Universidade Federal de Pernambuco
Centro de Informática

Mestrado em Ciências da Computação

AVALIAÇÃO DE PERFORMABILIDADE DE UM CALL CENTER DE EMERGÊNCIA

Marcus Auréllio de Queiroz Vieira Lima

DISSERTAÇÃO DE MESTRADO

Recife
Setembro de 2012

Universidade Federal de Pernambuco
Centro de Informática

Marcus Auréllio de Queiroz Vieira Lima

AVALIAÇÃO DE PERFORMABILIDADE DE UM CALL CENTER DE EMERGÊNCIA

Trabalho apresentado ao Programa de Mestrado em Ciências da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: *Prof. Dr. Paulo Romero Martins Maciel*

Recife
Setembro de 2012

Catalogação na fonte
Bibliotecária Jane Souto Maior, CRB4-571

Lima, Marcus Auréllio de Queiroz Vieira
Avaliação de performabilidade de um call center de emergência / Marcus Auréllio de Queiroz Vieira Lima. - Recife: O Autor, 2012.
xiii, 62 folhas: il., fig., tab.

Orientador: Paulo Romero Martins Maciel.
Dissertação (mestrado) - Universidade Federal de Pernambuco. CIn, Ciência da Computação, 2012.

Inclui bibliografia e apêndice.

1. Avaliação de desempenho. 2. Disponibilidade. 3. Redes de Petri. I. Maciel, Paulo Romero Martins (orientador). II. Título.

004.029 CDD (23. ed.) MEI2012 – 199

Dissertação de Mestrado apresentada por **Marcus Auréllio de Queiroz Vieira Lima** à Pós Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título "**Avaliação de Performabilidade de um Call Center de Emergência**" orientada pelo **Prof. Paulo Romero Martins Maciel** e aprovada pela Banca Examinadora formada pelos professores:

Prof. Ricardo Massa Ferreira Lima
Centro de Informática / UFPE

Prof. Francisco Vieira de Souza
Departamento de Informática e Estatística / UFPI

Prof. Paulo Romero Martins Maciel
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 13 de setembro de 2012.

Prof. Nelson Souto Rosa
Coordenador da Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco.

A Deus.
À Minha Família.
Aos Meus Amigos.
Ao Prof. Dr. Paulo Romero Martins Maciel, orientador.

AGRADECIMENTOS

Gostaria de agradecer a todos que contribuíram para o desenvolvimento deste trabalho. Ao professor Paulo Maciel, pela orientação, apoio e enorme paciência. Também gostaria de agradecer-lhe por todas as oportunidades de crescimento acadêmico e pessoal. Aos professores Ricardo Massa Ferreira e Francisco Vieira de Souza por terem aceitado o convite para compor a banca de defesa do mestrado. A todos do grupo MoDCS (*Modeling of Distributed and Concurrent Systems*), em especial a Almir Guimarães, Bruno Silva e Alysson Barros Silva pela contribuição para a realização deste trabalho. Agradeço também a equipe do call center estudado por toda colaboração com esta pesquisa. Agradeço a BR Voice, especialmente Fernando Arminante, pelo o incentivo recebido. Aos meus amigos, principalmente Jorge Rodrigues, Carlos Frederico, Cristiano Bertolini, Igor Galdino, Fernando Oliveira. Gostaria de agradecer à minha família, em especial à minha mãe, Telma Alves, à minha tia Tânia Alves, aos meus irmãos Alexandre e Katharine, à Jessyka Flavyanne pelo incentivo, carinho e amor nos momentos mais difíceis desta jornada. Agradeço, principalmente, a Deus, que colocou todas essas pessoas em meu caminho.

Se você pensar que pode ou que não pode, de qualquer forma, você estará certo.

—HENRY FORD

RESUMO

Call centers de emergência, geralmente, servem uma entidade pública como, por exemplo, uma cidade, município ou estado e prestam serviços às pessoas, a fim de ajudá-las em situações críticas. Sendo assim, a população tem acesso ao corpo de bombeiros, polícia civil e militar a partir de uma ligação de qualquer telefone. Tendo em vista que estes sistemas atendem pessoas nas circunstâncias mais extremas, eles devem ser altamente confiáveis. A disponibilidade e desempenho são aspectos-chaves em centros de chamadas de emergência. No ponto de vista de desempenho, os resultados das avaliações realizadas pelos projetistas tendem a ser otimistas, uma vez que o comportamento de falha e processos de reparo do sistema são ignorados. A abordagem em análise de performabilidade é de vital importância porque, em um sistema como esse, a degradação de desempenho, na presença de falhas, provoca interrupções. O tempo de inatividade (*downtime*) desses centros refere-se a um período de tempo em que estes serviços não fornecem sua principal função. Modelos de arquitetura, políticas de serviços e redundâncias são atributos para definir e avaliar o funcionamento global de tais sistemas. A região nordeste do Brasil tem indústrias que têm atraído muitos profissionais e contempla também eventos importantes. O aumento no fluxo estimado de pessoas ao longo dos anos requer expansões cuidadosamente planejadas e aperfeiçoamento no sistema de chamada de emergência, a fim de lidar com a demanda esperada e aumentar os níveis de performabilidade necessários. Este trabalho apresenta um modelo hierárquico, heterogêneo, para a avaliação de performabilidade e custo de *call center* de emergência, estuda um *call center*, localizado em uma grande cidade no Brasil, analisa o impacto das chamadas perdidas relacionadas ao tempo de inatividade, descartes e inclui aspectos como a importância para confiabilidade e custo. Os resultados obtidos neste trabalho podem ser usados para auxiliar nas decisões sobre intervenções em um *call center* e melhorar a sua performabilidade. O modelo também pode ser utilizado para outras classes de centros de emergência.

Palavras-chave: Avaliação de Performabilidade, Avaliação de Desempenho, Avaliação de Disponibilidade, Redes de Petri Estocásticas, *Call Center*.

ABSTRACT

Emergency call centers generally serve a public entity such as a city, county or state and provide services to people in order to help them in critical situations. Thus, the population has access to the fire department, civil and military police from a call from any telephone. Considering that these systems attend people in the most extreme circumstances, they must be highly reliable. The availability and performance are key aspects in emergency call centers. In the point of view of performance, the results of the evaluations performed by the designers tend to be optimistic since the behavior of failure and repair processes of the system are ignored. The approach in performability analysis is of vital importance because, in a such system, the performance degradation, in the presence of failures, causes outages. The inactivity time (downtime) of these centers refers to a period of time that these services do not provide their primary function. Architectural models, services policies and redundancy are attributes to define and evaluate the overall functioning of such systems. The northeastern region of Brazil has industries that have attracted many professionals and also contemplates important events. The estimated increase in the flow of people over the years requires carefully planned expansions and improvements in the emergency call system in order to handle the expected demand and increase levels of performability required. This work presents a hierarchical model, heterogeneous, to evaluate the performability and cost of emergency call center, it studies a call center, located in a large city in Brazil, it analyzes the impact of missed calls related to downtime, discards and includes aspects such as the reliability-importance and cost. The results obtained in this work can be used to support decisions about interventions in a call center and improve its performability. The model can also be used for other classes of emergency centers.

Keywords: Performability Evaluation, Performance Evaluation, Availability Evaluation, Stochastic Petri Nets, Call Center.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

ASTRO - *The Amazing Stochastic Petri Net/RBD Evaluator Tool.*

CO - *Cost of Ownership.*

CTMC - *Continuous Time Markov Chains*

FIFO - *First In, First Out.*

GSPN - *Generalized Stochastic Petri Nets*

HCPN - *Hierarchical Coloured Petri Nets*

FSPN - *Fluid Stochastic Petri Nets*

eDSPN - *Extended Deterministic and Stochastic Petri Nets*

MTTF - *Mean Time to Failure.*

MTTR - *Mean Time to Repair.*

QoS - *Quality of Services.*

RBD - *Reliability Block Diagram.*

RCI - *Reliability-Cost Importance.*

RI - *Reliability Importance.*

SLA - *Service Level Agreement.*

SPN - *Stochastic Petri Nets.*

TimeNET - *Timed Net Evalution Tool.*

UA - *Unavailability.*

UPS - *Uninterruptible Power Supply*

PABX - *Private Automatic Branch Exchange*

PSTN - *Public Switched Telephone Network*

CAPÍTULO 1

INTRODUÇÃO

Neste capítulo, são apresentadas as principais motivações e justificativas para realização desta Dissertação. Em seguida, são definidos os objetivos e os trabalhos relacionados. Finalmente, é descrita a estrutura do trabalho.

1.1 MOTIVAÇÃO E JUSTIFICATIVA

Call centers de emergência geralmente servem uma entidade pública, como uma cidade, município ou estado. Os serviços prestados são oferecidos à população, a fim de ajudá-la em situações críticas. Através de uma ligação para esses centros de emergência, uma pessoa pode ter acesso aos serviços do corpo de bombeiros, polícia civil e militar a partir de qualquer telefone.

Atualmente, um *call center* de emergência pode ser definido como um conjunto de subsistemas, tais como estrutura de energia, rede de dados, telecomunicações e de serviços de atendimento ao cliente, constituindo uma arquitetura computacional com componentes relacionados. O período de interrupção (*downtime*) desses *call centers* de emergência refere-se a um momento em que esses serviços não fornecem sua principal função. Assim, é necessário estabelecer este tempo com base nos recursos disponíveis de cada sistema.

Avaliação de performabilidade de tais sistemas é atividade chave no planejamento de capacidade do sistema. Um sistema bem projetado reduzirá os efeitos de uma falha, pois permite que os projetistas definam onde os investimentos devem ser aplicados.

A região nordeste do Brasil é considerada um dos principais polos de investimentos do país; sendo assim, está repleta de indústrias que têm atraído muitos profissionais e eventos importantes. O aumento no fluxo estimado de pessoas ao longo dos anos requer expansões cuidadosamente planejadas e adaptações do sistema de chamada de emergência, a fim de lidar com a demanda esperada e aumentar os níveis de performabilidade necessários.

É importante salientar que sistemas informatizados e de telecomunicações, do ponto de vista de desempenho, tendem a ser otimistas, uma vez que o comportamento da falha e reparo do sistema são ignorados. É relevante, portanto, considerar o desempenho, disponibilidade, capacidade e custo juntos. Este processo é importante porque, em um *call center* de emergência, a falha de um componente elétrico, *links* ou roteador e a ausência de redundância provocam interrupção parcial ou até total dos serviços providos pelo sistema. Em outras palavras, provoca a diminuição da capacidade e afeta a qualidade de serviço do sistema, bem como o seu desempenho.

Este trabalho apresenta um modelo hierárquico heterogêneo para a avaliação da performabilidade de *call centers* de emergência e analisa o impacto das chamadas perdidas relacionadas ao *downtime* e descartes.

1.2 OBJETIVOS

Este trabalho visa aliar a avaliação de performabilidade e custos de *call center* de emergência e destaca a importância dos componentes em relação ao desempenho do sistema. Almeja-se, assim, ser capaz de projetar sistemas que atendam aos requisitos de desempenho e minimize custos de implantação e operação.

Os objetivos específicos desta Dissertação são:

- Propor um modelo de diagrama de bloco de confiabilidade (*Reliability Block Diagram* - RBD) de Redes de Petri Estocásticas (*Stochastic Petri Nets* - SPN) para análise de performabilidade de *call centers* de emergência;
- Adotar um modelo de custo (*Cost of Ownership* - CO) para análise da eficácia dos gastos de uma organização de *call center* de emergência, uma vez que inclui despesas relacionadas a possuir e manter uma determinada arquitetura;
- Adotar a importância para a confiabilidade e custos (*Reliability-Cost Importance* - RCI) para determinar os componentes mais importantes do sistema, sendo possível relacioná-lo ao custo do sistema (*Cost of Ownership* - CO) e decidir quais componentes devem ser replicados aumentando, assim, diretamente a performabilidade.

1.3 TRABALHOS RELACIONADOS

A qualidade dos serviços (*Quality of Services* - QoS) oferecidos por um *call center* de emergência é um aspecto essencial e deve ser abordada pelos responsáveis pela tomada de decisões e pelos projetistas. Muitos trabalhos foram publicados sobre avaliação de *call centers*, porém poucos avaliaram a performabilidade de *call centers* de serviços de emergência. Os trabalhos descritos abaixo apresentam temática relacionada à desta pesquisa.

Em [Fanaeepour, Naghavian e Azgomi 2007], os autores fazem introdutoriamente uma análise da importância de um *call center* mediante a rápida expansão empresarial nos grandes centros industriais e comerciais, mostrando uma importante elevação na demanda desses serviços e custos associados. Foi realizado um levantamento da estrutura e da tecnologia de serviços de *call centers* considerando aspectos de desempenho e custos, por fim, são apontados alguns modelos e ferramentas disponíveis para a análise desses sistemas.

Os autores propõem um modelo em Redes de Petri Estocásticas Generalizadas (GSPN) para a análise de sistemas de *call centers*, utilizando a ferramenta Sharpe, com o objetivo final de minimizar a carga de trabalho dos agentes, a partir da análise da Qualidade de

Serviço (QoS) e eficiência operacional. Eles defendem que, como vantagens, tal modelo em GSPN permite a identificação completa comportamental qualitativa, bem como a análise do desempenho quantitativo associado aos custos da infraestrutura. Além disso, as alterações dinâmicas de estados podem ser explicitamente descritas por GSPNs, que indicam claramente as transições de estado com relação a disparos de eventos.

O modelo proposto por eles tem como foco o atendimento *self-serving* pelos usuários através de um sistema iterativo e foi analisado com conceitos de Redes de Petri. O trabalho demonstra como a carga de trabalho dos agentes diminui com os mecanismos de *self-serving* e distribuição automática. Contudo, seu trabalho limita-se ao desempenho qualitativo sobre chamadas que chegam, não realizando o estudo do impacto de chamadas por desistência, descartes, erros e ausência de falha e reparo no sistema.

Em [Pichitlamken et al. 2003], é realizada uma modelagem para simulação de *call center* onde foi avaliado o dimensionamento adequado do número de agentes bem como as seguintes medidas de desempenho: utilização dos agentes, taxa de desistência e QoS.

Os autores mencionaram como dificuldades para a validação do mesmo: (i) têm-se o número de entradas (chamadas) bem como a soma dos tempos de serviços das chamadas, porém, não se tem os tempos de chegada ou tempos de serviço de cada chamada e a falta dessas informações dificulta a análise dos dados porque os métodos de estimação de parâmetros geralmente não se aplicam. Além disso, a natureza estocástica das chamadas aumenta essa dificuldade em função da variação das taxas de chegada em relação aos dias da semana bem como ao longo do dia; (ii) para modelar o processo por simulação, faltariam dados como: quanto tempo o cliente está disposto a esperar antes de abandonar a fila? (iii) Quando o tráfego de entrada é baixo e alguns agentes estão ociosos, um discador automático realiza múltiplas chamadas de saída em paralelo (tentando alcançar potenciais clientes, por exemplo, para comercialização ou venda direta), a fim de aumentar a produtividade do centro. A atuação do discador e a comparação do desempenho obtido através da simulação com os valores empíricos podem apresentar discrepâncias em relação às quais não se tem certeza se são decorrentes da modelagem ou da falta de conhecimento em relação ao discador; (iv) tem-se ainda que a qualidade dos serviços (QoS, definida como a fração de clientes de entrada cujo tempo de espera é inferior a 20 segundos) é melhor do que o alvo (80%) e não se sabe se esse resultado é em função de uma resposta excessivamente rápida dos agentes, mesmo sem atender às necessidades do cliente; (v) o tempo em que o discador está disponível, provavelmente, não corresponde ao tempo programado para o atendimento.

O modelo proposto pelos autores tem grande potencial para o atendimento de chamadas do *call center*, mas se limita ao estudo de QoS, ocupação dos agentes, chamadas atendidas e descartadas e não mostra o custo para implantação do sistema. Nessa dissertação, foi realizada a avaliação da performabilidade do *call center* de emergência, compreendendo uma análise do impacto do *downtime* em seu desempenho através do número de chamadas válidas perdidas por ano. Sendo assim, foi possível compreender e aplicar os conceitos do trabalho nessa Dissertação, a qual possui uma abrangência bem maior e mais completa do *call center*.

[Aguir et al. 2004] fazem a representação de um sistema de um grande *call center* através de um modelo de cadeia de Markov de tempo contínuo (CTMC), defendendo que, dessa forma, em um sistema de carga computacional elevada, têm-se uma representação mais precisa. O trabalho foi desenvolvido em função da alta competitividade entre empresas desses serviços, no mercado europeu, em que o aumento no desempenho com a redução dos custos se torna um desafio. Para isso, os autores investigam em que medida ocorre o trabalho da equipe em relação à previsão do tempo de espera do cliente na fila e se é possível extrair as primeiras tentativas de chamadas do número total. Através de estimativas, analisa-se o sistema por meio de simulação, a fim de, por aproximação, obter-se a validação da pesquisa.

Na pesquisa foram coletados dados em intervalos de 2,5 minutos e realizadas 10000 repetições para essa validação. Os autores argumentam que a forma do padrão observado nas chamadas de chegada pode ser qualitativamente diferente das chamadas primárias (primeira tentativa de chamada), o que reforça a importância da modelagem do comportamento de novas tentativas na medida em que esses dados, em um *call center* de grande demanda, pode favorecer a alocação de recursos e de agentes, possibilitando uma melhor empregabilidade de todos esses recursos, garantindo um melhor desempenho. No *call center* de emergência estudado nesta Dissertação, verificou-se também a existência de um alto índice de desistências devido à impaciência dos usuários e alta taxa de descartes provocado pelo sistema, uma vez que o volume de chamadas recebidas é grande. Assim, os usuários realizam uma nova chamada acreditando que serão atendidos rapidamente, o que não ocorre, uma vez que o atendimento é do tipo FIFO.

O modelo proposto por [Aguir et al. 2004] foi utilizado para estimar as taxas reais de chegada baseadas nos dados de demanda onde as novas tentativas de chamadas de uma mesma pessoa não podem ser distinguidas da sua primeira tentativa. Tal característica é comum em *call centers* e desconsiderar esse fenômeno de novas tentativas pode provocar grandes distorções em análises de dimensionamento do número adequado de agentes. Esse problema também foi observado nesta Dissertação que utilizou um modelo de Redes de Petri Estocásticas (*Stochastic Petri Nets* - SPN) que avalia o desempenho do *call center*. Sendo assim, foram analisados o número total de chamadas válidas bem como as chamadas perdidas em função dos descartes se os agentes estiverem ocupados ou se a fila estiver cheia para a avaliação de performabilidade do *call center*.

Em [Arno, Gross e Schuerger 2006], verifica-se uma discussão sobre os conceitos de disponibilidade e confiabilidade. Os autores modelaram a estrutura de fornecimento de energia através de modelo RBD, composto por gerador, de um *call center* de emergência de grande porte. Primeiro, a disponibilidade do sistema foi avaliada tendo em vista que a interrupção, por 10 segundos, do fornecimento de energia, implicaria em um tempo de parada maior que os 10 segundos, uma vez que o reinício de todo o sistema e o *downtime* ocorreria em um tempo bem maior. Em função dos resultados da disponibilidade, os autores destacam que ao remodelar o RBD adicionando um módulo UPS verifica-se que a disponibilidade do sistema aumentou.

Além da avaliação da disponibilidade, os autores analisaram também a confiabilidade do sistema e os resultados mostraram que o sistema do *call center* não é tão confiável como o desejado. Sendo assim, eles ressaltam a importância da eliminação de pontos únicos de falhas para o aumento da confiabilidade, destacando ainda que a modelagem de confiabilidade é bastante eficaz quando usada para a comparação entre sistemas semelhantes. A estrutura de energia avaliada é semelhante a do *call center* de emergência estudado nesta Dissertação bem como a avaliação da disponibilidade através do modelo RBD. Contudo, eles utilizaram apenas a modelagem RBD a qual não consegue lidar com restrições de tempo de módulo ativo como o tempo de acionamento da redundância do gerador de energia. Esta Dissertação utiliza estratégias de modelagem hierárquicas e heterogêneas, composta por RBD e SPN, que são essenciais para representar sistemas com mecanismos de redundância ativos e que consideram esse tempo de acionamento na determinação da disponibilidade do sistema.

