. Por conseguinte, técnicas de avaliação de desempenho de sistemas computacionais têm sido bastante utilizadas em diversas áreas da computação e das engenharias com o propósito de aumentar o desempenho dessas aplicações e reduzir os custos [Jai91, MA05, MAFM99].

O uso de métodos formais para a análise de sistema de transferência eletrônica de fundos pode fornecer indicadores confiáveis de desempenho. Dentre as diversas abordagens se destacam as redes de Petri, essas que foram propostas por Carl Adam Petri em 1962 na sua tese de doutoramento [Pet62]. Segundo [MBC⁺95], as redes de Petri

permitem a especificação de sistemas concorrentes, assíncronos, distribuídos, paralelos, não-determinísticos e estocásticos. Desde então, as redes de Petri vêm sendo aplicadas nas mais diversas áreas, indo desde a ciência da computação, até as áreas de administração de empresas e às biológicas. Desde seu surgimento, diversas extensões foram propostas para esse formalismo. Entre as extensões existentes, estão as *Generalized Stochastic Petri nets* (GSPN), que incluem a noção de tempo estocástico. Esse tipo de rede permite a avaliação de sistemas a partir de métricas, por exemplo, a probabilidade de utilização de um recurso computacional (processador, disco, memória, entre outros.), ou ainda a disponibilidade e/ou confiabilidade desses recursos. Essas métricas podem ser computadas para um determinado intervalo de tempo (análises transientes), ou quando o sistema entra em equilíbrio (análises estacionárias) [GKZH95]. Além disso, satisfazendo algumas propriedades [Mur89], os resultados das métricas podem ser obtidos de forma analítica, como também por simulação do modelo.

Apesar da grande aplicabilidade e relativa facilidade de utilização de métricas, aplicar GSPN para a modelagem de sistemas, de maneira que o modelo respeite uma série de restrições e para possuir algumas propriedades desejáveis, por exemplo, *liveness* e *deadlock*, pode não ser uma tarefa trivial. Portanto, a geração de modelos que representam as propriedades desejáveis e que permitam a modelagem e análise de diferentes recursos computacionais, pode auxiliar na avaliação de desempenho dos sistemas de transferência eletrônica de fundos.

1.3 TRABALHOS RELACIONADOS

Segundo Jain [Jai91], ao contrário do senso comum, a avaliação de desempenho é uma arte. Como qualquer trabalho artístico, uma avaliação bem sucedida não pode ser produzida mecanicamente. Cada avaliação requer um íntimo conhecimento do sistema a ser modelado e uma cuidadosa seleção da metodologia, da carga de trabalho (*workload*) e do ferramental. Quando o sistema é apresentado pela primeira vez a um analista, a maioria dos problemas são expressos pelo usuário de uma maneira abstrata, como um rascunho. Definir o problema real e convertê-lo para uma forma que possa ser manipulado por ferramentas e técnicas conhecidas, no qual o tempo e outras restrições podem ser satisfeitas, é a principal parte da “arte” do analista.

A utilização de métodos formais para modelagem de desempenho de sistemas provê uma maneira eficiente para abstrair e gerar modelos com o intuito de representar os recursos destes sistemas [Ho92, Jai91, BGdMT06]. Os modelos de desempenho dos sistemas capturam os aspectos mais relevantes do comportamento destes, em termos de como é a

carga de trabalho a que este sistema está submetido, como também da disponibilidade dos recursos. Podem-se classificar os modelos de desempenho em dois tipos: modelos de simulação e modelos analíticos. Os modelos analíticos representam o comportamento do sistema através de um conjunto de formulações matemáticas, cujas soluções oferecem informações sobre as métricas de desempenho estabelecidas. Por outro lado, os modelos de simulação são programas que emulam o comportamento de um sistema simulando a geração e o processamento de eventos [MAFM99].

Entre tais formalismos, Redes de Petri Estocásticas Generalizadas (*Generalized Stochastic Petri Nets* - GSPNs) [MCB84] são amplamente utilizadas para modelar sistemas complexos, usando uma representação de natureza gráfica com suporte a métodos consistentes de análise formal, bem como permitem avaliar o desempenho destes. Desrochers et al. [DAJ95] apresentam uma proposta de modelagem de sistemas automatizados de manufatura através das GSPNs. Neste trabalho, os autores propõem a modelagem de sistemas de manufatura através de blocos básicos de redes de Petri. Dessa forma, a abordagem simplifica a atividade de construção do modelo e ajuda a gerenciar a complexidade da tarefa de modelar grandes sistemas.

Trivedi et al. [THGF96] apresentam métodos gerais, como diagrama de blocos de confiabilidade (RBDs), árvore de falha, grafo de confiabilidade, redes de Petri (SPN, GSPN e SRN) e combinação de outras técnicas adotadas para avaliação de dependabilidade. Para esses métodos, são consideradas dependências, compartilhamento de recursos, cobertura de falhas e modelos de espaço de estados, mas cada um na sua particularidade. Por sua vez, um exemplo de comunicação de redes, representando a relação de cliente/servidor, foi utilizado para ilustrar os métodos. O poder de análise, como também as limitações do modelo utilizado para encontrar a confiabilidade e a disponibilidade dos exemplos, foram apresentados.

Em [GDI09], N. Gharbi propõe uma abordagem para a modelagem e a análise de desempenho de *retrial systems*² com várias classes de clientes e servidores utilizando o formalismo *Colored Generalized Stochastic Petri Nets* (CGSPN). No entanto, esta abordagem proposta está restrita a apenas duas políticas de requisição (*Random Server* e *Fastest Free Server*), além de não oferecer meios para obtenção de métricas de desempenho relacionadas à confiabilidade.

Sousa [SMAC09] et al. sugerem, por conseguinte, uma metodologia para avaliação de desempenho baseada em dois modelos: um para avaliação de desempenho e outro para avaliação de dependabilidade. Essa abordagem analisa o impacto das ocorrências de

²Sistemas que possuem retorno de clientes.

falhas e de atividades de manutenção no desempenho de sistemas TEF. Contudo, a principal limitação desse trabalho é que o modelo de manutenção do sistema TEF não aborda políticas de manutenção no processo de formação, como também a duração do processo de diagnóstico da falha. Outra restrição é a ausência de um planejamento em relação ao tamanho da equipe de manutenção, como também ao período de recuperação. Catalan et al. [CU07] propõem um *framework* de uma máquina virtual baseada num determinado sistema. Neste trabalho, realizou-se uma análise e avaliação da eficiência do TP (*Transaction Processing*) *Monitor*³ que lidam com um grande número de transações *online* antes da execução. Essa abordagem possui limitação na análise de métricas complexas, por exemplo, confiabilidade e disponibilidade.

Kounev [KB06] também propõe uma ferramenta (SimQPN) de simulação de rede de filas para redes de Petri, integrando aspectos de comportamento do *hardware* e do *software* no modelo. A metodologia provê uma análise do padrão *Queueing Petri Net* (QPN) que também pode ser utilizada para planejamento de capacidade do *hardware* e/ou do *software*. No entanto, esse trabalho não fornece meios para representar as QPNs diretamente. Em vez disso, a ferramenta requer que os modelos de simulação sejam descritos usando uma linguagem de simulação complexa que consome tempo e que, além disso, é passível de erros. Outra desvantagem dessa abordagem é que os simuladores geralmente não são tão céleres e eficientes como os simuladores especializados, uma vez que esses, usualmente, não são otimizados para um determinado tipo de modelos.

1.4 OBJETIVOS

Esta dissertação objetiva apresentar modelos formais baseados nas *Generalized Stochastic Petri Nets* (GSPN) que represente os principais componentes que participam da infraestrutura computacional de um sistema TEF, para a partir desses modelos, realizar análises de desempenho, disponibilidade e confiabilidade. Em outras palavras, a partir dos modelos gerados, será possível realizar um planejamento de capacidade dos recursos computacionais, processador e disco, como também, a análise da degradação dos recursos em termos de disponibilidade e confiabilidade.

Esse modelo GSPN, gerado e refinado através do método de aproximação por fases [DAJ95], deverá ser capaz de fornecer métricas importantes para a análise de desempenho do sistema TEF. Exemplos dessas métricas são: o nível de utilização dos recursos (processador e disco), a disponibilidade e a confiabilidade do sistema TEF. Os resultados dessas métricas poderão ser utilizados em um planejamento de capacidade, proporcio-

³*Middleware* designado para executar a transação.

nando, assim, tomadas de decisões quanto à melhoria da infraestrutura.

Objetiva-se, também, propor uma metodologia para auxiliar na modelagem e avaliação de desempenho de sistema TEF. Essa metodologia apresentará um conjunto de modelos GSPN básicos gerados pelo método de aproximação de fase. Além disso, a metodologia será composta por diversas etapas que vão desde a medição das métricas de desempenho, até o refinamento e a análise dos resultados obtidos.

Especificando-se melhor, o trabalho ora posto possui os seguintes objetivos:

- definir uma metodologia que auxilie no processo de avaliação de desempenho de um sistema TEF;
- desenvolver um modelo GSPN para realizar análises de desempenho do sistema TEF;
- desenvolver modelos GSPN redundantes para avaliar os níveis de disponibilidade e confiabilidade do sistema TEF;
- realizar um planejamento de capacidade dos recursos processador e disco, para identificar gargalos no sistema;
- utilizar os resultados obtidos como ponto de apoio para tomada de decisão, melhorando assim o desempenho do sistema TEF.

Estudos de caso serão, pois, realizados com o intuito de validar o modelo proposto. A análise de desempenho através do modelo GSPN possibilitará um estudo detalhado do comportamento em termos de utilização dos recursos, disponibilidade e confiabilidade do sistema TEF. É importante destacar que em uma situação real, seria muito complexo realizar uma análise de desempenho e um planejamento de capacidade, visto que para executar essas atividades seriam necessários altíssimos custos e um tempo delongado.

Uma vez validado o modelo através da comparação entre dados reais medidos e modelados, este será utilizado para modelar um estudo de caso real. Neste trabalho, serão apresentados alguns estudos de caso conduzidos no Laboratório de Análise de Performance CIn/UFPE. Esse laboratório gerencia projetos de pesquisas para planejamento de capacidade de sistema TEF, desenvolvimento de aplicativos voltados para *software* livre, *mobile* e reconhecimento de imagem [Ita09].

Parte do trabalho relacionado ao descrito nesta dissertação, bem como os trabalhos correlatos, podem ser encontrados nas seguintes publicações: [ASM⁺09, SMA09, SMAC09, SAM⁺09, AMC⁺09, MMS⁺09].

1.5 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está organizado como segue. O Capítulo 2 introduz os conceitos fundamentais a serem utilizados na dissertação, tais como: processos estocásticos, cadeias de Markov, medição, sistemas de redundância e redes de Petri. O Capítulo 2.7.8 apresenta uma breve introdução sobre sistemas de transferência eletrônica de fundos. O Capítulo 3 detalha a metodologia proposta para a avaliação de desempenho do sistema TEF. O Capítulo 4 apresenta os modelos de desempenho proposto e descreve seus componentes. O Capítulo 5 apresenta os estudos de caso desenvolvidos no Laboratório de Performance CIn-Itautec. Finalmente, o Capítulo 6 apresenta as considerações finais e trabalhos futuros.

CAPÍTULO 2

FUNDAMENTOS

O capítulo ora exposto apresenta os principais conceitos desta dissertação. Primeiramente, abordam-se processo estocástico, cadeias de Markov e técnicas de medição. Posteriormente, introduzem-se conceitos sobre dependabilidade e os sistemas de redundância. Por fim, abordam-se os principais conceitos que envolvem as Redes de Petri (RdP), como os de propriedades estruturais e comportamentais, métodos de análise e sua aplicabilidade. Além disso, também apresenta-se uma extensão das RdP: as redes de Petri estocásticas generalizadas (GSPN). Por fim, conceitos sobre o sistema de transferência eletrônica de fundos.

2.1 INTRODUÇÃO

O estudo de sistemas envolve uma série de parâmetros de entrada e saída. A esses parâmetros, podem ser associadas variáveis classificadas como contínuas ou discretas. As variáveis contínuas podem assumir qualquer valor no conjunto dos números reais, enquanto as discretas assumem determinado subconjunto de valores, e suas variações são sempre em quantidades pré-definidas como, por exemplo, no conjunto de números inteiros.

Um modelo é uma representação de um ou mais pontos de vista de um sistema em um determinado nível de abstração. Em um modelo formal, um sistema é representado através de um conjunto de variáveis, funções e relações matemáticas que representa uma ou mais perspectivas do sistema. Enquanto um sistema é “algo real”, o modelo é uma “abstração”, como, por exemplo, um conjunto de equações matemáticas. A Figura 2.1 mostra a representação de um sistema através de um modelo.

A classificação apresentada não é exclusiva e tem a finalidade de descrever o escopo dos diferentes aspectos da teoria de sistemas, assim como ajudar a identificar as principais características dos sistemas. A Figura 2.2 fornece uma referência rápida para as classificações de sistemas [CAS99].

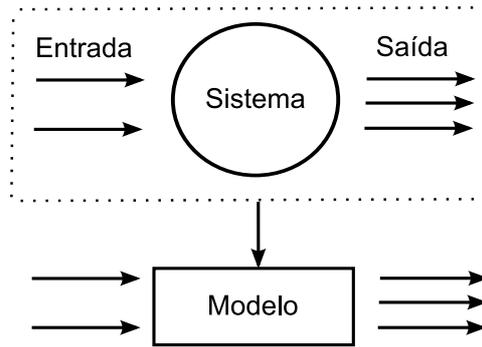


Figura 2.1 Processo de modelagem simples.

Diversos modelos de representação têm sido utilizados para representar sistemas computacionais. Dentre eles, redes de filas, álgebras de processos, autômatos e redes de Petri são os mais utilizados [CAS99].

2.2 PROCESSO ESTOCÁSTICO

A avaliação de desempenho e confiabilidade de sistemas computacionais através de métodos analíticos têm sido bastante utilizados nas áreas de ciência da computação e engenharias [Tri06]. Processos estocásticos são modelos matemáticos úteis para a descrição de fenômenos de natureza probabilística, como uma função de um parâmetro que usualmente tem o significado de tempo.

Um processo estocástico é uma família de variáveis aleatórias $\{X(t), t \in T\}$ definidas em um dado espaço de probabilidade e indexadas por um parâmetro t , em que t varia em um conjunto de índices T . Em um processo estocástico $\{X(t), t \in T\}$, o conjunto T é chamado de índices do processo estocástico. Os valores assumidos por $X(t)$ são chamados de estados e o conjunto de todos os possíveis estados é chamado de espaço de estados E do processo estocástico.

Se o conjunto dos índices T for discreto (enumerável), então o processo é dito ser um processo de parâmetro discreto ou tempo discreto. Outra denominação usual é a seqüência aleatória e é denotada por $\{X_n, n = 1, 2, \dots\}$, por exemplo. Se T for um conjunto contínuo, tem-se um processo de parâmetro contínuo ou tempo contínuo. Se o espaço de estados E for discreto, então o processo é dito ser um processo de estado discreto, também chamado de cadeia. Nesse caso, E é geralmente assumido ser $E = \{0, 1, 2, \dots\}$. Se o espaço de estados E for contínuo, então temos um processo de estado contínuo.

Classificam-se os processos estocásticos de acordo com o tipo da variável aleatória

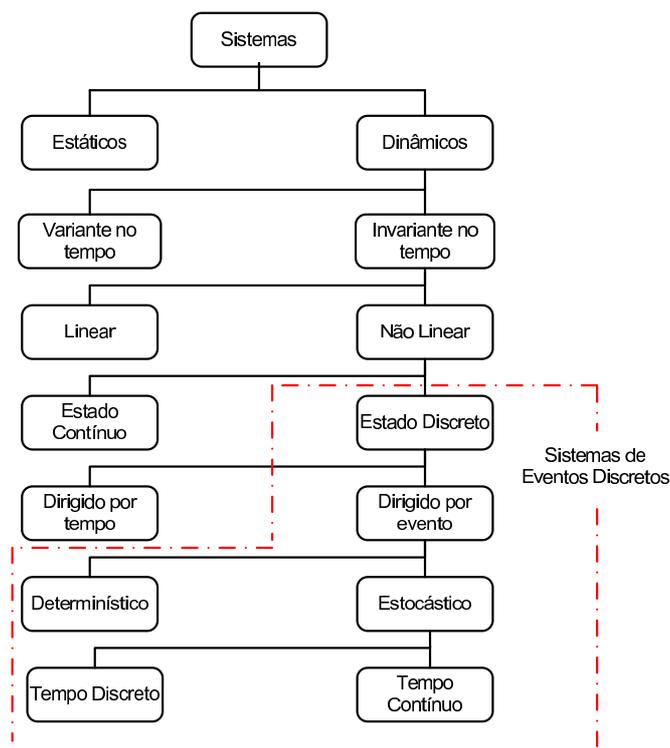


Figura 2.2 Principais classificações de sistemas.

associada ao processo. O processo estocástico de tempo contínuo tem associado uma variável aleatória de tempo contínuo, e o espaço de estados pode ser discreto ou contínuo. Enquanto o processo estocástico de tempo discreto tem associado uma variável aleatória de tempo discreto e, da mesma maneira, o espaço de estado pode ser discreto ou contínuo. Em geral, a descrição probabilística completa de um processo aleatório genérico não é factível. Processos Markovianos são uma classe especial de processo estocástico, para o qual, a descrição probabilística é simples e de relevância particular.

2.3 CADEIAS DE MARKOV

O formalismo de Cadeias de Markov [Ste, Tri06] é um formalismo matemático para a modelagem de sistemas. Com o uso desse formalismo, é possível descrever o funcionamento de um sistema utilizando um conjunto de estados e transições entre esses estados. As transições entre os estados são modeladas por um processo estocástico de tempo contínuo ou discreto definido por distribuições exponenciais ou geométricas, respectivamente.

Um modelo descrito através do formalismo de cadeia de Markov pode ser interpretado como uma máquina de estados, onde os nós (vértice de um grafo) desta representam os estados, e os arcos representam as transições entre os estados do modelo em cadeia de

Markov.

2.3.1 Escala de Tempo

Um modelo descrito pelo formalismo de cadeia de Markov [Ste, Tri06] pode ser classificado de acordo com a sua escala de tempo:

- Cadeias de Markov à escala de Tempo Contínua (CTMC - *Continuous Time Markov Chains*);
- Cadeias de Markov à escala de Tempo Discreta (DTMC - *Discrete Time Markov Chains*);

Os modelos em CTMC diferem basicamente dos modelos em DTMC por suas transições entre os estados poderem ocorrer em qualquer instante de tempo e não em pontos discretos de tempo.

2.3.2 Propriedades

Apresentar-se-ão, a seguir, as propriedades para construção de um modelo descrito pelas cadeias de Markov [Ste].

Os estados do modelo são discretos e enumeráveis. A escala de tempo para a transição entre os estados do modelo pode ser de forma contínua (CTMC) ou discreta (DTMC). A transição entre os estados do modelo depende exclusivamente do estado atual deste, sem importar quais formam os estados prévios ou futuros de tal modelo. A taxa (CTMC) ou probabilidade (DTMC) de transição de estados do modelo dá-se obedecendo a uma lei exponencial ou geométrica, respectivamente.

A representação gráfica de um modelo em Cadeia de Markov é similar a dos autômatos, onde é associado para cada estado do autômato um estado do modelo e para cada transição uma taxa (CTMC) ou probabilidade (DTMC). Um modelo em cadeia de Markov (CTMC) é representado matematicamente por uma matriz de transição de estados. A probabilidade de cada estado em regime estacionário (solução de um modelo em cadeia de Markov) é a solução do sistema da equação linear 2.1.

$$\pi Q = 0, \tag{2.1}$$

onde Q é a matriz de estados e π (vetor de probabilidade) é o autovetor correspondente ao autovalor unitário da matriz de transição. É importante ressaltar que a soma dos elementos do vetor de probabilidade π deve ser igual a 1, ou seja, $\|\pi\| = 1$.

Para os modelos em CTMC, a matriz de transição de estados Q é denominada de gerador infinitesimal, onde cada elemento não diagonal da linha i e coluna j da matriz representa a taxa de transição do estado i para o estado j do modelo. Os elementos diagonais de Q representam o ajuste necessário para que a soma dos elementos de cada linha seja zero. Para os modelos em DTMC, a matriz de transição de estados P é denominada de matriz estocástica, onde cada elemento representa a probabilidade de transição entre os estados do modelo.

2.4 MEDIÇÃO DE DESEMPENHO

A medição de desempenho é usada para compreender as características do sistema em operação, que está sendo construído ou prototipado. Existem alguns propósitos para medição de desempenho [JE06]:

- aplicar técnicas de melhorias *tuning* no sistema implementado;
- realizar melhorias na aplicação;
- validar modelos de desempenho que foram construídos anteriormente; e
- influenciar o projeto dos futuros sistemas a serem implementados.

Essencialmente, o processo envolve:

- entendimento de gargalos no sistema;
- compreensão da aplicação que está em execução no sistema e como este se comporta diante da carga de trabalho; e
- inovação das características do sistema para melhor aproveitar a carga de trabalho.

Segundo Jain [Jai91], a medição de desempenho consiste essencialmente na monitoração do sistema enquanto está sob a ação de uma carga de trabalho. Para adquirir resultados representativos, a carga de trabalho deve ser cuidadosamente selecionada. Essa carga é utilizada nos estudos de desempenho, podendo ser real ou sintética. Embora a carga de trabalho real seja uma boa escolha por representar de forma fiel o sistema, ocasionalmente esta opção não é a desejável. Isso acontece quando o tamanho da carga não é considerável, e também quando esses dados receberam muitas perturbações ou, até mesmo, por questões de acessibilidade destes. Devido a esses motivos, outros tipos de cargas de trabalho têm sido usadas:

- *Kernels*;
- Programas sintéticos;
- *Benchmarks*.

A escolha da carga de trabalho é tão importante quanto a definição de qual estratégia de medição deve seguir [Lil00]. As diferentes estratégias de medição têm em sua base o conceito de evento, onde este é uma mudança no estado do sistema. A definição precisa de um evento depende da métrica que está sendo medida. Por exemplo, um evento pode ser definido como um acesso ao disco, uma referência de memória, uma operação de comunicação de uma rede ou uma mudança interna de um processador. Esses diferentes tipos de métricas podem ser classificados em categorias baseadas no tipo de evento que compreende a métrica:

- Métricas de contagem de evento: são simples, pois apenas contam a quantidade de vezes que um determinado evento ocorre. Exemplos: quantidade de requisições de leitura/escrita de um disco;
- Métricas de evento secundário: registram os valores de alguns parâmetros secundários, sempre que um determinado evento ocorrer. Por exemplo, determinar o número médio de mensagens na fila que são enviadas para um *buffer* de uma porta de comunicação, é necessário registrar o número de mensagens na fila cada vez que uma mensagem foi enviada ou removida da fila. Assim, os eventos a serem monitorados serão o número de mensagens de entrada e saída da fila.
- *Profiles*: é usado para caracterizar o comportamento de um programa ou uma aplicação de um sistema. Normalmente, é usado para identificar onde o programa ou sistema está consumindo mais tempo.

Diferentes estratégias de medição atendem a diferentes tipo de eventos:

- Dirigida a evento: é a estratégia mais simples de medição de evento. Este tipo registra informações necessárias para calcular a métrica de desempenho sempre que um pré-evento ou eventos ocorrem. Uma vantagem dessa estratégia é que o *overhead* gerado ocorre apenas no registro da informação. Se o evento nunca ocorre, ou apenas raramente, a perturbação no sistema será relativamente pequena.

- *Tracing*: é uma estratégia similar à dirigida a evento, exceto porque ao invés de simplesmente gravar os eventos que ocorrem, uma parte do estado do sistema é registrada para identificar o evento. Portanto, é uma estratégia que requer mais armazenamento do que um simples contador de eventos.
- Amostragem: esta estratégia de medição registra em intervalos fixos de tempo a métrica de interesse. Como resultado, uma sobrecarga (*overhead*) pode ocorrer dependendo da frequência em que a medição é executada.
- Indireta: é uma estratégia que deve ser usada quando a métrica desejada não está acessível diretamente. Nesse caso, deve-se encontrar outra métrica que pode ser medida diretamente, a partir da qual se pode deduzir ou obter a métrica de desempenho desejada.

As características de cada estratégia de medição indicarão qual é mais ou menos apropriada para cada situação. Programas de *tracing* podem fornecer informações mais detalhadas sobre o sistema a ser monitorado. A ferramenta de medição do tipo dirigida a evento, por outro lado, normalmente fornece apenas um resumo de alto nível do comportamento do sistema, tais como contagem total ou média de durações. As informações fornecidas tanto por uma ferramenta de medição dirigida a evento quanto pela ferramenta *tracing* são exatas, porém, como o número preciso de tempos é executada numa certa seqüência de vezes, uma determinada subrotina é executada também. Por outro lado, as informações fornecidas pela estratégia de amostragem são de natureza estatística. Assim, repetindo o mesmo experimento com a medição dirigida a evento ou com a ferramenta de *tracing*, estes produzirão os mesmos resultados a cada vez que o experimento é realizado.

2.5 DEPENDABILIDADE

O objetivo de tolerância a falhas é a melhoria da dependabilidade (*dependability*). Esse termo indica a qualidade do serviço fornecido por um dado sistema e a confiança depositada nesse serviço.

Os principais atributos relacionados à dependabilidade [Pra96] são: confiabilidade, disponibilidade, segurança de funcionamento (*safety*), segurança (*security*), manutenibilidade, testabilidade e o comprometimento do desempenho (*performability*). Um resumo dos principais atributos é mostrado a seguir.

- Dependabilidade (*dependability*): qualidade do serviço fornecido por um dado sistema;

- Confiabilidade (*reliability*): probabilidade de um sistema realizar suas funções para as quais foram determinadas sem ou na presença de falhas, durante um intervalo de tempo e de acordo com as condições concebidas;
- Disponibilidade (*availability*): probabilidade do sistema estar operacional. Leva em consideração a alternância de períodos de funcionamento e reparo;
- *Safety*: probabilidade do sistema estar operacional e executar sua função corretamente ou descontinuar suas funções de forma a não provocar dano a outros sistema que dele dependam;
- *Security*: proteção contra falhas maliciosas, visando privacidade, autenticidade e integridade dos dados.

2.5.1 Medidas de Funcionamento

As medidas para avaliação de dependabilidade mais usadas na prática são: taxa de defeitos, MTTF, MTTR, MTBF. Todas essas medidas estão relacionadas à confiabilidade e disponibilidade. A seguir apresentar-se-á uma definição dessas medidas:

- MTTF (*mean time to failure*): tempo médio esperado até a primeira ocorrência de defeito;
- MTTR (*mean time to repair*): tempo médio para o reparo do sistema;
- MTBF (*mean time between failure*): tempo médio entre os defeitos do sistema.

Uma representação usual para a taxa de defeitos de componentes de *hardware* é dada pela curva da banheira. Na Figura 2.3 podem se distinguir três fases:

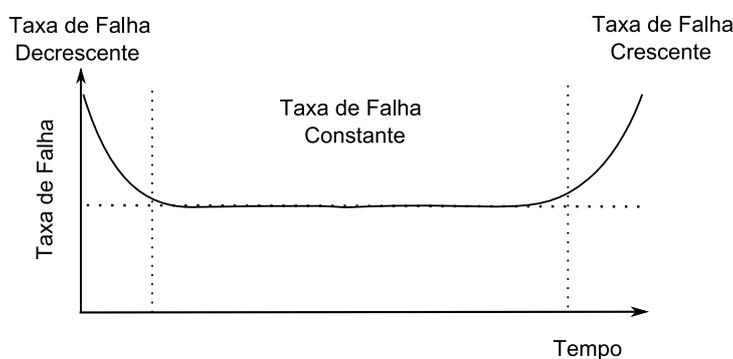


Figura 2.3 Curva da banheira.

Os componentes só apresentam taxa de defeitos constante durante um período de tempo conhecido como a vida útil, que segue uma fase com taxa de defeitos decrescentes, chamada de defeito prematuro. Para acelerar a fase de defeito prematuro, os fabricantes recorrem à técnicas de *burn-in*, onde é efetuada a remoção de componentes fracos pela colocação dos componentes em operação acelerada antes de colocá-los no mercado ou no produto final.

2.6 SISTEMAS DE REDUNDÂNCIA

Redundância é uma abordagem aplicada em vários domínios para aumentar a disponibilidade e confiabilidade dos sistemas. As tradicionais técnicas utilizadas são classificadas em quatro grupos: Prevenção a falhas, Remoção de Falhas, Previsão de Falhas e Tolerância a Falhas [GM02]. Ao contrário de outras técnicas, a tolerância de falha (redundância) visa permitir à entrega correta do serviço até mesmo na presença de falhas. Os recursos adicionais são adicionados para garantir a integridade da entrega do serviço. No entanto, adicionar redundância proporciona aumento nos custos e na complexidade do sistema. Nesse sentido, deve-se fazer um estudo detalhado do sistema abordado para que seja aplicada a técnica de redundância mais adequada.

A técnica de redundância pode ser estática ou dinâmica, e dependendo do tempo, as falhas podem ser detectadas ou mascaradas [ALRL04]. Em um mecanismo redundante estático, ambos os elementos principais (por exemplo, dispositivo, tarefa e serviço) e os redundantes estão permanentemente ativos. Portanto, no caso de falhas nesse mecanismo, os usuários não perceberão que ocorreu uma falha e nem a degradação de desempenho. A redundância modular tripla (*TMR*) (ver Figura 2.4) é o mais usual mecanismo de redundância. Esse mecanismo é composto por três módulos redundantes e um *voter*. O TMR consiste em avaliar o resultado produzido pelos três módulos e decidir a resposta correta através do mecanismo *voter*, desde que apenas um deles esteja em falha. Se um dos módulos ficar defeituoso, os dois restantes mascaram o resultado do módulo defeituoso quando é realizada a votação. A redundância modular N é uma generalização do TMR, no qual, em vez de considerar apenas três estruturas redundantes, N estruturas são consideradas.

O mecanismo de redundância dinâmica (redundância ativa) é caracterizado pelo mecanismo de detecção da falha e por uma ação de recuperação que consome um determinado tempo de ativação. O modelo *cold standby* (ver Figura 2.5) é uma solução que possui um módulo reserva desabilitado [Pra96]. Estes módulos podem ser colocados fora do sistema em boas condições de ambiente, com o propósito de preservar a confiabilidade dos seus

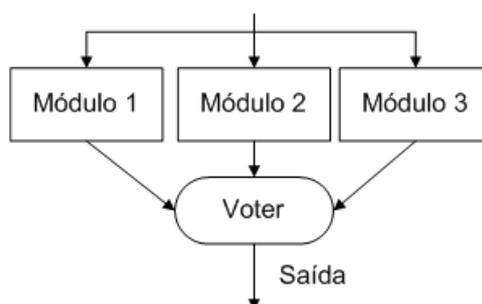


Figura 2.4 Mecanismo de redundância modular tripla.

componentes, uma vez que esses módulos são desabilitados em condições normais, e com a ocorrência de falhas no módulo principal, o módulo reserva deve ser inicializado com a finalidade de manter o sistema disponível. Esta operação pode levar muito tempo, por conseguinte, o desempenho do sistema é afetado. Esta solução pode não ser apropriada para algumas classes de sistemas em que as restrições de tempo são muito restritas.

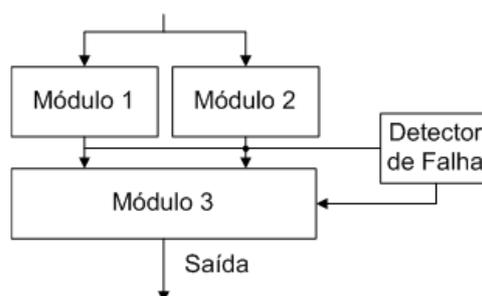


Figura 2.5 Mecanismo *Cold Standby*.

Uma alternativa para a redundância *cold standby* é o mecanismo *hot standby*, que adota a redundância estática; conseqüentemente, em vez de manter o módulo reserva desativado, o mecanismo *hot standby* mantém esse módulo redundante ativado de forma que o defeituoso possa ser substituído sem que necessite de um tempo extra para chavear para o reserva. Além desses mecanismos mencionados, outros métodos têm sido proposto, entre eles está o bloco de recuperação [RX94], *N*-versão programado [Avi85], e assim por diante.

2.7 REDES DE PETRI

O conceito das redes de Petri (RdP) foi inicialmente introduzido por Carl Adam na sua tese de doutoramento intitulada de *Kommunikation mit Automaten* (Comunicação com Autômatos), em 1962 na Universidade de Damstadt, Alemanha [Pet62]. Desde então,

esse formalismo tem sido amplamente utilizado em diferentes áreas, tais como a Ciência da Computação, Engenharia Elétrica, Administração, Química, entre outras.

Diversas variantes do modelo de RdP clássico têm sido desenvolvidas ao longo do tempo, tais como redes temporizadas [MFCH76], estocásticas [Mar89], alto-nível [Jen91] e orientadas a objetos [Jan98]. Isso é devido à necessidade de suprir as diferentes áreas de aplicação, além de prover facilidades de comunicação e transferência de métodos e ferramentas de uma área para outra.

Diversas características fazem das redes de Petri um formalismo atrativo para a modelagem e análise de sistemas concorrentes e distribuídos, dentre elas é possível destacar as seguintes [GV03]:

- redes de Petri são um conjunto de formalismos que tem uma representação gráfica;
- fornecem mecanismos de refinamento e abstração que são de grande importância para o projeto de sistemas complexos;
- existe uma grande variedade de ferramentas computacionais disponíveis para as RdPs, tanto comerciais quanto acadêmicas, para modelagem, análise e verificação;
- têm sido utilizadas em muitas áreas das ciências aplicadas e engenharias. Portanto, vários resultados são encontrados na literatura para os diferentes domínios da sua aplicação;
- existem várias extensões do modelo básico das RdPs, que permitem tanto a representação de características básicas no estudo da concorrência, como também possibilita a análise de problemas práticos das organizações.

Graficamente, as RdP são representadas por lugares (Figura 2.6(a)), transições (Figura 2.6(b)), arcos (Figura 2.6(c)) e *tokens* (Figura 2.6(d)). Uma RdP é um grafo dirigido, onde os lugares e transições são seus vértices, interligados através de arcos dirigidos. Se a origem de um arco for um lugar, seu destino precisa necessariamente ser uma transição, e vice-versa. A distribuição de *tokens* nos lugares da RdP determinam o estado do sistema.

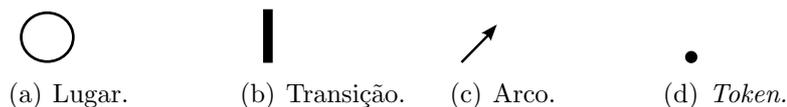


Figura 2.6 Elementos de uma rede de Petri.

Na Figura 2.7, apresenta-se uma RdP, que representa um simples modelo que envia uma transação para um servidor. Os lugares representam um PDV (Ponto de Venda) e

um servidor, enquanto que as transições (*Envia_Transação* ou *Recebe_Transação*) representam os eventos que alteram o estado da transação. Nesse exemplo, o arco dirigido do lugar *PDV* para a transição *Envia_Transação* indica que, para que uma transação seja enviada, é necessário que haja um *token* no lugar *PDV*. De forma similar, o arco dirigido do lugar *servidor* para a transição *Recebe_Transação* indica que, para receber uma transação, necessita-se que haja um *token* no lugar *servidor*. A localização do *token* na rede indicará, se a transação está no PDV (Figura 2.7(a)) ou no servidor (Figura 2.7(b)).

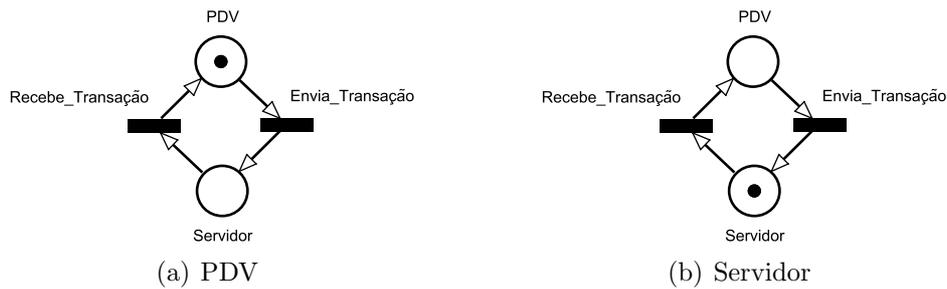


Figura 2.7 Exemplo de uma rede de Petri.

Na representação gráfica, um lugar pode ser conectado a uma transição através de múltiplos arcos (arcos multivalorados) que podem ser compactados em um único arco rotulado. Estes arcos podem ser substituídos por um único arco com um peso associado. Quando uma transição é disparada, ela consome os *tokens* dos lugares de entrada, colocando outros *tokens* nos lugares de saída. A quantidade de *tokens* consumidos e colocados nos lugares de saída é dada pelo peso do arco que conecta os lugares a esta transição. A Figura 2.8 apresenta um exemplo.

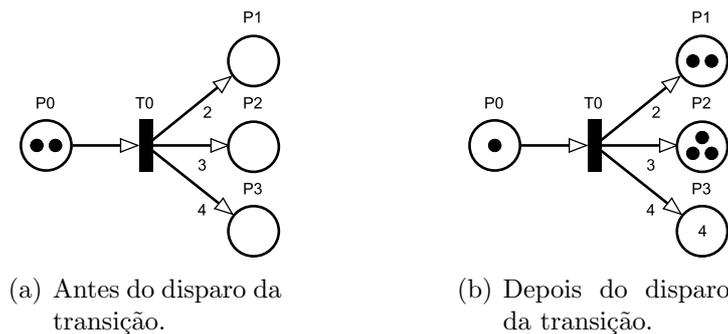


Figura 2.8 Exemplo de uma rede de Petri com peso nos arcos.

Dependendo do sistema modelado, as transições e os lugares de saída e entrada podem ter significados diferentes [Mur89], conforme descritos na Tabela 2.1.

Definição 2.1. (Redes de Petri). Uma rede de Petri (RdP) é uma 5-tupla $PN = \{P, T, I, O, \mu_0\}$, onde:

Tabela 2.1 Interpretações para os lugares e transições.

Lugares de Entrada	Transições	Lugares de Saída
pré-condições	eventos	pós-condições
dados de entrada	passo e computação	dados e saída
sinal de entrada	processamento de sinal	sinal de saída
disponibilidade de recursos	tarefa	liberação de recursos
condição	cláusula lógica	conclusões
<i>buffers</i>	processador	<i>buffers</i>

Fonte: [Mur89, p. 542]

- P é o conjunto de lugares;
- T é o conjunto de transições, $P \cap T = \emptyset$;
- $I, O : T \times P \rightarrow \mathbb{N}$ são funções que denotam os lugares de entrada e saída das transições, respectivamente;
- $\mu_0 : P \rightarrow \mathbb{N}$ é uma função que denota a marcação inicial dos lugares da rede.

As funções I e O descrevem, respectivamente, os arcos de entrada e saída de uma dada transição $t \in T$ para um lugar $p \in P$. Assim, para se obter o peso do arco que conecta p a t , utiliza-se a notação $I(t, p)$. Essa notação também pode ser utilizada para a função de arcos de saída O . É usual representar estas funções através de uma notação matricial. Nesta notação, I é denominada matriz de entrada e O é a matriz de saída.

Existem outras formas de representar os elementos de uma rede de Petri. Por exemplo, o conjunto de lugares de entrada e de saída de uma determinada transição pode ser definido conforme apresenta a Definição 2.2. Similarmente, o conjunto de transições de entrada e de saída de um determinado lugar pode ser definido conforme apresenta a Definição 2.3.

Definição 2.2. (Lugares de Entrada e de Saída) Os conjuntos de lugares de entrada e de saída de uma transição $t_i \in T$ podem ser representados da seguinte forma:

$$\bullet t = \{p \in P \mid I(p, t) > 0\}$$

$$t \bullet = \{p \in P \mid O(t, p) > 0\}$$

Definição 2.3. (Transições de Entrada e de Saída) Os conjuntos de transições de entrada e de saída de um lugar $p_i \in P$ podem ser representados da seguinte forma:

$$\bullet p_i = \{t \in T \mid O(t, p) > 0\}$$

$$p_i \bullet = \{t \in T \mid I(p, t) > 0\}$$

2.7.1 Rede de Petri Marcada

Uma marca (*token*) é um conceito primitivo em redes de Petri, tal qual lugar e transição. As marcas são informações atribuídas aos lugares. Uma marcação associa um k (Número Natural (\mathbb{N})) a cada lugar da rede. Abaixo são apresentadas as seguintes definições formais: marcação, vetor de marcação e rede de Petri marcada, respectivamente, nas Definições 2.4, 2.5 e 2.6.

Definição 2.4. (Marcação). Seja P o conjunto de lugares de uma RdP. Define-se formalmente marcação como uma função que mapeia o conjunto de lugares P a inteiros não-negativos $M : P \rightarrow \mathbb{N}$.

Definição 2.5. (Vetor Marcação) Seja P o conjunto de lugares de uma RdP. A marcação pode ser definida formalmente como um vetor $M = (M(p_1), \dots, M(p_n))$, onde $n = \#P$, para todo $p_i \in P$, tal que $M(p_i) \in \mathbb{N}$.

Definição 2.6. (Rede Marcada) Define-se uma RdP marcada pela dupla $RM(R; M_0)$, onde R é a estrutura da rede e M_0 é a marcação inicial.

2.7.2 Transições: Habilitação e Disparo

De forma geral, o comportamento dos mais diversos sistemas pode ser descrito em termo dos seus possíveis estados internos e processos de mudança desses estados. Para simular o comportamento dinâmico de um sistema, a marcação em uma rede de Petri é modificada de acordo com as regras descritas a seguir:

- uma transição $t \in T$ é habilitada, se cada lugar de entrada p da transição t ($\bullet t$) é marcado com no mínimo $w(p, t)$ *tokens*;
- uma transição habilitada pode ou não disparar. O disparo da transição depende da RdP que está sendo considerada. Em outras palavras, a transição pode estar associada a uma guarda temporal (redes temporais) ou a um predicado lógico, que indicará quando um evento associado deve ocorrer;
- o disparo de uma transição habilitada t remove $w(p, t)$ *tokens* dos lugares de entrada p da transição $t(\bullet)$, e adiciona $w(t, p)$ *tokens* nos lugares de saída p da transição $t(\bullet)$.

Definição 2.7. (Habilitação de Transição) Seja $PN = (P, T, F, W, M_0)$ uma rede, $t \in T$ uma transição e M_k uma marcação. Se $M_k[t >, M_k(p_i) \geq I(p_i, t), \forall p_i \in P$.

Definição 2.8. (Regras de Disparo de Transições) Seja $PN = (P, T, F, W, M_0)$ uma rede, $t \in T$ uma transição e uma marcação M_k . A transição t pode disparar quando ela estiver habilitada. Disparando uma transição habilitada, a marcação resultante é $M_i = M_k - I(p_j, t) + O(p_j, t), \forall p_i \in P$. Se uma marcação M_i é alcançada por M_k pelo disparo de uma transição t_i , ela é denotada por $M_k[t > M$.

Se uma transição t_1 está habilitada para uma marcação M e uma segunda transição t_2 está habilitada para a marcação M_1 , obtida após o disparo de t_1 , é dito que a seqüência $sq = t_1; t_2$ está habilitada para M . Em outras palavras, se $M[t_1 > M_1$ e $M_1[t_2 > M_2$ então $M[t_1; t_2 > M_2$. Portanto, é designado o disparo de uma seqüência $sq \in T$ por $M[sq > M'$.

Definição 2.9. (Seqüência Disparáveis) A seqüência sq está habilitada, possibilitando a obtenção de uma marcação M'' , ($M[sq > M''$) se, e somente se, ocorrem os casos abaixo:

- $sq = \lambda$, onde λ é uma seqüência vazia, tal que $M'' = M$;
- $sq = sq't$, onde sq é uma seqüência de transições $t \in T$ e existe M' tal que $M[sq' > M'$ e $M'[t > M''$.

Se $M[sq > M''$, diz-se que sq é uma seqüência disparável para M .

2.7.3 Grafo de Alcançabilidade

Geralmente um grafo rotulado e direcionado é adotado para representar todas as possíveis marcações que a rede de Petri pode alcançar. Este grafo é geralmente chamado de grafo de alcançabilidade.

Definição 2.10. (Grafo de Alcançabilidade) Podemos definir um grafo de alcançabilidade como sendo uma tupla (V, E) , onde V representa o conjunto de vértices representadas pelas marcações possíveis, e E é o conjunto de arestas rotuladas.

Como exemplo, considere $M = \{m_0 = |1, 0, 0|, m_1 = |0, 1, 0|, m_2 = |0, 0, 1|\}$ sendo o conjunto das marcações alcançáveis da rede de Petri representada na Figura 2.9. O respectivo grafo de alcançabilidade é representado na Figura 2.10.

2.7.4 Redes Elementares

As redes elementares são blocos básicos que permitem a modelagem de sistemas mais complexos. Nesta seção, são apresentadas algumas das redes elementares, tais como seqüência, distribuição, junção, escolha não-determinística, atribuição e confusão.

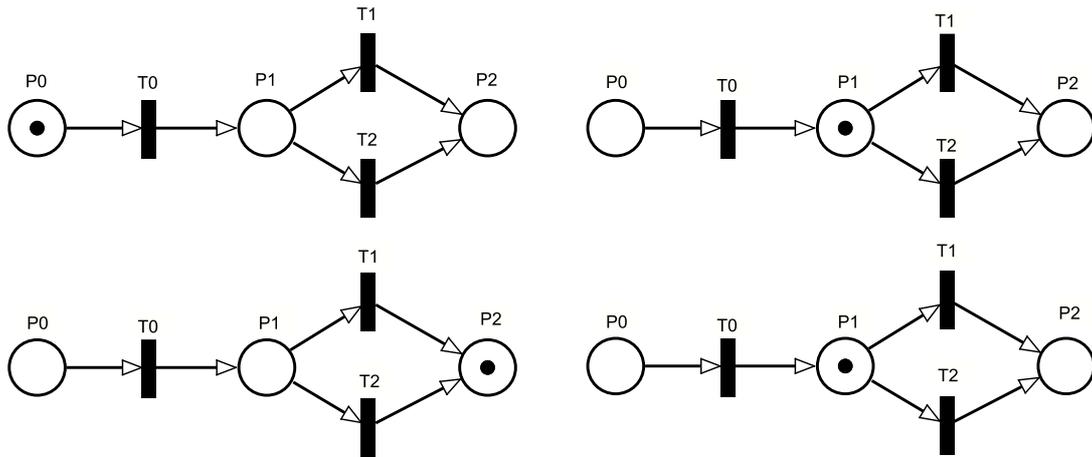


Figura 2.9 Exemplo de redes de Petri.

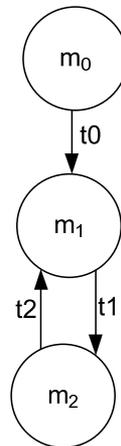


Figura 2.10 Exemplo de grafo de alcançabilidade.

2.7.4.1 Seqüência. A seqüência é uma rede que representa ações consecutivas, desde que uma dada condição seja satisfeita. Em outras palavras, após a execução de cada ação, uma nova condição poderá ser disparada, permitindo, assim, a execução de uma nova ação. A Figura 2.11 apresenta um exemplo dessa rede, onde um *token* no lugar p_0 habilita a transição t_0 , e com o disparo dessa transição, uma nova condição é estabelecida (p_1 é marcado). Essa nova condição pode permitir o disparo de uma nova condição associada ao lugar p_1 .

2.7.4.2 Distribuição. Esta rede permite a criação de processos paralelos a partir de um processo hierarquicamente superior. Como apresentado na Figura 2.12, o disparo da transição t_0 adiciona um *token* no lugar p_1 e outro no lugar p_2 . Essas novas condições (p_1 e p_2) permitem a execução de novos processos paralelos.

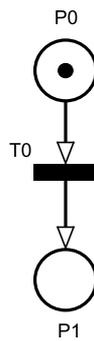


Figura 2.11 Seqüência.

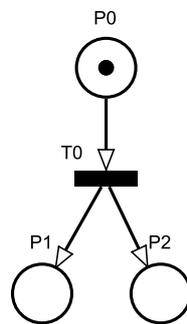


Figura 2.12 Distribuição.

2.7.4.3 Junção. Esta rede modela a sincronização de processos paralelos (ver Figura 2.13). Ela combina duas ou mais redes, permitindo que outro processo continue sua execução somente após o término de todos os processos paralelos que o antecedem. Como demonstrado na Figura 2.13, a transição t_0 estará habilitada, se ambas as pré-condições contiverem *tokens* (p_0 e p_1). Se essa condição for satisfeita, então a transição t_0 poderá ser disparada, retirando um *token* dos lugares p_0 e p_1 e colocando em p_2 .

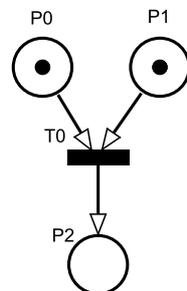


Figura 2.13 Junção.

2.7.4.4 Escolha Não-Determinística. Nesta subseção, é apresentada uma rede elementar que pode ser denominada de conflito, escolha ou decisão, dependendo da aplicação. A Figura 2.14 representa a escolha não determinística, onde o disparo de uma transição desabilita o disparo de uma outra transição. Havendo um *token* em p_0 , t_0 e t_1 , torna-se conflitante, isto é, o disparo de uma transição elimina a possibilidade da outra.

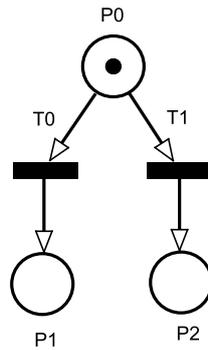


Figura 2.14 Escolha.