Nesta Dissertação, foram utilizados os resultados da avaliação de desempenho de um *call center* de emergência realizada por [Silva 2010] que é mais detalhada, apresentando um modelo SPN que estuda os passos das chamadas de emergência no *call center*, partindo da entrada destas chamadas até o atendimento final e despacho dos veículos de emergência. Esses dados foram utilizados em conjunto com os dados de avaliação de disponibilidade e o custo do sistema, para realizar a avaliação da performabilidade do *call center* de emergência que proporciona à tomada de decisão mais precisa e planejamento pelos projetistas.

1.4 ESTRUTURA DA DISSERTAÇÃO

O Capítulo 1 compreende esta introdução a respeito do tema estudado, apresenta os trabalhos relacionados e os objetivos desta Dissertação.

No Capítulo 2 são abordados os conceitos básicos sobre avaliação de dependabilidade, incluindo informações relacionadas. Além disso, são abordados diagramas de bloco de confiabilidade, compreendendo o conteúdo sobre importância para confiabilidade e custo. Em seguida, é apresentada uma visão geral sobre modelagem hierárquica e desempenho com conceitos sobre Redes de Petri Estocásticas. Por último, são descritos métodos para determinação do custo de propriedade do sistema.

No Capítulo 3 é descrita a estrutura do sistema de *call center* de emergência através de suas características, funcionamento lógico, componentes e suas funções. Em seguida, são discutidos os modelos desenvolvidos para a avaliação da dependabilidade e desempenho.

No Capítulo 4 são apresentados os resultados da principal contribuição desta Dissertação: avaliação da dependabilidade e do desempenho do *call center* de emergência. Em seguida, foram propostas melhorias na arquitetura do sistema com o objetivo de aumentar sua performabilidade. Além disso, foi realizada uma análise do desempenho do sistema ajustando a quantidade de agentes com o intuito de reduzir a quantidade de chamadas descartadas.

O Capítulo 5 compreende as considerações finais sobre o desenvolvimento deste trabalho, assim como as principais contribuições, abrangência e limitações encontradas. São propostos alguns trabalhos futuros que complementam este trabalho e que podem direcionar futuras pesquisas nesta área.

CAPÍTULO 2

FUNDAMENTOS

Neste capítulo são apresentados os conceitos básicos sobre dependabilidade, incluindo seus atributos e informações relacionadas. Além disso, são abordados diagrama de bloco de confiabilidade, compreendendo o conteúdo relacionado a importância para confiabilidade e custo. Em seguida, é apresentada uma visão geral sobre desempenho e conceitos de Redes de Petri Estocásticas. Por último, são descritos métodos para determinação do custo de propriedade do sistema. Todos estes conceitos são fundamentais para o entendimento do formalismo adotado e da pesquisa realizada.

2.1 CONCEITOS BÁSICOS SOBRE DEPENDABILIDADE

Dependabilidade é um conceito genérico que engloba os seguintes atributos: disponibilidade: probabilidade de que o sistema esteja operando corretamente; confiabilidade: probabilidade de que o sistema não falhe até o tempo t, segurança: ausência de consequências catastróficas para o(s) usuário(s) e o ambiente; confidencialidade: ausência de divulgação não autorizada de informações; integridade: ausência de alterações impróprias no estado do sistema; manutenabilidade: capacidade de sofrer reparos e modificações.

A confiabilidade e a disponibilidade podem ser calculadas através de modelos combinatórios, como RBD [Maciel P. e Fernandes 2007], ou dos modelos baseados em estados, por exemplo, SPN [Maciel P. e Fernandes 2007]. O RBD permite representar componentes e estabelecer equações que permitam o cálculo da disponibilidade e da confiabilidade. No entanto, modelos RBD não representam, adequadamente, dependências de falhas e reparação.

Por outro lado, os métodos baseados em estados [Maciel P. e Fernandes 2007] podem representar dependências, permitindo assim a representação de mecanismos complexos redundantes, bem como políticas de manutenção sofisticadas. Alguns desses formalismos permitem tanto a análise numérica quanto a simulação estocástica nos modelos.

Em um sistema, os tipos de componentes, a quantidade e a qualidade destes e a maneira que eles estão dispostos afetam diretamente a sua dependabilidade. A principal meta para o estudo de dependabilidade é definir um modelo que represente os modos de falha e a capacidade de reparo do sistema baseado nos componentes ou subsistemas dos quais o sistema é composto [Corporation 2003].

A confiabilidade é a probabilidade do sistema desempenhar adequadamente o seu propósito específico por um dado período, até que ocorra a primeira falha [Maciel P. e Kim 2011] e [Xie Yuan Shun Dai 2004].

A função para confiabilidade $R(t)$, apresentada na Equação (2.1), é a probabilidade do sistema estar operando corretamente, a partir de um intervalo zero até o tempo t [Maciel P. e Kim 2011], onde T é uma variável aleatória representando o tempo de falha ou tempo para falha.

$$R(t) = P(T > t), t \geq 0 \tag{2.1}$$

A probabilidade de ocorrer falhas, inverso da confiabilidade, é representada pela Equação (2.2), onde T é uma variável aleatória que representa o tempo para ocorrência de falha no sistema.

$$F(t) = 1 - R(t) = P(T \leq t) \tag{2.2}$$

A Equação (2.3) representa a confiabilidade considerando a função de densidade $f(t)$ do tempo para ocorrência de falha no sistema.

$$R(t) = P(T \geq t) = \int_{t}^{\infty} f(t)dt \tag{2.3}$$

O Tempo Médio entre Falhas (*Mean Time to Failure* - MTTF) é o tempo médio para a ocorrência de falhas no sistema. Seja $f(t)$ uma função de densidade de probabilidade. Assim, a confiabilidade (R) é obtida e o MTTF pode ser calculado utilizando a equação abaixo [Kuo e Zuo 2003]:

$$MTTF = \int_{0}^{\infty} R(t)dt = \int_{0}^{\infty} f(t)dt \tag{2.4}$$

A disponibilidade de estado estacionário de um sistema é definida como a fração de tempo que o sistema está disponível para atender às demandas dos usuários. O período em que o sistema não está disponível é chamado de *downtime*; o período em que o sistema está disponível é chamado *uptime* [Maciel P. e Kim 2011]. A disponibilidade pode ser obtida pelo Tempo Médio entre Falhas (MTTF) e Tempo Médio para Reparo (*Mean Time to Repair* - MTTR) através da seguinte equação:

$$Disponibilidade = \frac{MTTF}{MTTF + MTTR} \tag{2.5}$$

Considerando que UA (*downtime*) = 1 - Disponibilidade e a Equação (2.6), a seguinte equação é derivada [Kuo e Zuo 2003]:

$$MTTR = MTTF \times \frac{UA}{Disponibilidade} \tag{2.6}$$

2.2 DIAGRAMA DE BLOCOS DE CONFIABILIDADE

Diagrama de Blocos de Confiabilidade é frequentemente utilizado para descrever a relação entre o funcionamento dos componentes de um sistema [Kuo e Zuo 2003]. O RBD tem sido utilizado para representar estruturas em série, paralelo, série-paralelo, *bridge*, *k-out-of-n* e estruturas de redes de forma geral [Kuo e Zuo 2003]. As estruturas em série e paralelo, apresentadas na Figura 2.1, são as mais comumente utilizadas e a maioria das outras estruturas são derivadas a partir dessas duas.

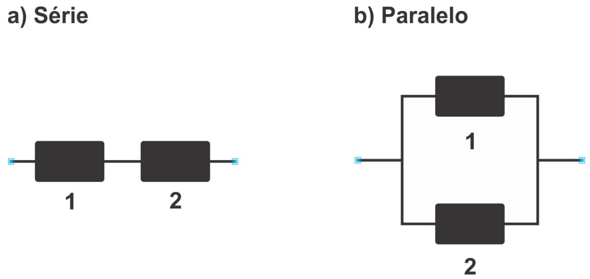


Figura 2.1 Diagrama de Blocos de Confiabilidade.

O RBD é composto por um conjunto de blocos funcionais conectados de acordo com o efeito da falha de cada bloco sobre a confiabilidade do sistema. RBD é um modelo orientado a sucesso. Em RBDs, o estado do sistema é descrito como uma função booleana dos estados dos seus componentes ou subsistemas, onde a função é avaliada como verdadeiro sempre que pelo menos o número mínimo de componentes está operacionalmente habilitado para executar a funcionalidade desejada.

RBDs têm um vértice de origem e um de destino, um conjunto de blocos (normalmente retângulos), onde cada bloco representa um componente, arcos conectando os componentes e os vértices. Graficamente, quando um componente está funcionando, o bloco b_i pode ser substituído por um arco, caso contrário o bloco b_i é removido. O sistema está funcionando adequadamente quando existe pelo menos um caminho do nó de origem para o nó de destino.

Assim, para a estrutura em série apresentada na Figura 2.1(a), se um único componente falhar, o funcionamento de todo o sistema é interrompido. Assumindo um sistema com n componentes, a confiabilidade do sistema é obtida através da Equação (2.7):

$$R_s(t) = \prod_{i=1}^{n} R_i(t) \tag{2.7}$$

onde $R_i(t)$ corresponde a confiabilidade do bloco b_i no instante de tempo t.

Para uma estrutura em paralelo (Figura 2.1(b)), pelo menos um componente deve está operacional para que todo sistema esteja operacional. Levando-se em conta n componentes, a confiabilidade do sistema é obtida através da Equação (2.8):

$$R_s(t) = 1 - \prod_{i=1}^{n}(1 - R_i(t)) \tag{2.8}$$

onde $R_i(t)$ corresponde a confiabilidade do bloco b_i no instante de tempo t.

Os blocos em qualquer estrutura em série ou em paralelo podem ser combinados em um novo bloco com a expressão de confiabilidade da equação acima. Usando essas combinações, qualquer sistema série-paralelo pode ser eventualmente transformado em um bloco. A sua confiabilidade pode ser facilmente calculada usando iterativamente as equações apresentadas [Xie Yuan Shun Dai 2004].

As funções estrutural e lógica são funções que representam a estrutura e permitem conhecer o estado do sistema através de sua soluções. Estas funções são utilizadas para auxiliar e avaliar a dependabilidade dos modelos série e paralelo. Ambas as funções possuem basicamente o mesmo objetivo; fica a critério do projetista adotar uma ou outra. A seguir, são apresentadas as funções.

2.2.1 Função Estrutural

Esta função representa, na forma de uma expressão matemática, a estrutura do sistema. Ela é utilizada para indicar a relação entre o estado do sistema e o estado dos componentes [Kuo e Zuo 2003].

Para obter essa função, o estado de cada componente corresponde a uma variável que pode ter apenas dois valores possíveis. Seja x_i a variável que representa o estado do componente i para $1 \leq i \leq n$, onde n é o número de componentes do sistema. Então o estado x_i do componente i é apresentado por:

$$x_i = \begin{cases} 1, & \text{se o componente } i \text{ está funcionando,} \\ 0, & \text{se o componente } i \text{ está falho.} \end{cases} \tag{2.9}$$

O vetor $x = (x_1, x_2, ..., x_n)$ representa o estado de todos os componentes. O estado do sistema é determinado pelos estados dos componentes. Seja ϕ a variável que representa o estado do sistema como um todo. Esta variável é definida como:

$$\phi = \begin{cases} 1, & \text{se o sistema está funcionando,} \\ 0, & \text{se o sistema está falho.} \end{cases} \tag{2.10}$$

Se os estados dos componentes são conhecidos, então o estado do sistema também pode ser obtido [Kuo e Zuo 2003]. O estado do sistema é uma função determinística dos estados dos componentes. Dessa forma, é possível apresentar a Equação (2.11).

$$\phi = \phi(x) = \phi(x_1, x_2, ..., x_n) \tag{2.11}$$

onde $\phi(x)$ é a função estrutural do sistema. Cada estrutura tem uma única função estrutural $\phi(x)$.

As regras de formação das funções estruturais em série e paralelo são apresentadas, respectivamente, através das Equações (2.12) e (2.13).

Sejam n componentes $x_1, x_2, ..., x_n$ em série, a função estrutural $\phi(x)$ desses componentes é representada por:

$$\phi(x) = \prod_{i=1}^{n} x_i = min\{x_1, x_2, ..., x_n\} \tag{2.12}$$

Sejam n componentes $x_1, x_2, ...x_n$ em paralelo, a função estrutural $\phi(x)$ desses componentes é representada por:

$$\phi(x) = 1 - \prod_{i=1}^{n}(1 - x_i) = max\{x_1, x_2, ..., x_n\} \tag{2.13}$$

2.2.2 Função Lógica

A função lógica define o estado do sistema. Esta função é representada por expressões booleanas. Em algumas situações, simplificar a função estrutural pode não ser uma tarefa fácil. A função lógica pode ser aplicada para simplificar as funções geradas para os sistemas através de álgebra booleana [Maciel P. e Kim 2011]. As notações para a função lógica são apresentadas a seguir:

x_i : evento em que o componente i funciona, $1 \leq i \leq n$, sendo n o número de componentes;

$\bar{x}_i$: complemento de x_i, indicando que o componente i é defeituoso, $1 \leq i \leq n$;

- $S(x_1, x_2, ..., x_n)$: o evento em que o sistema com os componentes $\{1, 2, ..., n\}$ funciona.
- $\bar{S}(x_1, x_2, ..., x_n)$: o complemento de S$(x_1, x_2, ..., x_n)$, indicando o evento em que o sistema com os componentes $\{1, 2, ..., n\}$ é falho.

Segundo [Maciel P. e Kim 2011] as notações, a função lógica de um sistema em série e paralelo com n componentes são apresentadas nas Equações (2.14) e (2.15).

$$S_{serie} = x_1 x_2 ... x_n. \tag{2.14}$$

$$\bar{S}_{paralelo} = \bar{x}_1 \bar{x}_2 ... \bar{x}_n. \tag{2.15}$$

2.2.3 Redução do Diagrama de Blocos de Confiabilidade

Os modelos de diagrama de blocos de confiabilidade podem ser grandes, complexos e de difícil compreensão. Reduzir estes modelos possibilita diminuir a complexidade, diminuindo o tamanho e facilita o entendimento do sistema.

A aplicação de redução (ou simplificação) no modelo (que possui as taxas constantes) permite obter um modelo de alto nível, a partir de um modelo detalhado. Com a redução, é possível trabalhar com modelos mais simples, mantendo as mesmas características (do ponto de vista de dependabilidade) do modelo original.

A redução pode ser aplicada em estruturas série, paralelo ou *bridge*. Assim, é possível reduzir a estrutura paralelo (para um bloco) obtendo uma estrutura em série com dois blocos simples e então aplicar a redução na estrutura série [Kuo e Zuo 2003, Dhillon 2002].

A redução pode ser aplicada sucessivamente até que o modelo se torne apenas um bloco contendo a taxa de falha e a taxa de reparo do sistema. Todas as equações são consideradas para distribuição do tipo exponencial [Maciel P. e Kim 2011].

O cálculo do tempo médio para reparo dos subsistemas utiliza a Equação (2.16). Esta equação pode ser aplicada a estruturas em série, paralelo ou *bridge*.

$$MTTR = \frac{\sum_{k}^{i=1} \lambda_i CMT_i}{\sum_{k}^{i=1} \lambda_i} \tag{2.16}$$

onde k é o número de componentes, λ_i é a taxa de falha do componente i e CMT_i é o tempo requerido para reparo do componente i.

Considerando-se os tempos como sendo exponencialmente distribuídos, o MTTF de uma composição em série é:

$$MTTF_s = \frac{1}{\sum_{i=1}^{n} \lambda_i} \tag{2.17}$$

onde λ_i é a taxa de falha do componente i.

Por outro lado, de acordo com [Kuo e Zuo 2003, Dhillon 2002], o cálculo do tempo médio para falha de composições em paralelo pode ser determinado através de várias equações dependendo da estrutura e dos parâmetros:

- Para sistemas cujos componentes possuem a mesma taxa de falha, a Equação (2.18) é utilizada.

$$MTTF_s = \frac{1}{\lambda} \sum_{i=1}^{n} \frac{1}{i} \tag{2.18}$$

- Caso o subsistema seja composto por apenas dois componentes com taxas de falha diferentes, o MTTF pode ser obtido utilizando a Equação (2.19).

$$MTTF_s = \frac{1}{\lambda_1} + \frac{1}{\lambda_2} + \frac{1}{\lambda_1 + \lambda_2} \tag{2.19}$$

- Quando o subsistema possui somente três componentes com taxas de falha diferentes, o valor do MTTF pode ser calculado a partir da Equação (2.20).

$$MTTF_s = \frac{1}{\lambda_1} + \frac{1}{\lambda_2} + \frac{1}{\lambda_3} - \frac{1}{\lambda_1 + \lambda_2} - \frac{1}{\lambda_1 + \lambda_3} - \frac{1}{\lambda_2 + \lambda_3} + \frac{1}{\lambda_1 + \lambda_2 + \lambda_3} \tag{2.20}$$

- Quando o subsistema possui n componentes connectados em parallel, o MTTF do subsistema é determinando através da confiabilidade do sistema em paralelo no tempo t conforme demonstrado na Equação (2.21).

$$MTTF_s = \int_0^\infty R_s(t)dt = \int_0^\infty [1 - \prod_{i=1}^{n}(1 - e^{-\lambda_i t})]dt \tag{2.21}$$

2.3 REDES DE PETRI ESTOCÁSTICAS

Redes de Petri é uma família de formalismos gráficos para modelagem, análise e simulação de diversos tipos de sistemas [Murata 1989] e [Maciel P. R. M. e Cunha 1996]. Redes de Petri incorporam uma noção de estado local e uma regra para mudança de estado (disparo de transição) que lhes permite capturar tanto as características estáticas quanto dinâmicas de um sistema real [Murata 1989].

A introdução de conceitos de tempo em modelos de Redes de Petri foram propostas mais tarde por [Ramchandani 1974], [Merlin e Farber 1976] e [Sifakis 1977] sob pontos de vista distintos. [Molloy 1981] bem como [Florin e Natkin 1989] propuseram modelos de Redes de Petri nos quais tempos estocásticos foram considerados [Maciel P. R. M. e Cunha 1996]. Os dois últimos trabalhos abrem a possibilidade de relacionar a teoria das Redes de Petri e modelagem estocástica [Molloy 1981] e [Florin e Natkin 1989]. Atualmente, esses modelos, bem como suas extensões são, genericamente, chamados *Stochastic Petri Net* (SPN).

Há muitas maneiras diferentes de representar SPNs. Este trabalho adota uma definição genérica adotada em [German 2000], na qual SPN é uma tupla definida da seguinte forma:

Definição 1. *Uma tupla SPN = (P, T, I, O, H, $\prod$, G, M_o, Atts) é uma Rede de Petri Estocástica, onde:*

- P = $\{p_1, p_2, ..., p_n\}$ é o conjunto de lugares,
- T = $\{t_1, t_2, ..., t_n\}$ é o conjunto de transições imediatas e temporizadas,
- I $\in (\aleph \rightarrow \aleph)^{n \times m}$ é a matriz que representa os arcos de entrada (que podem ser dependentes de marcações),

- O $\in (\aleph \rightarrow \aleph)^{n\times m}$ é a matriz que representa os arcos de saída (que podem ser dependentes de marcações),
- H $\in (\aleph \rightarrow \aleph)^{n\times m}$ é a matriz que representa os arcos inibidores (que podem ser dependentes de marcações),
- $\prod \in \aleph^n$ é um vetor que associa o nível de prioridade a cada transição,
- G $\in (\aleph^n \rightarrow \{true, false\})^m$ é o vetor que associa uma condição de guarda relacionada à marcação do lugar a cada transição,
- $M_o \in \aleph^n$ é o vetor que define uma marcação inicial para cada lugar (estado inicial),
- Atts = (*Dist*, *Markdep*, *Policy*, *Concurrency*, *W*)^n compreende o conjunto de atributos associados às transições, onde:

Dist $\in \aleph^n \rightarrow$ F é uma função de distribuição de probabilidade associada ao tempo de uma transição, sendo que 0$\leq$ F $\leq \infty$. Esta distribuição pode ser dependente de marcação;

Markdep $\in$ *constante*, *enabdepg*, define se a distribuição de probabilidade associada ao tempo de uma transição é constante ou dependente de marcação (*enabdepg* - a distribuição depende da condição de habilitação atual);

Policy $\in$ *fprd*, *prsg* define a política de memória adotada pela transição (*fprd* - *preemptive repeat different*, valor padrão, de significado idêntico à *race enabling policy*; *prsg* - *preemptive resume*, corresponde à *age memory policy*);

Concurrency $\in$ *ss*, *is* é o grau de concorrência das transições, onde *ss* representa a semântica *single server* e *is* representa a semântica *infinity server*.

W $\in \aleph^n \rightarrow R^+$ é a função peso, que associa um peso (w_t) às transições imediatas e uma taxa λ_t às transições temporizadas, onde:

$$\pi(t) = \begin{cases} \geq 1, & \text{se } t \text{ é uma transição imediata,} \\ 0, & \text{caso contrário.} \end{cases} \tag{2.22}$$

Se t é uma transição temporizada, então λ_t será o valor do parâmetro da função densidade probabilidade exponencial. Se t é uma transição imediata, então W_t será um peso, que é usado para o cálculo das probabilidades de disparo das transições imediatas em conflitos. Os arcos inibidores são usados para prevenir transições de serem habilitadas quando certa condição é verdadeira.

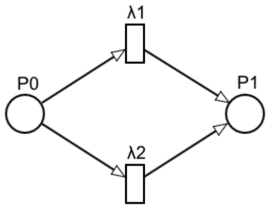
Em muitas circunstâncias, pode ser adequado representar a marcação inicial como um mapeamento a partir do conjunto de lugares para números naturais (m_0 : P $\rightarrow \aleph$), onde $m_0(p_i)$ denota uma marcação acessível (estado alcançável) do lugar P_i. Neste trabalho, a notação $\#P_i$ também foi adotada para representar $m(p_i)$.

2.3.1 Moment Matching

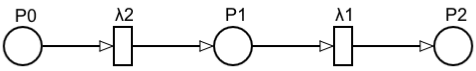
Modelos SPN consideram somente transições imediatas e temporizadas com tempos de disparo distribuídos exponencialmente. Essas transições modelam ações, atividades e eventos.

Muitas atividades podem ser modeladas através do uso dos construtores *throughput subnets* e *s-transitions*. Esses construtores são utilizados para representar distribuições expolinomiais de probabilidades, tais quais as distribuições *Erlang*, Hipoexponencial e Hiperexponencial [Watson J.R. e Desrochers 1991].

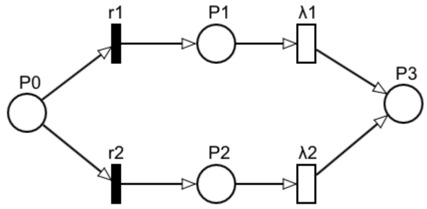
Combinações de lugares, transições exponenciais e transições imediatas podem ser usadas entre dois lugares para representar diferentes tipos de distribuições. As Figuras 2.2(a), 2.2(b) e 2.2(c) representam três *throughput subnets* formadas por conexões série e paralelo.



(a) Conexão paralela



(b) Conexão série



(c) Conexão paralela

Figura 2.2 *Throughput Subnets.*

A Figura 2.2(a) descreve uma *throughput subnet* formada por duas transições exponenciais em paralelo com taxas λ_1 e λ_2 , respectivamente.

Uma marcação no lugar P0 aparecerá no lugar P1 após a ocorrência de dois curtos *delays*, τ_1 e τ_2. O resultado da função de densidade com esse *delay* é definido pela Equação 2.23.

$$\tau = min(\tau_1, \tau_2) \tag{2.23}$$

A função de densidade para esses tempos é dada pela Equação (2.24). Essas transições exponenciais em paralelo são equivalentes a uma transição exponencial com taxa $\lambda_1 + \lambda_2$.

$$f\tau(t) = (\lambda_1 + \lambda_2) \exp^{-(\lambda_1+\lambda_2)^t}, t \geq 0 \tag{2.24}$$

A Figura 2.2(b) descreve uma *throughput subnet* formada por duas transições exponenciais em série com os parâmetros λ_1 e λ_2, respectivamente. Uma marcação no lugar P0 aparecerá no lugar P2 após o disparo das transições exponenciais, as quais têm um tempo associado $\tau = \tau_1 + \tau_2$, cuja função de densidade é dada pela Equação (2.25) [Watson J.R. e Desrochers 1991].

$$f\tau(t) = (f\tau_1 \star f\tau_2)(t) = \frac{\lambda_1\lambda_2(\exp^{-\lambda_1 t} - \exp^{-\lambda_2 t})}{(\lambda_2 - \lambda_1)}, t \geq 0 \tag{2.25}$$

onde: $\star$ é o operador de convolução[1]. Para o caso onde $\lambda_1 = \lambda_2 = ... = \lambda_n = \lambda$, a função densidade é dada pela Equação (2.26).

$$f\tau(t) = \frac{\lambda^n}{(n-1)!} t^{n-1} \exp^{-\lambda t}, t > 0 \tag{2.26}$$

Essa expressão representa uma distribuição do tipo *Erlang* de ordem n. Uma distribuição do tipo *Erlang* é especificada por dois parâmetros $\lambda > 0$ e $n > 0$.

A Figura 2.2(c) descreve uma *throughput subnet* formada por duas subredes paralelas, cada uma contendo uma transição imediata e uma transição exponencial. Uma marcação no lugar P0 aparecerá no lugar P3 após o disparo das transições imediatas e exponenciais em cada sub-rede. A probabilidade de cada sub-rede é determinada pelos pesos r_1 e r_2 das transições imediatas. A função de densidade dos tempos associados as transições exponenciais é dada pela Equação (2.27), que é uma distribuição hiperexponencial.

$$f\tau(t) = r_1 f\tau_1(t) + r_2 f\tau_2(t) = r_1\lambda_1 \exp^{-\lambda_1 t} + r_2\lambda_2 \exp^{-\lambda_2 t}, t > 0 \tag{2.27}$$

Essa *throughput subnet* implementa uma função de densidade com tempo hiperexponencial, cuja distribuição hiperexponencial é descrita pela Equação (2.28).

$r_j, j = 1...n,$

$\lambda_j, j = 1...n.$

$$\sum r_j = 1 \tag{2.28}$$

A técnica de aproximação de fases pode ser aplicada para modelar ações, atividades e eventos não-exponenciais através do *moment matching*. O método apresentado calcula o primeiro momento em torno da origem (média) e o segundo momento central (variância) e estima os momentos respectivos da *s-transition* [Watson J.R. e Desrochers 1991].

Dados de desempenho ou dependabilidade medidos ou obtidos dos sistemas (distribuição empírica) com média μ_D e desvio-padrão σ_D podem ter seu comportamento estocástico aproximados através da técnica de aproximação de fases. O inverso do coeficiente de variação dos dados medidos (Equação 2.29) permite a seleção da distribuição expolinomial que melhor se adapta à distribuição empírica.

$$\frac{1}{CV} = \frac{\mu_D}{\sigma_D} \tag{2.29}$$

A Redes de Petri descrita na Figura 2.3 representa uma atividade com distribuição de probabilidade genérica.

[1]Convolução é um operador linear que, a partir de duas funções dadas, resulta numa terceira que mede a área subentendida pela superposição das mesmas em função do deslocamento existente entre delas.

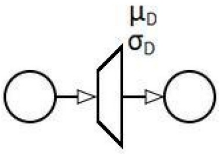


Figura 2.3 Distribuição Empírica.

Dependendo do valor do inverso do coeficiente de variação dos dados medidos (Equação 2.29), a respectiva atividade tem uma dessas distribuições atribuídas: *Erlang*, Hipoexponencial ou Hiperexponencial.

Quando o inverso do coeficiente de variação é um número inteiro e diferente de um, os dados devem ser caracterizados através da distribuição *Erlang* que é representada por uma sequência de transições exponenciais, cujo tamanho é calculado através da Equação (2.30).

$$\gamma = (\frac{\mu_D}{\sigma_D})^2 \tag{2.30}$$

A taxa de cada transição exponencial é calculada através da Equação (2.31).

$$\lambda = \frac{\gamma}{\mu_D} \tag{2.31}$$

Os modelos de Redes de Petri descritos na Figura 2.4 representam uma distribuição *Erlang*.

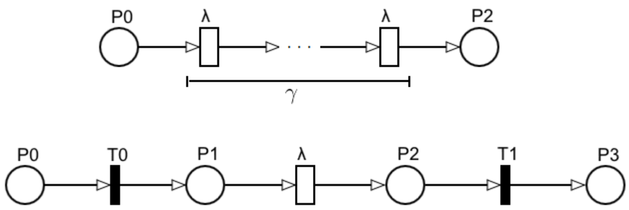


Figura 2.4 Distribuição *Erlang*.

Quando o inverso do coeficiente de variação é um número maior que um (mas não é um número inteiro), os dados são representados através da distribuição hipoexponencial, a qual é representada por uma sequência de transições exponenciais, cujo tamanho é calculado através da Equação (2.32).

$$(\frac{\mu_D}{\sigma_D})^2 - 1 \leq \gamma < (\frac{\mu_D}{\sigma_D})^2 \tag{2.32}$$

As taxas das transições exponenciais são calculadas através das Equações (2.33) e (2.34).

$$\lambda_1 = \frac{1}{\mu_1} \tag{2.33}$$

$$\lambda_2 = \frac{1}{\mu_2} \tag{2.34}$$

Onde os tempos médios atribuídos às transições exponenciais são calculados através das Equações (2.35) e (2.36).

$$\mu_1 = \frac{\mu_D \mp \sqrt{\gamma(\gamma+1)\sigma_D^2 - \gamma\mu_D^2}}{\gamma+1} \tag{2.35}$$

$$\mu_2 = \frac{\gamma\mu_D \pm \sqrt{\gamma(\gamma+1)\sigma_D^2 - \gamma\mu_D^2}}{\gamma+1} \tag{2.36}$$

Os modelos de Redes de Petri apresentados na Figura 2.5 descrevem uma distribuição hipoexponencial.

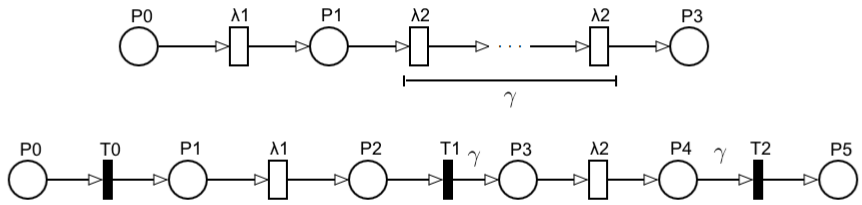


Figura 2.5 Distribuição Hipoexponencial.

Quando o inverso do coeficiente de variação é um número menor que um, os dados devem ser caracterizados através de uma distribuição hiperexponencial. A taxa da transição exponencial deve ser calculada através da Equação (2.37) e os pesos das transições imediatas são calculados através das Equações (2.38) e (2.39).

$$\lambda_h = \frac{2\mu_D}{(\mu_D^2 + \sigma_D^2)} \tag{2.37}$$

$$r_1 = \frac{2\mu_D^2}{(\mu_D^2 + \sigma_D^2)} \tag{2.38}$$

$$r_2 = 1 - r_1 \tag{2.39}$$

O modelo de Redes de Petri que representa esta distribuição hiperexponential é descrito na Figura 2.6.

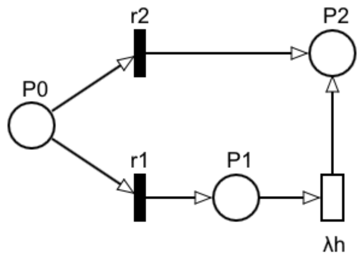


Figura 2.6 Distribuição Hiperexponencial.

2.4 MODELAGEM HIERÁRQUICA

Modelos de dependabilidade, como RBD, árvores de falhas e gráficos de confiabilidade, chamados métodos combinatórios, têm sido amplamente adotados. No entanto, esses modelos não conseguem representar completamente sistemas complexos e dependências de reparação.

Métodos baseados em estados, por outro lado, podem representar essas dependências, permitindo assim que a representação de complexos mecanismos redundantes, bem como políticas de manutenção sofisticadas. No entanto, eles sofrem com a complexidade do espaço de estados [Maciel P. e Fernandes 2007].

Portanto, as estratégias de modelagem hierárquicas e heterogêneas (baseadas em estados e modelos combinatórios) são essenciais para representar grandes sistemas com mecanismos de redundância ativos e políticas de manutenção [Trivedi et al. 2009]. A Figura 2.7 apresenta um exemplo de modelo hierárquico que utiliza RBD e SPN.

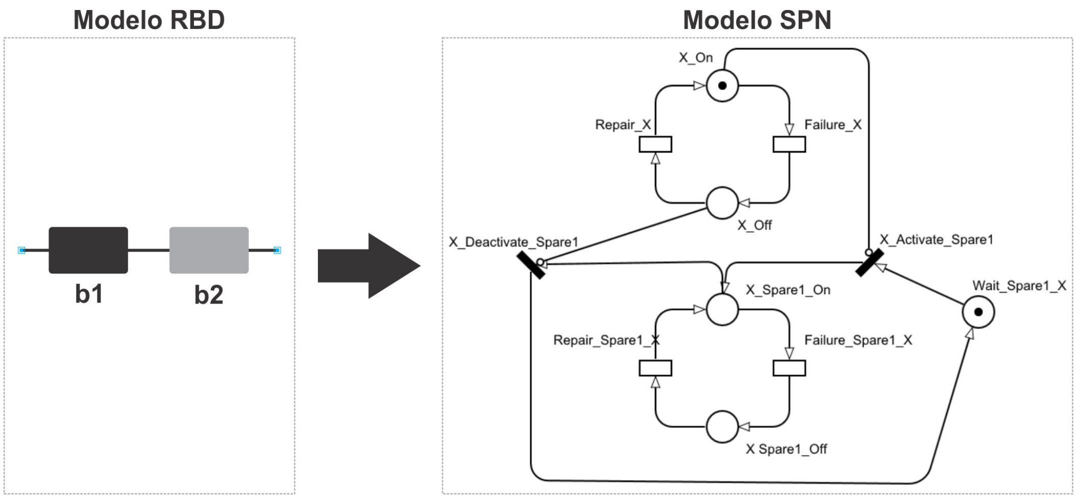


Figura 2.7 Modelagem Hierárquica Heterogênea.

O modelo RBD da Figura 2.7 é composto por dois blocos em série b1 e b2 representados pela expressão (2.12) e (2.14) onde o bloco b2 compõe uma hierarquia heterogênea com mecanismo de redundância com dois módulos apresentados no modelo SPN.

O modelo RBD é um modelo estático que não consegue englobar a parte dinâmica do sistema, ou seja, não é capaz de representar o tempo para ativar um mecanismo de redundância ativo (*Mean Time to Activate*) [Maciel P. e Fernandes 2007]. Segundo os autores, o modelo SPN é composto por dois módulos redundantes configurados em *cold standby*. Para representar essa hierarquia heterogênea, adotam-se as Redes de Petri Estocásticas.

O modelo SPN inclui quatro lugares; são eles: X_On, X_Off, X_Spare1_On, X_Spare1_Off que representam os estados operacionais e de falha dos módulos principais e de redundância, respectivamente.

O módulo de redundância (*Spare1*) é inicialmente desativado, representado pela ausência de *token* nos lugares X_Spare1_On e X_Spare1_Off. Uma falha do módulo principal é representada pelo disparo da transição X_Activate_Spare1. Como mencionado anteriormente, além de parâmetros MTTF e MTTR, também é necessário representar o tempo médio para ativação. Vale observar que o MTTF e MTTR atribuídos ao módulo principal podem ser diferentes do módulo de redundância.

A avaliação da disponibilidade do modelo hierárquico da Figura 2.7 é realizada através da avaliação do modelo RBD e SPN separadamente. O modelo SPN permite obter os valores do MTTF e MTTR através das Equações (2.4) e (2.6). Em seguida, esses resultados são atribuídos ao modelo RBD para análise da disponibilidade total do sistema.

Portanto, a modelagem hierárquica permite que o projetista avalie uma determinada arquitetura considerando as restrições de tempo do sistema como, por exemplo, o tempo necessário para a ativação de mecanismo de redundância e políticas de manutenção.

2.5 DESEMPENHO

A avaliação de desempenho de sistemas computacionais consiste em um conjunto de critérios e técnicas classificadas como as baseadas em medição e modelagem. As técnicas baseadas em modelagem podem ser classificadas como analíticas e simulação [Lilja 2000] e [Jain 1991]. No projeto para aquisição e utilização de sistemas computacionais, o objetivo do projetista é obter o melhor desempenho para determinar o custo [Jain 1991].

A medição de desempenho envolve essencialmente a monitoração do sistema enquanto está sob ação de uma carga de trabalho. Para adquirir resultados representativos, a carga de trabalho deve ser cuidadosamente selecionada por parte do projetista e utilizada nos estudos de desempenho, podendo ser real ou sintética [Lilja 2000] e [Jain 1991].

Embora a carga de trabalho real seja uma boa escolha por representar, de forma fiel, ela não pode ser repetida e, portanto, não é geralmente adequada para utilização. 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. Devido a esses motivos, uma carga sintética, cujas características são semelhantes às da carga de trabalho real, pode ser aplicada repetidamente de uma maneira controlada, desenvolvida e usada para estudos [Jain 1991] e [Lilja 2000].

A principal razão para a utilização de uma carga de trabalho sintética é que ela é uma representação ou modelo da carga de trabalho real. A carga de trabalho pode ser facilmente modificada sem afetar a operação e pode ser facilmente portada para sistemas diferentes, devido ao seu pequeno tamanho e ela pode ter embutida capacidades de medição [Jain 1991].

A escolha da carga de trabalho é tão importante quanto a definição de qual estratégia de medição deve ser seguida. As estratégias de medição têm em sua base o conceito de evento, que é uma mudança no estado do sistema. A definição precisa de um evento depende da métrica que está sendo medida. Os tipos de métricas podem ser classificados em categorias baseadas no tipo de evento que compreende a métrica [Lilja 2000]:

- Métricas baseadas em contagem de evento representam o registro da quantidade de vezes em que um determinado evento ocorre. Exemplo: quantidade de requisições de leitura/escrita de um disco;
- Métricas baseadas em evento secundário representam o registro da quantidade de vezes em que um evento ocorre devido à ocorrência de outro evento. Exemplo: para determinar a quantidade média de requisições enfileiradas em um *buffer*, será necessário registrar as requisições à medida que estas são adicionadas ou removidas do *buffer*. Assim, os eventos a serem monitorados serão as operações de enfileiramento e desenfileiramento e a métrica será a quantidade média de requisições na fila;
- *Profiles* caracterizam o comportamento de um programa ou uma aplicação de um sistema. Usualmente é capaz de identificar detalhadamente em quais operações o programa ou sistema está consumindo mais tempo;
- Métrica dirigida a evento é a estratégia que registra as informações necessárias para o cálculo da métrica de desempenho sempre que o evento pré-selecionado ou eventos ocorrem. Uma vantagem dela é que a perturbação (*overhead*) gerada na medição ocorre apenas durante o registro do evento. Se o evento nunca ocorrer, ou apenas ocorrer raramente, a perturbação no sistema será relativamente pequena;
- Métrica de *Tracing* é uma estratégia similar à dirigida a evento, exceto que, em vez de simplesmente registrar o que o evento ocorreu, 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 é a estratégia que registra os dados do sistema em intervalos fixos de tempo, independentemente da ocorrência do evento. Como resultado, uma perturbação (*overhead*) pode ocorrer dependendo da frequência em que a medição é executada. Essa estratégia de medição produz um resumo estatístico do comportamento global do sistema. Eventos que ocorrem com pouca frequência são perdidos devido a esta aproximação estatística;
- A 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.

A modelagem analítica utiliza um conjunto de equações e funções matemáticas para descrever o comportamento de um sistema. Apesar desses modelos considerarem parâmetros específicos de um sistema, podem ser facilmente adaptados para outros sistemas. Durante a construção dos modelos, deve-se levar em consideração a sua complexidade e praticidade.