2.7.4.5 Atribuição. Atribuição é uma rede elementar que permite que dois ou mais processos habilitem um terceiro processo. Na Figura 2.15, tanto a transição t_0 quanto a transição t_1 são independentes, porém ambas têm um lugar de saída em comum. Portanto, após o disparo de qualquer uma dessas transições, cria-se uma condição (p_2 é marcado) que possibilita o disparo de uma outra transição.

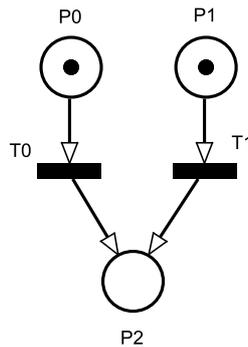


Figura 2.15 Atribuição.

2.7.4.6 Confusion. Este padrão de rede modela a situação onde dois eventos estão ao mesmo tempo em conflito e em concorrência. A Figura 2.16 apresenta dois tipos de *confusion*: (a) simétrico e (b) assimétrico. No modelo da Figura 2.16 (a), as transições t_0

e t_2 são concorrentes, e cada uma dessas transições está em conflito efetivo com t_1 , pois o disparo de t_1 impossibilita o disparo de t_0 e t_2 . Por outro lado, na Figura 2.16 (b), t_0 e t_2 são concorrentes. No entanto, se t_2 disparar antes de t_0 e t_1 , haverá conflito efetivo.

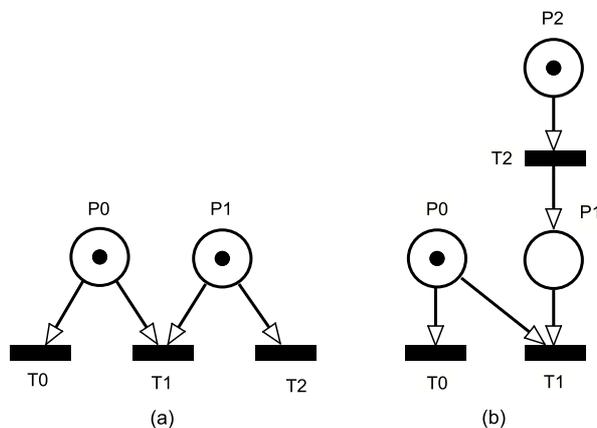


Figura 2.16 Confusão simétrico e assimétrico.

2.7.5 Propriedades das Redes de Petri

Diversas propriedades podem ser obtidas a partir dos modelos, permitindo assim revelar as mais diversas características do sistema. Essas propriedades podem ser subdivididas em comportamentais e estruturais, as quais são descritas nas seções seguintes.

2.7.5.1 Propriedades Comportamentais. Subscvem-se aqui as principais propriedades comportamentais baseadas em [Mur89, MLC96]. As propriedades comportamentais são aquelas que dependem da marcação.

Alcançabilidade

A propriedade de alcançabilidade indica a possibilidade de atingir uma determinada marcação pelo disparo de um número finito de transições a partir de uma dada marcação inicial. Uma marcação M_0 é dita alcançável a partir de M_i , se existir uma seqüência de disparo que transforme M_0 em M_i . A seqüência de disparo é denotada pelo conjunto $\sigma = \{t_1, t_2, \dots, t_n\}$. Nesse caso, M_i é alcançável a partir de M_0 por σ . Onde σ é formalmente descrito por $M_0[\sigma > M_i$.

Limitação e *Safeness*

Uma rede é dita *k-limitada*, se todos os seus lugares forem limitados, ou seja, o número de

tokens em cada lugar não deve ultrapassar um número finito k , para qualquer marcação alcançável a partir de M_0 . Uma rede de Petri é dita *safeness*, se $k = 1$.

Liveness

Uma rede é dita *live* se não importam quais marcações sejam alcançáveis a partir de um marcação inicial m_0 , se for possível disparar qualquer transição através do disparo de alguma seqüência de transições $L(M_0)$. O conceito de *deadlock* está fortemente conectado ao conceito de *liveness*. No entanto, o fato de um sistema ser livre de *deadlock* não resulta que este seja *liveness*. Contudo, um sistema *liveness* implica em um sistema livre de *deadlocks*. A análise de *liveness* de uma rede permite verificar se os eventos modelados efetivamente ocorrem durante o funcionamento do sistema, ou se foram definidos eventos mortos no modelo [Mur89].

Cobertura

A propriedade de cobertura está fortemente conectada ao conceito de alcançabilidade e *liveness*, apresentados nesta seção, respectivamente. Quando se deseja saber se alguma marcação M_i pode ser obtida a partir de uma marcação M_j , tem-se o problema denominado cobertura de uma marcação. Uma marcação M_i é dita coberta se existe uma marcação M_j tal que $M_j \geq M_i$. Fora isso, em alguns sistemas, deseja-se apenas observar o comportamento de determinados lugares. Para isso, restringe-se a pesquisa a apenas um conjunto de lugares de particular interesse (cobertura de submarcações).

Reversibilidade e Home State

Uma rede é dita reversível se, para cada marcação M em $R(M_0)$, M_0 é alcançável a partir de M . Assim, a rede possui a capacidade de retornar à marcação inicial. Além disso, em algumas aplicações não é necessário voltar à marcação inicial, mas sim a uma marcação específica. Essa marcação específica é denominada *Home State*.

2.7.5.2 Propriedades Estruturais As propriedades estruturais são aquelas que não dependem da marcação, ou seja, possuem dependência exclusivamente da topologia da rede. Abaixo serão descritas as principais propriedades estruturais baseadas em [Mur89, MLC96].

- **Limitação Estrutural.** Uma rede é dita limitada estrutural, se o número de *tokens* é limitado para qualquer marcação inicial.

- **Conservação.** A conservação é uma importante propriedade das RdP, pois permite a verificação da não destruição de recursos através da conservação de *tokens*.
- **Repetitividade.** Uma rede é considerada repetitiva, se para uma marcação e uma seqüência de transições disparáveis, todas as transições dessa rede são disparadas ilimitadamente.
- **Consistência.** Uma rede é dita consistente, se dada uma seqüência de transições disparáveis a partir de uma marcação inicial M_0 , ele retorna a M_0 , porém todas as transições da rede são disparadas pelo menos uma vez.

2.7.6 Métodos de Análise

Os métodos de análise das RdP podem ser classificados em três grupos: análise baseada na árvore de cobertura, os métodos baseados na equação fundamental das RdP e na técnica complementar. Nesta dissertação, são introduzidos os seguintes grupos: análise baseada na árvore de cobertura e as técnicas de redução.

2.7.6.1 Árvore de Cobertura. O método de análise denominado árvore de cobertura baseia-se nas construções de uma árvore que possibilite a representação de todas as possíveis marcações de uma rede [MLC96].

Para uma dada rede de Petri, com uma marcação inicial, é possível obter diversas marcações para um grande número de transições potencialmente habilitadas. Essas marcações podem ser representadas por uma árvore, onde os nós são as marcações e os arcos as transições disparadas. Para uma rede de Petri ilimitada, a árvore de cobertura é denominada árvore de alcançabilidade, dado que esta contém todas as possíveis marcações da rede. A árvore de cobertura é um gráfico utilizado para representar finitamente um número infinito de marcações. Para possibilitar a representação finita das marcações, através da árvore, é utilizado o símbolo ω . A Figura 2.17 (b) mostra a árvore de cobertura para o exemplo da Figura 2.17 (a).

Definição 2.11. (Árvore de Cobertura) Seja $RM = (P, T, F, W, M_0)$ uma rede de Petri marcada. Define-se árvore de cobertura pelo par $AC = (S, A)$, onde S representa as marcações e A os arcos rotulados por $t_j \in T$.

Algumas propriedades podem ser analisadas usando árvore de cobertura, tais como *limitação*, *segurança*, *deadlock* e *alcançabilidade*. A identificação dessas propriedades é realizada da seguinte forma:

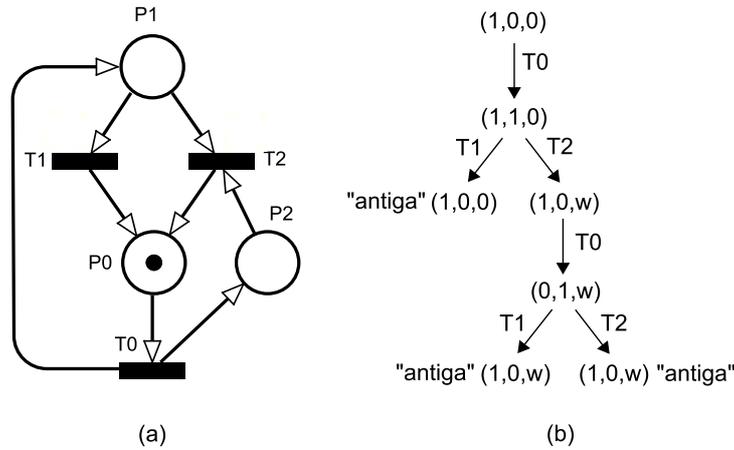


Figura 2.17 Árvore de cobertura.

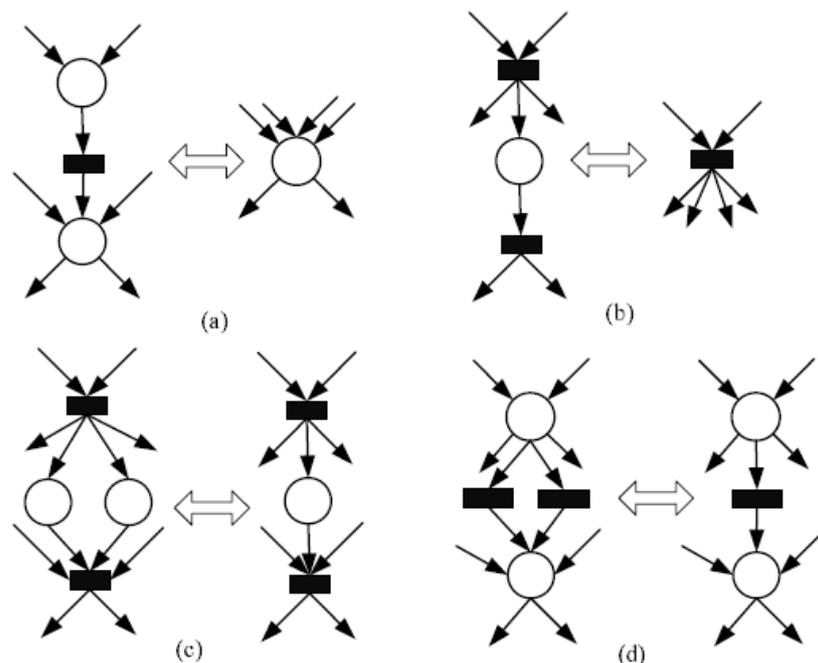
- uma rede (N, M_0) é ilimitada, se ω não aparecer na árvore de cobertura;
- uma rede (N, M_0) é segura, se apenas 0's e 1's aparecem na árvore de cobertura;
- uma transição t é morta, se ela não aparecer na árvore de cobertura;
- se uma marcação M é alcançável a partir de M_0 , então existe o nó M' , tal que $M \leq M'$.

O principal problema desse método consiste na alta complexidade computacional, mesmo quando se usam técnicas de redução das RdP.

2.7.6.2 Redução. Reduções das redes de Petri são transformações aplicadas ao modelo de um sistema com o objetivo de simplificá-lo, preservando as propriedades do sistema a ser analisado. Normalmente, essa técnica é utilizada para facilitar a análise de sistemas complexos. Por outro lado, é possível transformar um modelo abstrato em um modelo refinado, mantendo suas propriedades [Mur89]. Existem várias técnicas de transformação para redes de Petri. A Figura 2.18 apresenta algumas das possíveis transformações.

2.7.7 Generalized Stochastic Petri Nets (GSPN)

A Definição 2.12 apresenta as GSPNs. Essa definição foi criada com base nos trabalhos de [CM00, MBC⁺95] visando incorporar na própria definição os conceitos de prioridade, peso, arcos inibidores, transições estocásticas, semântica de servidor, guardas e pesos dos arcos dependentes da marcação.



Fonte: [Mur89, p. 553]

Figura 2.18 Redução e refinamento para as redes de Petri.

Definição 2.12. (*Generalized Stochastic Petri Nets*). Uma *Generalized Stochastic Petri Net* (GSPN) é uma 10-tupla $N = \{P, T, \pi, I, O, H, g, ss, \mu_0, W\}$, onde:

- P é o conjunto de lugares;
- T é conjunto de transições, $P \cap T = \emptyset$;
- $\pi : T \rightarrow \mathbb{N}$ é a função de prioridade, onde:

$$\pi(t) = \begin{cases} \geq 1, & \text{se } t \text{ é uma transição imediata;} \\ 0, & \text{caso contrário.} \end{cases}$$
- $I, O, H : T \times P \times \mathbb{N}^{|P|} \rightarrow \mathbb{N}$ são funções dependentes da marcação que denotam os lugares de entrada, saída e de inibição das transições, respectivamente;
- $g : T \times \mathbb{N}^{|P|} \rightarrow \{true, false\}$ é a função de guarda das transições;
- $ss : T \rightarrow \mathbb{N}^* \cup \{\infty\}$ é a função que determina a semântica de servidor da transição;
- $\mu_0 : P \rightarrow \mathbb{N}$ é a função que denota a marcação inicial dos lugares da rede;
- $W : T \rightarrow \mathbb{R}^+$ é a função de pesos, que representa os pesos (w_t) das transições imediatas e as taxas λ_t das transições estocásticas, onde:

$$W(t) = \begin{cases} w_t \geq 0, & \text{se } t \text{ for uma transi\~{c}o\~{e} imediata;} \\ \lambda_t > 0, & \text{caso contr\~{a}rio.} \end{cases}$$

Nas GSPNs podem existir transi\~{c}o\~{e}s imediatas e estoc\~{a}sticas. As transi\~{c}o\~{e}s imediatas possuem tempo de retardo associado igual \~{a} zero, enquanto que \~{a}s transi\~{c}o\~{e}s estoc\~{a}sticas possuem um tempo que representa uma distribui\~{c}o\~{e} exponencial. Os arcos inibidores t\~{e}m o objetivo de de inibir o disparo das transi\~{c}o\~{e}s da rede. De maneira semelhante aos arcos tradicionais, estes arcos tamb\~{e}m podem ter um peso associado. Graficamente, as transi\~{c}o\~{e}s estoc\~{a}sticas e os arcos inibidores costumam ser representados conforme mostra a Figura 2.19.

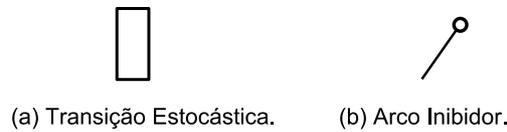


Figura 2.19 Elementos GSPN que estendem o comportamento das rede de Petri.

A sem\~{a}ntica de servidor indica a quantidade m\~{a}xima de disparos paralelos que uma transi\~{c}o\~{e} pode realizar [BGdMT06]. Se $ss(t)$ for igual a 1, k ou ∞ , onde $1 < k < \infty$, a sem\~{a}ntica ser\~{a} *single server*, *k-server* ou *infinite server*, respectivamente.

As prioridades e os pesos s\~{a}o atributos associados \~{a}s transi\~{c}o\~{e}s. Em uma GSPN, a prioridade das transi\~{c}o\~{e}s estoc\~{a}sticas ser\~{a} sempre zero, que \~{e} a menor prioridade poss\~{i}vel, enquanto a das transi\~{c}o\~{e}s imediatas ser\~{a} sempre maior que zero. Assim, caso a GSPN alcance uma marca\~{c}o\~{e} capaz de habilitar tanto uma transi\~{c}o\~{e} imediata quanto uma transi\~{c}o\~{e} estoc\~{a}stica, a transi\~{c}o\~{e} imediata sempre ser\~{a} disparada primeiro. De maneira an\~{a}loga, se em uma dada marca\~{c}o\~{e} existem duas transi\~{c}o\~{e}s imediatas habilitadas, a transi\~{c}o\~{e} com prioridade mais alta sempre ser\~{a} disparada primeiro.

No entanto, se o conflito ocorre entre duas transi\~{c}o\~{e}s imediatas que possuam uma mesma prioridade, este conflito ser\~{a} resolvido com base em seus pesos. Os pesos s\~{a}o utilizados para resolver o conflito de maneira probabil\~{i}stica. Se, por exemplo, em uma dada marca\~{c}o\~{e} de uma RdP, ocorre um conflito entre as transi\~{c}o\~{e}s t_1 e t_2 e $\pi(t_1) = \pi(t_2)$, ent\~{a}o para resolver este conflito probabilisticamente atrav\~{e}s de seus pesos. Assim, se $w_{t_1} = 4$ e $w_{t_2} = 6$, e estas transi\~{c}o\~{e}s s\~{a}o as \~{u}nicas habilitadas, a probabilidade de ocorrer o disparo de t_1 ou t_2 \~{e} dada, respectivamente, por $4/(4+6)$ e $6/(4+6)$.

As defini\~{c}o\~{e}s das fun\~{c}o\~{e}s de arcos I , O e H forma estendidas nas GSPNs, para contemplar a depend\~{e}ncia da marca\~{c}o\~{e}. Assim, os pesos dos arcos podem ser definidos como um inteiro positivo constante para qualquer marca\~{c}o\~{e}, ou ainda como uma express\~{a}o dependente de marca\~{c}o\~{e} da rede, de forma que a avalia\~{c}o\~{e} desta express\~{a}o resulte em um

número natural. Assim, para uma marcação $\mu \in RS$, denota-se o peso de um arco de entrada de uma transição $t \in T$ a partir de um lugar $p \in P$, como $I(t, p, \mu)$. Abaixo, são apresentadas novamente algumas definições que fazem parte dos conceitos adicionais às GSPNs.

Definição 2.13. (Transição de Entrada, de Saída e Inibidas) Os conjuntos de transições de entrada e de saída de um lugar $p \in P$ são, respectivamente: $\bullet p = \{t \in T \mid O(t, p, \mu) > 0, \exists \mu \in RS\}$, $p^\bullet = \{t \in T \mid I(t, p, \mu) > 0, \exists \mu \in RS\}$ e ${}^\circ p = \{t \in T \mid H(t, p, \mu) > 0, \exists \mu \in RS\}$.

Definição 2.14. (Lugar de Entrada, de Saída e Inibidores) Os conjuntos de lugares de entrada, de saída e inibidores de uma transição $t \in T$ são, respectivamente: $\bullet t = \{p \in P \mid I(t, p, \mu) > 0, \exists \mu \in RS\}$, $t^\bullet = \{p \in P \mid O(t, p, \mu) > 0, \exists \mu \in RS\}$ e ${}^\circ t = \{p \in P \mid F(t, p, \mu) > 0, \exists \mu \in RS\}$.

Definição 2.15. (Transição Disparável) Uma transição $t \in T$ está habilitada em uma marcação μ_i se, e somente se, todas as condições abaixo forem satisfeitas:

- $\mu_i \geq I(t, p, \mu_i), \forall p \in \bullet t;$
- $\mu_i < H(t, p, \mu_i), \forall p \in {}^\circ t;$
- $g(t, \mu_i) = true.$

Definição 2.16. (Disparo) Se t é uma transição que está habilitada em uma marcação μ_i , o disparo desta transição gera uma nova marcação μ_j tal que:

- $\mu_j(p) = \mu_i(p) - O(t, p, \mu_i), \forall p \in \bullet t.$
- $\mu_j(p) = \mu_i(p) + I(t, p, \mu_i), \forall p \in t^\bullet.$

2.7.8 Aproximação por Fases

A técnica de aproximação por fases [DAJ95, MR93] tem sido comumente utilizada para representar o comportamento de uma distribuição desconhecida. Essa distribuição desconhecida pode ser aproximada para outra distribuição conhecida, por exemplo, Erlang, Hipo-exponencial e Hiper-exponencial [Tri06]. É importante mencionar que essa técnica tem sido usada com sucesso na modelagem de atividades não-exponenciais.

As transições das redes de Petri estocásticas podem ser imediatas ou estocásticas, cujo tempo apresenta um atraso exponencialmente distribuído. Portanto, a técnica de aproximação por fases é utilizada para representar tempos que sigam outra distribuição [DAJ95, MR93].

A aproximação é realizada através da construção de uma sub-rede cujo *throughput* segue a distribuição desejada, conforme mostra a Figura 2.20. Esta sub-rede é construída

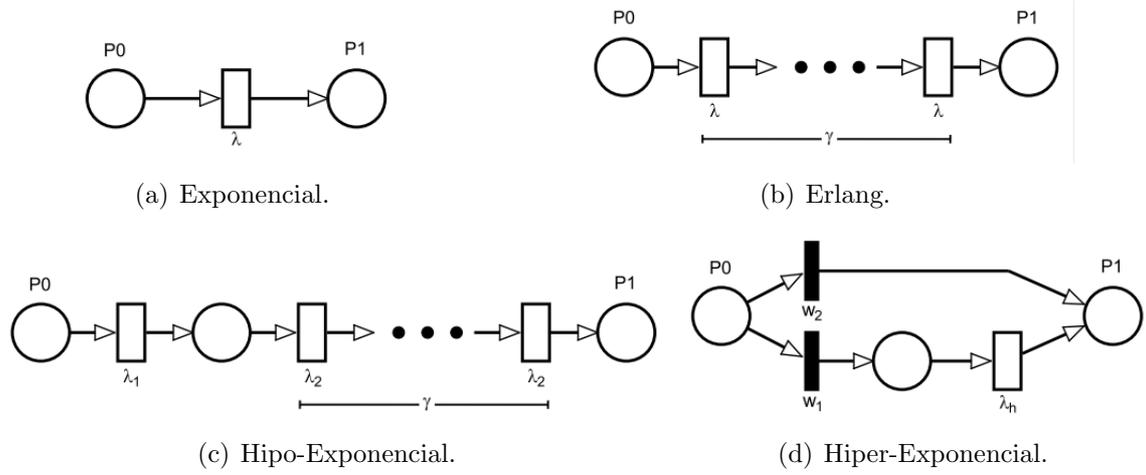


Figura 2.20 Sub-rede GSPN para representar distribuições polinômio-exponenciais utilizando *moment matching*.

utilizando-se o *moment matching*, que consiste em aproximar os dois primeiros momentos da distribuição, calculando-se a *média* (μ_D) e o *desvio padrão* (σ).

A partir dos valores calculados para μ_D e σ_D , utilizam-se os passos e equações a seguir para realizar a aproximação por fases [DAJ95].

1. Se $\mu_D / \sigma_D = 1$, utiliza-se a subrede que representa uma Distribuição Exponencial (Ver Figura 2.20(a)). É usada uma transição exponencial com taxa $\lambda = \mu_D$;
2. Se $\mu_D / \sigma_D \in \mathbb{Z} \wedge \mu_D / \sigma_D \neq 1$, utiliza-se a subrede que representa uma Distribuição Erlang (Ver Figura 2.20(b)) pela Equação (2.2). A taxa da transição exponencial é calculada pela Equação (2.3).

$$\gamma = \left(\frac{\mu_D}{\sigma_D}\right)^2 \quad (2.2)$$

$$\lambda = \frac{\gamma}{\mu_D} \quad (2.3)$$

3. Se $\mu_D / \sigma_D > 1$, utiliza-se a subrede que representa uma Distribuição Hipo-Exponencial (Ver Figura 2.20(c)) pela Equação (2.4). A taxa das transições exponenciais são calculadas pelas Equações (2.5) e (2.6). Os respectivos *delays* (valores medidos) atribuídos à transição exponencial são calculados pelas Equações (2.7) e (2.8).

$$\left(\frac{\mu_D}{\sigma}\right)^2 - 1 \leq \gamma < \left(\frac{\mu_D}{\sigma}\right)^2 \quad (2.4)$$

$$\lambda_1 = \frac{1}{\mu_1} \quad (2.5)$$

$$\lambda_2 = \frac{1}{\mu_2} \quad (2.6)$$

$$\mu_1 = \mu_D \mp \frac{\sqrt{\gamma(\gamma+1)\sigma^2 - \gamma\mu^2}}{\gamma+1} \quad (2.7)$$

$$\mu_2 = \gamma\mu_D \pm \frac{\sqrt{\gamma(\gamma+1)\sigma^2 - \gamma\mu^2}}{\gamma+1} \quad (2.8)$$

4. Se $\mu_D / \sigma_D < 1$, utiliza-se a subrede que representa uma Distribuição Hiper-Exponencial (Ver Figura 2.20(d)). A taxa da transição exponencial deve ser calculada com a Equação (2.9) e os pesos das transições imediatas são calculados pelas Equação (2.10) e (2.11).

$$\lambda_h = \frac{2\mu_D}{\mu_D^2 + \sigma_D^2} \quad (2.9)$$

$$w_1 = \frac{2\mu_D^2}{\mu_D^2 + \sigma_D^2} \quad (2.10)$$

$$w_2 = 1 - w_1 \quad (2.11)$$

2.8 SISTEMA TEF

Desde os primórdios a sociedade utiliza o comércio como ato de troca, venda e compra de valores e de produtos. Os povos antigos praticavam o comércio como meio de sobrevivência, trocando objetos ou alimentos. Com o passar do tempo, a simples troca de mercadorias deixou de ser suficiente, e com novas descobertas após expedições a novos mundos, o homem passou a utilizar pedras e metais preciosos como objeto de troca. Visando viabilizá-las, e tendo em vista uma padronização, a moeda foi criada até evoluir à forma do dinheiro utilizada hodiernamente [Web06].