Os modelos analíticos permitem uma análise ampla e aprofundada em relação aos efeitos causados pelos parâmetros definidos nas equações sobre a aplicação. Além disso, também se podem estabelecer possíveis relacionamentos entre cada um dos parâmetros considerados. Essa modelagem, quando comparada às demais técnicas de avaliação de desempenho, apresenta menor custo de execução. Para validar os resultados alcançados através dos modelos elaborados, a modelagem analítica pode compará-los aos valores reais medidos em testes experimentais [Lilja 2000].

A simulação é utilizada tanto em avaliação desempenho, quanto na validação de modelos analíticos. Ao contrário das medições analíticas, as simulações baseiam-se em modelos abstratos do sistema e não exigem que o sistema esteja totalmente implantado para que sejam aplicadas. Assim, os modelos utilizados durante a simulação são desenvolvidos através da abstração de características essenciais do sistema, sendo que a complexidade e o grau de abstração dele podem mudar de um sistema para outro. Durante a simulação, controlam-se, com maior eficiência, os valores assumidos por parâmetros do sistema. Portanto, fica mais fácil obter informações relevantes para a avaliação de desempenho [Lilja 2000].

2.6 IMPORTÂNCIA PARA CONFIABILIDADE E CUSTOS

A importância para confiabilidade (*Reliability Importance* - RI), ou a importância Birnbaum [Kuo e Zuo 2003], é uma medida utilizada para avaliar, como o seu nome sugere, a importância do componente para confiabilidade, permitindo estabelecer comparações entre as funções de cada uma delas no sistema e determinar quais componentes que devem ser redesenhados, a fim de melhorar a confiabilidade do sistema.

De acordo com [Kuo e Zuo 2003], o índice da importância para confiabilidade (I_i^B) do componente i é definido pela Equação (2.40):

$$I_i^B = \frac{\partial R_s(p)}{\partial p_i} \tag{2.40}$$

onde I_i^B é o índice da importância para confiabilidade do componente i, p é o vetor de confiabilidade dos componentes, p_i é a confiabilidade do componente i e R_s é a confiabilidade do sistema.

Em outras palavras, o RI do componente i é igual a quantidade do aumento da confiabilidade do sistema quando a confiabilidade do componente i é melhorada por uma unidade. Baseado nesta interpretação e considerando que $0 \leq p_i \leq 1$, a importância para confiabilidade de um componente i pode ser definida pela Equação [Kuo e Zuo 2003]:

$$I_i^B = R_s(1_i, p^i) - R_s(0_i, p^i) \tag{2.41}$$

onde p^i representa o vetor de confiabilidade dos componentes com o i-ésimo componente removido; 0_i representa a condição quando o componente i é falho e 1_i a condição quando o componente i está sempre no modo operacional.

Essa abordagem é adotada para observar o comportamento dos valores do RI para cada componente, onde equipamentos com menor confiabilidade são substituídos ou replicados para atender os requisitos estabelecidos pelo projetista para o sistema.

A Figura 2.8(a) apresenta um exemplo de análise da importância para confiabilidade em um período de dez anos (tempo igual a 87600 horas) em um modelo RBD em série através dos componentes A1, B1 e C1, onde foram obtidos os valores 1.0, 0.7495 e 0.7462 respectivamente. A análise dos resultados indica que o projetista deve adicionar redundância no componente A1.

Após replicar o componente A1 (Figura 2.8(b)), o valor da importância para confiabilidade do componente B1 passa a ser RI=1.0, de modo que este componente se torna o mais importante (nesta nova estrutura). O segundo lugar passa a ser do componente C1 com RI=0.9956.

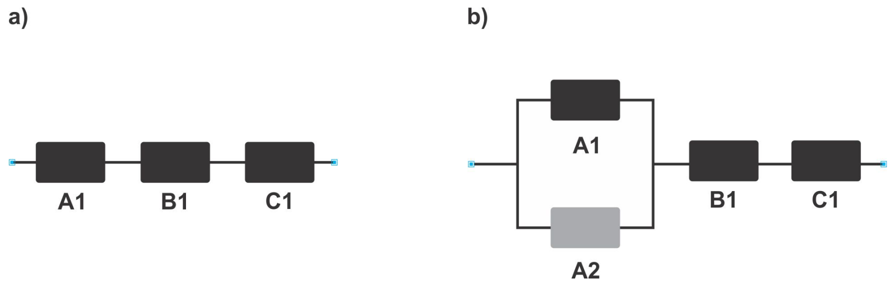


Figura 2.8 Importância para Confiabilidade (RI).

Em sumo, a importância para a confiabilidade de cada componente é apresentada da seguinte forma: é calculado o valor para a confiabilidade do sistema quando o componente está funcionando, em seguida é calculado este valor quando o componente falha. O valor da importância é dado pela diferença entre o valor quando o componente funciona e quando ele falha conforme apresentado na Equação 2.41.

Os resultados são apresentados como valores absolutos e normalizados em relação ao de maior valor (Tabela 2.1) utilizando a Equação (2.42):

$$In_i = \frac{I_i}{I_x} \tag{2.42}$$

onde In_i é o índice normalizado para o componente i; I_i é o valor do índice não normalizado para o componente i e, I_x é o valor do maior índice não normalizado entre os componentes.

O índice RI, apesar de ser um bom indicador para melhorar a disponibilidade e/ou confiabilidade, considera que os recursos financeiros são ilimitados. Investir muito dinheiro para melhorar um componente indicado como o mais importante pode não ser viável. [Figueiredo et al. 2011] propôs o índice de Importância para a Confiabilidade e Custos (*Reliability-Cost Importance* - RCI) que se destina a quantificar a importância do componente para o sistema relativo ao custo de aquisição do equipamento. Sendo assim, o componente mais importante é aquele que tem o maior RI combinado com um custo mais baixo, quando comparado com outros componentes.

O método proposto provê uma lista ordenada de componentes indicando o índice de impacto no sistema quando melhorado. Calculando o custo do componente em relação aos outros, o índice de importância para a confiabilidade quantifica o componente e favorece a confiabilidade do sistema, investindo menos recursos financeiros do que em outras soluções. A Equação (2.43) expressa esta relação [Figueiredo et al. 2011]:

$$RCI_i = I_i^B \times (1 - \frac{UC}{CO}) \tag{2.43}$$

onde RCI_i é o índice de Importância para a Confiabilidade e Custo do componente i, I_i^B é o valor não normalizado da importância para a confiabilidade do componente i. Custo unitário (UC) é o custo de aquisição do componente i e CO é o custo de propriedade do sistema e suporte.

Através dos componentes A1, B1 e C1 da Figura 2.8(a), o índice de importância para a confiabilidade e custos foi calculado. Em uma estrutura em série, considerando importância para a confiabilidade, o componente menos confiável é o que tem o mais alto valor de RI. Já quando é considerado RCI, o componente mais importante pode ser outro que tenha um valor alto de RI (mas não o mais alto) e um custo maior (quando comparado aos outros), como pode ser observado na Tabela 2.1.

Tabela 2.1 Índice de Importância para a Confiabilidade e Custos.

	(RI)	**(RCI)**	**Custo**
Componente A1	1,0	0,4536	R$4.487,00
Componente B1	0,7495	0,6598	R$983,00
Componente C1	0,7462	0,4970	R$2.742,00

A Tabela 2.1 apresenta os valores das métricas de confiabilidade (tempo igual a 87600 horas), o RI, RCI e Custo para os três componentes. Analisando os resultados dessa tabela, o componente mais importante (considerando a confiabilidade e custo) é o componente B1 com o RCI=0,6598 e custo R$983,00 em relação aos outros dois componentes. É possível observar que o resultado é diferente daquele indicado pela importância para a confiabilidade (componente A1).

Como apresentado, este índice se mostra uma opção interessante para auxiliar projetistas a tomarem decisões na fase de projeto, pois indica como os componentes contribuem para a confiabilidade do sistema, além de indicar onde aplicar melhorias.

Além de estar relacionado com a confiabilidade, o RCI também possui uma relação com os custos da arquitetura definidos na próxima seção.

2.7 CUSTO DE PROPRIEDADE

Custo de propriedade (*Cost of Ownership* - CO) é um modelo desenvolvido para a análise dos custos diretos e indiretos para adquirir e usar *hardware* e *software* dentro de uma organização [David, Schuff e Louis 2002] e [Group 2011].

Este modelo foi derivado do custo total de propriedade (*Total Cost of Ownership* - TCO) que é composto pelos custos relacionados a aquisição de equipamentos (*hardware* e *software*) e pelos custos de administração. Esse último tipo é dividido em custos de controle (centralização e padronização) e custos operacionais (suporte, avaliação, instalação/*upgrade*, treinamento, *downtime*, *futz*, auditoria e consumo de energia [David, Schuff e Louis 2002].

Contudo, neste trabalho, foi utilizada uma abordagem que considerou somente os custos relacionados aos preços de varejo dos equipamentos e do suporte contratado para estes componentes, e por esse motivo, esse trabalho adota o termo custo de propriedade (CO) [David, Schuff e Louis 2002].

O CO está relacionado com o índice de Importância para a Confiabilidade e Custo do componente (RCI) uma vez que este último tem como propósito determinar a importância do componente para o sistema em função do custo de aquisição do equipamento. Sendo assim, o RCI indica qual componente proporcionará maior impacto (na performabilidade do sistema) quando melhorado (normalmente replicado) considerando o custo do componente em relação ao custo dos outros componentes [Figueiredo et al. 2011].

O custo de propriedade é definido pela equação (2.44):

$$CO = (H_{ware} + S_{ware}) + \frac{\sum_{i=1}^{n} C_{suport}}{i} \tag{2.44}$$

onde $H_{ware} \in R^{+}$ é o custo de aquisição do *Hardware*, $S_{ware} \in R^{+}$ é o custo de aquisição do *Software*, $C_{suport} \in R^{*+}$ é o custo de suporte no mês, $n \in R^{*+}$ é o número total de meses e $i \in R^{*+}$ é a taxa de inflação do período.

2.8 CONSIDERAÇÕES FINAIS

Este capítulo apresentou os principais conceitos que envolvem esta Dissertação. Primeiramente, foram discutidos conceitos sobre dependabilidade. Posteriormente foram apresentados conceitos relacionados à diagrama de bloco de confiabilidade e a importância para confiabilidade e custo. Por fim, foi apresentada uma visão geral sobre desempenho e conceitos sobre Redes de Petri Estocásticas e métodos para determinação do custo de propriedade do sistema.

CAPÍTULO 3

MODELOS

Este capítulo apresenta a descrição do sistema de *call center* de emergência, bem como as suas características, operações, componentes e funções. Em seguida, são descritos os modelos para avaliação da disponibilidade e desempenho. O estudo foi conduzido em um *call center* de emergência, localizado no nordeste do Brasil, que tem propriedades comumente encontradas em *call centers* em todo o mundo.

3.1 DESCRIÇÃO DO SISTEMA

O *call center* de emergência estudado possui uma infraestrutura própria que é apresentada na Figura 3.1. O sistema é composto pela Estrutura de Energia, Estrutura de Rede, Estrutura de Voz e Estrutura de Atendimento ao Usuário.

A Estrutura de Energia é composta pelo Transformador, SubPainel e UPS (*Uninterruptible Power Supply*) que possui um banco de baterias e fornece energia aos *PowerStrips* que provêem energia aos componentes conectados a eles. A Estrutura de Rede consiste em Roteador, *Swicth* e um *link* de dados contratado por uma operadora de telefonia. A Estrutura de Voz é composta pelo PABX (*Private Automatic Branch Exchange*) e um link de uma PSTN (*Public Switched Telephone Network*).

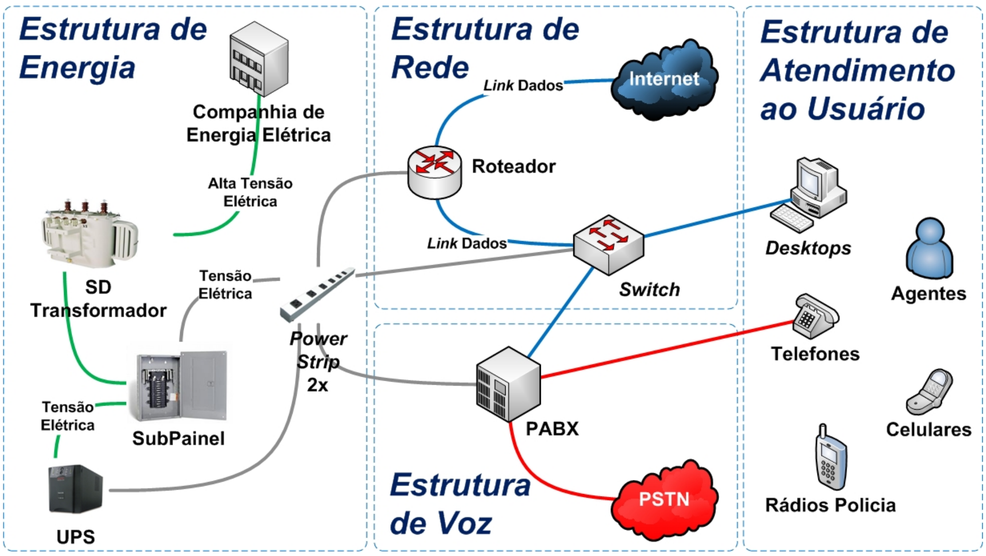


Figura 3.1 Estrutura do *call center*.

O Sistema PABX gerencia e transfere as chamadas que chegam no *call center* de emergência. Essas chamadas são inseridas em uma fila, que tem um tamanho pré determinado e podem ser atendidas imediatamente se existir algum agente disponível ou aguardar até serem atendidas ou não. Sempre que o número de chamadas exceder o tamanho da fila, a chamada é automaticamente descartada.

A Estrutura de Atendimento ao Usuário tem quinze rádios, quinze *desktops* e quatro celulares para comunicação interna entre os membros de cada categoria: Polícia Civil, Militar, Científica e Bombeiros.

A Figura 3.2 mostra o processo de atendimento de chamadas de emergência. Quando um usuário faz uma chamada existem três possibilidades:

1. Ele é atendido imediatamente se existir algum agente disponível e a fila não estiver cheia.

2. Ele aguarda na fila, se todos os agentes estiverem ocupados, mas existir lugar disponível na fila.

3. Sua chamada é descartada se não existir espaço disponível na fila.

Entretanto, no caso de o usuário aguardar na fila, existem duas possibilidades:

1. Ele aguarda ser atendido, quando o agente ficar disponível.

2. Ele abandona o sistema sem ser atendido.

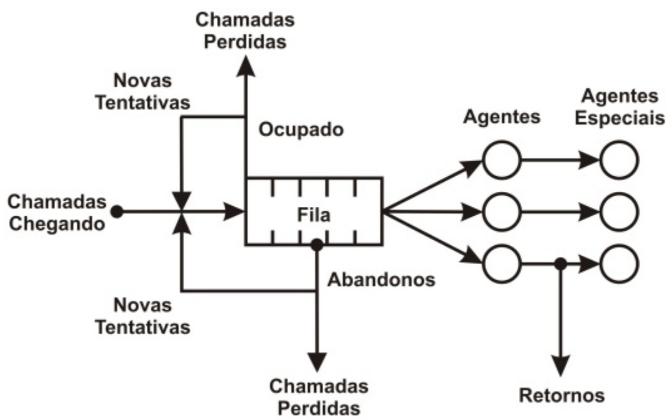


Figura 3.2 Esquema operacional de um *call center* de emergência simples.

Existe também a possibilidade de uma nova tentativa de chamadas por usuários tanto no caso de chamadas descartadas (chamadas perdidas) quanto em abandonos. O problema principal encontrado no *call center* de emergência é o grande número de chamadas chegando na central. Muitas dessas chamadas deveriam ser tratadas imediatamente ou deveriam aguardar apenas alguns segundos na fila. Os usuários podem ficar impacientes por entrar na fila e esperar muitos minutos para serem atendidos. Quando o usuário deixa a fila e tenta reconectar, ele entra no fim da fila, pois o tratamento das chamadas no sistema é do tipo primeiro que entra é o primeiro que sai (*First In First Out* - FIFO).

Uma questão séria enfrentada pela gestão operacional do *call center* de emergência é a taxa de trotes. Essas chamadas mantêm os agentes ocupados enquanto chamadas válidas podem ser descartadas caso a fila esteja cheia. Isso resulta, na maioria dos casos, em novas tentativas de chamadas, pois os usuários necessitam urgentemente do serviço.

O atendimento do usuário é realizado por um agente responsável pela triagem do serviço. Essa triagem determina quais agentes serão contactados de acordo com a área (Polícia Civil, Militar, Científica e Bombeiros) e transfere as chamadas para eles, os quais acionarão veículos para o local da ocorrência.

3.2 MODELO DE DISPONIBILIDADE

Esta seção apresenta um modelo para avaliação de disponibilidade para um *call center* de emergência através de modelagem hierárquica heterogênea com RBD e SPN.

O modelo RBD, apresentado na Figura 3.3, consiste em vários blocos conectados em série-paralelo e foi dividido em duas partes: SubModelo (A) e SubModelo (B). O SubModelo (B) consiste em oito blocos: SubModeloParalelo*PowerStrip* e PABX compreende a Estruturas de Voz, Roteador e *Switch* correspondem a Estruturas de Redes e SubModeloParalelo(PMilitar, PCivil, DBombeiro e PCientífica) representa a Estrutura de Atendimento ao Usuário, enquanto o SubModelo (A) representa a Estrutura de Energia.

Estes blocos são o resultado da aplicação de redução em série-paralelo da estrutura do *call center* apresentado na Figura 3.1 através das Equações (2.16), (2.17), (2.18), (2.20) e (2.21).

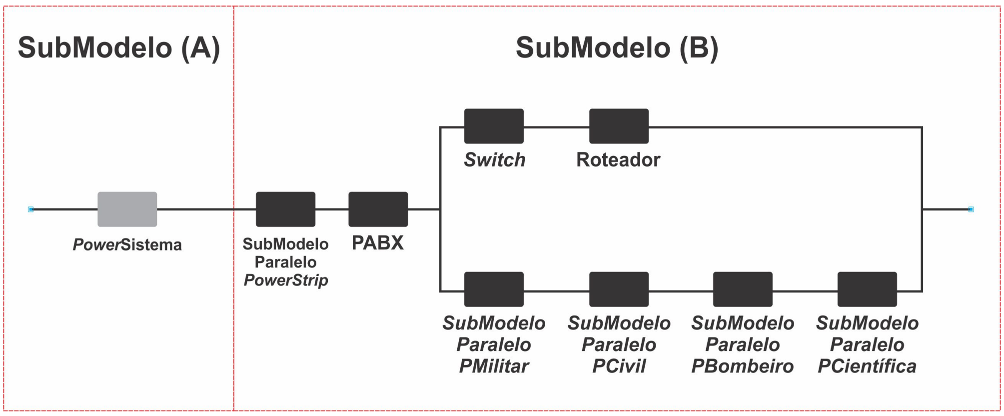


Figura 3.3 Modelo RBD da arquitetura (A).

No SubModelo (B) foi aplicada uma redução em paralelo nos dois componentes *Power-*

Strip gerando o bloco equivalente SubModeloParalelo*PowerStrip*. O segundo bloco é o PABX que não possui redundância, pois é um equipamento robusto, caro e tolerante a falha. Por estas razões, os administradores do *call center* de emergência decidiram manter apenas um PABX com a possibilidade de adicionar outro componente no futuro, caso estudos comprovem sua necessidade.

O terceiro e quarto blocos são representados por *Switch* e Roteador em série respectivamente. O quinto, sexto, sétimo e oitavo blocos correspondem ao Atendimento ao Usuário, representado pelos os blocos SubModeloParalelo(PMilitar, PCivil, DBombeiro e PCientífica) que correspondem aos equipamentos da Polícia Militar, Polícia Civil, Bombeiros e da Polícia Científica que representa à transmissão de voz de rádio e telefone celular usado para a comunicação interna.

A Figura 3.4 é a representação hierárquica heterogênea do bloco *Power*Sistema do modelo RBD por meio da modelagem SPN (Ver Seção 2.4) do SubModelo (A) que representa a Estrutura de Energia.