Com as mudanças sociais existentes no mundo moderno, as pessoas têm buscado maneiras mais cômodas, seguras e eficazes para realizar suas transações comerciais. Nesse

sentido, as empresas têm se tornado mais dinâmicas e competitivas para acompanhar a economia globalizada. A crescente evolução tecnológica do sistema de pagamento tem levado a um crescimento cada vez mais expressivo dos meios de pagamento eletrônico, por exemplo, cartões de crédito, débito e transferências, entre outros [ABE09].

2.8.1 Pagamento Eletrônico

Atualmente, as empresas têm oferecido aos seus clientes diversas opções de pagamento eletrônico, com o intuito de facilitar e agilizar as transações comerciais. Nessa perspectiva de evolução do mercado de pagamento eletrônico, as empresas que não aderirem a esse tipo de negócio poderão ser afetadas negativamente, devido a uma maior exigência dos clientes [Alb00].

O pagamento eletrônico é qualquer tipo de pagamento que não utiliza papel moeda. Os tipos de pagamentos eletrônicos incluem: cartões de crédito e débito, rede ACH (*Automated Clearing House*)¹ e por meio de aparelhos telefônicos.

2.8.2 Benefícios do pagamento eletrônico

O pagamento eletrônico é muito prático, conveniente e seguro para o cliente, pois na maioria dos casos, o cliente fornece apenas as informações do cartão de crédito/débito e seu endereço de entrega. Os pagamentos eletrônicos, diminuem consideravelmente os custos até para as empresas, já que, as transações são processadas de forma eletrônica, além de diminuir falhas humanas e reduzir a utilização de papéis. Portanto, oferecer meios de pagamento eletrônico torna o processo de atendimento ao cliente mais rápido e ágil [Alb00].

Por apresentar benefícios fundamentais, o pagamento eletrônico tem se tornando um meio de pagamento em constante crescimento. No Brasil, o setor de pagamento eletrônico de fundos através do uso de cartões foi responsável em 2008 por uma movimentação financeira em torno de R\$ 375 bilhões de reais, e até o final do primeiro trimestre de 2009, o número de estabelecimentos que aceitam cartões superaram o número de 1,4 milhões [ABE09]. O setor de supermercados no Brasil, registrou um crescimento na participação dos pagamentos realizados com o uso de cartões (crédito e débito). Em 2006, a participação nos pagamentos chegou a 47,3% e no ano seguinte alcançou o percentual de 54,1% [ABR09]. Outro fator importante, para esse crescimento foi a facilidade de crédito concedido para pessoas físicas por instituições financeiras que aumentou o poder

¹O sistema ACH compreende o depósito direto, débito direto e os cheques eletrônicos (e-cheques).

compra dos consumidores [BCB09]. Isso demonstra cada vez mais que o mercado de pagamento eletrônico possui potencial de crescimento e aceitação.

2.8.3 Transferência Eletrônica de Fundos

O Sistema de transferência eletrônica de fundos funciona como um cheque eletrônico, por meio de cartões magnéticos, as empresas podem receber o pagamento das compras de seus clientes. Através do processamento eletrônico, com débito e crédito automáticos em conta bancária, as compras são aprovadas. O cliente quando realiza compras ou pagamentos com o cartão bancário, gera uma mensagem que é enviada eletronicamente à empresa que gerencia o sistema TEF, que são conferidos automaticamente pelo banco: senha, validade do cartão, lista negra e limite de compra de cada cliente. O débito acontece no momento da transação e o crédito somente na data previamente agendada [Cos02].

Segundo Albertin [Alb00], qualquer transferência ou pagamento por meio de um terminal eletrônico, instrumento telefônico, computador ou fita magnética, exige autorização de uma instituição financeira para debitar ou creditar em uma conta bancária. A primeira experiência surgiu na década de 1940, quando o primeiro cartão magnético foi lançado nos Estados Unidos. Atualmente, o sistema TEF utiliza modernas tecnologias da computação e das telecomunicações para garantir que as transferências de dinheiro ou ativos financeiros sejam executadas com rapidez e segurança. A Tabela 2.2 apresenta uma classificação das categorias TEF.

O TEF discado é utilizado em qualquer tipo de ponto de venda. Sua arquitetura dispõe de *Pinpad* (aparelho de leitura dos cartões), *software* gerenciador, que faz a comunicação entre o servidor TEF e as autorizadoras, e o *software* de cada bandeira. O TEF discado, utiliza uma conexão discada comum por meio de um modem instalado no próprio computador do ponto de venda [Visa].

O TEF dedicado é utilizado por clientes que possuam um número grande de pontos de venda. A arquitetura é igual ao discado, diferenciando-se, somente, no tipo de conexão. O TEF dedicado, utiliza uma linha privativa direta, *X-25* ou *Frame Relay* [Ita99], que o mantém conectado constantemente às autorizadoras, tornando o processo mais célere. A desvantagem é o elevado custo de manutenção [Visa].

O TEF IP (Internet Protocol) é destinado a empresas que cujo fluxo de transações seja alto. O TEF IP possui maior rapidez, agilidade e baixos custos de telefonia, pois utiliza a internet como meio de comunicação [SBH06, Visa].

Tabela 2.2 Categorias de pagamento TEF.

Fonte: [Alb00, p. 121]

Categorias	Subcategorias	Exemplos
1. Pagamentos Bancários e Financeiros	Pagamento em grandes escala Pagamento em pequena escala Home Banking	Transferência de banco para banco ATM (<i>Cash dispenser</i>) Pagamento de Contas
2. Pagamento de Varejo	Cartão de crédito Cartões de crédito/débito Cartões de pagamento	Visa, Mastercard, American Express
3. Pagamento de Comércio	Sistemas de pagamento baseados em sinal:	<i>Digi Cash</i>
Eletrônico <i>on-line</i>	Dinheiro eletrônico Cheques eletrônicos Cartões inteligentes ou cartões de débito	<i>Net Cheque</i> <i>Mondex Eletronic</i> <i>Currency Card</i>
	Sistemas de pagamento baseados em cartão de crédito: Cartões de crédito criptografados Número de autorizações de terceiros	Criptografia baseada na web <i>First Virtual</i>

2.8.4 Componentes do Sistema TEF

O sistema de transferência eletrônica de fundos é composto por três componentes: ponto de venda (PDV), servidor TEF e autorizadora [Ita99]. A seguir, são apresentadas as descrições de cada componente:

- **Ponto de Venda (PDV):** Terminal em que o funcionário efetua a venda. No PDV é montada a transação, que é enviada para o servidor TEF. Através do sistema utilizado no PDV, o funcionário registra todas as informações de entrada e de recebimento de dinheiro, de saída e de baixa no controle de estoque dos produtos que estão sendo vendidos;
- **Servidor TEF:** Servidor onde são instalados o gerenciador das transações TEF e o banco de dados. Nesse servidor, as transações são processadas e armazenadas. Além disso, esse servidor é responsável pela comunicação entre as autorizadoras;

- **Autorizadora:** A rede autorizadora é a empresa responsável pela comunicação da operação de compra entre o banco do cliente e o estabelecimento que oferece o serviço. Pode-se citar como exemplo: Amex [AME], Redecard [Red], Tecban [Tec], entre outros.

A Figura 2.21 apresenta uma visão da arquitetura desse sistema. O PDV envia a transação comercial de uma compra (Fluxo a.), que é processada e registrada no servidor TEF, para em seguida, ser enviada para a rede autorizadora (Fluxo b.). Esta, por sua vez, verifica junto ao banco do cliente a permissão para efetivação da compra. Em seguida, uma mensagem é retornada para o servidor TEF (Fluxo c.), que armazena dados da mensagem de aceitação ou negação da compra. Por fim, o servidor TEF envia essa mensagem para o PDV (Fluxo d.), informando ao cliente a aprovação ou não da compra [Ita99]. As mensagens utilizam a norma ISO 8583 [fs09] para especificar o protocolo do fluxo das transações financeiras com cartões.

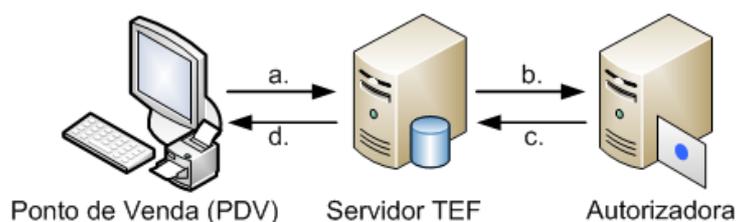


Figura 2.21 Fluxo dos processos do sistema TEF.

2.9 CONSIDERAÇÕES FINAIS

Este capítulo apresentou os principais conceitos que envolvem esta dissertação. Primeiramente, apresentamos conceitos de processo estocástico, cadeias de Markov, estratégias de medição de dados, dependabilidade e sistemas de redundância. Subseqüentemente, as redes de Petri foram introduzidas, mostrando que estas são uma ferramenta bem utilizada para modelagem e análise de vários tipos de sistemas, tais como os concorrentes, assíncronos, distribuídos, paralelos, não-determinísticos e estocásticos. Além disso, a técnica de aproximação por fases foi apresentada. Por fim, os conceitos sobre sistema de transferência eletrônica de fundos.

CAPÍTULO 3

METODOLOGIA DE AVALIAÇÃO DE DESEMPENHO

Este capítulo apresenta uma metodologia para auxiliar na modelagem e avaliação de desempenho de sistema de transferência eletrônica de fundos. A metodologia proposta apresenta diversas etapas, desde o entendimento do ambiente, seleção de métricas de desempenho até a validação do modelo e interpretação dos resultados.

3.1 INTRODUÇÃO

A integração dos serviços de pagamentos aliados ao avanço das novas tecnologias tem fomentado o crescimento e a complexidade das transações eletrônicas. Assim, em um mercado competitivo, as empresas devem oferecer sistemas confiáveis e com alta disponibilidade, isso porque, a indisponibilidade de um serviço contratado pode levar a sérios prejuízos econômicos. Como apresentado anteriormente, é de particular interesse deste trabalho analisar métricas relacionadas a níveis de utilização dos recursos, disponibilidade e confiabilidade do sistema TEF. Para tanto, é proposta a geração de modelos de desempenho baseados em *Generalized Stochastic Petri Nets* (GSPN).

Este trabalho está inserido no contexto de avaliação de desempenho de sistema de transferência eletrônica de fundos. A Figura 3.1 ilustra a metodologia adotada e contextualiza o ambiente no qual este trabalho está inserido, destacando suas principais atividades em: conhecimento e levantamento de informações do sistema, seleção das métricas de desempenho, criação do modelo de redes de Petri abstrato, medição, tratamento estatístico dos dados, refinamento e análise do modelo de redes de Petri, validação qualitativa e quantitativa, escolha do método de avaliação e interpretação dos resultados. É importante ressaltar que esta metodologia pode ser utilizada para auxiliar na modelagem de diversos sistemas que apresentem características semelhantes.

- **Conhecimento e levantamento de informações do sistema:** compreende o entendimento geral do sistema de transferência eletrônica de fundos e de seus componentes. O analista pode selecionar alguns componentes específicos para uma avaliação;

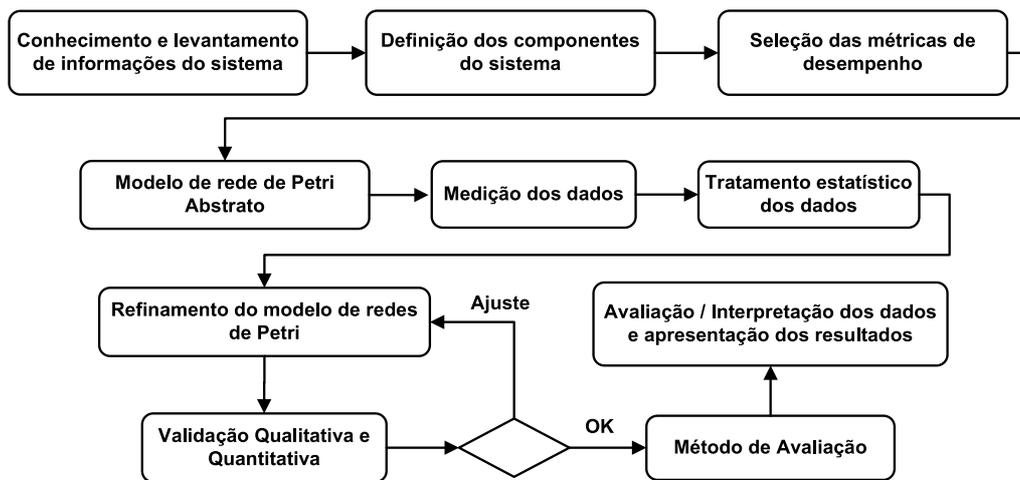


Figura 3.1 Fluxograma da metodologia de avaliação de desempenho.

- **Seleção das métricas de desempenho:** corresponde à seleção das métricas de desempenho relevantes para avaliação do sistema TEF, tais como utilização e *throughput* dos recursos;
- **Criação do modelo de redes de Petri abstrato:** o fluxo dos processos do sistema (TEF) são transformados em modelos de redes de Petri abstratos, permitindo assim, que este fluxo seja validado. Além disso, são criadas as métricas selecionadas;
- **Medição:** corresponde à configuração do ambiente para medição utilizando um gerador de transações genérico a partir de um *trace* de carga real. Os dados obtidos são utilizados para modelagem e validação do modelo;
- **Tratamento estatístico dos dados:** os dados medidos são processados por um tratamento estatístico para que os dados anormais (*outliers*) sejam analisados, garantindo, assim, que os dados medidos sejam confiáveis;
- **Refinamento e análise do modelo de redes de Petri:** o modelo de redes de Petri abstrato é refinado, com a finalidade de tornar o modelo mais representativo;
- **Validação Qualitativa e Quantitativa:** realiza análises e verificações das propriedades da rede de Petri, com o intuito de verificar as propriedades estruturais e comportamentais;
- **Escolha do método de Avaliação:** é definida a técnica de avaliação que deverá ser aplicada para que se obtenha as métricas do modelo, podendo essa técnica ser analisada em regime transiente ou estacionária;

- **Avaliação/interpretação dos dados e apresentação dos resultados:** os dados produzidos pelos experimentos são interpretados e os resultados serão apresentados.

3.2 CONHECIMENTO E LEVANTAMENTO DE INFORMAÇÕES DO SISTEMA

A primeira etapa da metodologia adota nesta dissertação tem como finalidade o entendimento e levantamento de informações do sistema. Essa etapa proverá uma base de conhecimento inicial sobre o sistema a ser modelado. É importante salientar que nesta fase são adquiridos os insumos necessários para a modelagem do sistema. Caso não sejam bem definidos esses insumos, o avaliador provavelmente cometerá erros de interpretação e, conseqüentemente, comprometerá a execução das etapas seguintes.

A metodologia ora apresentada baseia-se na modelagem de um sistema de transferência eletrônica de fundos através de redes de Petri. A partir dessas redes de Petri pode-se obter métricas que indicam o funcionamento do sistema. Uma vez analisado o ambiente, pode-se avaliar diferentes cenários visando encontrar uma configuração adequada. Com o estudo consolidado acerca do ambiente, o projetista terá insumos para a elaboração dos cenários a serem adotados no sistema real.

A área de atuação da companhia e os tipos de produtos vendidos podem influenciar nos tipos de cenários a serem criados. Produtos vendidos em farmácias, por exemplo, possuem um volume de saída menor do que os produtos alimentícios vendidos em supermercados. De forma semelhante, empresas que vendem passagens aéreas, também possuem tráfego diferenciado. Portanto, para cada área de atuação da empresa, determinadas cargas podem caracterizar diferentes situações de negócios. Estas cargas, podem assumir comportamentos, por exemplo, de distribuições poli-exponenciais e de tráfego em rajada ao longo de um período.

Uma companhia que utiliza pagamento eletrônico pode ser composta por diversas outras companhias, conhecidas como filiais, onde cada uma dessas poderá possuir um determinado número de ponto de venda (PDV). Os pontos de venda, por sua vez, possuem um tipo de fluxo de transações que varia de acordo com as características do negócio e com o tipo de transação (crédito ou débito). Assim, na configuração do ambiente, essas informações são importantes para definir quais frequências de envio de transações eletrônicas devem-se utilizar.

As informações elaboradas nesta etapa da metodologia servirão de suporte para as próximas etapas, mais especificamente na escolha do fluxo dos processos TEF que será abordado durante a execução da metodologia.

3.3 SELEÇÃO DAS MÉTRICAS DE DESEMPENHO

Após o entendimento dos aspectos do sistema TEF, o avaliador deve identificar as métricas de interesse para representar o desempenho do sistema. As métricas devem ser selecionadas para cada dispositivo. A seguir, são listadas algumas métricas relevantes:

- **Tempo de serviço (*Service Time*):** é o tempo em que uma transação gasta para ser executada pelo sistema.
- **Vazão (*Throughput*):** é a quantidade de operações que são executadas em um determinado período de tempo;
- **Tempo de fila (*Queue Time*):** é o tempo em que uma transação aguarda na fila para ser atendida;
- **Tempo de Resposta (*Response Time*):** compreende o tempo entre a chegada da transação e a resposta do sistema, ou seja, a soma do tempo na fila e o tempo de serviço;
- **Utilização (*Utilization*):** representa o percentual de tempo em que o sistema ficou ocupado executando uma transação. O recurso com maior utilização pode ser considerado o gargalo do sistema. Adicionalmente, se esse recurso for considerado o gargalo do sistema, ele deverá ter uma maior atenção por parte do avaliador no sentido de otimizá-lo;
- **Disponibilidade (*availability*):** probabilidade do sistema estar operacional em um determinado instante de tempo, isto é, alternância de períodos de funcionamento e reparo;
- **Confiabilidade (*reliability*):** capacidade necessária de um sistema ou componente atender suas funções, sob condições definidas, durante um determinado período de tempo.

As métricas escolhidas para avaliar o desempenho do sistema TEF deverão ser posteriormente convertidas para o formato GSPN, para que sejam calculadas a partir dos modelos em redes de Petri propostos [Zim01].

3.4 CRIAÇÃO DO MODELO DE REDE DE PETRI ABSTRATO

Esta fase concerne à geração do modelo em redes de Petri abstrato que represente o sistema a ser avaliado. Esse modelo é denominado abstrato, pois não possui os detalhes que descrevem o desempenho do sistema. Posteriormente, estas características serão incluídas no modelo para que esse tenha uma maior representação. Estas características serão representadas através da utilização da técnica de aproximação por fases [DAJ95, MR93]. A decisão de como o modelo abstrato deve ser gerado é bastante relevante, visto que, a escolha da granularidade irá determinar o detalhamento dos componentes dos modelos.

Os parâmetros necessários para cada componente serão descritos de maneira mais detalhada na Seção 4.1 do Capítulo 4. A Figura 3.2 apresenta uma rede de Petri abstrata para um sistema de transferência eletrônica de fundos. Esse modelo representa o envio de transações para um servidor TEF, que são processadas e armazenadas nos seus respectivos recursos (processador e disco). As transições do modelo abstrato processador e disco TEF, utilizam transições genéricas (trapezoidal), que na próxima etapa estas serão refinadas [DAJ95].

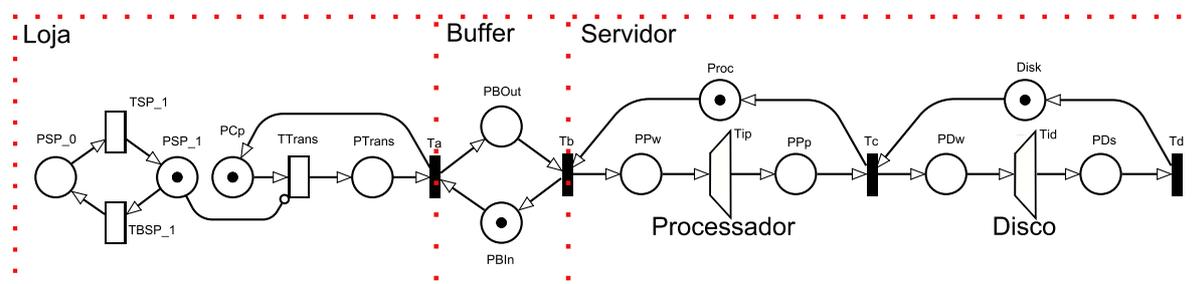


Figura 3.2 Modelo redes de Petri abstrato para um sistema TEF.

A geração do modelo abstrato é definida com base nos componentes delineados na fase de entendimento do ambiente. O processo de modelagem é fundamental na geração dos modelos, possibilitando a validação e análise qualitativa, de maneira que os modelos refinados possam ser construídos a partir do modelo “abstrato” com boas características, tais como, ausência de *deadlock*, reversibilidade, limitação, entre outras [Mur89, MLC96]. Após a criação do modelo abstrato, pode-se realizar um *token game* [Zim01] sobre o modelo em redes de Petri, visando validar o seu comportamento abstrato.

3.5 MEDIÇÃO

Para que o sistema seja refinado, é necessário que se obtenha ou que se colete dados acerca do sistema. Os dados coletados influenciarão nas demais etapas da metodologia. Sendo assim, esta fase deve ser executada de maneira criteriosa, de forma que erros sejam evitados para que não interfiram nos resultados da avaliação.

Para a obtenção dos dados, é possível encontrar dados históricos disponíveis e confiáveis, e dados históricos não disponíveis e não confiáveis. Quando os dados históricos estão disponíveis e foram coletados de maneira apropriada, faz-se necessário que os dados sejam organizados e analisados. A análise consiste em avaliar se os dados apresentados realmente representam a realidade das transações TEF.

Antes mesmo de medir os dados, o avaliador deverá escolher a ferramenta a ser utilizada na obtenção desses dados e especificar alguns parâmetros referentes à medição. Os parâmetros a serem definidos serão: intervalo de medição, intervalo de coleta das amostras, tamanho da amostra e o formato de armazenamento dos dados medidos. Esses parâmetros são importantes, pois podem influenciar na interpretação dos dados. Em outras palavras, um dimensionamento equivocado na quantidade de amostras, por exemplo, poderá interferir no ambiente e, conseqüentemente, na qualidade dos dados medidos.

Na obtenção dos dados, é necessária a montagem e configuração do *hardware/software* do sistema. Essa configuração irá influenciar diretamente nos resultados da avaliação do sistema. As características dos componentes de *hardware* (processador, disco e memória) e o tipo de arranjo dos discos são considerados parâmetros de *hardware*. O número de companhias, a quantidade de PDVs que enviam transações eletrônicas e o Sistema de Gerenciamento do Banco de Dados (SGDB) são parâmetros de configuração de *software*. Assim, as combinações desses parâmetros possibilitam ao avaliador, criar diversos cenários, uma vez que cada cenário resultará em resultados de desempenho diferentes.

Após a obtenção dos dados, seja através de dados históricos ou da medição, é fundamental que estejam organizados e armazenados para análise e, posteriormente, sejam tratados estatisticamente. Dessa forma, os dados obtidos nessa etapa da metodologia devem ser organizados para que o nível de abstração utilizado seja respeitado.

3.6 TRATAMENTO ESTATÍSTICO DOS DADOS

Esta seção descreve a aplicação dos métodos estatísticos para análise dos dados coletados na fase anterior da metodologia, permitindo assim que os dados medidos não apresentem anormalidades, ou seja, discrepâncias estatísticas.

Antes da análise estatística dos dados, faz-se uma análise exploratória deles [Tri05]. Análise exploratória permite avaliar distorções, presenças de viés e inconsistência nos dados. As distorções dos dados podem estar associadas à presença de valores extremos (*outliers*), por exemplo. Uma maneira de detectar essas distorções é utilizar o cálculo de limites superior e inferior, cujos dados além ou aquém destes respectivos limites são caracterizados como *outliers* [Tri05]. O limites inferior e superior são calculados da forma a seguir:

1. $Q1 - 1,5 \times IIQ$ -Intervalo Interquarti (limite inferior) e $Q3 + 1,5 \times IIQ$ (limite superior) para calcular os *soft outliers*;
2. $Q1 - 3 \times IIQ$ -Intervalo Interquarti (limite inferior) e $Q3 + 3 \times IIQ$ (limite superior) para calcular os *hard outliers*.

A Figura 3.3 apresenta um diagrama de caixa (*box-plot*) que destaca os valores mínimo, máximo, primeiro quartil, terceiro quartil, mediana, e *outliers*.