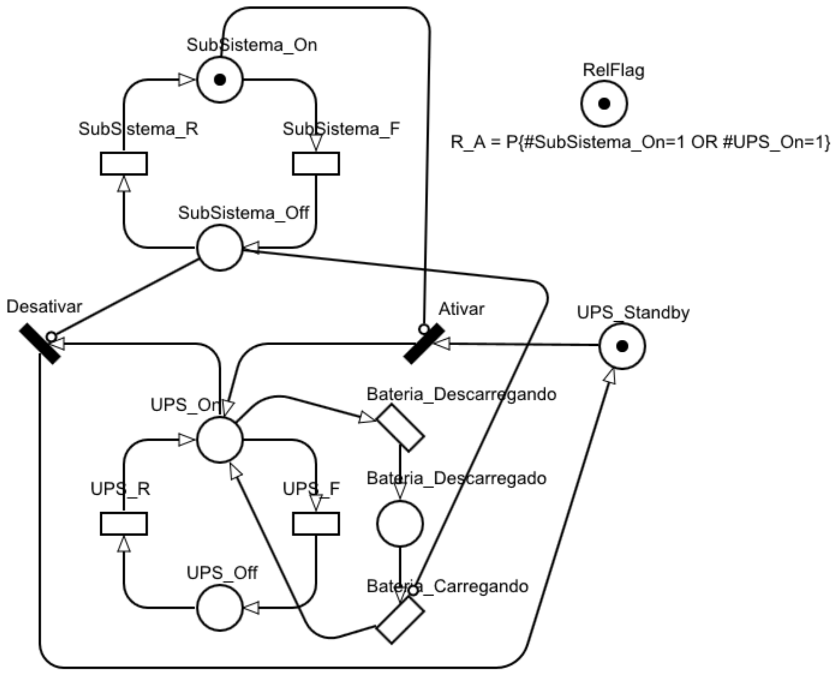


Figura 3.4 Modelagem Hierárquica e Heterogênea do SubModelo (A).

A modelagem é composta por dois módulos, o SubSistema, que representa o Transformador e SubPainel, estiver *down*, o *call center* funcionará somente se o módulo UPS estiver *up* (considerando que os demais componentes estão funcionando). Se ambos módulos estiverem *down*, o serviço de emergência não funcionará.

O modelo SPN é constituído por seis lugares: SubSistema_On, SubSistema_Off, UPS_On, UPS_Off, Bateria_Descarregado e UPS_Standy que representam tanto os estados operacional e de falha [Maciel P. e Fernandes 2007] do módulo principal quanto do módulo de redundância.

O lugar Bateria_Descarregado representa a descarga da bateria (cuja vida da bateria é de duas horas). Representa-se a desativação do módulo de redundância (UPS) através da ausência de *tokens* nos lugares UPS_On, UPS_Off e Bateria_Descarregado.

A falha do módulo principal (SubSistema) é representada pelo disparo da transição Ativar. O disparo dessa transição representa o início da operação e é descrito por uma transição imediata (im), uma vez que o tempo de disparo é igual a zero [Maciel P. e Fernandes 2007].

Os atributos relacionados a cada transição do modelo descrito nesta seção são apresentados na Tabela 3.1.

Tabela 3.1 Atributos das transições do modelo SPN *cold standby*.

Transição	Tipo	*Delay* ou Peso
SubSistema_F	exp	MTTF
SubSistema_R	exp	MTTR
UPS_F	exp	MTTF
UPS_R	exp	IF #RelFlag=1: ($10^{n=8}\times$ MTTF) ELSE MTTR;[1]
Bateria_Descarregando	exp	TempoDescarga
Bateria_Carregando	exp	IF #RelFlag=1: ($10^{n=8}\times$ TempoDescarga) ELSE TempoCarga;
Ativar	im	1
Desativar	im	1

As transições temporizadas SubSistema_F, SubSistema_R, UPS_F, UPS_R, Bateria_Descarregando e Bateria_Carregando seguem uma distribuição exponencial (exp) enquanto que Desativar é transição imediata (im). A expressão P{#SubSistema_On = 1 OR #UPS_On = 1} permite determinar a disponibilidade do SubModelo (A) (análise estacionária) se #RelFlag = 0; por outro lado, se #RelFlag = 1, a mesma expressão permite obter a confiabilidade, caso uma análise transiente seja realizada.

3.3 MODELO DE DESEMPENHO

O SPN descrito na Figura 3.5 é um modelo de desempenho [Silva 2010] que permite estimar a duração média das chamadas, o número de descartes, o número de desistências, de trotes, solicitação de informações, ligações válidas e por engano. Estas métricas são importantes para verificar a relação entre o número de chamadas e números de agentes necessários no centro de emergência. O modelo desenvolvido por [Silva 2010] sofreu alguns ajustes e está sendo usado nesta Dissertação para permitir a análise de determinados

[1] O valor n=8 foi adotado para simular um estado absorvente. Assim, o MTTR é igual a $10^8\times$ MTTF [Zimmermann e Knoke 2007].

cenários de interesse.

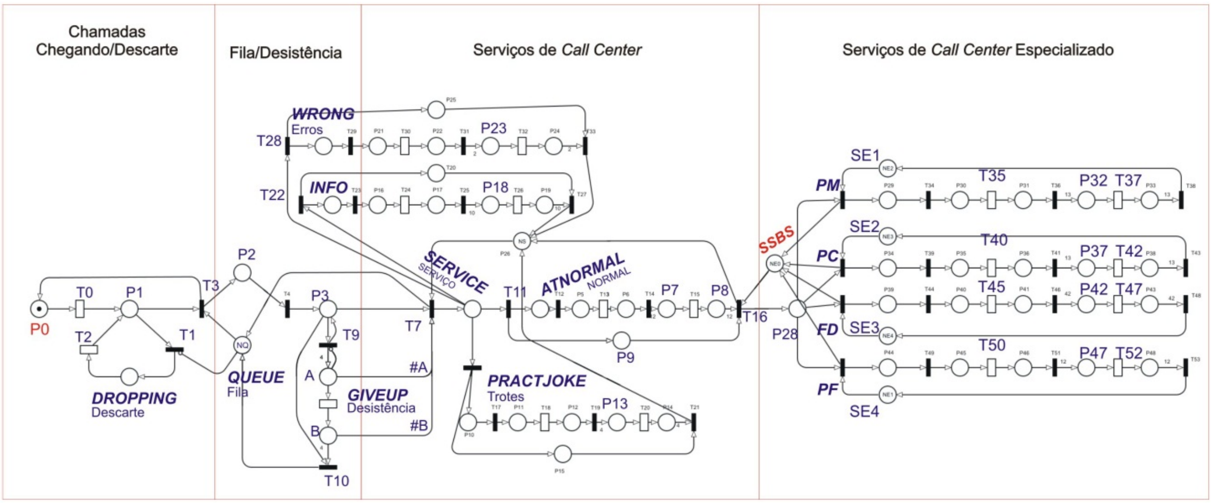


Figura 3.5 Modelo SPN.

Através desse modelo, é possível detectar o tempo ocioso dos agentes, o número de chamadas descartadas, tamanho da fila de espera, o tempo de espera e número de descartes. O modelo também permite estudar a probabilidade de desistência das chamadas do *call center*, dimensionar o número de agentes civis e especializados dentro dos requisitos exigidos a fim de evitar mau dimensionamento. A quantidade de agentes utilizados para atender todas essas chamadas obtidas pode ser usada para encontrar a distribuição dos agentes no *call center*.

Através dos números de ligações destinadas à Polícia Militar, à Polícia Civil, aos Bombeiros e Polícia Científica é possível verificar a demanda para cada serviço. Finalmente, o número de agentes atendendo a esses chamados permite constatar se esse número é ideal e se estes agentes estão sobre carregados ou ociosos.

O modelo proposto é composto por quatro sub-modelos ou sub-redes (Chamadas Chegando/Descarte, Fila/Desistência, Serviço de *call center*, Serviço de *call center* Especializado) que descrevem o comportamento do *call center* de emergência. Os *tokens*, que representam as chamadas de emergência, são iniciados no lugar P0.

O disparo da transição temporizada T0 (possui um *delay* que representa os próximos *tokens* no modelo) torna o lugar P1 marcado. No lugar P1, é possível verificar a existência de *tokens* no lugar NQ (capacidade máxima do número de *tokens* neste modelo que é iqual a trinta) através do arco inibidor conectado entre a transição imediata T1 e o lugar NQ. Se a quantidade de *tokens* no lugar NQ for menor que trinta, a transição imediata, T3, é habilitada; caso seja igual a trinta, T1 é habilitada.

Se a quantidade de *tokens* no lugar NQ for iqual a trinta, o disparo da transição imediata T1 produz uma marca no lugar DROPPING (Descarte). Em seguida, o modelo contabilizará os *tokens* através da expressão P{#DESCARTE>0} × (1/T2), apresentada na

Tabela 3.4, e retorna os *tokens* para o lugar P1, através do *delay*, que representa uma nova tentativa de chamada, da transição temporizada, T2. Porém, se no lugar NQ a quantidade de *tokens* presentes for menor que trinta (caracterizando espaço disponível no lugar NQ do modelo), o lugar P2 é marcado e logo depois o lugar P3 é marcado onde os *tokens* entram na sub-rede Fila/Desistência.

É interessante notar que o *delay* da desistência é representado por uma distribuição *Erlang*, composta pela sub-rede que contém as transições T9, T10, DESISTÊNCIA; os lugares A, B (e os respectivos arcos). A distribuição poliexponencial, definida em [Bolch G. Greiner e Trivedi 2006], foi a que melhor se ajustou aos dados medidos e foi definida de acordo com o método *moment matching* [Desrochers e Al-Jaar 1995].

A sub-rede Fila/Desistência simula o processo de espera em uma fila FIFO, onde há a possibilidade da chamada ser atendida ou o usuário desistir. Após o disparo da transição T4, o *token* passa para o lugar P3. Nesse momento, as transições T9 e T7 são habilitadas; o disparo da transição imedidata T9 representa uma desistência enquanto o disparo da transição T7 representa o início do serviço de atendimento de chamadas.

A sub-rede Serviço de *call center* é responsável por representar as classes de chamadas. Este modelo trata os trotes, as chamadas erradas, solicitação de informações e serviços normais. A razão entre estas classes de chamadas é representada pelos pesos atribuídos às transições T11, T16, T22 e T28 (ver Tabela 3.2).

Tabela 3.2 Pesos das transições.

Tipo de Chamadas	Transições	Peso
Válidas	T11	W11
Trotes	T16	W16
Erros	T22	W22
Informações	T28	W28

Os serviços especializados são representados pela sub-rede Serviço de *call center* Especializado, onde as chamadas, representadas por *tokens*, são armazenadas no lugar P28. Esta sub-rede representa quatro classes de serviços, ou seja, Polícia Militar (MP), Polícia Civil (CP), Bombeiros (FD) e da Polícia Científica (FP). As relações de serviço são definidas com base no peso atribuídos às transições MP, CP, FD e FP.

A marcação do lugar SSBS (*Specialized Service Buffer Size*) representa o espaço livre na fila de atendimento especializado. O lugar P28 representa a fila de serviços especializados (*Specialized Service Queue* - SSQ). As marcações dos lugares SE1, SE2, SE3 e SE4 representam o número de atendentes disponíveis de cada serviço.

A expressão representada na Tabela 3.3 permite calcular o número de chamadas atendidas em um determinado período (1440 minutos). MP corresponde às chamadas para a Polícia Militar e é calculada através da probabilidade de ter um ou mais *tokens* no lugar P32 (P{#P32>0}) multiplicado pela taxa de serviço (1/W(T37)) e pelo período.

O número total de chamadas para a Polícia Civil, Científica e Bombeiros é calculado da mesma forma que as chamadas para a Polícia Militar, variando apenas o lugar e as taxas de serviço (ver Tabela 3.3). Os atrasos (*delays*) atribuídos as transições T35, T37, T40, T42, T45, T47, T50 e T52 foram obtidos pela aplicação do método *moment matching* às medidas de tempo coletadas a partir do sistema real.

Tabela 3.3 Total de chamadas.

Métrica	Expressão	Peso
MP	P{#P32>0} × 1/W(T37) × Período;	W37
CP	P{#P37>0} × 1/W(T42) × Período;	W42
FD	P{#P42>0} × 1/W(T47) × Período;	W47
FP	P{#P47>0} × 1/W(T52) × Período;	W52

A Tabela 3.4 mostra as expressões das probabilidades de serviço normal (ATNORMAL), trotes (TROTE), informação (INFO), chamadas erradas (ERRO), desistências (DESISTÊNCIA) e chamadas descartadas (DESCARTE).

Tabela 3.4 Medidas dos serviços.

Serviços Medidos	Expressão	Peso
ATNORMAL	P{#P7>0} × 1/W(T15) × Período;	W15
TROTE	P{#P13>0} × 1/W(T20) × Período;	W20
ERRO	P{#P23>0} × 1/W(T32) × Período;	W32
INFO	P{#P18>0} × 1/W(T26) × Período;	W26
DESISTÊNCIA	P{#B>0} × (1/DESISTÊNCIA)	
DESCARTE	P{#DESCARTE>0} × (1/T2)	

A Tabela 3.5 apresenta as expressões das utilizações dos agentes especializados da Polícia Militar (UMP), Polícia Civil (UCP), Bombeiros (UF) e Polícia Científica (UFP).

Tabela 3.5 Utilização dos agentes especializados.

Métrica	Expressão
UMP	MP - E{#SE2}
UCP	CP - E{#SE3}
UFD	FD - E{#SE4}
UFP	FP - E{#SE1}

Segundo [Silva 2010], a Polícia Militar tem dez agentes especializados, a Polícia Civil e os Bombeiros têm dois cada e a Polícia Científica tem um agente [Silva 2010]. Estes valores podem ser alterados de acordo com a necessidade do serviço.

O processo de validação foi realizado através da comparação dos resultados das métricas de desempenho obtidas utilizando o modelo de desempenho do call center de emergência proposto e dos resultados das medições dos dados reais [Silva 2010]. A tabela 3.6 apresenta o erro máximo relativo entre os valores obtidos para a Polícia Militar, Civil, Científica, Bombeiros, chamadas válidas, trotes, solicitações de informações e chamadas erradas.

Tabela 3.6 Erro relativo máximo.

Categoria	**Erro Finais de Semana**	**Erro Dias úteis**
Polícia Militar	4,21%	10,11%
Polícia Civil	3,81%	10,47%
Bombeiros	10,25%	10,02%
Polícia Científica	10,23%	11,55%
Chamadas Válidas	4,52%	5,15%
Trotes	2,74%	3,05%
Informações	7,31%	5,92%
Chamadas Erradas	11,34%	10,90%

3.4 CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentada a descrição da estrutura do *call center* de emergência estudado nesta Dissertação. Em seguida, foram descritos os modelos utilizados para a avaliação da disponibilidade e desempenho do *call center* de emergência.

CAPÍTULO 4

AVALIAÇÃO E PLANEJAMENTO

Neste capítulo, é apresentada a metodologia utilizada para a avaliação de performabilidade da arquitetura mostrada no Capítulo 3 que utiliza uma estratégia para decomposição e composição, cujo objetivo é reduzir a complexidade do processo de avaliação. Além disso, a viabilidade da arquitetura foi verificada em relação aos requisitos do *call center* de emergência (*downtime* aceitável e maior número possível de chamadas atendidas), através das métricas de disponibilidade e desempenho.

4.1 METODOLOGIA DE AVALIAÇÃO

O desenvolvimento de técnicas, estratégias e modelos que proporcionem meios para avaliação da disponibilidade e desempenho de *call center* de emergência são de fundamental importância, uma vez que os recursos computacionais devem ser usados de forma a atender os níveis de serviços estabelecidos e que se faça um uso eficiente da infraestrutura computacional.

Em *call center* de emergência, eventos de falhas e atividades de reparo podem causar degradação no desempenho do sistema, portanto, a análise de desempenho sem considerar os efeitos da disponibilidade pode ser imprecisa. A metodologia adotada para avaliação de performabilidade de *call center* de emergência combina os resultados da avaliação da disponibilidade e os resultados da avaliação de desempenho através de uma estratégia de decomposição e composição.

Foram seguidas quatro etapas para analisar a performabilidade e os custos de infraestrutura do *call center* de emergência. A primeira etapa diz respeito ao entendimento do sistema, através de seus componentes, interfaces e interações.

A segunda etapa tem como objetivo a definição dos modelos de performabilidade (RBD e SPN) com o suporte das ferramentas adotadas e apresentadas no Apêndice A. Uma vez que os modelos são criados e validados, a avaliação da arquitetura é realizada (terceira etapa) e os custos são calculados (quarta etapa).

Através do conjunto de resultados, o projetista deve verificar se os resultados cumprem os seus requisitos. Se os valores estimados são viáveis, a infraestrutura existente é aceita. Caso contrário, se a arquitetura não atender aos requisitos, o projetista deve propor outra alternativa.

A etapa de avaliação da arquitetura compreende a análise do efeito dos eventos de falhas, das atividades de reparo e desempenho no sistemas. Nessa etapa, técnicas de avaliação

Figura 4.1 Fluxo para avaliação da performabilidade.

são aplicadas levando em consideração os requisitos da infraestrutura disponíveis. Além disso, a abordagem de modelagem proposta pode também ser utilizada para a avaliação de uma variedade de centros de emergência. A Figura 4.1 mostra o diagrama de fluxo utilizado para avaliação de performabilidade do *call center* de emergência.

4.1.1 Estratégia de Decomposição e Composição

A análise de performabilidade do sistema é realizada através da composição dos resultados obtidos da avaliação da disponibilidade e desempenho. Uma técnica de decomposição hierárquica divide o modelo de performabilidade em dois modelos distintos [Haverkort e Niemegeers 1996].

A estratégia adotada combina um modelo de disponibilidade, o qual considera eventos de falhas e os processos de reparo no sistema, a um conjunto de modelos de desempenho [Sahner R. e Puliafito 1996] e [Puliafito, Riccobene e Scarpa 1996].

A análise de performabilidade da infraestrutura do sistema descreve o efeito da disponibilidade no desempenho do sistema através de métricas de avaliação [Sousa 2009]. Essas métricas são calculadas, independentemente, a partir dos modelos e posteriormente combinadas para mostrar o efeito da disponibilidade no desempenho do *call center* de emergência. A Figura 4.2 apresenta o diagrama de atividades da metodologia para Decomposição e Composição.

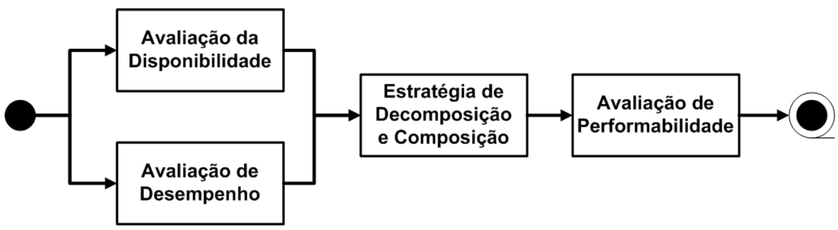


Figura 4.2 Fluxo para decomposição e composição [Sousa 2009].

4.2 AVALIAÇÃO DA DISPONIBILIDADE

A análise de performabilidade do sistema foi realizada através da composição dos resultados obtidos através dos modelos desenvolvidos para a avaliação da disponibilidade e desempenho conforme apresentado na Seção 4.1. Para avaliar a disponibilidade da arquitetura foi necessário realizar reduções no modelo RBD devido a sua complexidade.

Uma redução em série foi aplicada no SubModelo (B) nos blocos SubModeloParalelo*PowerStrip* e PABX obtendo o bloco B1. Em seguida, foi aplicada outra redução em série nos blocos Roteador e *Switch* resultando no bloco B2 e outra redução nos blocos SubModeloParalelo(PMilitar, PCivil, DBombeiro e PCientífica) obtendo o bloco B3. Esse processo de redução é demonstrado na Figura 4.3(a).

Além disso, foi aplicada redução em paralelo nos blocos B2 e B3, conforme ilustrado na Figura 4.3(b), resultando no bloco B4 (com taxas correspondentes a estrutura em paralelo). Logo após, foi aplicada outra redução em série nos blocos B1 e B4 apresentados na Figura 4.3(c). O SubModelo (B) foi substituído por um único bloco equivalente.