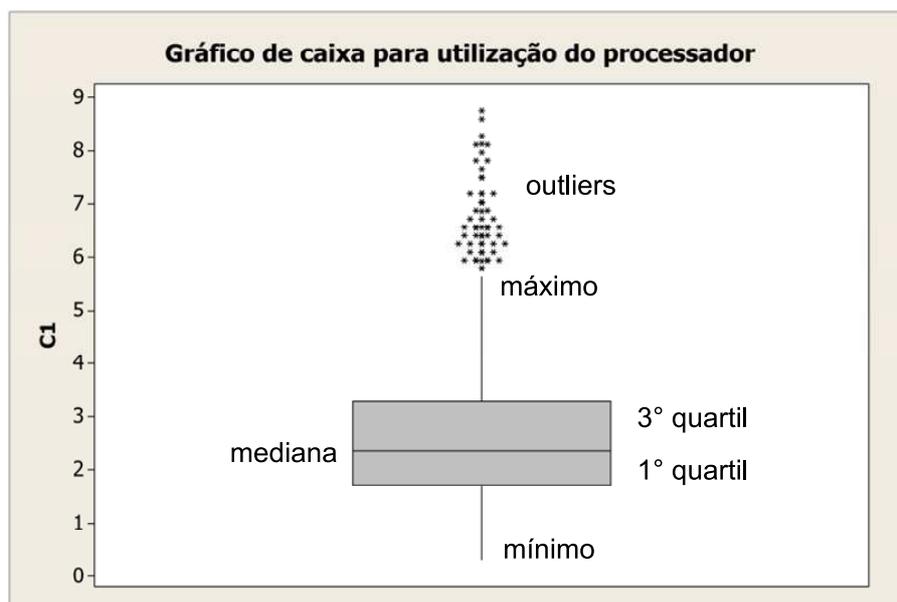


Figura 3.3 Gráfico de caixa para os dados da utilização do processador.

Os valores apresentados no diagrama de bloco são referentes aos dados medidos no sistema TEF. Estes valores correspondem à métrica de desempenho de utilização do processador. A frequência de envio de transações utilizada para esse exemplo foi de 900 (*tpm*). Os *outliers* são representados por um ponto, e a cada remoção destes, todas as estatísticas são calculadas novamente. Em uma determinada situação, os valores extremos

podem ser mantidos pelo avaliador quando se deseja analisar outros comportamentos do sistema. A análise estatística dos dados deve ser realizada após a análise exploratória.

3.7 REFINAMENTO DO MODELO DE REDES DE PETRI

Nesta etapa, o modelo abstrato deve ser refinado de maneira a obter o modelo de redes de Petri refinado, que represente o sistema de transferência eletrônica de fundos. O modelo inicialmente proposto poderá gerar outros modelos refinados em função das estatísticas obtidas na fase de medição. Na Seção 2.7.8 são apresentados alguns exemplos de modelos GSPNs que podem ser usados para o refinamento do modelo abstrato. Estes modelos são definidos através das estatísticas obtidas (média e desvio-padrão).

Para o refinamento do modelo abstrato, o avaliador deve observar se o nível de abstração utilizado está adequado para avaliar o sistema de forma precisa. Diante dessa observação, as estatísticas média (μ_λ) e desvio-padrão (σ_λ) são utilizadas para encontrar uma distribuição de probabilidade que represente as métricas desejadas. Esta abordagem é conhecida como aproximação por fases [DAJ95, MR93] e tem sido usada com sucesso para a modelagem de atividades não exponenciais.

A Figura 3.4 ilustra o modelo em redes de Petri refinado equivalente ao modelo abstrato da subrede processador TEF apresentado anteriormente. A técnica de aproximação por fases foi aplicada à transição original *Tip*. Visando aproximar o valor do tempo de processamento das transações, a transição *Tip* foi aproximada para uma distribuição hipo-exponencial.

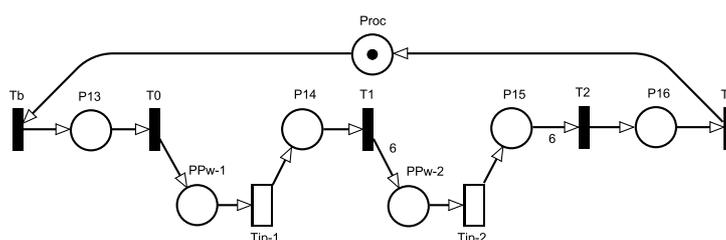


Figura 3.4 Modelo GSPN refinado para o componente processador do sistema TEF.

Existem pontos considerados importantes na escolha de uma aproximação por fase, como a qualidade da aproximação, as medidas de aproximação, o número de estados da aproximação e a facilidade de obtenção do modelo resultante, visto que, alguns modelos usados para a aproximação podem produzir resultados mais satisfatórios em relação aos outros.

3.8 VALIDAÇÃO QUALITATIVA E QUANTITATIVA

Nesta etapa, o modelo refinado é analisado, verificado e validado. Esta atividade visa analisar e verificar um conjunto de propriedades estruturais e comportamentais associadas ao modelo de rede de Petri e também validar o comportamento abstrato da rede de Petri através do *token game* [Zim01].

A análise qualitativa captura aspectos lógicos da evolução do sistema. Entre as propriedades de interesse, pode-se ressaltar a análise da existência de *deadlock*, *liveness*, limitação e reversibilidade. Essas propriedades estão detalhas no Capítulo 2.

A validação quantitativa tem por objetivo prover evidências que corroborem a representatividade do sistema pelo modelo. A representatividade do modelo mencionada diz respeito à capacidade de se replicar os valores das métricas do sistema através do modelo concebido. Para isso, cenários especialmente planejados devem ser especificados. O valor médio de utilização do processador para uma dada frequência de envio das transações é um valor que pode ser obtido facilmente, tanto no modelo quanto através de medição do sistema. A utilização do processador foi considerada como uma métrica para validação do modelo.

Além do valor médio de utilização do processador, também é possível validar o valor médio de utilização do disco. Essa validação pode ser realizada considerando o valor médio estimado no modelo em relação aos dados medidos no sistema real. Os resultados dessas métricas devem ser obtidos utilizando uma avaliação estacionária, ou uma avaliação transiente.

Para que o modelo seja considerado válido, é necessário que este proporcione o cálculo de métricas com erros de exatidão dentro de níveis considerados aceitáveis. Caso não se consiga obter os valores próximos aos medidos, é necessário revisar o modelo e o processo de refinamento, visando encontrar erros cometidos nessa etapa.

3.9 ESCOLHA DO MÉTODO DE AVALIAÇÃO

Nesta etapa, deve-se definir a técnica de avaliação que deverá ser aplicada para que se obtenham os valores das métricas do modelo associados aos cenários de interesse. O avaliador deve levar em consideração os recursos de *hardware* disponíveis, como também a estrutura dos modelos e as métricas que se desejam obter.

Os modelos podem ser avaliados, considerando-se o regime transiente ou estado estacionário. Nem todos os modelos de rede de Petri gerados propiciam o cálculo de métricas em regime estacionário, dado que nem todos os modelos alcançam esse estado. Para que

as métricas possam ser calculadas no estado estacionário, é necessário que a rede de Petri seja limitada. Como dito anteriormente, os modelos propostos neste trabalho são estruturalmente limitados, permitindo, assim, o cálculo de métricas no estado estacionário.

Na avaliação de estado transiente, os valores obtidos das métricas são calculados considerando-se um intervalo de tempo. Já a avaliação de estado estacionário, computa as métricas desprezando os efeitos transientes iniciais do modelo. Desse modo, métricas transientes ou estacionárias podem ser calculadas através de métodos (analíticos/numérico) pela simulação do modelo. Através dos métodos numéricos, as métricas são computadas analisando-se o grafo de alcançabilidade, sendo esse grafo computado apenas para modelos limitados. Assim, quando a propriedade limitação é garantida o modelo em análise não ultrapassa um número finito de estados. Como o modelo gerado neste trabalho é estruturalmente limitado, esta será uma opção adotada para o cálculo das métricas.

Para definir qual técnica de avaliação utilizar, deve-se considerar também a configuração dos recursos de *hardware* disponíveis, visto que, na análise, o tamanho da memória deve ser suficiente para o armazenamento do espaço de estados gerados. Apesar disso, caso o espaço de estados seja suficiente grande, pode ser computacionalmente inviável utilizar essa opção. Por sua vez, a simulação do modelo pode ser considerada para obtenção das métricas.

3.10 AVALIAÇÃO E INTERPRETAÇÃO DOS RESULTADOS

O objetivo desta última etapa consiste em realizar uma avaliação quantitativa dos modelos das redes de Petri geradas, com o intuito de se obter resultados das métricas definidas nas etapas anteriores. Após analisar um cenário inicial, é possível avaliar diversos outros cenários sobre o mesmo modelo de redes de Petri. Nesse sentido, quando se deseja comparar os resultados de diferentes cenários de forma a encontrar o mais significativo, deve-se utilizar o mesmo conjunto de métricas, e apenas um único parâmetro da rede deve sofrer variação com um determinado incremento.

Sendo assim, os cenários criados devem representar situações hipotéticas para o sistema de transferência eletrônica de fundos, onde os resultados destes definem como o sistema TEF se comporta, caso os parâmetros do sistema fossem alterados. É importante que os cenários não sejam criados de forma aleatória. A fase de entendimento do ambiente deve fornecer insumos suficientes para que o avaliador tenha um ponto de partida, mesmo que a princípio essas informações sejam superficiais. Por dessas, o avaliador poderá definir quais são os cenários mais representativos a serem executados. A análise dos diversos cenários tem como objetivo criar diversas situações hipotéticas, que através

de interpretações, produzirão informações para tomada de decisão.

Os resultados obtidos poderão auxiliar num processo de tomada de decisão. Cabe ao avaliador interpretar os dados e apontar as possíveis soluções para um problema analisado. As métricas podem, por exemplo, apontar para um possível gargalo no sistema, ou ainda, a necessidade de melhorias no desempenho do sistema. Além disso, a criação de um documento com as conclusões dos cenários com os gráficos e tabelas, pode auxiliar a apresentação desses resultados.

3.11 CONSIDERAÇÕES FINAIS

Este capítulo apresentou a metodologia adotada para avaliação de desempenho de sistemas de transferência eletrônica. Em resumo, essa metodologia se iniciou a partir da compreensão geral do sistema avaliado, identificação de seus componentes e dos critérios de desempenho que serão utilizados na geração do modelo. Dessa forma, a metodologia proposta apresenta um conjunto de atividades definidas para o processo de avaliação.

Este capítulo apresenta os modelos de redes de Petri para a modelagem de sistemas de transferência eletrônica de fundos. Além disso, é proposto outros modelos para representar os sistemas de redundância *cold standby*, *hot standby* e *TMR* de todo o sistema TEF. Para esses modelos são apresentadas suas descrições e métricas de desempenho.

4.1 MODELOS

Esta seção apresenta os modelos para os componentes que fazem parte do sistema de envio de transações eletrônicas de fundos. Os modelos representam uma loja, *buffer*, processador TEF e disco TEF. Além desses, são apresentados os modelos *cold standby*, *hot standby* e *TMR* - *Triple modular redundancy* para representar tipos de redundância do sistema TEF.

O trabalho em comento propõe a utilização de redes de Petri com tempos estocásticos para a modelagem de um sistema de transferência de fundos [ASM⁺09]. Especificamente, serão empregadas as *Generalized Stochastic Petri nets* (GSPN) para especificação dos modelos.

A seguir, serão apresentados os modelos GSPNs. Para cada modelo, será apresentada a sua representação gráfica e a sua descrição. Como também suas métricas de desempenho.

4.1.1 Loja

A loja compreende parte do sistema TEF, que tem por finalidade permitir que seus clientes efetuem pagamentos destinados a bens e serviços sem a necessidade do uso do papel moeda. A loja é formada por pontos de vendas (PDVs), esses que enviam as transações eletrônicas de pagamento comerciais para que sejam processadas e armazenadas no servidor TEF.

O modelo GSPN loja (Ver Figura 4.1) representa uma loja que envia transações eletrônicas. Esse modelo também utiliza um mecanismo de variação no envio das transa-

ções comerciais em tráfego de rajada. No modelo proposto, é abstraído a representação dos PDVs, já que o mecanismo de envio das transações permite determinar o número de transações que podem ser enviadas em um intervalo.

A Figura 4.1 apresenta o modelo que representa uma loja do sistema TEF.

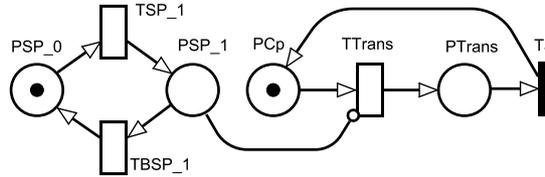


Figura 4.1 Modelo GSPN para a loja.

A transição $TTrans$ representa o envio das transações comerciais para o servidor TEF. O disparo dessa transição é gerado a uma taxa de λ_{TTrans} . Para representar o envio das transações de forma exponencial, são necessárias que as transições TSP_1 e $TBSP_1$ tenham a mesma taxa da transição $TTrans$. No entanto, caso o avaliador deseje representar o envio das transações em tráfego de rajada, é importante incrementar diferentes taxas nas transições TSP_1 e $TBSP_1$. Transições essas que representam o mecanismo de variação no envio das transações em regime de rajada. Adicionalmente, com o disparo da transição $TTrans$, um *token* é armazenado no lugar $PTrans$, que em seguida é disparado pela transição Ta , sendo consumido do lugar $PTrans$ e armazenado no lugar PCp e na fila representada pelo modelo *buffer*.

Assim, a partir do modelo que representa a loja, o avaliador pode definir diversos cenários de interesse com ou sem a representação do tráfego em rajada para análise. Além disso, o avaliador pode adicionar outros modelos lojas em paralelo para representar negócios de naturezas diferentes.

4.1.2 Buffer

A fila é um fator significativo no desempenho do sistema TEF. O modelo *buffer* representa o mecanismo de armazenamento e repasse das transações para o servidor TEF. Esse modelo de fila é implementado utilizando a política de filas FIFO (*First In First Out*). É possível, através desse modelo, variar a quantidade de transações comerciais que podem permanecer na fila antes de serem atendidas, sendo, dessa forma, uma fila limitada. A Figura 4.2 apresenta o modelo que representa a fila do sistema TEF.

O lugar $PBin$ representa o local onde é definida a capacidade da fila do modelo. O número de *tokens* no lugar $PBin$ determina a quantidade de transações que podem ser armazenadas na fila e repassadas para o servidor TEF. Conseqüentemente, cada *token*

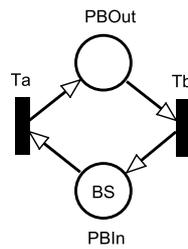


Figura 4.2 Modelo GSPN para o *buffer*.

armazenado no lugar *PBOut* representa que uma transação será processada pelo servidor TEF. Para utilizar o modelo *buffer* em conjunto com o outro que representa a loja, faz-se necessária uma fusão entre a transição *Ta* referente ao modelo loja e a transição *Ta* referente ao modelo *buffer*. Para que o modelo loja envie uma transação para o modelo *buffer*, é necessário que o lugar *PBIn* tenha pelo menos um *token* armazenado e que a transição *Ta* esteja habilitada.

A ausência de *tokens* em *PBIn* denota que o servidor TEF está atendendo às transações no limite de sua capacidade. Por conseguinte, caso o servidor TEF finalize o processamento de n transações, a transição *Tb* é então disparada e n *tokens* são armazenados em *PBIn*. Portanto, novas transações poderão ser enviadas ao servidor TEF para serem processadas.

4.1.3 Processador TEF

O processador é considerado um componente determinante no desempenho do sistema TEF, pois é tido como um dos componentes que recebe uma maior demanda de atividades. Este recurso é responsável pelo processamento das transações comerciais enviadas pelas lojas. Através deste modelo, também é possível variar a quantidade de transações comerciais que podem ser processadas em paralelo. A Figura 4.3 apresenta o modelo para representar o processamento das transações no sistema TEF.

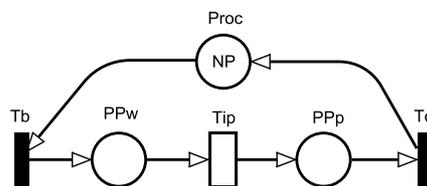


Figura 4.3 Modelo GSPN para o processador TEF.

Um *token* no lugar *Proc* denota que o recurso processador está ocioso. O disparo da transição *Tb* representa a solicitação para processar uma transação comercial e o disparo

de T_{ip} indica que a transação está sendo processada. Esse processamento é gerado a uma taxa de $\lambda_{T_{ip}}$. Por conseguinte, quando o modelo processador estiver em uso, o lugar $Proc$ estará sem *token* armazenado. Para que uma nova transação seja processada, é necessário haver o disparo da transição T_c . Portanto, o disparo da transição T_c indica que a atividade de processamento foi concluída. Um *token* armazenado no lugar $Proc$ indica também essa atividade. É relevante mencionar que a técnica de aproximação por fases apresentada no Capítulo 2 pode ser aplicada à transição T_{ip} . Para que o avaliador utilize o modelo processador, é necessário realizar uma fusão com o modelo *buffer*. Essa fusão é realizada através da transição T_b referente ao modelo *buffer* e a transição T_b referente ao modelo que representa o processador.

Após definido o modelo que representa o processamento das transações, é necessário definir a métrica relacionada a esse componente. A Tabela 4.1 apresenta as métricas utilizadas.

Tabela 4.1 Métricas do modelo processador TEF.

Métrica	Expressão
Utilização do processador TEF	$P\{\#Proc = 0\}$
<i>Throughput</i> do processador TEF	$P\{\#PPp > 0\} \times \lambda_{T_{ip}}$

4.1.4 Disco TEF

Para o sistema TEF, o disco é um importante componente que manipula as operações de leitura e escrita das transações comerciais. No mesmo sistema, o disco é considerado um componente crítico para o desempenho do sistema, pois cada transação computada pelo processador é registrada no disco. Esse modelo permite variar o grau de paralelismo em que as transações comerciais são armazenadas no disco. A Figura 4.4 apresenta o modelo para o disco TEF.

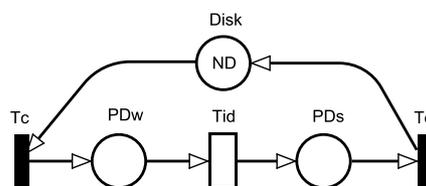


Figura 4.4 Modelo GSPN para o disco TEF.

Para utilizar esse modelo em conjunto com o modelo processador, deverá ser realizada

uma fusão entre a transição Tc do modelo processador e a transição Tc referente ao modelo disco. Um *token* no lugar $Disk$ denota que esse recurso está ocioso. A habilitação da transição Tc representa a solicitação de leitura/escrita de uma transação comercial e o disparo de Tid representa a atividade de leitura/escrita dessa transação no disco. Essa atividade é gerada a uma taxa de λ_{Tid} . Caso o lugar $Disk$ não tenha *token* armazenados, o recurso disco estará em uso. Para que uma nova transação seja lida/escrita no disco, é necessária que a transição Td dispare, liberando assim o recurso. Adicionalmente, com o disparo da transição Td , um *token* é armazenado no lugar $Disk$ indicando que o recurso está disponível. É importante salientar que a técnica de aproximação por fases apresentada no Capítulo 2 pode ser aplicada à transição Tid .

Após definido o modelo disco que representa a leitura e escrita das transações, é necessário definir as métricas relacionadas a esse componente. A Tabela 4.2 apresenta essas métricas.

Tabela 4.2 Métricas do modelo disco TEF.

Métrica	Expressão
Utilização do disco TEF	$P\{\#Disk = 0\}$
Throughput do disco TEF	$P\{\#PDs > 0\} \times \lambda_{Tid}$

4.1.5 Modelos de Redundância

Redundância em ambientes computacionais é essencialmente importante, pois em um mercado competitivo, a presença de falhas podem comprometer a integridade do serviço disponibilizado, como também ocasionar prejuízos financeiros. Neste sentido, visando aumentar a disponibilidade e a confiabilidade dos sistemas TEF, são propostos alguns modelos para representar os sistemas de redundância, *Hot standby*, *TMR* e *Cold standby*.

4.1.5.1 Modelo Cold standby. O sistema *cold standby* [KA80] é composto de dois componentes: o módulo principal e reserva. O módulo de reserva só é ativado quando o módulo principal falha. Para que o módulo reserva entre em funcionamento, é necessário um período entre a ativação. Por esse motivo, a redundância *cold standby* é denominada passiva. A Figura 5.13 apresenta o modelo para a redundância *cold standby*.

O lugar C_ON representa o estado de funcionamento do módulo principal. Para que o módulo principal falhe, é necessária que a transição $MTBF_C$ seja disparada.

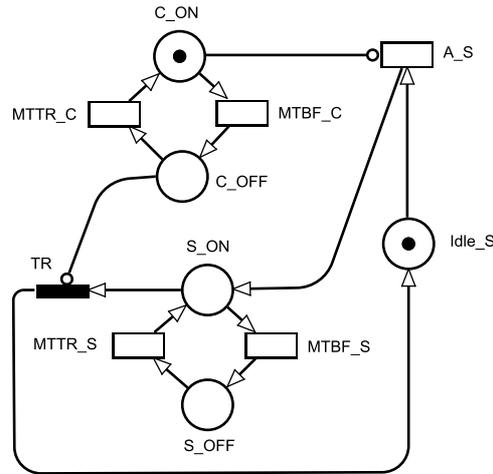


Figura 4.5 Modelo GSPN *cold standby* para o sistema TEF.

Essa falha é gerada a uma taxa de λ_{MTBF_C} . Com o disparo de $MTBF_C$, um *token* é removido do lugar C_ON e armazenado em seguida no lugar C_OFF . Com a ausência de um *token* no lugar C_ON , a transição A_S é habilitada. O disparo desta transição é responsável pela ativação do módulo reserva. Em seguida, com o disparo da transição A_S , um *token* é consumido do lugar $Idle_S$ e armazenado no lugar S_ON . Um *token* armazenado no lugar S_ON representa a ativação do módulo reserva. Já um *token* no lugar $Idle_S$ representa o sistema reserva desabilitado. O lugar S_ON representa o estado de funcionamento do módulo reserva e o disparo da transição $MTBF_S$ indica a falha desse módulo.

As transições $MTTR_C$ e $MTTR_S$ com as taxas $(\lambda_{MTTR_C}, \lambda_{MTTR_S})$ representam respectivamente o reparo dos módulos principal e reserva. Com o disparo da transição $MTTR_C$, o módulo principal é reparado e um *token* é consumido do lugar C_OFF e armazenado no lugar C_ON . No entanto, pelo disparo da transição $MTTR_S$ o módulo reserva é reparado e *token* é armazenado em S_ON . Um *token* armazenado no lugar S_ON ocasiona a habilitação da transição TR . O disparo desta transição é responsável pela desativação do módulo reserva, visto que, para esse tipo de redundância apenas um módulo deve ficar em funcionamento. Assim, após o disparo da transição TR , um *token* do lugar S_ON é consumido e armazenado no lugar $Idle_S$, indicando o desligamento do módulo reserva.

As métricas que definem a disponibilidade e a confiabilidade são apresentadas na Tabela 4.3. Relevante dizer que para computar a métrica de confiabilidade através do modelo, é necessário garantir que não aconteça o disparo das transições de reparo $MTTR_C$ e $MTTR_S$. Para isso, o peso do arco de entrada de cada transição de reparo é alterado

para um valor inteiro positivo maior que um. A condição de disparo, dessa forma, não será satisfeita e a transição de reparo não poderá ser disparada, o que possibilita o cálculo da confiabilidade.

Tabela 4.3 Métricas do modelo *cold standby*.

Métrica	Expressão
Disponibilidade	$P\{\#C_ON > 0 \text{ OR } \#S_ON > 0\}$
Confiabilidade	$P\{\#C_ON > 0 \text{ OR } \#S_ON > 0\}$

4.1.5.2 Modelo Hot standby. O tipo de redundância *hot standby* [KA80] é utilizado em sistemas que necessitam de um alto nível de disponibilidade. A redundância *hot standby* possui como principal característica, se comparado ao sistema *cold standby*, a ausência de um tempo de ativação. Esse mecanismo utiliza o módulo reserva em sincronismo com o módulo principal, ou seja, os dois módulos funcionam de forma paralela. Caso ocorra uma falha no módulo principal, o módulo reserva automaticamente assume a função do módulo principal sem que o sistema perceba essa mudança. A Figura 4.6 apresenta o modelo para a redundância *hot standby*.

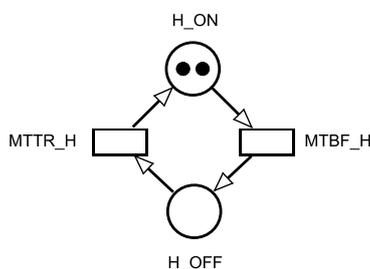


Figura 4.6 Modelo GSPN *Hot standby* para o sistema TEF.

No lugar H_ON são armazenados dois *tokens* com o intuito de representar o funcionamento dos módulos principal e reserva em paralelo. Os lugares H_ON e H_OFF representam, respectivamente, os estados de funcionamento e de falha. A transição $MTBF_H$ e $MTTR_H$ representam, respectivamente, o tempo médio entre falhas e o tempo médio de reparo do modelo. Estas transições são disparadas com as taxas λ_{MTBF_H} e λ_{MTTR_H} . Para que o modelo *hot standby* entre em estado de completa falha, ou seja, o módulo principal e reserva falhem, é necessário que os dois *tokens* sejam consumidos do lugar H_ON e em seguida armazenados no lugar H_OFF . Caso apenas um *token* seja removido do lugar H_ON pelo disparo da transição $MTBF_H$, o modelo continua operando.