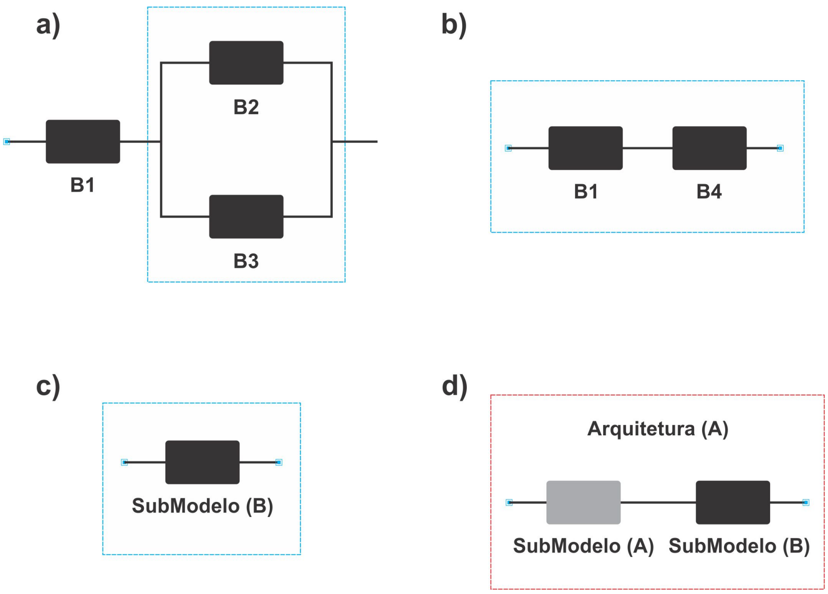


Figura 4.3 Redução do modelo RBD da arquitetura (A).

Dessa forma, os valores dos MTTFs, fornecidos pelos fabricantes, e MTTRs, estabelecidos no SLA, apresentados na Tabela 4.1 foram atribuídos ao RBD e SPN. Depois desse processo, utilizando a modelagem hierárquica apresentada no Capítulo 3, os modelos consistem em somente dois blocos (ver Figura 4.3(d)) conectados em série denominado arquitetura (A). Sendo assim, foi necessário obter o valor do MTTF do modelo SPN da Figura 3.4, que corresponde a 15631094,2193 horas e que foi obtido através da Equação (2.4). O MTTR foi determinado através da Equação (2.6) e o valor obtido foi 1708,9780 horas. A disponibilidade da arquitetura (A) corresponde a 99.959%, o *downtime* é igual a 3 hora e 35 minutos por ano.

Tabela 4.1 MTTFs e MTTRs dos blocos dos SubModelos (A) e (B).

SubModelo		MTTF(horas)	MTTR(horas)
(A)	SubSistema	732249,5012	82,0075
	UPS	250000	8
(B)	*Switch*	26298	4,37
	Roteador	35064	4
	PABX	39598	2
	SubModeloParalelo*PowerStrip*	3.4871×10^{13}	3,8
	SubModeloParaleloPMilitar	696,7710	954,3153
	SubModeloParaleloPCivil	3110,5848	1477,3897
	SubModeloParaleloDBombeiro	3110,5848	1477,3897
	SubModeloParaleloPCientífica	1915,7483	1548,3278

De acordo com a classificação de disponibilidade apresentada em [Gray e Siewiorek 1991], pode-se verificar que a arquitetura está entre as classes bem-gerenciada e tolerante a falha. Portanto, o *downtime* é considerado alto, uma vez que se trata de um *call center* de emergência e o número diário de chamadas espera-se que aumente, pois nos próximos anos o estado receberá um número maior de turistas, profissionais bem como realizar eventos importantes.

Considerando o alto *downtime* do *call center* de emergência, a arquitetura deve ser melhorada para fornecer energia e assim, manter os computadores e equipamentos de telecomunicação necessários funcionando para atender as solicitações de emergência. A perda de energia (*downtime*) por 3h 35min interrompe toda a operação do sistema e, sendo assim, muitas ligações são perdidas e a população fica sem o atendimento que necessita urgentemente.

O índice RCI determina a importância do componente para a confiabilidade do sistema em função do seu custo de aquisição. Para obter o valor do RCI de cada equipamento do sistema, é necessário encontrar o valor do RI deles. O RCI permitiu estabelecer comparações entre os componentes da arquitetura e ajudou a determinar quais deveriam ser adicionados ou removidos para aumentar a disponibilidade do sistema. Os valores do RI e RCI da arquitetura (A) são apresentados na Tabela 4.2.

Quando ocorre a falha no módulo SubSistema, o módulo UPS continua a fornecer eletricidade (representa a Estrutura de Energia) com autonomia de duas horas, somente quando descarrega é que o sistema vai para *down*. Sendo assim, é necessário determinar o RCI da arquitetura (A) em dez anos, pois a cada intervalo de tempo igual a esse período, os equipamentos mais importantes do sistema são substituídos.

A partir dos resultados da Tabela 4.2, é possível concluir que SubSistema é um módulo importante do modelo SPN uma vez que ele possui RI=1,0 e RCI=0,4712. Se ocorrer uma falha nesse módulo, a UPS é usada ininterruptamente e sem oscilações, porém ela possui uma autonomia de 2 horas e seu tempo de carga é de 48 horas. Após esse período, o *call center* de emergência se torna inoperante até que o componente seja reparado.

Além disso, alguns imprevistos como componentes queimados ou danificados durante a manutenção do módulo UPS podem provocar o *downtime*, pois não possui redundância. Considerando os motivos expostos, é proposta a aquisição de um gerador.

Tabela 4.2 RI e RCI da arquitetura (A) no período de dez anos.

SubModelo		(RI)	(RCI)	Custo
(A)	SubSistema	1,0	0,4712	$8.416,00
	UPS	0,3814	0,2017	$7.500,00
(B)	*Switch*	1,0	0,9995	$3.148,00
	Roteador	0,4348	0,4346	$3.899,00
	PABX	0,3267	0,2529	$1.468.515,04
	SubModeloParalelo*PowerStrip*	0,0358	0,0358	$119,98
	SubModeloParaleloPMilitar	-	-	$10.384,50
	SubModeloParaleloPCivil	-	-	$2.112,50
	SubModeloParaleloDBombeiro	-	-	$2.112,50
	SubModeloParaleloPCientífica	-	-	$1.078,50

A análise dos valores do RCI do RBD da arquitetura mostrou que o Switch (RI=1.0 e RCI=0.9995), Roteador (RI=0.4348 e RCI=0.4346), PABX (RI=0,3267 e RCI=0,2529) e o SubModeloParalelo*PowerStrip* (RI=0,0358 e RCI=0,0358) são os mais importantes, e, portanto, é necessário adicionar redundância nesses blocos.

Baseado nos resultados do RCI, neste trabalho foi proposta uma nova arquitetura (B) que consiste em algumas mudanças na arquitetura (A). Propõe-se adicionar redundância no *Switch*, Roteador e no SubModeloParalelo*PowerStrip*, e assim, o sistema consistiria em dois *switches* conectados em paralelo, dois roteadores em paralelo e quatro *PowerStrip* conectados em paralelo. Contudo, não é possível aplicar redundância no PABX, dado seu alto custo. Por esta razão, foi proposta uma solução com um custo muito menor que consiste na alteração do contrato de manutenção para reduzir o MTTR e firmar os novos requisitos de manutenabilidade no *Service Level Agreement* (SLA) do respectivo contrato. No novo contrato, a cláusula do SLA deve especificar que o MTTR não deve ser superior a uma hora.

A modelagem SPN da arquitetura (B), apresentada na Figura 4.4, foi proposta com o objetivo de aumentar a disponibilidade do *call center* de emergência. O modelo tem um gerador que assegura redundância para o suprimento contínuo de energia em caso de falha na estrutura de energia. A escolha desse componente ocorreu devido a seu baixo custo de aquisição e manutenção, em vez de contratar outra companhia de energia cuja instalação é, em geral, inviável. Além disso, é importante salientar que as dependências entre os demais blocos do SubModelo (B) não foram alteradas.

O modelo SPN que representa a Estrutura de Energia é composto pelos módulo principal SubSistema. Por outro lado, Gerador e UPS são os módulos de redundância não ativos. O *call center* estará *down* se todos os três módulos estiverem *down* ao mesmo tempo. Foi proposto aumentar a autonomia do módulo UPS de 2 para 6 horas, assim os administradores seriam contemplados com mais tempo para resolver os problemas com o gerador e os outros componentes sem interromper sua operação.

A Tabela 4.3 apresenta os valores do MTTF e MTTR do módulo Gerador da arquitetura (B).

Tabela 4.3 MTTFs e MTTRs do módulo Gerador da arquitetura (B).

Gerador	
MTTF (horas)	43830
MTTR (horas)	3,67

É importante observar que quando o módulo SubSistema falha, ocorre o disparo automático do módulo Gerador. Entretanto, existe um tempo necessário para que o sistema entre em funcionamento. Este tempo é imperceptível para o usuário uma vez que o módulo UPS proporciona eletricidade contínua e sem interrupções por um período de 6 horas.

A Figura 4.4 apresenta o modelo SPN que inclui dez lugares: SubSistema_On, SubSistema_Off, Gerador_On, Gerador_Off, Gerador_Standby, UPS_On, UPS_Off, UPS_Standby, Bateria_Descarregado e Esperar. Esses lugares representam os estados operacional e de falha tanto do módulo principal quanto dos módulos de redundância, respectivamente. Além disso, o lugar Bateria_Descarregado representa a descarga da bateria e o lugar Esperar representa o retorno do fornecimento de energia possibilitando a recarga da bateria no modelo.

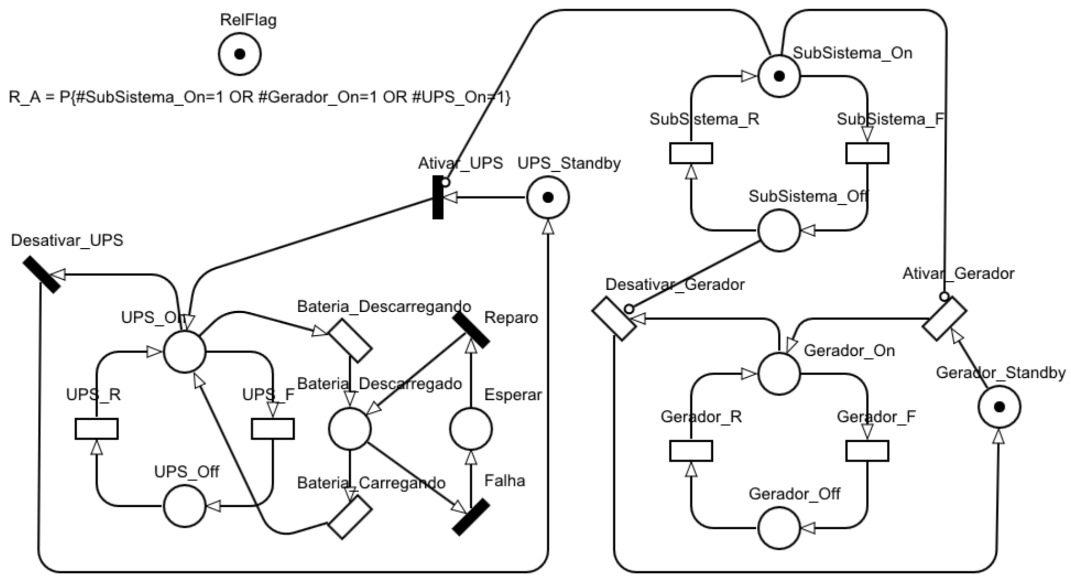


Figura 4.4 SubModelo (A) da arquitetura (B).

A falha do módulo principal (SubSistema) é representada pelo disparo da transição Ativar_UPS. O disparo dessa transição representa o início da operação, torna o lugar UPS_On marcado (representa o fornecimento de energia contínua) e é descrito por uma transição imediata (im) uma vez que o tempo de disparo é igual a zero. Logo, o disparo da transição Ativar_Gerador ocasiona uma marca no lugar Gerador_On e é descrito por uma transição temporizada (exp), pois o tempo de disparo é de aproximadamente dois minutos [Maciel P. e Fernandes 2007]. Após o disparo da transição Ativar_Gerador, o *token* no lugar UPS_On retorna para o lugar UPS_Standby através do disparo da transição imediata Desativar_UPS.

Quando os lugares SubSistema_Off e Gerador_Off estiverem marcados ocorrerá o disparo da transição Falha tornando o lugar Esperar marcado. Por outro lado, a transição Reparo (representa a recarga da bateria assim que o fornecimento de energia for restabelecido) será habilitada quando o lugar SubSistema_On ou Gerador_On estiver marcado. Além disso, as transições imediatas Bateria_Descarregando e Bateria_Carregando têm uma duração de 6 horas e 144 horas respectivamente. As transições temporizadas, Gerador_F, Gerador_R, Desativar_Gerador, UPS_F, UPS_R, Bateria_Descarregando e Bateria_Carregando têm uma distribuição exponencial (exp) enquanto que Desativar_UPS, Falha e Reparo são transições imediatas (im). Os atributos relacionados com cada transição do modelo descrito neste capítulo são apresentados na Tabela 4.4.

Tabela 4.4 Atributos e transições do SPN da arquitetura (B).

Transição	**Tipo**	**Delay ou Peso**
SubSistema_F	exp	MTTF
SubSistema_R	exp	MTTR
Ativar_Gerator	exp	Tempo Médio Ativação
Desativar_Gerator	exp	Tempo Médio Desativação
Gerador_F	exp	MTTF
Gerador_R	exp	IF #RelFlag=1: ($10^{n=8}\times$MTTF) ELSE MTTR;
UPS_F	exp	MTTF
UPS_R	exp	IF #RelFlag=1: ($10^{n=8}\times$ MTTF) ELSE MTTR;
Bateria_Descarregando	exp	TempoDescarga
Bateria_Carregando	exp	IF #RelFlag=1: ($10^{n=8}\times$ TempoDescarga) ELSE TempoCarga;
Falha	(SubSistema_On=0 AND Gerador_On=0)	
Reparo	SubSistema_On=1 OR Gerador_On=1	
Ativar_UPS	Gerador_On=0 OR (SubSistema_Off=1 AND Gerador_Off=1)	
Desativar_UPS	(SubSistema_Off=1 AND Gerador_On=1) OR SubSistema_On=1	

A disponibilidade do SubModelo (A) da arquitetura (B) foi determinada através da expressão P{#SubSistema_On =1 OU #Gerador_On =1 OU #UPS_On =1} no modelo SPN sendo #RelFlag=0 (análise estacionária). Por outro lado, se #RelFlag=1, essa mesma expressão permite obter a confiabilidade, caso uma análise transiente seja realizada. O valor do MTTF foi obtido através da Equação (2.4) e seu valor corresponde a 61579973,9852 horas. O MTTR foi determinado através da Equação (2.6) e seu valor é igual a 0,5009 horas.

Os valores do MTTF, MTTR, disponibilidade, *downtime* e custo de propriedade das arquiteturas (A) e (B) são apresentados na Tabela 4.5. Segundo a classificação usada neste trabalho, disponível em [Gray e Siewiorek 1991], verifica-se que a arquitetura (B) pertence à classe alta disponibilidade.

Tabela 4.5 Valores do MTTF, MTTR, Disponibilidade e CO das arquiteturas (A) e (B).

	Arq(A) bem-gerenciada	**Arq(B) alta-disponibilidade**
MTTF Sistema (horas)	11.792,4123	174.561,1506
MTTR Sistema (horas)	4,8354	1,0034
Disponibilidade(%)	99,959	99,999
Downtime (hora/ano)	3h 35min	5min
CO (US$)	US$6.503.432,02	US$8.379.171,98

Para determinar o custo de propriedade (CO) do sistema, foi considerado um período de um ano e utilizada a Equação (2.44). Este cálculo envolve o custo de aquisição de *hardware* e *software* do sistema (uma única vez), os custos de suporte (mensal), a taxa de inflação e o período analisado. Portanto, foi realizada uma cotação em dólar do custo de aquisição dos componentes da arquitetura (A). O custo total do *hardware* obtido foi de US$ 1.475.682,02, o custo total do *software* foi de US$ 23,000.00. Por outro lado, o custo de suporte mensal do sistema é US$ 13.346,00. Além disso, a taxa de inflação utilizada foi de 3,2% a.a. no ano base de 2011 [Bureau Labor Statistics 2011].

Os valores dos custos de aquisição de *hardware*, *software* e suporte mensal obtidos através da cotação e a taxa de inflação foram substituídos na Equação (2.44) obtendo o valor de CO igual a US$ 6.503.432,02. O CO da arquitetura (B) foi determinado de uma maneira semelhante de arquitetura (A). Os valores de cada variável da Equação (2.44) bem como o valor final de CO de ambas as arquiteturas são mostrados na Tabela 4.6.

Tabela 4.6 Custo de Propriedade das arquiteturas (A) e (B) em dólar.

	Arquitetura (A)	**Arquitetura (B)**
H_{ware}	US$1.475.682,02	US$1.499.421,98
S_{ware}	US$23.000,00	US$0,00
$C_{suporte}$	US$13.346,00	US$18.346,00
Período n	12 months	
Taxa de Inflação i	3,2% a.a.	
Total	US$6.503.432,02	US$8.379.171,98

O MTTF da arquitetura (B) é maior do que arquitetura (A), o que significa que (B) leva mais tempo a falhar porque tem redundância nos blocos com maior RCI. O *downtime* da arquitetura (B) é de 5 minutos o que significa que seu valor é 43 vezes menor que o *downtime* da arquitetura (A).

4.3 AVALIAÇÃO DE DESEMPENHO

[Silva 2010] conduziu uma análise de desempenho do *call center* de emergência. Nesse trabalho, um estudo do comportamento do sistema foi realizado para suportar a tomada de decisão no *call center*. Assim, o impacto do *downtime* na quantidade de chamadas foi avaliado em relação ao número de descartes.

[Silva 2010] analisou as chamadas no período de Janeiro de 2008 a Dezembro de 2009 uma vez que esse período possui maior intensidade de ligações permitindo que a amostra fosse composta por uma ampla gama de dados. Sendo assim, [Silva 2010] concluiu que nos dias úteis, o número de chamadas varia entre 6.500 a 8.500 diariamente. Nos finais de semana, o número de chamadas aumenta entre 9.500 e 10.500 chamadas por dia. Portanto, o número médio de chamadas por dia foi obtido através de uma média ponderada e corresponde a 8.219,18.

Além disso, [Silva 2010] afirma que uma média de 41% do número total de 3.000.000,70 chamadas no ano em dias úteis dos meses estudados consistiram em trotes (Figura 4.5). O foco dessa Dissertação é o número de chamadas totais versus quantidade de agentes e válidas por departamento do *call center* de emergência: Polícia Militar, Civil, Científica e Bombeiros.

A Figura 4.5 apresenta as probabilidades de chamada de cada categoria (válidas, trotes, erradas e informação). Além disso, são apresentadas as probabilidades de cada tipo de chamada válida por categoria (Polícia Militar, Civil, Científica e Bombeiros).

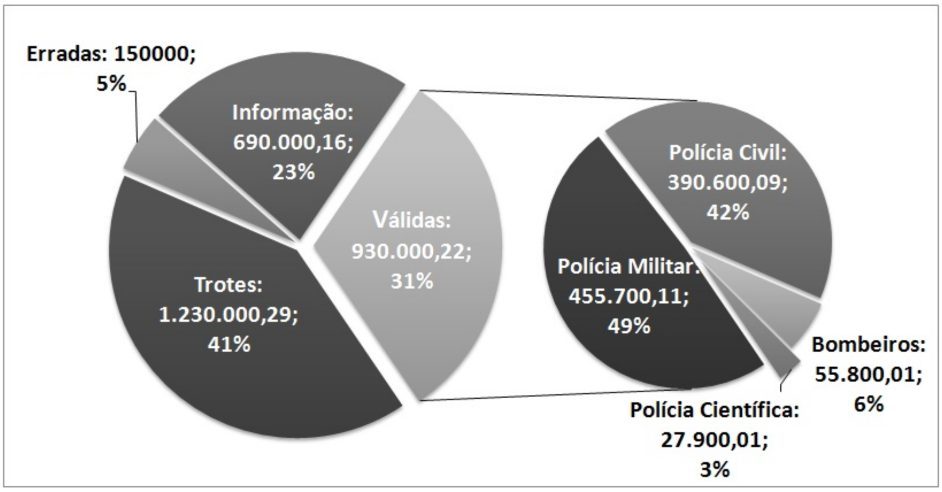


Figura 4.5 Classificação das chamadas do *call center* de emergência.