Para que seja garantida a paralelização das atividades de falha e reparação dos módulos, as transições devem ser utilizadas com a semântica de disparo *Infinite Server* [Zim01].

Sendo definido o modelo *hot standby*, as métricas relacionadas à disponibilidade e à confiabilidade precisam ser definidas. A Tabela 4.4 apresenta essas métricas. Para computar a métrica de confiabilidade através do modelo, é necessário garantir que não aconteça o disparo da transição de reparo $MTTR_H$. Para isso, o peso do arco de entrada da transição de reparo é alterado para um valor inteiro positivo maior que três. A condição de disparo, dessa forma, não será satisfeita e a transição de reparo não poderá ser disparada, o que possibilita o cálculo da confiabilidade.

Tabela 4.4 Métricas do modelo *hot standby*.

Métrica	Expressão
Disponibilidade	$P\{\#H_ON > 0\}$
Confiabilidade	$P\{\#H_ON > 0\}$

4.1.5.3 Modelo TMR. O modelo *TMR* (*Triple Modular Redundancy*) foi originalmente desenvolvido por Von Neumann [SS98]. O mecanismo de redundância TMR possui características que permitem que até um dos três módulos que compõem este componente possa falhar e assim mesmo o componente não falhe. Esta técnica é baseada no mascaramento dos componentes em falha de maneira que o comportamento não seja propagado. A técnica *TMR* é muito empregada em sistemas onde a disponibilidade e restrições temporal são fatores cruciais. A Figura 4.7 apresenta o modelo para a redundância *TMR*.

Os lugares T_ON e T_OFF representam, respectivamente, os estados operacional e de falha para cada um dos três módulos do componente TMR. Cada módulo é representado por um *token* no lugar T_ON . De forma semelhante ao modelo *hot standby*, o modelo *TMR* não necessita de tempo para ativação do módulo reserva. Assim, caso algum módulo falhe, a mudança do módulo principal para o reserva acontece de forma imediata. O disparo da transição $MTBF_T$ indica que um dos três módulos falhou e o disparo da transição $MTTR_T$ indica que este módulo é reparado. No entanto, caso aconteça apenas uma falha, o modelo continua operando. Assim, para que o modelo *TMR* pare de funcionar, é necessário que os três *tokens* sejam removidos do lugar T_ON e armazenados no lugar T_OFF . Para garantir a paralelização dos eventos de falha e reparação dos módulos, as transições $MTTR_T$ e $MTBF_T$ devem utilizar a semântica de disparo *Infinite Server* [Zim01].

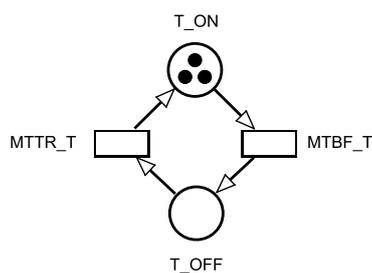


Figura 4.7 Modelo GSPN *TMR* para o sistema TEF.

Sendo definido o modelo *TMR*, as métricas relacionadas à disponibilidade e à confiabilidade precisam ser definidas. A Tabela 4.5 apresenta essas métricas. Para que seja computada a métrica de confiabilidade através deste modelo, é necessário garantir que não aconteça o disparo da transição de reparo *MTTR_T*. Para isso, o peso do arco de entrada da transição de reparo é alterado para um valor inteiro positivo maior que quatro. A condição de disparo, dessa forma, não será satisfeita e a transição de reparo não poderá ser disparada, o que possibilita o cálculo da confiabilidade.

Tabela 4.5 Métricas do modelo *TMR*.

Métrica	Expressão
Disponibilidade	$P\{\#H_ON > 0\}$
Confiabilidade	$P\{\#H_ON > 0\}$

4.2 CONSIDERAÇÕES FINAIS

O capítulo ora posto apresentou os modelos propostos para modelagem do sistema de transferência eletrônica de fundos. Foram apresentados os seguintes modelos: loja, *buffer*, processador TEF, disco TEF e suas respectivas métricas. Além desses, foi apresentado modelos com o mecanismo de redundância *cold standby*, *hot standby* e *TMR* para representar todo o sistema TEF.

ESTUDO DE CASO

Neste capítulo serão apresentados os estudos de caso realizados no laboratório de análise de performance CIn-Itautec. O objetivo dos estudos de caso é validar tanto os modelos GSPN propostos, quanto a metodologia de avaliação. Para tanto, todas as etapas da metodologia foram utilizadas para auxiliar na avaliação desses estudos, a fim de demonstrar a aplicabilidade destes. Além disso, as seguintes métricas de desempenho do sistema TEF foram analisadas: utilização, *throughput*, disponibilidade e confiabilidade.

5.1 INTRODUÇÃO

O método adotado para validar o modelo GSPN e a metodologia de avaliação de desempenho do sistema de transferência eletrônica, consiste na comparação dos resultados obtidos através dos modelos GSPN propostos com os dados medidos num sistema TEF real. Nesse sentido, procura-se demonstrar a aplicabilidade do modelo GSPN e da metodologia desenvolvida criando-se cenários que representem situações reais. Três estudos de caso serão apresentados: (i) avaliação de desempenho dos recursos processador e disco, (ii) os recursos são avaliados utilizando um mecanismo que varia o envio das transações em tráfego de rajada, e (iii) avaliação dos níveis de disponibilidade e confiabilidade do sistema TEF.

5.1.1 Conhecimento e levantamento de informações do sistema

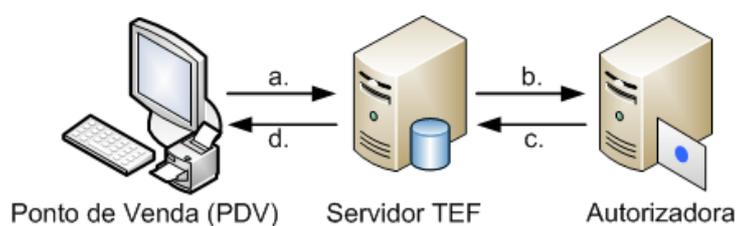
Como resultado da primeira etapa da metodologia, foi possível identificar os componentes de um sistema TEF, a interação destes e o fluxo dos processos envolvidos no envio de uma transação de pagamento eletrônico. Nessa primeira etapa, foram levantados e definidos o tipo de companhia, a quantidade de filiais, o número de PDVs e o *trace* de carga. A Tabela 5.1 apresenta um resumo do ambiente TEF analisado.

A Figura 5.1 apresenta um fluxo simplificado dos processos do sistema TEF. Esse fluxo é composto por quatro etapas: companhia para servidor TEF, servidor TEF para autorizadora, autorizadora para servidor TEF e servidor TEF para companhia. No entanto,

Tabela 5.1 Resumo do ambiente TEF.

Tipo	# PDV	# Filiais
Loja de departamento	3345	75

este trabalho foca no fluxo da companhia para o servidor TEF. Em outras palavras, não foram consideradas as operações que acontecem entre o servidor TEF e a autorizadora, visto que, as informações processadas nesse fluxo são restritas apenas às autorizadoras de cartões.

**Figura 5.1** Fluxo dos processos do sistema TEF.

Os recursos identificados para análise de desempenho do sistema TEF foram processador e disco. A escolha desses recursos foi motivada por serem considerados os componentes de maior relevância.

5.1.2 Seleção das métricas de desempenho

Na segunda etapa da metodologia, foram selecionadas as métricas utilizadas na avaliação do sistema. Para analisar o sistema, foi necessário escolher a ferramenta de medição. A ferramenta escolhida foi o *Perfmon* [Bat03], por ser prática e por possuir diversas métricas possibilitando o monitoramento de desempenho. O *Perfmon* é uma ferramenta de monitoramento de desempenho do sistema operacional *Windows* que permite examinar recursos, tais como: processador, disco, memória, rede, entre outros.

As métricas de desempenho selecionadas no *Perfmon* para analisar os componentes foram:

- **Tempo de processador.** Indica a porcentagem do tempo em que o processador estava ocupado servindo alguma solicitação, ou seja, a utilização do processador do sistema.
- **Tempo ocioso do disco.** Indica a porcentagem do tempo em que o disco estava ocioso, em outras palavras, sem executar leitura e/ou escrita.

- **Média de transferências/seg no disco.** Indica o tempo médio em que uma requisição de I/O (leitura e escrita) leva para cada transferência.
- **Comprimento médio da fila de disco.** Indica o número médio de requisições de I/O (leitura e escrita) em espera na fila do disco.

Além das métricas apresentadas acima, é importante ressaltar que o tempo médio de serviço do processador e do disco foi definido de forma indireta [Lil00]. O tempo médio de resposta do processador foi obtido utilizando-se a métrica *tempo de processador* dividida pela quantidade de transações TEF. Esse experimento será explicado de forma detalhada mais adiante. No entanto, o tempo médio de serviço do disco foi obtido pela subtração da métrica *Média de transferências/seg no disco* pelo *Comprimento médio da fila de disco* [Fri05]. Por fim, o *throughput* do processador e disco foi calculado dividindo-se a utilização pelo tempo de serviço de cada recurso [Fri05].

5.2 CRIAÇÃO DO MODELO DE REDES DE PETRI ABSTRATO

A terceira etapa da metodologia consiste na criação de um modelo abstrato do sistema TEF. As informações obtidas na fase de entendimento do ambiente, forneceram elementos suficientes para a construção do modelo GSPN. A Figura 5.2 ilustra o modelo abstrato do sistema de transferência eletrônica de fundos. Cada um dos componentes do sistema TEF como loja, buffer e servidor (processado e disco) foram conectados de forma seqüencial aos demais. Além disso, cada um desses componentes podem ser replicados e executados em paralelo.

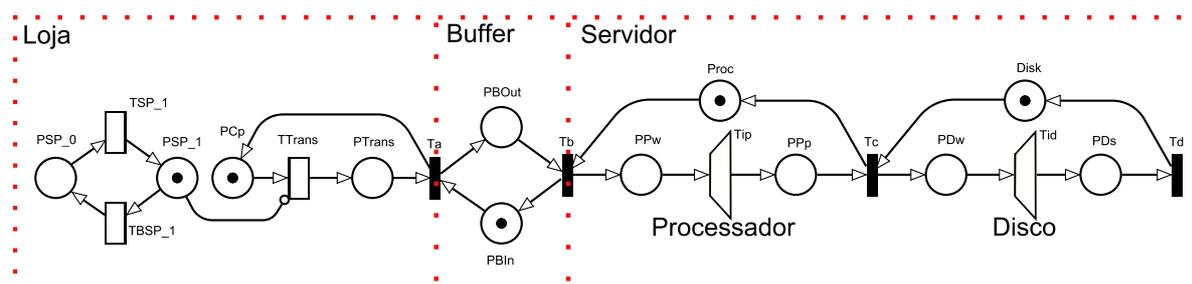


Figura 5.2 Modelo redes de Petri abstrato do sistema TEF.

A validação do modelo abstrato foi realizada através de um *token game*, cujo intuito foi reproduzir o funcionamento genérico das transações sob a óptica do desempenho. As métricas selecionadas para a avaliação do modelo foram: utilização do processador e utilização e do disco. A Tabela 5.2 apresenta um resumo das métricas selecionadas para

o modelo abstrato do sistema TEF. As métricas 1 e 2 representam, respectivamente, a probabilidade do recurso processador e disco TEF estarem sendo utilizados. Por fim, as métricas 2 e 3 representam, respectivamente, o *throughput* do processador e do disco TEF. O *throughput* do processador TEF é computado pela probabilidade de haver *tokens* armazenados no lugar *PPs* e em seguida multiplicada pela taxa da transição *TiP* e o *throughput* do disco TEF é calculado pela probabilidade de haver *tokens* armazenados no lugar *PDs* e em seguida multiplicada pela taxa da transição *Tid*.

Tabela 5.2 Resumo das métricas para o modelo abstrato do sistema TEF.

#	Métrica	Expressão
1	Utilização do processador TEF	$P(\#Proc = 0)$
2	Utilização do disco TEF	$P(\#Disk = 0)$
3	<i>Throughput</i> do processador	$P\{\#PDs > 0\} \times \lambda_{TiP}$
4	<i>Throughput</i> do disco	$P\{\#PDs > 0\} \times \lambda_{Tid}$

5.3 MEDIÇÃO

Na quarta etapa da metodologia, o ambiente foi montado e as métricas de interesse foram coletadas através do monitoramento do ambiente. Primeiramente, configurou-se um ambiente para emular um sistema TEF operacional, ou seja, um ambiente real composto por pontos de vendas de lojas reais, autorizadoras, entre outros. As aplicações responsáveis pelo gerenciamento das transações comerciais foram executadas em um servidor Intel Xeon Dual Core 2800, com 4GB de memória RAM utilizando um sistema operacional *Windows Server 2003*. A aplicação responsável pela geração de cargas que enviou as transações comerciais foi executada em uma máquina Intel Xeon Dual Core 3200, com 3GB de memória RAM utilizando um sistema operacional *Windows Server 2003*. A Figura 5.3 ilustra o ambiente de medição no qual os experimentos foram conduzidos.

Para se obter os dados com pouca interferência externa, os processos que não eram estritamente necessários para a realização do experimento foram encerrados. Os experimentos foram realizados em dois servidores, onde no primeiro estava a aplicação TEF e o SGDB; e no segundo estavam instalados os clientes, responsáveis pelo envio das transações de pagamento (geração de carga). Vale salientar que os dados do *trace* utilizados para gerar a carga das transações, foram extraídos de um cliente real da Itautec. Por motivos de privacidade, não será citado o nome do cliente neste trabalho.

Após o ambiente estabilizado, ou seja, quando o servidor da autorizadora estava dis-

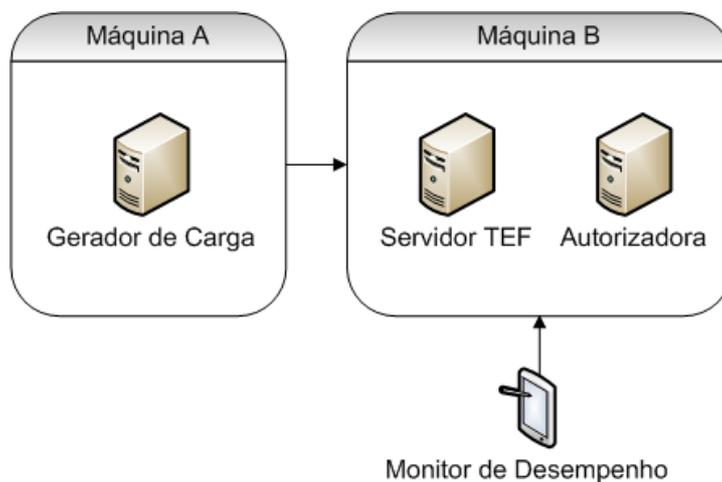


Figura 5.3 Ambiente de medição.

ponível e o servidor TEF tinha todos os PDVs conectados, foi iniciada a medição das métricas de desempenho utilizando o monitor de desempenho do *windows* (*perfmon*). A duração da medição foi de duas horas e o intervalo de coleta dos dados foi de cinco segundos. O intervalo de coleta foi definido após diversos experimentos, onde inicialmente o tempo adotado foi de um segundo. Após análises das medições percebeu-se que este intervalo mudava a característica dos dados. Assim, foi determinado um intervalo de amostras que não afetasse a qualidade dos dados medidos, visto que, quanto maior o número de amostras, maior o *overhead* da ferramenta. O período de duração da medição foi definido por se tratar de um intervalo considerável para um ambiente comercial. Os dados coletados foram armazenados em *logs*. É importante salientar que os dados coletados foram armazenados em uma partição do disco isolada do ambiente instalado, a fim de evitar que os dados medidos fossem afetados.

O primeiro experimento realizado objetivou encontrar o tempo médio de serviço em que uma transação leva para ser processada e o tempo médio de serviço em que a transação consome na leitura/escrita do disco. A obtenção desses valores tem o objetivo de “alimentar” o modelo proposto, para que este represente de forma significativa um ambiente real. Assim, o primeiro experimento foi configurado utilizando uma companhia com apenas um PDV que enviava apenas uma transação comercial por minuto. O tempo médio de serviço do processador foi calculado utilizando-se a métrica *tempo de processador* multiplicado pelo tempo de medição, que em seguida foi dividido pela quantidade de amostras e novamente dividido pela quantidade de transações (tpm). Desta forma, foi possível coletar dados referentes ao desempenho de apenas uma transação. No entanto, através desse experimento, foi possível capturar o tempo de serviço sem que houvesse interferência,

pois o ambiente estava sob baixa demanda.

O segundo experimento teve como objetivo criar cenários para obter os dados para a validação do modelo, uma vez que já foram obtidos os valores dos tempos de serviço. Foram criados sete cenários com as seguintes frequências: 100, 200, 300, 400, 500, 600 e 700 tpm (transações por minuto). Estas frequências foram definidas por se tratarem de valores representativos a diversos tipos de negócios. Nesse segundo experimento, foram utilizados a mesma duração de medição e o intervalo de amostras adotados no primeiro. A Tabela 5.3 apresenta a configuração referente ao segundo experimento.

Tabela 5.3 Configuração do 2º Experimento.

TPM	Filiais	PDV
100	75	3345
200	75	3345
300	75	3345
400	75	3345
500	75	3345
600	75	3345
700	75	3345

5.4 TRATAMENTO ESTATÍSTICO DOS DADOS

Na quinta etapa da metodologia foram analisados os dados medidos. Como discutido anteriormente, conduziu-se uma análise exploratória cujo objetivo foi, eventualmente, remover os valores extremos (*outliers*). Os *outliers* foram identificados e ao serem retirados, os dados passaram a não representar a realidade. Portanto, verificou-se que os *outliers* eram naturais ao ambiente e representavam a realidade do ambiente estudado.

A Tabela 5.4 apresenta os dados referentes ao primeiro experimento executado na fase de medição. Os parâmetros (μ_D) e (σ_D) representam, respectivamente, a média e o desvio-padrão referentes ao tempo médio de serviço dos recursos computacionais. O tempo de serviço está associado à atividade de processamento e armazenamento das transações nos componentes processador e disco.

5.5 REFINAMENTO E ANÁLISE DO MODELO DE REDES DE PETRI

Esta etapa da metodologia concerne ao refinamento do modelo de redes de Petri abstrato. Para refinar os modelos, as estatísticas média (μ) e desvio padrão (σ) foram utilizadas

Tabela 5.4 Média e Desvio Padrão.

Tempo de Serviço	μ_D (s)	σ_D (s)
<i>Processador</i>	1,3110E-03	5,080E-04
<i>Disco</i>	2,7560E-03	3,530E-04

para definir o modelo refinado. A técnica de aproximação por fases [DAJ95] foi utilizada para determinar qual distribuição de probabilidade expolinomial era mais adequada para representar os dados medidos. Essa técnica comparou a média (μ) e o desvio padrão (σ) para obter os tempos a serem associados às transições genéricas T_{ip} e T_{id} do modelo abstrato. Conforme apresentado no Capítulo 2, a técnica de aproximação por fases representa funções expolinomiais como Erlang, Hipo-exponencial e Hiper-exponencial [BGdMT06].

Utilizando as estatísticas da Tabela 5.4, foram determinadas as distribuições de probabilidade adequadas para modelagem das subredes. A Tabela 5.5 apresenta os parâmetros utilizados nas subredes processador e disco. Através da Equação (2.4) apresentada no Capítulo 2, a distribuição Hipo-exponencial foi escolhida. Essa distribuição foi modelada através de duas transições temporizadas com peso nos arcos. Os tempos μ_1 e μ_2 foram calculados através das Equações (2.5 e 2.6) e representam uma aproximação do tempo médio de serviço das transições T_{ip} e T_{id} , e os pesos dos arcos são representados por γ (Ver Equação 2.4).

Tabela 5.5 Parâmetros da distribuição.

Transição	μ_1 (s)	μ_2 (s)	γ	Distribuição desejada
T_{ip}	8,00E-05	5,40E-04	6	Hipo-exponencial
T_{id}	1,00E-05	5,00E-05	61	Hipo-exponencial

A Figura 5.4 apresenta o modelo refinado do sistema TEF. As transições T_{ip} e T_{id} foram refinadas para representar distribuições Hipo-exponenciais. Assim, os valores de μ_1 calculados para as transições T_{ip} e T_{id} foram utilizados nas transições Tip-1 e Tid-1. De forma similar, os valores de μ_2 foram utilizados nas transições Tip-2 e Tid-2.

5.6 VALIDAÇÃO QUALITATIVA E QUANTITATIVA

Nesta etapa, o modelo refinado foi analisado e verificado. As análises verificaram um conjunto de propriedades estruturais e comportamentais associadas ao modelo de rede de

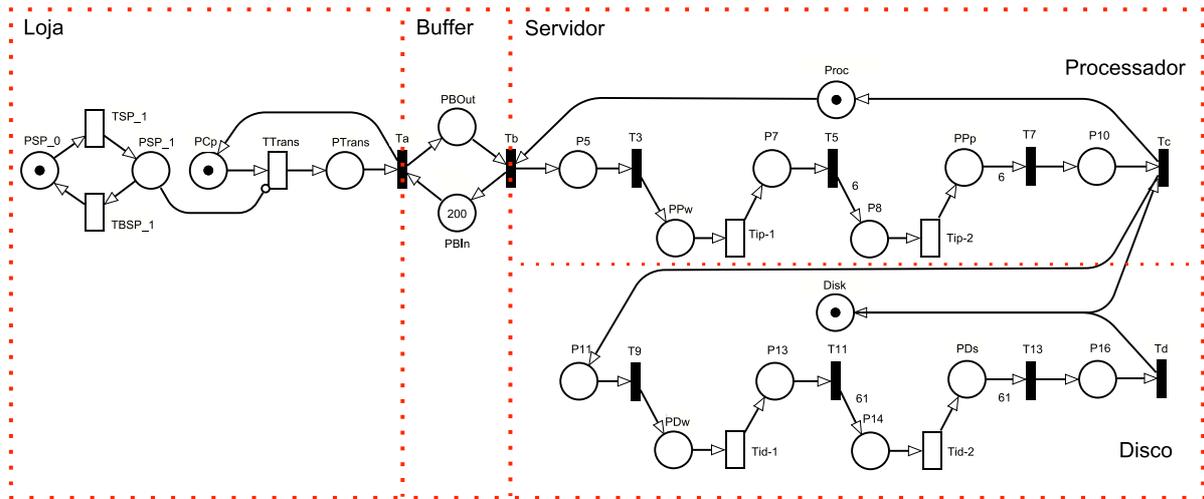


Figura 5.4 Modelo redes de Petri refinado do sistema TEF.

Petri. A análise qualitativa foi executada através da ferramenta *Peptool* (*Programming Environment based on Petri Nets*) [Gra97], objetivando analisar propriedades comportamentais e estruturais. As seguintes propriedades de interesse foram satisfeitas: *Deadlock Freedom*, Reversibilidade, *Liveness* e Limitação.

Como a propriedade estrutural limitação foi garantida pelo modelo, o tipo de avaliação quantitativa escolhida foi a análise estacionária. Esse método de análise apenas é utilizado quando as redes geradas são estruturalmente limitadas. Um *tokengame* também foi executado para validar o comportamento da rede de Petri.

Antes da realização da etapa de refinamento do modelo abstrato foi realizado um experimento visando sua validação quantitativa. Para isso, foram utilizados os valores que representam os tempos de serviço dos componentes processador e disco TEF (Ver Tabela 5.4) no modelo abstrato (Ver Figura 5.2). No entanto, os resultados alcançados a partir deste modelo apresentaram erros entre 96% e 100%. Neste sentido, foi necessário a utilização da técnica de aproximação por fases para o refinamento do modelo abstrato com a finalidade de aproximar o comportamento do sistema TEF real ao modelo GSPN. É importante mencionar que, esta técnica vem sendo utilizada com sucesso na modelagem de sistemas.

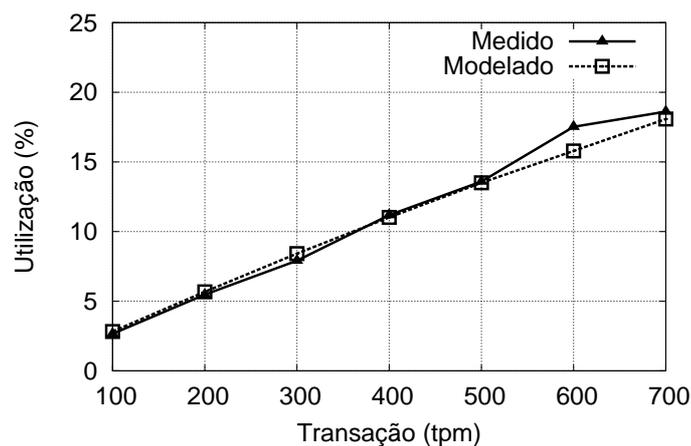
Para a validação quantitativa, foram especificados 7 cenários representativos. As frequências de envio das transações comerciais utilizadas em cada cenário foram 100, 200, 300, 400, 500, 600 e 700 tpms (transações por minuto). Como citado anteriormente, estas frequências foram adotadas por abrangerem vários tipos de negócios. A Tabela 5.6 ilustra a configuração da companhia.

Tabela 5.6 Experimento 2.

TPM	Filiais	PDV
100	75	3345
200	75	3345
300	75	3345
400	75	3345
500	75	3345
600	75	3345
700	75	3345

Para validar o modelo refinado, os dados medidos pelo sistema foram comparados com os dados calculados pelo modelo. As métricas adotadas para a validação do modelo foram: utilização e *throughput* do processador e do disco. Adicionalmente, as análises do modelo TEF foram executadas através da ferramenta TimeNET [GKZH95].

As Figuras 5.5, 5.6, 5.7 e 5.8 apresentam a comparação dos valores medidos do sistema TEF real com os valores calculados do modelo. Estas mostram respectivamente a comparação das seguintes métricas: utilização do processador, utilização e *throughput* do disco. O erro médio encontrado na métrica utilização do processador foi de 0,16% e para o *throughput* deste foi 3,22%. Por fim, a métrica utilização do disco foi 0,87% e o *throughput* deste foi 0,03%.

**Figura 5.5** Utilização do Processador TEF.

Os resultados obtidos com o modelo e as respectivas medições do sistema foram comparados por meio do Teste T-emparelhado [Lil00]. Pôde-se constatar com 95% de grau de

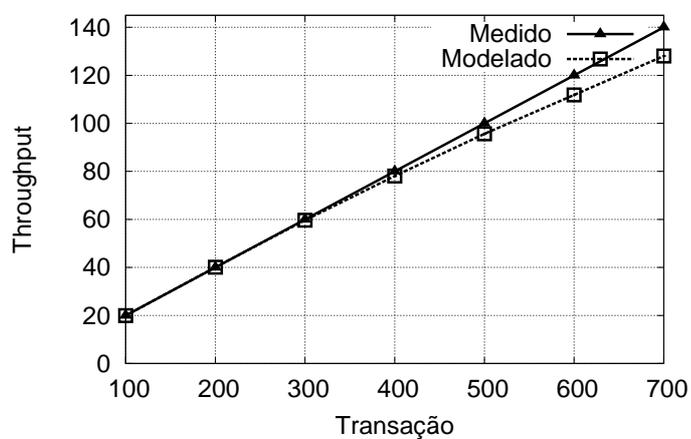


Figura 5.6 *Throughput* do Processador TEF.

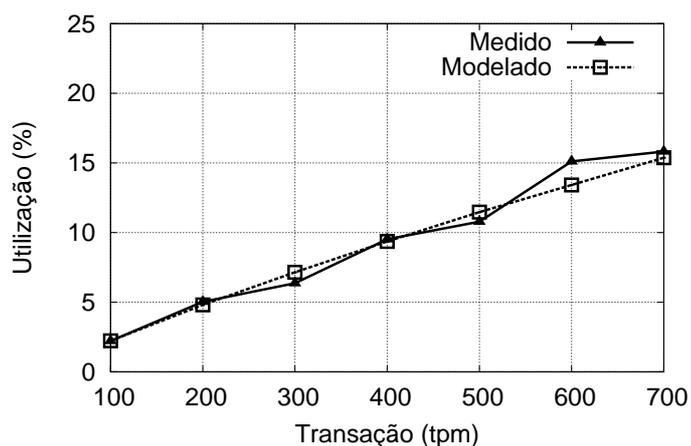


Figura 5.7 Utilização do Disco TEF.

confiança, que os resultados não evidenciam qualquer discrepância entre os dados medidos e os valores obtidos do modelo.

5.7 ESCOLHA DO MÉTODO DE AVALIAÇÃO E INTERPRETAÇÃO DOS RESULTADOS

Nesta etapa, definiu-se a técnica de avaliação a ser utilizada para obtenção valores das métricas do modelo, como também são apresentados os resultados relativos à degradação de desempenho (*performability*), considerando diferentes cenários, como também o método de avaliação escolhido. Esta avaliação teve como objetivo criar diversos cenários interes-

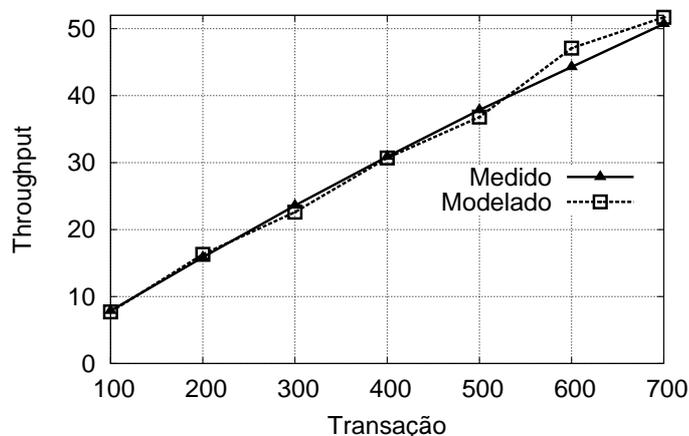


Figura 5.8 *Throughput* do Disco TEF.

santes para analisar o desempenho do sistema em estudo, produzindo assim informações revelantes para tomada de decisão. O estudo de caso, aqui apresentado, foi conduzindo através de um gerador de carga utilizando *traces* de um sistema TEF real. As transações comerciais fornecidas pelo gerador de carga são enviadas para o servidor TEF por meio de um mecanismo de variação das transações. Inicialmente, um estudo de caso foi executado para analisar o desempenho dos recursos computacionais processador e disco. Subseqüentemente, e de maneira similar, ambos os recursos processador e disco foram analisados, considerando-se o impacto do desempenho causado pelo mecanismo de variação de envio das transações comerciais em rajada. Finalmente, foi realizado um estudo de confiabilidade e disponibilidade utilizando o modelo de redundância *cold standby* para representar o sistema TEF. Para este estudo, foram utilizados os resultados obtidos da avaliação de desempenho do primeiro cenário.

O método de avaliação utilizado nos cenários 1 e 2 foi a análise estacionária, visto que o modelo proposto garante a propriedade estrutural limitação. No entanto, no cenário 3 foi utilizado a análise transiente, cujo propósito foi obter os resultados em um dado intervalo de tempo determinado.

5.7.1 Cenário 1: Análise de Desempenho dos Recursos Processador e Disco do Sistema TEF

O primeiro cenário descreve uma loja de departamentos contendo vários pontos de vendas que registram transações comerciais de crédito e débito. O objetivo desse cenário foi identificar possíveis pontos de saturação nos recursos computacionais processador e disco

do sistema TEF. As frequências das transações adotadas nesse cenário foram: 1500, 2500, 3500, 4500, 5500, 6500, 7000, e 7500 tpms. Na frequência 2500 tpm, o nível de utilização do disco atingiu o valor de 52% e o processador 61%. Na frequência 5500 tpm, o nível de utilização do processador chegou a 82%, enquanto o nível de utilização do disco foi de 52%. Estes resultados revelam que os níveis de utilização do processador e disco, respectivamente, excederam os valores de 80% e 50%. Conseqüentemente, estes recursos estão próximos da região crítica de operação (saturação) [Fri05]. Portanto, o avaliador deve analisar esta situação com cautela, uma vez que, os níveis de utilização dos recursos estão próximos da saturação. Assim, é sugerido aplicar técnicas de *tuning*, *upgrade* no sistema ou até mesmo uma substituição dos recursos como medidas preventivas para evitar que o sistema entre em saturação.

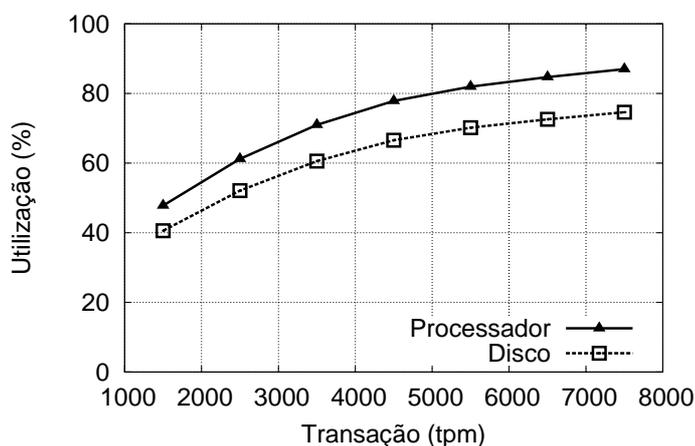


Figura 5.9 Cenário 1 (Utilização do Processador e Disco TEF).

A Figura 5.9 ilustra os níveis de utilização para cada frequência adotada no Cenário 1. Esse cenário pode ser utilizado para um planejamento de capacidade, já que esses resultados não foram medidos em um ambiente real, por se tratar de um volume de transações relativamente alto para a loja de departamento avaliado. Com isso, o avaliador pode prever através do modelo o possível comportamento do desempenho, caso o volume de vendas cresça devido ao aquecimento do mercado. É importante enfatizar que o nível de utilização de 50% para disco, por exemplo, e 80% para o processamento de atividades no processador, são consideradas críticas (gargalo) [Fri05].

A Figura 5.10 ilustra o *Throughput* obtido do modelo processador e disco TEF. Os resultados mostram a variação do *Throughput* em função do número de transações enviadas para serem processadas e lidas ou escritas, respectivamente pelo processador e disco. A

partir de 2500 transações, o *throughput* destes apresenta um crescimento que em seguida tende a entrar em estado estacionário.

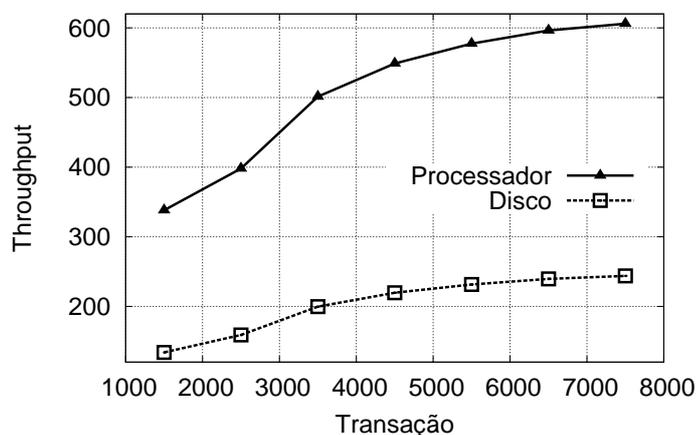


Figura 5.10 Cenário 1 (*Throughput* do Processador e Disco TEF).

5.7.2 Cenário 2: Avaliação de Desempenho considerando Tráfego em Rajada do Sistema TEF

O objetivo do segundo cenário é avaliar o impacto do desempenho causado pelo mecanismo de envio das transações comerciais em tráfego de rajada. Assim, foi possível encontrar níveis de utilização para uma mesma frequência de envio das transações comerciais. Para executar esse cenário, os tempos utilizados nas transições *TBSP_1* e *SP_1* (ver Figura 5.4) foram incrementados, visto que as transições dessa subrede permitem esse tipo de comportamento. Para esse cenário, foram criados cinco experimentos (*B1* a *B5*), para cada frequência (1500, 2500, 3500, 4500, 5500, 6500, 7000, e 7500 tpms) com o propósito de analisar o impacto dos níveis de utilização dos recursos.

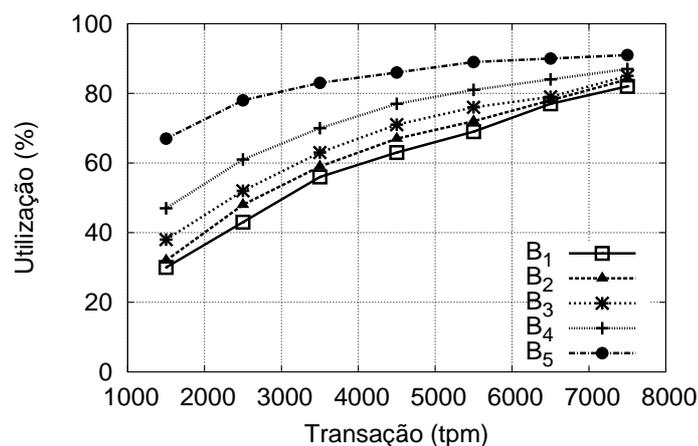
Dessa forma, foi possível estimar o nível de utilização considerando uma carga de trabalho particular. A Tabela 5.7 apresenta um sumário desse cenário com os tempos usados para cada frequência de envio das transações. Os valores dos tempos desse experimento foram baseados em dados de empresas reais, sendo omitido neste trabalho o nome da empresa por questões de privacidade.

Para a frequência 2500, os valores do processamento das transações demonstraram um aumento nos níveis de utilização do recurso processador. Os valores encontrados nesta frequência foram: 43,3% (*B1*), 48,0% (*B2*), 52,2% (*B3*), 61,2% (*B4*) e 78,0% (*B5*). Esses resultados demonstraram que, considerando o experimento com o tempo *B5*, o sistema

Tabela 5.7 Resumo dos tempos para cada tpm.

Tpm	B_1 (s)	B_2 (s)	B_3 (s)	B_4 (s)	B_5 (s)
1500	0,200	0,400	0,600	0,800	1,000
2500	0,120	0,240	0,360	0,480	0,600
3500	0,085	0,171	0,257	0,342	0,428
4500	0,066	0,133	0,200	0,266	0,333
5500	0,054	0,109	0,163	0,218	0,272
6500	0,046	0,092	0,138	0,184	0,230
7500	0,040	0,080	0,120	0,160	0,200

aproximou-se da região crítica de processamento [Fri05]. Nesses casos, é necessário que o avaliador aplique técnicas de *tuning* ou reduza a carga do sistema TEF. A Figura 5.11 ilustra os resultados para o segundo cenário.

**Figura 5.11** Cenário 2 (Utilização do Processador TEF).

Da mesma forma, a frequência de 2500 tpms mostrou um crescimento dos níveis de utilização do disco, os quais foram: 36,8% (B1), 40,9% (B2), 44,5% (B3), 52,1% (B4) e 64,6% (B5). Os resultados demonstram que no experimento com o tempo B_4 com a frequência 2500, o disco aproximou-se da região crítica de operação. A Figura 5.12 ilustra os resultados do disco para o cenário 2.

Os cenários apresentados nas Figuras 5.9, 5.11 e 5.12 mostram a análise de tendência do *throughput* relacionadas à variação das transações em rajada. Os resultados obtidos desses cenários têm o propósito de encontrar níveis de utilização críticos dos recursos (processador e disco) para um planejamento das infra-estruturas TEF. Além disso, é

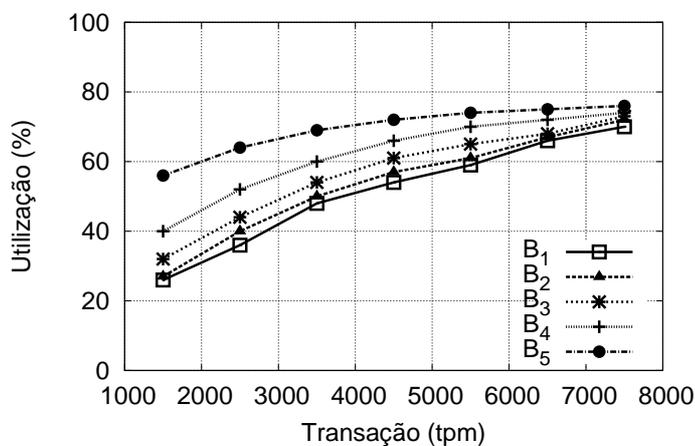


Figura 5.12 Cenário 2 (Utilização do Disco TEF).

possível utilizar esses resultados como ponto de apoio para tomada de decisão, com o intuito de adotar medidas de otimização do sistema.

5.7.3 Cenário 3: Avaliação da Disponibilidade e Confiabilidade do Sistema TEF

O terceiro cenário tem como propósito analisar a degradação do desempenho (*performance*) do sistema TEF, considerando diferentes cláusulas de nível SLA (*Service Level Agreement*). Nesse sentido, foi adotado um modelo com redundância *cold standby* (ver Figura 5.13) visando melhorar os níveis de disponibilidade e de confiabilidade do sistema TEF. Para isso, foram realizados experimentos com esse modelo buscando alcançar bons níveis de disponibilidade e confiabilidade. O modelo GPSN *cold standby* foi detalhado no Capítulo 4. O objetivo desse cenário foi também analisar o impacto que falhas provocam no sistema TEF, a partir de métricas de *throughput*, disponibilidade e confiabilidade. Assim, para analisar a degradação do sistema foi necessário combinar os resultados obtidos dos cenários de desempenho com os obtidos do modelo de redundância.

Adicionalmente, esse cenário considerou os resultados de desempenho (*throughput*) obtidos no primeiro cenário (ver Figura 5.10). Para isso, foi definido analisar apenas dois valores do *throughput* dos recursos processador e disco, ou seja, os resultados com menor e maior intensidade no intervalo das frequências 1500 a 7500 tpms.

Para analisar os níveis de disponibilidade e confiabilidade do modelo GPSN *cold standby*, foram adotadas três SLA. Em cada SLA utilizou componentes com diferentes níveis de confiabilidade, visando analisar o comportamento destes. A Tabela 5.8 apresenta esse acordos de nível de serviço, além dos parâmetros de *dependability*, como por

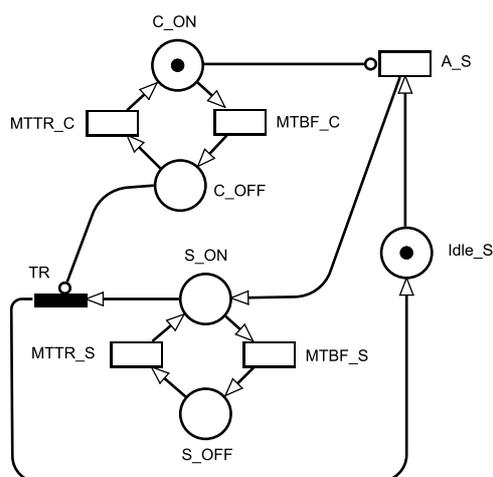


Figura 5.13 Modelo GSPN *cold standby* para representar o sistema TEF.

exemplo, os tempos de falha e de reparo para o módulo principal ($MTBF_C$, $MTTR_C$) e reserva ($MTBF_S$, $MTTR_S$), e por fim, o tempo de ativação do módulo reserva ($Activate_S$).

Tabela 5.8 Resumo dos parâmetros de Dependabilidade (horas).

SLA	Cold StandBy		Spare		
	MTTR_C	MTBF_C	MTTR_S	MTBF_S	Activate_S
1	8	12700	10	9700	0,125
2	8	1700	10	800	0,125
3	8	360	10	140	0,125

A Figura 5.14 ilustra o comportamento da disponibilidade relacionada a cada acordo de nível de serviço considerando o modelo *cold standby*. Como pode ser observada na Figura 5.14, no período de um ano ocorre uma pequena degradação no desempenho do sistema proporcionada por falhas neste, mas levando em consideração que o módulo reserva foi ativado. No entanto, a partir do segundo ano, é possível perceber que os níveis de disponibilidade para cada SLA tornam-se constantes. O sistema TEF apresentou menor degradação com o SLA 1 do que os SLAs 2 e 3. Portanto, através dessa análise foi possível perceber que para cada SLA adotada no modelo de redundância, diferentes níveis de disponibilidade foram alcançados. Assim, dependendo do nível de complexidade do sistema, o avaliador poderá obter, através do modelo, níveis de disponibilidade adequados a cada SLA. Portanto, quanto maior a disponibilidade, maior será a probabilidade das transações serem atendidas.

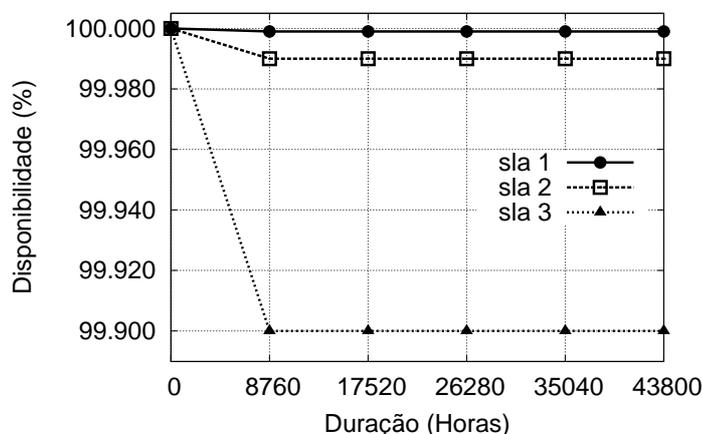


Figura 5.14 Disponibilidade do sistema TEF.

A Tabela 5.9 apresenta os resultados de disponibilidade do sistema TEF, usando cada acordo de serviço e comparando os resultados com o seu correspondente tempo de parada por ano (*downtime*). A partir dos dados de disponibilidade obtidos foi possível classificar os tipos de complexidade do sistema, como também obter o máximo tempo em que o sistema fica parado (*downtime*). Com o *SLA* 1, por exemplo, o sistema obteve um tempo médio total de parada de cinco minutos em um ano.

Tabela 5.9 Resultados de disponibilidade do sistema TEF.

SLA	Tipo do sistema	Disponibilidade (%)	#9s	Indisponibilidade (min/ano)
1	<i>Alta Disponibilidade</i>	99,999	5	5
2	<i>Tolerantes a falha</i>	99,990	4	52
3	<i>Bem gerenciável</i>	99,900	3	525

O resultado da análise de confiabilidade do sistema TEF é apresentada na Figura 5.15. Para esta análise, o período de tempo considerado foi de cinco anos. Considerou-se, também, que o sistema TEF falhou e não houve reparo tanto do módulo principal como do módulo reserva. É importante ressaltar que os resultados de confiabilidade alcançados com os SLA 2 e 3 (ver Figura 5.15) são bem similares, já que tendem ao fim do primeiro ano.

No entanto, considerando os níveis de serviço obtidos com o SLA 1, no primeiro ano de análise este manteve a confiabilidade em 80%. Sendo assim, o SLA 1 utilizado na

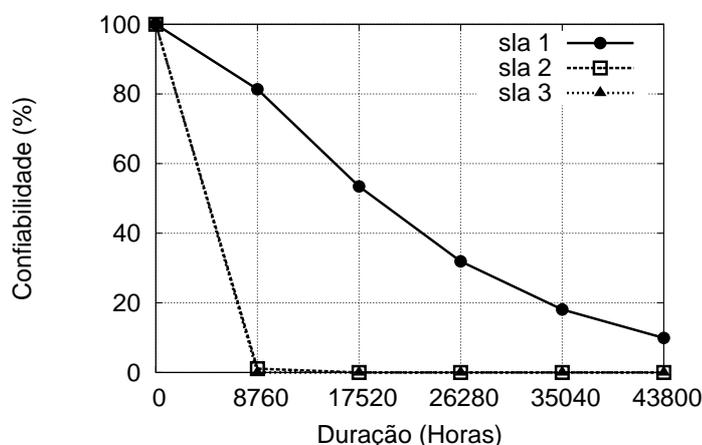
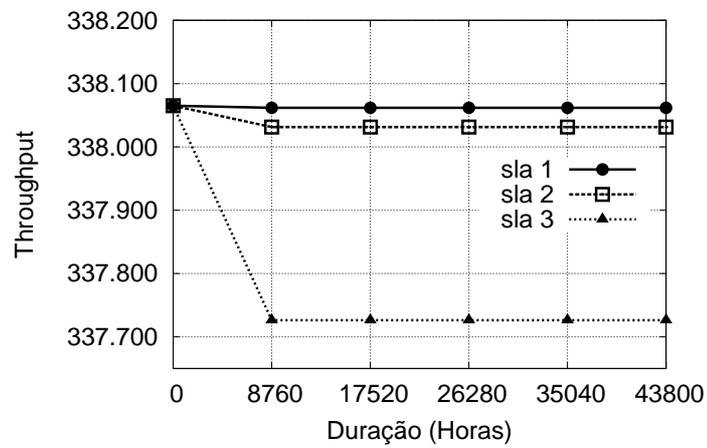
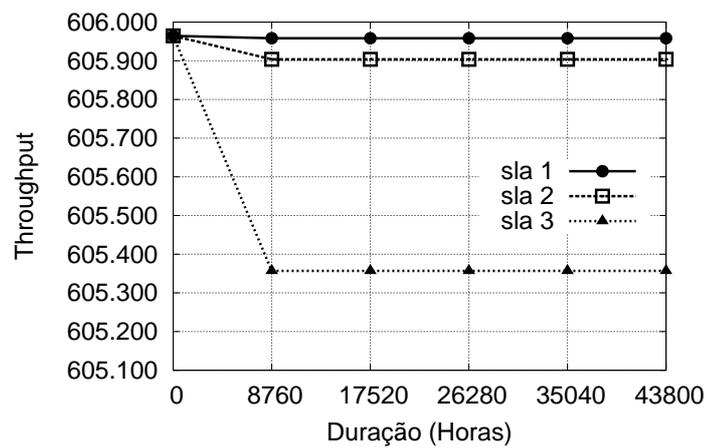


Figura 5.15 Confiabilidade do sistema TEF.