Para determinar o número total de chamadas perdidas devido ao *downtime* e descarte, foram utilizados os modelos de disponibilidade das arquiteturas (A) e (B), bem como o modelo de desempenho (mostrado na Figura 3.5). Os resultados dessa análise estão apresentados nas Figuras 4.6, 4.7 e 4.8.

A Figura 4.6 apresenta o número total de chamadas válidas perdidas devido ao *downtime* por ano em ambas as arquiteturas. Na arquitetura (A) o número total de chamadas válidas perdidas por ano é 386,5959 e seu *downtime* é de 3 hora e 35 minutos por ano. Por outro lado, a arquitetura (B) apresenta 9,4292 chamadas válidas perdidas por ano e seu *downtime* é de apenas 5 minutos por ano.

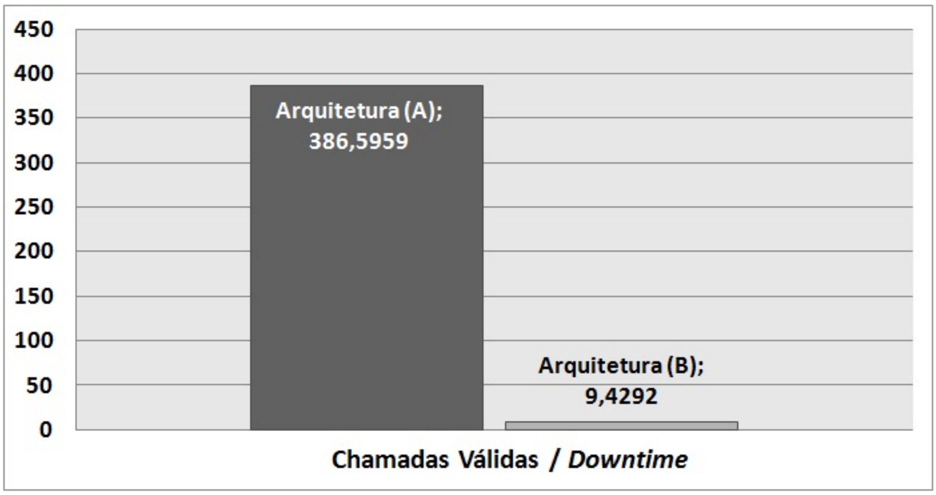


Figura 4.6 Total de chamadas válidas perdidas por ano devido ao *downtime* das arquiteturas (A) e (B).

É importante analisar o número de chamadas válidas por ano em cada categoria (Polícia Militar, Civil, Científica e Bombeiros) devido somente ao *downtime* e considerando 15 agentes atendendo as chamadas porque esse resultado mostra o impacto da disponibilidade nas arquiteturas (A) e (B). Esses resultados estão apresentados na Figura 4.7.

É possível concluir, após analisar esse gráfico, que a arquitetura (B) proposta proporciona uma redução significante no número de chamadas válidas perdidas porque possui uma performabilidade maior.

A Figura 4.8 mostra o número total de chamadas descartadas por ano nas arquiteturas (A) e (B) para 15 a 90 agentes. Esses resultados mostram que o número de descartes diminui consideravelmente devido ao aumento do número de agentes.

A quantidade de descartes da arquitetura (B) é maior que a quantidade da arquitetura (A) devido a esta arquitetura possuir um *downtime* de apenas 5 minutos, isto é, o sistema funciona por mais tempo que a arquitetura (A) e recebe um número maior de chamadas. Por outro lado, como o *downtime* da arquitetura (A) é de 3 hora e 35 minutos, o sistema descarta menos chamadas, contudo, o número de chamadas perdidas devido ao *downtime* é maior.

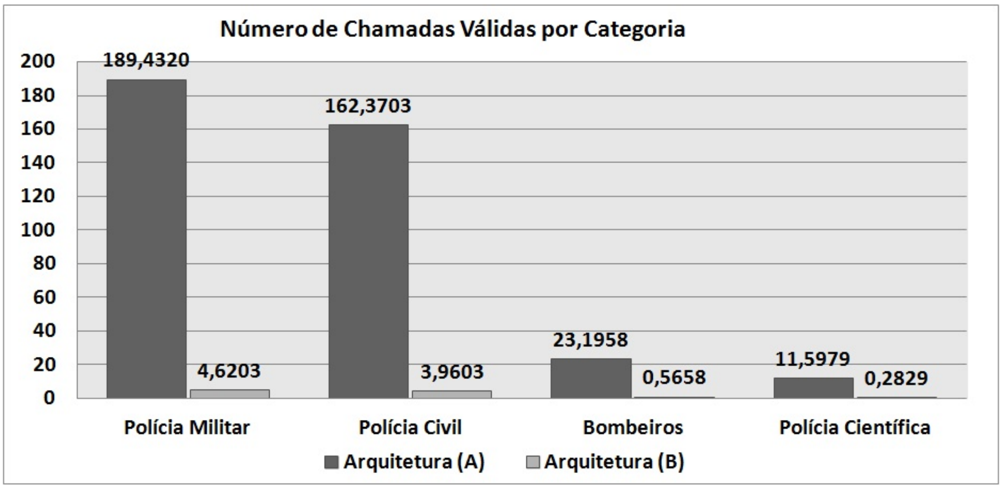


Figura 4.7 Número de chamadas válidas perdidas por ano em cada categoria em ambas arquiteturas devido ao *downtime*.

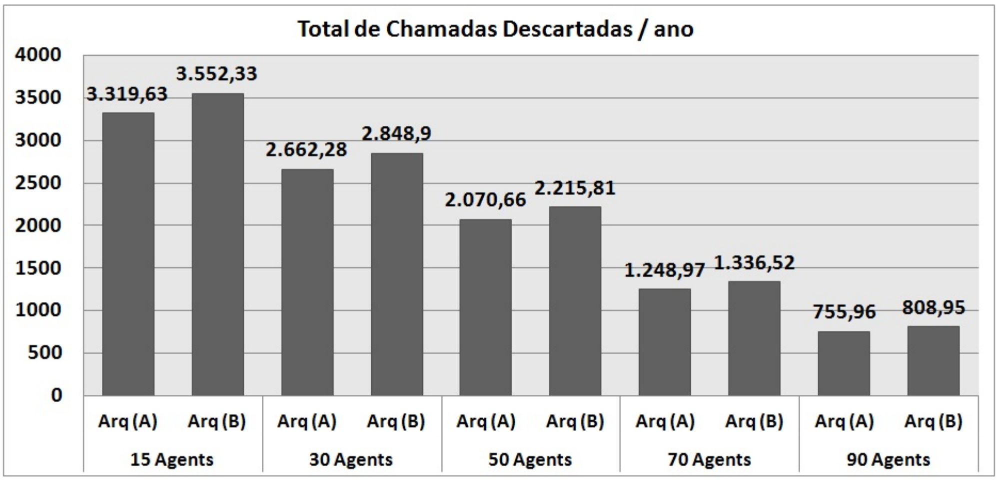


Figura 4.8 Número de chamadas perdidas devido ao descarte nas arquiteturas (A) e (B) para vários agentes.

A Figura 4.9 apresenta a taxa em porcentagem de utilização dos agentes, considerando o *downtime* nas duas arquiteturas, atendendo 3.000.000,70 chamadas de emergência por ano. A partir da análise da Figura 4.9, verifica-se que entre 30 a 50 agentes a porcentagem de utilização cresce notadamente. Além disso, verificou-se que entre 70 a 100 agentes, a taxa de utilização começa a se estabilizar.

Os resultados da avaliação de desempenho da arquitetura (A) indicam que o número de chamadas perdidas por ano devido ao *downtime* é um valor alto uma vez que se trata de

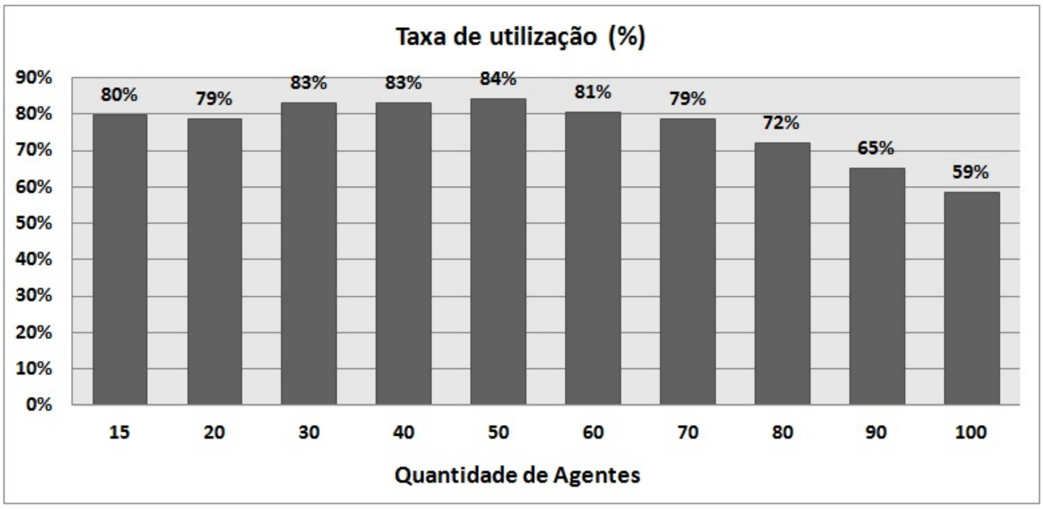


Figura 4.9 Taxa de utilização dos agentes em relação ao *downtime* nas aquiteturas (A) e (B).

pessoas que necessitam do atendimento do serviço de emergência. Os resultados da análise da disponibilidade da arquitetura (A) mostram que o MTTF e MTTR correspondem a 11.792,4123 e 4,8354 horas, respectivamente. Esses valores implicam em aproximadamente três noves de disponibilidade que resulta em 3 hora e 35 minutos de *downtime* por ano e provoca um descarte de aproximadamente 386,5959 chamadas válidas por ano.

Por outro lado, o MTTF da arquitetura (B) é maior que o da arquitetura (A), o que significa que ela demora mais tempo para falhar uma vez que foi adicionada redundância em seus componentes. A disponibilidade da arquitetura (B) é de cinco noves, o que provoca um *downtime* de 5 minutos por ano e é responsável pelo descarte de 9,4292 chamadas válidas por ano.

Portanto, os resultados da arquitetura (B) mostraram-se mais eficientes que os da arquitetura (A) uma vez que eles proporcionam uma redução de 97.56% no número de chamadas perdidas por ano devido ao *downtime*. O investimento requerido para a implantação completa da arquitetura (B), cuja disponibilidade é 99.999%, é US$8.379.171,98 em um ano, assumindo que a taxa de inflação é de 3.2% a.a. Esse valor corresponde a uma diferença de aproximadamente US$1.875.739,96 em relação ao custo de propriedade da arquitetura (A) que possui disponibilidade de 99.959%.

4.4 CONSIDERAÇÕES FINAIS

Este capítulo apresentou a avaliação da dependabilidade do *call center* de emergência através das métricas MTTF, MTTR, Disponibilidade, *Downtime* e CO. Além disso, foram propostas mudanças na arquitetura do sistema de forma a aumentar a sua disponibilidade, e, consequetemente reduzir o *downtime*. Também foi apresentado o modelo do novo subsistema.

Este capítulo apresentou também os resultados da avaliação de desempenho do sistema através das métricas: número total de chamadas válidas perdidas devido ao *downtime*, número de chamadas válidas por cada tipo de categoria e do número de chamadas descartadas por ano.

Finalmente, foi realizada uma comparação entre a arquitetura (A) do *call center* e a arquitetura (B) proposta, que é a principal contribuição deste trabalho, destacando a sua eficiência e capacidade de atender mais usuários do *call center* de emergência.

CAPÍTULO 5

CONCLUSÕES E TRABALHOS FUTUROS

O crescimento populacional expressivo nos últimos anos provocou o aumento dos índices de desastres, violência e outros delitos. Consequentemente, existe uma grande demanda pelos serviços de emergência e o *call center* é o primeiro contato que a população realiza. O usuário quer velocidade no atendimento e melhor desempenho, mas ele vem apenas com serviços confiáveis. Isso exige uma reformulação fundamental do modelo de desempenho tradicional, que ignora falha ou o processo de recuperação e concentra-se, principalmente, na contenção de recursos. Para implementar um sistema com aplicação prática, questões sobre capacidade, disponibilidade e desempenho devem ser consideradas de forma integrada.

Este trabalho propôs modelos para o estudo de performabilidade do *call center* de emergência. Além disso, foram computados os custos (CO) da arquitetura, através de um modelo apresentado para análise da eficácia dos gastos. Além disso, aplica a importância para a confiabilidade e custos (RCI) para determinar os componentes são mais importantes do sistema, sendo possível relacioná-lo ao custo do sistema (CO) e decidir quais componentes são importantes para o fun-cionamento contínuo do sistema. Ao avaliar a disponibilidade, desempenho e custo dos modelos, é possível obter medidas de performabilidade, tais como MTTF, MTTR, CO, RCI bem como a quantidade de chamadas perdidas devido aos descartes e ao *downtime*. Além disso, pode-se obter também o número de chamadas válidas perdidas e o impacto do número de agentes sobre o total de chamadas perdidas. Este trabalho visa aliar a avaliação de performabilidade e custos de *call center* de emergência e destaca a importância dos componentes em relação ao desempenho do sistema. Almeja-se, assim, ser capaz de projetar sistemas que atendam aos requisitos de desempenho e minimize custos de implantação e operação.

O estudo de caso realizado compreendeu a avaliação dos modelos em termos de performabilidade. Os resultados obtidos mostraram que o *downtime* é de uma hora e trinta e sete minutos e provoca um descarte considerável de chamadas válidas perdidas por ano, por isso, baseado nos resultados da arquitetura (A) foi proposta uma nova arquitetura (B) para melhorar a performabilidade do sistema com *downtime* de cinco minutos por ano, sendo responsável pela redução de descarte de chamadas válidas por ano. No entanto, a implementação da nova arquitetura requer investimento para efetuar as mudanças necessárias. Por outro lado, a análise dos resultados apresentados da performabilidade da arquitetura (B) justifica o investimento. Dessa forma, os projetistas têm a possibilidade de escolher, de acordo com os requisitos e orçamento, qual arquitetura melhor se adequa às necessidades. Os resultados obtidos neste trabalho podem ser usados para fornecer suporte para as decisões sobre as intervenções no *call center* para melhorar a sua performabilidade. Espera-se que os modelos apresentados no presente documento sejam úteis em uma variedade de *call centers* de emergência.

5.1 CONTRIBUIÇÕES

As principais contribuições desse trabalho consistem nas proposições de:

- a adaptação de uma metodologia para avaliação de performabilidade de *call centers* de emergência, composta por um método para avaliação de disponibilidade, um método para avaliação de desempenho e uma técnica de decomposição e composição. Os métodos têm o objetivo de avaliar o desempenho considerando a ocorrência de eventos de falhas, atividades de reparo e custo de propriedade (CO) tendo como base o menor número de descartes e *downtime* e dimensionamento adequado de agentes no *call center* de emergência.
- um modelo hierárquico para representar os componentes e as falhas do *call center* de emergência para avaliação de performabilidade. Os modelos foram utilizados para predizer o desempenho e a disponibilidade do sistema e determinar os componentes críticos a partir do ponto de vista da importância para a confiabilidade e custos (RCI).

5.2 TRABALHOS FUTUROS

Como trabalhos futuros, propõe-se estender este trabalho para outros atributos da avaliação da dependabilidade do *call center* de emergência. Sugere-se a avaliação da confiabilidade para determinar a probabilidade do sistema executar sua função por um intervalo específico sob condições pré-estabelecidas.

Sugere-se ainda como trabalho futuro, propor um atributo de manutenabilidade utilizando o modelo de performabilidade proposto aqui para determinar o quão rapidamente e economicamente as falhas podem ser evitadas através de manutenção preventiva ou a operação do sistema pode ser restaurada após a falha.

O controle da umidade em *call centers* é essencial para alcançar alta disponibilidade. Ar contendo muito ou pouco vapor de água contribui diretamente para redução da produtividade e *downtime*. Existe uma relação de interdependência entre a gestão de umidade e procedimentos de gestão de refrigeração de ar uma vez que a desumidificação do ar sempre reduz a capacidade de TI de remoção de calor. Quando projetado corretamente, um sistema de gerenciamento de ar pode reduzir os custos operacionais, reduzir o investimento de aquisição de novos equipamentos, aumentar a densidade da potência (watts/metro quadrado) do *call center* e reduzir as interrupções de calor relacionados processamento ou falhas.

REFERÊNCIAS BIBLIOGRÁFICAS

[Aguir et al. 2004]AGUIR, S. et al. The impact of retrials on call center performance. *OR Spectrum*, Springer Berlin / Heidelberg, v. 26, p. 353–376, 2004. ISSN 0171-6468.

[Arno, Gross e Schuerger 2006]ARNO, R.; GROSS, P.; SCHUERGER, R. What five 9's really means and managing expectations. In: *Industry Applications Conference, 2006. 41st IAS Annual Meeting. Conference Record of the 2006 IEEE*. [S.l.: s.n.], 2006. v. 1, p. 270 –275. ISSN 0197-2618.

[Arno, Stoyas e Schuerger 2011]ARNO, R.; STOYAS, E.; SCHUERGER, R. Nec article 708. *Industry Applications Magazine, IEEE*, v. 17, n. 1, p. 20–25, jan.-feb. 2011. ISSN 1077-2618.

[Association 2008]ASSOCIATION, N. F. P. *NFPA 70: National Electrical Code*. 2008. ed. Massachusetts: Thomson Learning, 2008.

[Athey e Stern 1998]ATHEY, S.; STERN, S. The adoption and impact of advanced emergency response services. *NBER Working Paper No. w6595. Massachusetts Avenue Cambridge*, June 1998.

[Avizienis, Laprie e Randell 2001]AVIZIENIS, A.; LAPRIE, J.; RANDELL, B. *Fundamental Concepts of Dependability*. 2001. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.6074>.

[Bolch G. Greiner e Trivedi 2006]BOLCH G. GREINER, S. M. H.; TRIVEDI, K. Queueing networks and markov chains: Modeling and performance evaluation with computer science applications. John Wiley & Sons, 2006.

[Bureau Labor Statistics 2011]Bureau Labor Statistics. *Databases, Table e Calculators by Subject*. 2011. Acesso em: 10 dez. 2011. Disponível em: <http://www.bls.gov/data/inflation_calculator.htm>.

[Ciardo e Trivedi 1993]CIARDO, G.; TRIVEDI, K. S. A decomposition approach for stochastic reward net models. *Perform. Eval.*, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 18, n. 1, p. 37–59, jul. 1993. ISSN 0166-5316. Disponível em: <http://dx.doi.org/10.1016/0166-5316(93)90026-Q>.

[Corporation 2003]CORPORATION, R. System analysis reference: Reliability availability and optimization. *ReliaSoft Publishing*, 2003.

[David, Schuff e Louis 2002]DAVID, J. S.; SCHUFF, D.; LOUIS, R. S. Managing your total it cost of ownership. *Commun. ACM*, ACM, New York, NY, USA, v. 45, n. 1, p. 101–106, jan. 2002. ISSN 0001-0782. Disponível em: <http://doi.acm.org/10.1145/502269.502273>.

[Desrochers e Al-Jaar 1995]DESROCHERS, A.; AL-JAAR, R. *Applications of Petri Nets in Manufacturing Systems: Modeling, Control, and Performance Analysis*. New York: IEEE Press, 1995.

[Dhillon 2002]DHILLON, B. *Engineering maintenance: a modern approach*. United States of America: CRC Press, 2002.

[Distefano 2008]DISTEFANO, S. Investigating fault tolerant computing systems reliability. In: *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*. [S.l.: s.n.], 2008. p. 1–8. ISSN 1530-2075.

[Fanaeepour, Naghavian e Azgomi 2007]FANAEEPOUR, M.; NAGHAVIAN, L.; AZGOMI, M. Modeling and evaluation of call centers with gspn models. In: *Computer Systems and Applications, 2007. AICCSA '07. IEEE/ACS International Conference on*. [S.l.: s.n.], 2007. p. 619–622.

[Figueiredo et al. 2011]FIGUEIREDO, J. et al. Estimating reliability importance and total cost of acquisition for data center power infrastructures. In: *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*. [S.l.: s.n.], 2011. p. 421–426. ISSN 1062-922X.

[Florin e Natkin 1989]FLORIN, G.; NATKIN, S. Matrix product form solution for closed synchronized queuing networks. In: *Petri Nets and Performance Models, 1989. PNPM89., Proceedings of the Third International Workshop on*. [S.l.: s.n.], 1989. p. 29–37.

[German 2000]GERMAN, R. *Performance Analysis of Communicating Systems - Modeling with Non-Markovian Stochastic Petri Nets*. New York: John Wiley & Sons, 2000.

[Gray e Siewiorek 1991]GRAY, J.; SIEWIOREK, D. P. High-availability computer systems. *Computer*, IEEE Computer Society Press, Los Alamitos, CA, USA, v. 24, n. 9, p. 39–48, set. 1991. ISSN 0018-9162. Disponível em: <http://dx.doi.org/10.1109/2.84898>.

[Group 2011]Group. *Advisory Company*. 2011. Acesso em: 15 dez. 2011. Disponível em: <http://www.gartner.com>.

[Haverkort e Niemegeers 1996]HAVERKORT, B. R.; NIEMEGEERS, I. G. Performability modelling tools and techniques. *Performance Evaluation*, v. 25, n. 1, p. 17–40, 1996.

[Jain 1991]JAIN, R. *The Art of Computer Systems Performance Analysis*. New York: John Wiley & Sons, 1991.

[Kleinrock 1975]KLEINROCK, L. *Queueing systems volume 1: Theory*. New York: John Wiley & Sons, 1975.

[Kuo e Zuo 2003]KUO, W.; ZUO, M. *Optimal reliability modeling: principles and applications*. New Jersey: John Wiley & Sons, 2003.

[Lilja 2000]LILJA, D. *Measuring Computer Performance: A Practitioner's Guide*. United Kingdon: Cambridge University Press, 2000.

[Maciel P. R. M. e Cunha 1996]MACIEL P. R. M., L. R. D.; CUNHA, P. R. F. *Uma Introdução às Redes de Petri e Aplicações*. Campinas-SP: Sociedade Brasileira de Computação, 1996.

[Maciel P. e Fernandes 2007]MACIEL P., R. N.; FERNANDES, S. Reliability and availability evaluation of communication networks. Preprint submitted to Elsevier, 2007.

[Maciel P. e Kim 2011]MACIEL P., T. K. J. R.; KIM, D. Performance and dependability in service computing: Concepts, techniques and research directions. In: _____. United States of America: IGI Global, 2011. cap. Dependability Modeling, p. 53–97.

[Merlin e Farber 1976]MERLIN, P.; FARBER, D. Recoverability of communication protocols–implications of a theoretical study. *Communications, IEEE Transactions on*, v. 24, n. 9, p. 1036–1043, sep 1976. ISSN 0090-6778.

[Molloy 1981]MOLLOY, M. K. *On the integration of delay and throughput measures in distributed processing models*. Tese (Doutorado), 1981. AAI8201138.

[Mundi, I. 2011]Mundi, I. *Site contains detailed country statistics, charts, and maps compiled from multiple sources*. 2011. Acesso em: 10 dez. 2011. Disponível em: <http://www.indexmundi.com>.

[Murata 1989]MURATA, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, v. 77, n. 4, p. 541–580, apr 1989. ISSN 0018-9219.

[Nielsen 2010]NIELSEN, T. B. *Call center capacity planning*. Tese (Doutorado) — Technical University of Denmark, Denmark, 2010.

[O'Reilly, Richman e Kelic 2007]O'REILLY, G.; RICHMAN, S.; KELIC, A. Power, telecommunications, and emergency services in a converged network world. In: *Design and Reliable Communication Networks, 2007. DRCN 2007. 6th International Workshop on*. [S.l.: s.n.], 2007. p. 1–6.

[Pichitlamken et al. 2003]PICHITLAMKEN, J. et al. Modelling and simulation of a telephone call center. In: *Simulation Conference, 2003. Proceedings of the 2003 Winter*. [S.l.: s.n.], 2003. v. 2, p. 1805–1812.

[Puliafito, Riccobene e Scarpa 1996]PULIAFITO, A.; RICCOBENE, S.; SCARPA, M. Evaluation of performability parameters in client-server environments. *The Computer Journal*, v. 39, n. 8, p. 647–662, 1996. Disponível em: <http://comjnl.oxfordjournals.org/content/39/8/647.abstract>.

[Ramchandani 1974]RAMCHANDANI, C. *ANALYSIS OF ASYNCHRONOUS CONCURRENT SYSTEMS BY TIMED PETRI NETS*. Cambridge, MA, USA, 1974.

[S. e Obaidat 2010]S., M.; OBAIDAT, N. A. B. *Fundamentals of Performance Evaluation of Computer and Telecomunication Systems*. New Jersey: John Wiley & Sons, 2010.

[Sahner R. e Puliafito 1996]SAHNER R., T. K.; PULIAFITO, A. *Performance and Reliability Analysis of Computer Systems: An Example-based Approach Using the SHARPE Software Package*. United States of America: Kluwer Academic Publishers, 1996.

[Sallhammar 2007]SALLHAMMAR, K. *Stochastic Models for Combined Security and Dependability Evaluation*. Tese (Doutorado) — Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Telematics, Trondheim, Norway, June 2007. Disponível em: <http://urn.ub.uu.se/resolve?urn=urn:nbn:no:ntnu:diva-1927>.

[Santos, Clarke e Nel 2007]SANTOS, M. dos; CLARKE, W.; NEL, A. Enhancing telecommunications business operations and service level agreements by incorporating operational risk management. In: *AFRICON 2007*. [S.l.: s.n.], 2007. p. 1–7.

[Shooman 2002]SHOOMAN, M. *Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design*. New York: John Wiley e Sons, 2002.

[Sifakis 1977]SIFAKIS, J. Use of petri nets for performance evaluation. In: *Proceedings of the Third International Symposium on Measuring, Modelling and Evaluating Computer Systems*. Amsterdam, The Netherlands: North-Holland Publishing Co., 1977. p. 75–93.

[Silva 2010]SILVA, A. B. *Avaliação de desempenho e planejamento de capacidade em call centers de serviços de emergência*. Dissertação (Mestrado) — Centro de Informática, Universidade Federal de Pernambuco, Recife, 2010.

[Silva et al. 2010]SILVA, B. et al. Astro: A tool for dependability evaluation of data center infrastructures. In: *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*. [S.l.: s.n.], 2010. p. 783 –790. ISSN 1062-922X.

[Sousa 2009]SOUSA, E. T. G. *Avaliação do Impacto de uma Política de Manutenção na Performabilidade de Sistemas de Transferencia Eletronica de Fundos*. Dissertação (Mestrado) — Master's thesis, Centro de Informática, Universidade Federal de Pernambuco, Recife, 2009.

[Telefonica 2011]Telefonica. *Descritivo de serviço colocation telefônica*. 2011. Acesso em: 10 dez. 2011. Disponível em: <http://www.telefonica.net.br/empresas/docs/Contrato_CIS_Housing.pdf>.

[Trivedi et al. 1994]TRIVEDI, K. et al. Techniques and tools for reliability and performance evaluation: Problems and perspectives. In: HARING, G.; KOTSIS, G. (Ed.). *Computer Performance Evaluation Modelling Techniques and Tools*. [S.l.]: Springer Berlin / Heidelberg, 1994, (Lecture Notes in Computer Science, v. 794). p. 1–24. ISBN 978-3-540-58021-8.

[Trivedi et al. 2009]TRIVEDI, K. et al. Dependability and security models. In: *Design of Reliable Communication Networks, 2009. DRCN 2009. 7th International Workshop on*. [S.l.: s.n.], 2009. p. 11–20.

[Watson J.R. e Desrochers 1991]WATSON J.R., I.; DESROCHERS, A. Applying generalized stochastic petri nets to manufacturing systems containing nonexponential transition functions. *Systems, Man and Cybernetics, IEEE Transactions on*, v. 21, n. 5, p. 1008 –1017, sep/oct 1991. ISSN 0018-9472.

[Xie Yuan Shun Dai 2004]XIE YUAN SHUN DAI, K.-L. P. M. *Computing System Reliability: Models and Analysis*. Review published in ijpe. [S.l.]: Plenum Publishers, 2004. 308 p. ISSN 030648496X.

[Zimmermann e Knoke 2007]ZIMMERMANN, A.; KNOKE, M. Timenet 4.0: A software tool for the performability evaluation with stochastic and colored petri nets. 2007.

Este volume foi tipografado em LaTeX na classe UFPEThesis (www.cin.ufpe.br/~paguso/ufpethesis).

APÊNDICE A

FERRAMENTAS PARA AVALIAÇÃO DE PERFORMABILIDADE

Este capítulo apresenta as principais características e funcionalidades das ferramentas para avaliação da dependabilidade e desempenho denominadas ASTRO e TimeNet, respectivamente, que permitem o suporte à análise e/ou simulação através de modelos RBD e/ou SPN.

Ao realizar uma avaliação sobre performabilidade é necessário realizar a criação de modelos de performabilidade. Uma vez que o modelo é especificado, os projetistas necessitam verificar vários *trade-offs* e selecionar uma solução viável considerando a utilização de ferramentas apropriadas que auxiliem na avaliação das métricas desejadas, tais como desempenho e disponibilidade. As ferramentas são importantes neste contexto para automatizar várias atividades de projeto e obter resultados de forma mais rápida e eficaz.

A.1 FERRAMENTA ASTRO

A ferramenta ASTRO, desenvolvida sobre o núcleo da ferramenta Mercury, foi projetada para fornecer suporte à avaliação de dependabilidade e sustentabilidade através da utilização de modelos RBD, SPN e ambientes de alto nível para avaliação de infraestruturas *data center* [Silva et al. 2010]. Apesar de ser sido construída originalmente para sistemas de *data center*, é possível utilizá-la para avaliar *call centers*, pois os componentes presentes em ambas estruturas são semelhantes.

O ASTRO foi desenvolvido utilizando a linguagem Java, sendo assim, esta ferramenta é multi-plataforma, ou seja, pode ser executada em qualquer sistema operacional que possua uma Máquina Virtual Java instalada como o Linux, Windows, Mac ou Solaris.

De acordo com [Silva et al. 2010], as principais funcionalidades do ASTRO são:

- Avaliação da infraestrutura de fornecimento energético de sistemas *data center*: permite avaliar o sistema em termos de dependabilidade e sustentabilidade através de modelos de alto nível que representam a estrutura do sistema de potência do *data center*.
- Avaliação da infraestrutura de resfriamento de sistemas *data center*: permite a avaliação de métricas de sustentabilidade e dependabilidade através de um ambiente semelhante ao sistema de potência.
- Simulação estacionária e transiente de modelos SPN: permite obter métricas de dependabilidade através de simulação estacionária ou transiente, utilizando Redes de Petri Estocásticas (SPN).
- Modelagem e avaliação de modelos RBD: permite a representação e avaliação de RBDs. Além disso, o ambiente também permite funcionalidades como: experimentação de cenários, geração de funções lógicas e estruturais, avaliação do impacto dos componentes e avaliação de RBD por limites.

A.1.1 Conversão de modelos para SPN e RBD

Segundo [Silva et al. 2010], especificamente no procedimento de conversão para modelos RBD, cada componente gera um novo bloco, contendo os valores de tempo médio de falha e reparo do componente. Estes blocos são organizados em estruturas em série, paralelo ou estrutura correspondente, representando o modo operacional definido pelo projetista (veja Figura A.1), possibilitando a avaliação de forma analítica do modelo.

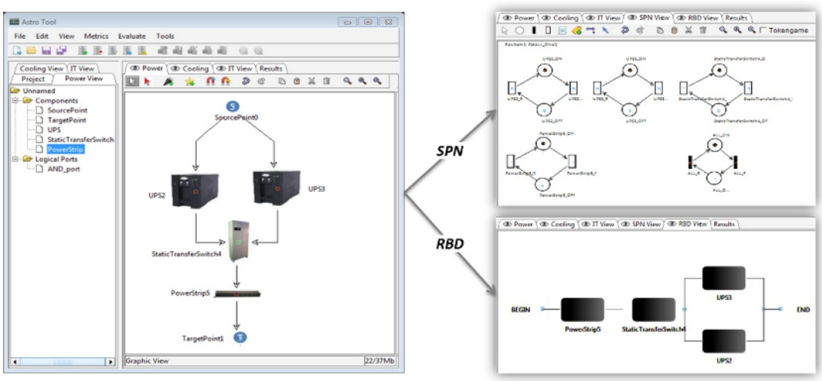


Figura A.1 Conversão do modelo de alto nível para RBD e SPN [Silva et al. 2010].

No processo de conversão para modelos SPN, cada componente disposto na visão de alto nível passa a representar um componente simples em SPN. Com todos os componentes criados, um novo componente simples, chamado componente agregador é adicionado, contendo, na sua transições de falha, uma "expressão de guarda", que representa o modo operacional.

A.1.2 Editor e Avaliador RBD

O Editor e avaliador RBD permite a modelagem e análise dos atributos de dependabilidade por meio de diagramas de blocos para a confiabilidade. Adicionalmente, o ambiente RBD permite o cálculo da importância dos componentes de um determinado sistema, geração de funções estruturais e lógicas, bem como avaliação dos limites de confiabilidade do sistema [Figueiredo et al. 2011].

Segundo [Silva et al. 2010], nos modelos RBD as métricas são avaliadas por meio de equações de forma fechadas (*closed-form*), o que permite a obtenção dos resultados de forma mais rápida, comparando-se com simulação em SPN. Os atributos disponibilidade instantânea e estacionária, confiabilidade, tempo médio para falha, tempo médio para reparo e *downtime* podem ser obtidos utilizando modelos RBD na ferramenta.

A ferramenta ASTRO permite a representação deste formalismo, considerando composições série e paralelo, *k-out-of-n (K de N)*, *Bridge* e composições destas, mas também modelos simples compostos por um único componente. Na visão de RBD, o modelo contém um nó de entrada (indicando o início) e um nó de saída (indicando o fim) [Silva et al. 2010].

O ambiente RBD da ferramenta ASTRO possui como funcionalidades a avaliação de métricas de dependabilidade, a experimentação de cenários, a avaliação de importância dos componentes, a avaliação dos atributos de dependabilidade por aproximação, as reduções do modelo e a geração de funções lógica e estrutural. Todas estas funcionalidades foram de grande importância para o estudo e análise do sistema de *call center* de emergência desta Dissertação.

Segundo [Silva et al. 2010], os modelos RBD e SPN podem ser gerados automaticamente para as estruturas modeladas no editor de alto nível. Especificamente no procedimento de conversão para modelos RBD, cada componente gera um novo bloco, contendo os valores de tempo médio de falha e tempo médio de reparo (do componente). Estes blocos são organizados em estruturas série, paralelo ou estrutura correspondente, representando o modo operacional definido pelo usuário, possibilitando a avaliação de forma analítica do modelo. Maiores detalhes sobre o ASTRO podem ser encontrados em [Silva et al. 2010] e [Figueiredo et al. 2011].

Segundo [Silva et al. 2010], a avaliação de desempenho no editor SPN é realizada através de técnicas de simulação, que é o processo através do qual um modelo é avaliado numericamente. A simulação pode ser transiente ou estacionária sendo que ambas as formas utilizam as mesmas funções básicas, no entanto métricas dependentes do tempo são obtidas através de simulações transientes, enquanto que as métricas de estado estacionário são resultado de simulações estacionárias.

A simulação transiente analisa o comportamento de um determinado modelo a partir do instante inicial até um determinado instante de tempo, ou seja, considera o período transiente do sistema. Este tipo de simulação pode ser utilizado para responder a perguntas do tipo: qual é a probabilidade de que após uma semana de funcionamento, o sistema ainda esteja operacional? As medidas de desempenho são computadas para o ponto final no tempo [Silva et al. 2010].

Por outro lado, a simulação estacionária avalia o desempenho do sistema após os efeitos transitórios iniciais passarem, ou seja, em um modo de funcionamento equilibrado. A simulação do estado estacionário pode ser utilizada para responder a perguntas típicas como: qual será a largura de banda máxima de um canal de comunicação? Qual a probabilidade do sistema estar funcionando em um tempo qualquer? São questões como essas que a simulação estacionária está disposta a resolver [Silva et al. 2010].

O ASTRO não realiza análise estacionária e transiente para modelos baseados em estados, portanto, foi adotada nessa Dissertação uma ferramenta que contempla essas análises em modelos SPN que é a ferramenta TimeNet.

A.2 FERRAMENTA TIMENET

O TimeNET (*timed net evaluation tool*) é o resultado da junção de um conjunto de ferramentas, com a finalidade de oferecer o suporte para a criação, edição, simulação e análise de Redes de Petri Estocásticas. A ferramenta é uma evolução do DSPNexpress, cuja influência maior foi da ferramenta GreatSPN.

A ferramenta TimeNET, é um kit de de pacotes gráficas e interativas para a modelagem com Redes de Petri Estocásticas (SPN). A versão TimeNET 4.0 (versão estável disponível desde 2007), totalmente reescrita em JAVA, inclui uma interface gráfica do usuário e fornece suporte ao sistema operacional Microsoft Windows. Esta ferramenta suporta uma nova classe de Redes de Petri Coloridas Estocásticas (SCPNs). A simulação discreta de eventos padrão foi implementada para a avaliação de desempenho dos modelos SCPN.

SCPNs permitem distribuições arbitrárias de atrasos de disparo, incluindo atrasos zero, complexos tipos de *token*, guardas globais, guardas de tempo e marcação prioridades de transição dependentes. Classes de Redes de Petri são definidas por um esquema extensível XML em TimeNET que afeta o comportamento da interface gráfica do usuário. Um modelo é um documento XML bem formado que é validado automaticamente.

A ferramenta permite a criação de transições com tempos de disparo determinísticos, distribuídos exponencialmente e também com uma classe especial de distribuição, a distribuição expolinomial. Esta distribuição, em particular, contém diversas distribuições conhecidas (e.g. determinística, uniforme, triangular, exponencial, etc.) [Zimmermann e Knoke 2007]. O TimeNET suporta as modelagem das seguintes redes:

- Redes de Petri Coloridas Hierárquicas (*Hierarchical Coloured Petri Nets* - HCPN): esta classe de Redes de Petri pode conter *tokens* sem tipo, que não armazenam valores, bem como *tokens* coloridos, que possuem tipos e, portanto, são capazes de armazenar valores.

- Redes de Petri Estocásticas Fluidas (*Fluid Stochastic Petri Nets* - FSPN): este formalismo é uma extensão da classe de Redes de Petri GSPN, no qual não só se conhece lugares que contêm determinado número de *tokens*, como também permite que um ou mais lugares tenha uma quantidade contínua de fluido, que é representado por um número real não negativo, em vez de apenas um número discreto de *tokens*. Este fluido pode ser transferido através de arcos fluidos, em um fluxo contínuo, tanto quanto a transição em questão permitir. Também é permitido que uma determinada quantidade de fluido seja depositada ou removida de uma vez. São bastante úteis na avaliação de sistemas que envolvem componentes como tempo, líquidos, temperaturas, etc.

- Redes de Petri Determinísticas e Estocásticas Extendidas (*Extended Deterministic and*

Stochastic Petri Nets - eDSPN): esta classe de Redes de Petri consiste em uma extensão à classe GSPN. Nela, são permitidas transições com distribuições de disparo imediatas (zero), exponencialmente distribuídas, determinísticas.

As classes de redes expostas acima são compostas basicamente por elementos comuns às Redes de Petri Estocásticas, com algumas especializações. Em geral, uma Redes de Petri Estocásticas no TimeNET é composta por Lugares, Transições, Arcos e Inibidores e outros parâmetros. Este trabalho apresenta um enfoque maior na classe de redes eDSPN no TimeNET.