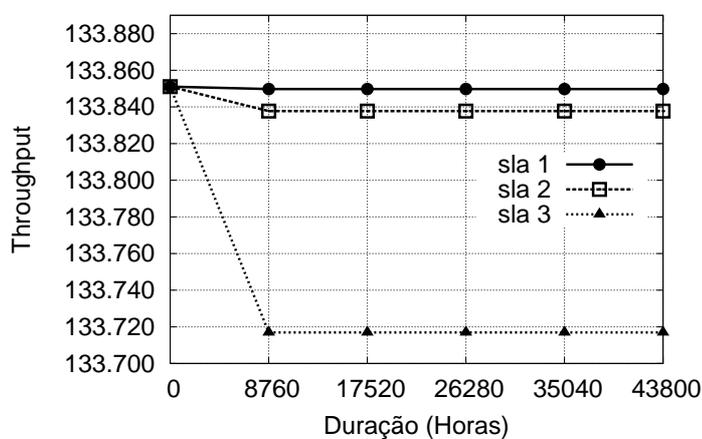
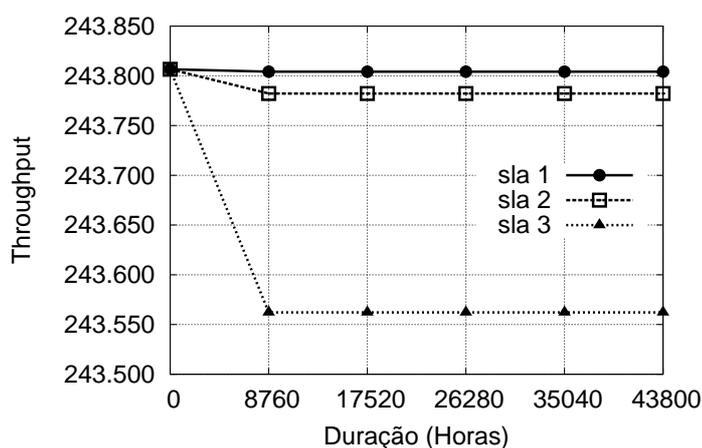
redundância apresentou os maiores níveis de confiabilidade em relação aos SLA 2 e 3, pois estes possuem MTBFs menores. Portanto, com o modelo, o avaliador será capaz de criar diversas situações com o intuito de alcançar SLAs desejáveis.

As análises de *performability* foram conduzidas através dos modelos de dependabilidade (Ver Figura 5.13) e de desempenho (Ver Figura 5.4). Os resultados obtidos por estes foram combinados com o intuito de estimar os níveis de degradação do desempenho do sistema TEF. As Figuras 5.16(a) e 5.16(b) apresentam os efeitos de degradação do desempenho do processador dada a ocorrência de falhas. Para esta análise, foram utilizados apenas o maior e o menor nível crítico de *throughput* obtidos do primeiro cenário (ver Figura 5.10). As Figuras 5.16(a) e 5.16(b) mostram como os níveis de desempenho decrescem no primeiro ano, mas que, em seguida, esses níveis entram em estado estacionário. Essa mudança de comportamento foi devida à ativação do módulo reserva e à posterior recuperação do sistema. Assim, devido a ativação desse mecanismo, o nível de disponibilidade foi mantido em equilíbrio. Similarmente, as Figuras 5.17(a) e 5.17(b), ilustram os resultados de degradação do desempenho no recurso disco. O tempo de avaliação considerado nesse cenário foi de cinco anos, ou seja, 43.800 horas.

Sistemas de pagamento eletrônico têm demonstrado um extraordinário crescimento durante a última década. Neste sentido, prover serviços com níveis de satisfação rigorosos, alta disponibilidade, confiabilidade, segurança, escalabilidade a um custo adequado são aspectos fundamentais para o lucro das empresas. A proposta de modelagem através desse método apresentou resultados representativos para a previsão do comportamento do sistema TEF em termos de desempenho, disponibilidade e confiabilidade. É importante

(a) Mínimo valor de *throughput*.(b) Máximo valor de *throughput*.**Figura 5.16** Resultados de *Performability* (Processador TEF).

destacar que a avaliação desses cenários sem a utilização dos modelos probabilísticos seria uma tarefa complexa, dispendiosa financeiramente e custosa.

(a) Mínimo valor de *throughput*.(b) Máximo valor de *throughput*.**Figura 5.17** Resultados de *Performability* (Disco TEF).

5.8 CONSIDERAÇÕES FINAIS

Este capítulo apresentou os resultados obtidos na realização dos estudos de caso. Através dos estudos aqui apresentados, foi possível não somente validar o modelo GSPN proposto por este trabalho, como também validar a metodologia de avaliação. Com os resultados dos experimentos, foi possível também avaliar o sistema TEF em termos de desempenho, *throughput*, disponibilidade, confiabilidade e *performability* através dos cenários propostos.

CONCLUSÃO E TRABALHOS FUTUROS

Com o aumento da complexidade e diversidade dos sistemas computacionais aliados à grande competitividade do mercado, cada vez mais as empresas estão se tornando dependentes das transações eletrônicas. Nesse sentido, as empresas buscam cada vez mais alcançar uma maior produtividade por menores custos e tempo possíveis. Por isso, faz-se necessária a análise de sistemas de pagamento eletrônico, visto que, no atual cenário competitivo, as falhas e a indisponibilidade desses sistemas podem resultar no insucesso da empresa e até mesmo resultar em perdas de grandes quantias financeiras. Assim, a utilização de métodos formais para análise de desempenho é uma tarefa fundamental nos dias atuais, pois os modelos formais são matematicamente fundamentados e permitem que avaliações tanto qualitativas como quantitativas possam ser realizadas. Portanto, modelos de desempenho são bastante úteis para o entendimento e previsão do comportamento de sistemas de transferência eletrônica de fundos.

O trabalho em comento apresentou uma abordagem de sistemas de transferência eletrônica de fundos baseada em *Generalized Stochastic Petri nets* (GSPN), com o objetivo de realizar avaliações de desempenho nos recursos computacionais. Além disso, uma metodologia foi apresentada com o intuito de auxiliar no processo de avaliação de desempenho do sistema TEF. Essa metodologia é composta por uma série de passos que envolvem desde o entendimento do ambiente, modelagem, até a geração e análise do modelo GSPN. Com a utilização das GSPNs como ferramenta de modelagem e análise, é possível aferir métricas de maneira probabilística. Por exemplo, pode-se encontrar a probabilidade de um recurso computacional estar processando transações, indicando assim o seu nível de utilização.

Com o intuito de validar tanto o modelo GSPN proposto, quanto a metodologia de avaliação, foram realizados três estudos de caso. O primeiro estudo teve o objetivo de identificar possíveis pontos de saturação nos recursos computacionais (processador e disco) do sistema TEF. O segundo estudo de caso avaliou o impacto do desempenho causado pelo envio de transações comerciais em tráfego de rajada, encontrando, assim, níveis de utilização para uma mesma frequência de envio das transações comerciais. Por fim, o último estudo de caso analisou a degradação do desempenho do sistema TEF

considerando diferentes SLAs e níveis de falha. Os resultados obtidos através dos modelos GSPNs demonstram a grande aplicabilidade em análise de desempenho de sistemas TEF, visto que a avaliação desses cenários sem a utilização dos modelos seria uma tarefa complexa, dispendiosa financeiramente e custosa.

6.1 CONTRIBUIÇÕES, LIMITAÇÕES E DIFICULDADES

As contribuições deste trabalho são as seguintes:

- Proposição de modelos GSPNs para representar o funcionamento do processamento e armazenamento das transações comerciais do sistema TEF. Através desses modelos, foi possível aferir métricas de desempenho como, por exemplo, a utilização, o *throughput*, a disponibilidade e a confiabilidade do sistema TEF.
- Desenvolvimento de uma metodologia para auxiliar o processo de avaliação do modelo de desempenho de sistemas TEF. Essa metodologia é composta por uma série de etapas que envolvem desde o entendimento do ambiente até a geração e a análise dos modelos GSPNs. Com a aplicação da metodologia proposta, vários problemas relacionados à modelagem de sistemas TEF poderão ser solucionados;
- Planejamento de capacidade da carga suportada pelos recursos processador e disco do sistema TEF. Com o modelo proposto, é possível prever quais os níveis de utilização e o comportamento destes recursos quando submetidos a uma determinada demanda de transações. Através dos resultados obtidos, o avaliador poderá tomar algumas medidas para melhorar o sistema. Atualização ou substituição de componentes podem ser utilizadas como medidas de prevenção;
- Publicação de artigo em conferência internacional Qualis B referente ao primeiro modelo que gerou esta dissertação [SAM⁺09]; Publicação de artigo em conferência internacional Qualis B referente ao primeiro e segundo estudo de caso apresentados nesta dissertação [ASM⁺09] e Submissão de artigo em um jornal internacional Qualis A referente aos três estudos de caso apresentados.

As limitações deste trabalho são as seguintes:

- A medição do ambiente TEF é realizada de forma manual, portanto, falhas podem ocorrer na configuração do ambiente e na medição, e conseqüentemente, erros podem ser inclusos nos resultados obtidos;

- Alguns componentes foram omitidos do modelo, tais como memória e rede. Isso ocorreu devido à complexidade da modelagem e por serem recursos que não são considerados críticos para análise em particular.

As dificuldades deste trabalho são as seguintes:

- Dificuldade em encontrar trabalhos relacionados à avaliação de desempenho de sistemas TEF. Essa limitação é devida a esse tipo de sistema ser tido como sigiloso, tendo em vista a tamanha dificuldade em localizar material bibliográfico. Conseqüentemente, poucas pesquisas na área têm sido desenvolvidas ou divulgadas;
- Escassez de ferramentas *open source* para medição do ambiente real;
- Na fase de validação do modelo de desempenho refinado, foram necessários diversos ajustes nos parâmetros do modelo até que os erros em relação aos dados medidos fossem aceitáveis. Sendo assim, custoso essa fase;
- Ausência de algumas métricas na ferramenta de medição utilizada. Neste sentido, foi necessário utilizar a estratégia de medição indireta para obter por exemplo a métrica tempo de serviço.

6.2 TRABALHOS FUTUROS

Outros estudos podem ser produzidos através do modelo proposto e da metodologia de avaliação propostos. A seguir, alguns deles são apresentados:

- Considerar outros modelos de redundância para cada recurso computacional (processador e disco), fazendo um estudo comparativo de custo e benefício;
- Um aspecto importante que pode ser considerado no modelo diz respeito a um mecanismo que represente o descarte de transações excedentes. Em outras palavras, caso a fila alcance um número de transações acima da capacidade determinada, estas transações excedentes seriam descartadas. Assim, é possível agregar novas métricas no modelo, como por exemplo, o número de transações descartadas;
- Contemplar outros componentes computacionais levando em considerações outros cenários.
- Criar um mecanismo para classificar o *trace* de cargas, com o intuito de identificar e agrupar os tipos de transações comerciais. Dessa forma, com a classificação da carga será possível criar modelos que representem diferentes tipos de negócios;

- Implementar uma ferramenta que auxilie o processo de avaliação de desempenho dos dados medidos e na geração dos modelos GSPN. Posto isso, após o processo de medição com a ferramenta, será possível organizar e tratar os dados medidos de forma mais eficiente. O tempo na análise dos dados será reduzido, como também na fase de refinamento dos modelos.

REFERÊNCIAS BIBLIOGRÁFICAS

- [ABE09] ABECS. Associação brasileira das empresas de cartões de crédito e serviços. <http://www.abecs.org.br>. Último acesso em 01 de Julho de 2009., 2009.
- [ABR09] ABRAS. Associação brasileira de supermercados. <http://www.abrasnet.com.br>. Último acesso em 10 de Março de 2009., 2009.
- [Alb00] A.L. Albertin. *Comércio eletrônico modelo, aspectos e contribuições de sua aplicação*. Atlas, 2000.
- [ALRL04] A. Avizienis, J.C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on dependable and secure computing*, 1(1):11–33, 2004.
- [AMC⁺09] Ermeson Andrade, Paulo Maciel, Gustavo Callou, Bruno Nogueira, and Carlos Araújo. Mapping uml sequence diagram to time petri net for requirement validation of embedded real-time systems with energy constraints. In *SAC '09: Proceedings of the 2009 ACM symposium on Applied Computing*, pages 377–381, New York, NY, USA, 2009. ACM.
- [AME] AMEX. American express. <https://www.americanexpress.com>. Último acesso em 20 de Junho de 2009.
- [ASM⁺09] Carlos Araújo, Erica Sousa, Paulo Maciel, Fábio Chicout, and Ermeson Andrade. Performance Modeling for Evaluation and Planning of Electronic Funds Transfer Systems with Bursty Arrival Traffic. In *INTENSIVE '09: Proceedings of the 2009 First International Conference on Intensive Applications and Services*, pages 65–70. IEEE Proceedings of INTENSIVE, Valência, April 2009.
- [Avi85] A. Avizienis. The N-version approach to fault-tolerant software. *IEEE Transactions on software engineering*, pages 1491–1501, 1985.

- [Bal01] G. Balbo. Introduction to Stochastic Petri Nets. *Lectures on Formal Methods and Performance Analysis: First EEF/Euro Summer School on Trends in Computer Science, Berg en Dal, The Netherlands, July 3-7, 2000: Revised Lectures*, 2001.
- [Bat03] J. Battisti. *Windows Server 2003: Complete Course*. Axcel Books, 2003.
- [BCB09] BCB. Banco central do brasil. <http://www.bcb.gov.br>. Último acesso em 12 de Abril de 2009., 2009.
- [BGdMT06] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi. *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. John Wiley and Sons, New York, NY, USA, 2 edition, 2006.
- [Bid02] H. Bidgoli. *Electronic commerce: principles and practice*. Academic Press, 2002.
- [BRB00] Iain MacRury Barry Richards and Jackie Botterill. *The Dynamics of Advertising*. Routledge, 2000.
- [C.08] Gomez C. Financial markets, institutions & financial services. 2008.
- [CAS99] C.G. CASSANDRAS. LAFORTUNE; Stéphane;”Introduction to Discrete Event Systems”, book, 1999.
- [Clu] Diners Club. Diners club internacinal. <http://www.diners.com.br>. Último acesso em 20 de Junho de 2009.
- [CM00] G. Ciardo and A.S. Miner. Structural approaches for SPN analysis. *High Performance Computing 2000, Grand Challenges in Computer Simulation*, pages 345–356, 2000.
- [Cos02] F.J.C.L. Costa. *Introdução à Administração de Materiais em Sistemas Informatizados*. Fábio Costa, 2002.
- [CU07] M.L. Catalan and H. Umeno. VM-based Benchmark and Analysis System for Testing Online Transaction Processing. In *Proceedings of the Second International Conference on Innovative Computing, Informatio and Control*. IEEE Computer Society Washington, DC, USA, 2007.

- [DAJ95] A.A. Desrochers and R.Y. Al-Jaar. *Applications of Petri Nets in Manufacturing Systems: Modeling, Control, and Performance Analysis*. IEEE Press, 1995.
- [Fri05] M. Friedman. *Microsoft Windows Server 2003 Performance Guide*. Microsoft Press Redmond, WA, US, 2005.
- [fS09] International Organization for Standardization. Iso 8583-1:2003. financial transaction card originated messages. interchange message specifications. <http://www.iso.org>, 2009.
- [GDI09] N. Gharbi, C. Dutheillet, and M. Ioualalen. Colored stochastic Petri nets for modelling and analysis of multiclass retrial systems. *Mathematical and Computer Modelling*, 2009.
- [GKZH95] R. German, C. Kelling, A. Zimmermann, and G. Hommel. TimeNET-a toolkit for evaluating non-Markovian stochastic Petrinets. In *Petri Nets and Performance Models, 1995., Proceedings of the Sixth International Workshop on*, pages 210–211, 1995.
- [GM02] J.C. Geffroy and G. Motet. *Design of dependable computing systems*. Kluwer Academic Publishers Hingham, MA, USA, 2002.
- [Gra97] B. Grahlmann. The PEP tool. *Lecture Notes in Computer Science*, 1254:440–443, 1997.
- [GS99] J.K. Galbraith and J.K.G.A.Z. Sanvicente. *Moeda: De onde veio para onde foi*. Cengage Learning Editores, 1899.
- [GV03] C. Girault and R. Valk. *Petri nets for systems engineering: a guide to modeling, verification, and applications*. Springer, 2003.
- [Ho92] Y.C. Ho. *Discrete event dynamic systems: analyzing complexity and performance in the modern world*. Institute of Electrical and Electronics Engineers, 1992.
- [ita] Itaotec. <http://www.itaotec.com.br>. Último acesso em 20 de Março de 2009.
- [Ita99] Itaotec. *Guia de referência para o Scope - Solução para pagamento Eletrônico*. 1999.

- [Ita09] Laboratório Itaotec. Laboratório de análise de performance cin/ufpe. <http://www.itaotec.cin.ufpe.br>. Último acesso em 20 de Março de 2009., 2009.
- [Jai91] R. Jain. *The art of computer systems performance analysis*. John Wiley & Sons New York, 1991.
- [Jan98] C.V. Janousek. *Modelling Objects by Petri Nets*. PhD thesis, PhD thesis, Department of Computer Science and Engineering, Technical University of Brno, Czech Republic,(In Czech). CiteSeer. IST-Copyright Penn State and NEC, 1998.
- [JE06] L.K. John and L. Eeckhout. *Performance evaluation and benchmarking*. CRC Press, 2006.
- [Jen91] K. Jensen. Coloured Petri Nets: A High Level Language for System Design and Analysis. *High-level Petri nets: theory and application*, page 44, 1991.
- [KA80] A. Kumar and M. Agarwal. A review of standby redundant systems. *IEEE Trans. Reliab*, 27(4):290–4, 1980.
- [KB06] Samuel Kounev and Alejandro Buchmann. Simqpn: a tool and methodology for analyzing queueing petri net models by means of simulation. *Performance Evaluation*, 63(4):364–394, 2006.
- [KH80] M. Kameyama and T. Higuchi. Design of dependent-failure-tolerant micro-computer system using triple-modular redundancy. *IEEE Transactions on Computers*, 100(29):202–206, 1980.
- [KJ08] Engel J. Kocak, T. and Buboltz J. Characterization of Server Performance Bottlenecks in Distributed Interactive Simulation Environments. *Simulation Modeling Practices and Theory*, 2008.
- [Kla03] P. Klaffke. Spree: A cultural history of shopping. 2003.
- [Kot99] P. Kotler. Marketing para o século XXI. *São Paulo: Futura*, page 305, 1999.
- [Lil00] D.J. Lilja. *Measuring Computer Performance: A Practitioner’s Guide*. Cambridge University Press, 2000.

- [MA98] D.A. Menascé and V.A.F. Almeida. *Capacity Planning for Web Performance: Metrics, Models, and Methods*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1998.
- [MA05] D.A. Menascé and V.A.F. Almeida. *Performance by Design: Computer Capacity Planning by Example*. Prentice Hall PTR, 2005.
- [MAD94] D.A. Menascé, V.A.F. Almeida, and L.W. Dowdy. *Capacity Planning and Performance Modeling: from Mainframes to Client-Server Systems*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1994.
- [MAFM99] D.A. Menascé, V.A.F. Almeida, R. Fonseca, and M.A. Mendes. A Methodology for Workload Characterization of E-Commerce Sites. *Proceedings of the 1st ACM Conference on Electronic Commerce*, pages 119–128, 1999.
- [Mar89] M.A. Marsan. Stochastic petri nets: an elementary introduction. *Advances in Petri Nets*, 424:1–29, 1989.
- [Mar03] T. Maria. Tratamento contábil das operações com cartões de crédito nas instituições financeiras. 2003.
- [Mas] MasterCard. Mastercard worldwide. <http://www.mastercard.com>. Último acesso em 20 de Junho de 2009.
- [MBC⁺95] M. A. Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. John Wiley and Sons, 1995.
- [MCB84] M.A. Marsan, G. Conte, and G. Balbo. A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. *ACM Transactions on Computer Systems (TOCS)*, 2(2):93–122, 1984.
- [MFCH76] P. Merlin, D. Farber, I.B.M.T.J.W.R. Center, and Y. Heights. Recoverability of communication protocols—implications of a theoretical study. *Communications, IEEE Transactions on [legacy, pre-1988]*, 24(9):1036–1043, 1976.
- [MLC96] P. Maciel, R. Lins, and P. Cunha. *Introdução às Redes de Petri e Aplicações*. Campinas - SP: Sociedade Brasileira de Computação, 1996.

- [MMS⁺09] M. Marwah, P. Maciel, A. Shah, R. Sharma, T. Christian, V. Almeida, C. Araújo, E. Souza, G. Callou, B. Silva, Galdino S., and Pires J. Quantifying the sustainability impact of data center availability. In *The First Workshop on GreenMetrics 2009, (SIGMETRICS/PERFORMANCE)*. ACM Proceeding of SIGMETRICS/PERFORMANCE, Seattle, 2009.
- [MR93] M. Malhotra and A. Reibman. Selecting and implementing phase approximations for semi-Markov models. *Communications in statistics. Stochastic models*, 9(4):473–506, 1993.
- [Mur89] T. Murata. Petri nets: Properties, Analysis and Applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.
- [O’b01] J.A. O’Brien. Sistemas de informação e as decisões gerenciais na era da Internet. *São Paulo: Saraiva*, pages 02–03, 2001.
- [Pet62] C. A. Petri. *Kommunikation mit Automaten*. PhD thesis, Bonn: Institut für Instrumentelle Mathematik, Schriften des IIM Nr. 2, 1962.
- [Pra96] D.K. Pradhan. *Fault-tolerant computer system design*. 1996.
- [PSST06] G. Pacifici, W. Segmuller, M. Spreitzer, and A. Tantawi. Dynamic Estimation of CPU Demand of Web Traffic. *Proceedings of the 1st International Conference on Performance Evaluation Methodologies and Tools*, 2006.
- [Red] RedeCard. Redecard. <https://www.redecard.com.br>. Último acesso em 20 de Junho de 2009.
- [RX94] B. Randell and J. Xu. The evolution of the recovery block concept. *Software Fault Tolerance*, pages 1–22, 1994.
- [SAM⁺09] Erica Sousa, Carlos Araujo, Paulo Maciel, Gabriel Alves, and Fábio Chicout. Performance Modeling for Evaluation and Planning of Electronic Funds Transfer Systems. In *The First IEEE Workshop on Performance evaluation of communications in distributed systems and Web based service architectures, (ISCC)*. IEEE Proceeding of ISCC, Sousse, 2009.
- [SBH06] R.JL Sastre, SM Bascon, and F.JL Herrero. New electronic funds transfer services over IP. In *IEEE Mediterranean Electrotechnical Conference, 2006. MELECON 2006*, pages 733–736, 2006.

- [SMA09] Erica Sousa, Paulo Maciel, and Carlos Araujo. Performability Evaluation of EFT Systems using Exponential Stochastic Models. In *International Conference on Systems, Man, and Cybernetics, (SMC)*. IEEE Proceeding of SMC, Texas, 2009.
- [SMAC09] Erica Sousa, Paulo Maciel, Carlos Araujo, and Fábio Chicout. Performability Evaluation EFT Systems for SLA Assurance. In *Fifth International Workshop on System Management Techniques, Processes, and Services (SMTPS)*. IEEE Proceeding of SMTPS, Roma, 2009.
- [SS98] D.P. Siewiorek and R.S. Swarz. *Reliable computer systems: design and evaluation*. AK Peters, Ltd., 1998.
- [Ste] WJ Stewart. Introduction to the numerical solution of Markov chains. 1994.
- [Tec] Tecban. Tecnologia bancária s.a. <http://www.tecban.com.br>. Último acesso em 20 de Junho de 2009.
- [THGF96] K.S. Trivedi, S. Hunter, S. Garg, and R. Fricks. Reliability analysis techniques explored through a communication network example. In *International Workshop on Computer-Aided Design, Test, and Evaluation for Dependability*. Citeseer, 1996.
- [Tri05] M.F. Triola. *Introdução à estatística. (Tradução Vera Regina de Farias e Flores; revisão técnica Ana Maria Lima de Farias e Flores)*. Livros técnicos e Científicos, 2005.
- [Tri06] K.S. Trivedi. *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*. John Wiley and Sons, 2nd Edition, 2006.
- [Visa] Visa. Soluções tef. <http://www.visanet.com.br>. Último acesso em 06 de Junho de 2009.
- [Visb] Visa. Visa inc. <http://www.visa.com.br>. Último acesso em 20 de Junho de 2009.
- [Web06] M. Weber. *História geral da economia*. Centauro, 2006.
- [WeS98] W. Weiszflog and A.P. e Silva. *Michaelis: moderno dicionário da língua portuguesa*. Melhoramentos, 1998.

- [Zim01] A. Zimmerman. TimeNET 3.0 User Manual. A Software Tool for the Performability Evaluation with Stochastic Petri Nets. *Performance Evaluation Group, TU Berlin*, 2001.

