MARCELO CARNEIRO DO AMARAL

REASON - AVALIAÇÃO DE CONFIABILIDADE E DISPONIBILIDADE EM REDES DE COMPUTADORES SUSTENTÁVEIS.

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Engenharia Elétrica.

São Paulo
2014

MARCELO CARNEIRO DO AMARAL

REASON - AVALIAÇÃO DE CONFIABILIDADE E DISPONIBILIDADE EM REDES DE COMPUTADORES SUSTENTÁVEIS.

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Engenharia Elétrica.

Área de Concentração:

Sistemas Digitais

Orientador:

Prof. Dr. Tereza Cristina Melo de Brito Carvalho

São Paulo
2014

Aos meus pais, mestres, irmã e namorada.

AGRADECIMENTOS

A minha orientadora Prof. Dr. Tereza Cristina Melo de Brito Carvalho, sou grato pela orientação, pela disposição e atenção no acompanhamento das diversas etapas desta pesquisa. Além disso, pelas observações perspicazes que contribuíram e vêm contribuindo para meu amadurecimento e crescimento pessoal.

Aos professores do Departamento de Engenharia de Computação e Sistemas Digitais que contribuíram em meu aprofundamento no tema abordado.

Aos demais colegas de pós-graduação e trabalho de pesquisa, pelas críticas e sugestões.

A minha namorada pelo apoio e compreensão durante o desenvolvimento deste trabalho e a minha família em geral, responsável pela essência de minha formação.

A Fundação para o Desenvolvimento Tecnológico da Engenharia - FDTE, juntamente com a Ericsson Research, pelo apoio financeiro.

Ao Laboratório de Sustentabilidade LASSU/PCS/POLI/USP, pelo compartilhamento de um ambiente propício à pesquisa acadêmica e da estrutura física necessária para a execução do trabalho.

A Ciência enseja a mais ousada metafísica de nosso tempo. É um constructo puramente humano, alimentada pela fé de que se sonharmos, buscarmos a descoberta, explicarmos e sonharmos novamente, desbravando repetidamente territórios desconhecidos, o mundo se tornará algo mais compreensível e nós capturaremos a verdadeira estranheza do Universo. E toda essa estranheza se mostrará interconectada e fará sentido.

(Edward O. Wilson)

A meta do homem é viver cada momento com liberdade, sinceridade e responsabilidade. Desta maneira estará realizando, nas possibilidades de sua natureza, sua tarefa evolucionária.

(Rollo May)

RESUMO

Redes de computadores orientadas à sustentabilidade ou eficientes energeticamente têm a capacidade de adaptar dinamicamente os modos de consumo de energia dos seus dispositivos de acordo com a demanda do tráfego da rede. Por exemplo, colocar no estado "dormente" os dispositivos que estão abaixo de um nível de utilização predeterminado, considerado de baixa carga. Neste cenário, existem novos desafios no que diz respeito ao modo como confiabilidade e disponibilidade da rede são avaliadas. O cálculo de confiabilidade e disponibilidade é comumente realizado através das técnicas de cadeia de Markov, ou Conjuntos-Conexos e Conjuntos-Desconexos. Porém, tradicionalmente, estas técnicas são baseadas em valores estáticos e não levam em consideração as mudanças dinâmicas que são inseridas no contexto de redes sustentáveis. Desta forma, este trabalho tem como principais objetivos prover um método capaz de avaliar o impacto na confiabilidade ou disponibilidade da rede, quando alguns dispositivos são colocados e tirados de modos de economia de energia, e apresentar a relação de compromisso entre economia de energia, confiabilidade e disponibilidade da rede. O método proposto, chamado REASoN, é uma composição dos dois métodos supracitados, que foram estendidos de forma a considerar no cálculo a dinamicidade dos ajustes dos níveis de energia. Para fins de avaliação, o trabalho realiza um cálculo numérico empregando o método REASoN, em que foi avaliada a confiabilidade dos dispositivos quando colocados no estado "dormente". Os impactos de operações de eficiência energética nas métricas de confiabilidade são expressos como mudanças na quinta casa decimal da confiabilidade da rede como 52 minutos de inatividade de componentes da rede e na quarta casa decimal da disponibilidade com 8h de inatividade. Para uma empresa de transações bancárias, 8h de inatividade pode significar R$ 1 bilhão de perda. O trabalho analisa, também, a implementação do REASoN dentro do contexto de um sistema de gerenciamento de rede orientado a sustentabilidade. Os resultados mostram que, quando o sistema não prioriza disponibilidade e confiabilidade, a economia de energia é de 43%. Já quando a disponibilidade e confiabilidade são priorizadas, a economia é de 27%, um valor representativo. Concluímos que o REASoN é uma ferramenta de grande utilidade para a tomada de decisão em redes de computadores sustentáveis, servindo, como base, para uma análise mais acurada sobre o impacto de mecanismos de economia de energia.

Palavras-chave: Disponibilidade; Confiabilidade; Redes de Computadores; Sustentabilidade; Eficiência energética; e Gerenciamento de rede baseado em políticas.

ABSTRACT

Sustainability-oriented computer networks or energy efficient networks have the ability to dynamically adapt the network device power modes according to the demand of network traffic. For example, putting to sleep devices that handle traffic below a pre-defined threshold. In this scenario, new challenges related to the reliability and availability evaluation of the network arise. The calculation of reliability and availability is commonly accomplished through techniques such as Markov chain, or Cut-Set and Tie-Set. However, traditionally these techniques are based on static values and do not take into account the dynamic changes that happen in the context of sustainable networks. Thus, the main objective of this thesis are to provide a method for assessing the impact on reliability or availability of the network when some devices are put to sleep, and to second present the trade-off between saving energy and changing the reliability and availability of the network. The proposed method, called REASoN, is a composition of two known methods: Markov chain and Cut-Set and Tie-Set, which were extended in order to consider in the reliability and availability calculation the dynamic adjustments of the energy levels. To evaluate the proposed method, the work performs a numerical evaluation deploying REASoN. In addition, we evaluated the reliability of the network. Results shows that the impacts of energy efficiency are expressed as changes in the fifth decimal place of network reliability as 52 minutes downtime of network components and in the fourth decimal place of availability with 8h of inactivity. For trading operations, 8h of downtime can mean R$ 1 billion loss. The work also analyzes REASoN implemented in a network management system oriented to sustainability, called SustNMS. The results show that when the system prioritizes energy efficiency and accepts reduction of availability and reliability, the energy savings reach 43%. However, if the system prioritizes availability and reliability the energy savings reach 27%. Hence, REASoN is a powerful tool that can be used for decision making in sustainability-oriented computer networks, achieving a more accurate analysis of the impacts of saving energy.

Keywords: Availability; Reliability; Computer networks; Sustainability; Energy efficiency; and Policy oriented network management.

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

A(t)	*Availability at instant of time t*
BGP	*Border Gateway Protocol*
CCR	*Consumer Consumption Rating*
CE	*Control Engine*
CLI	*Command Line Interface*
CO_2	*Carbon Dioxide*
DCiE	*Data Center infrastructure Efficiency*
DCP	*Data Center Productivity*
DiffServ	*Differentiated Services*
DU	*Device Updater*
ECR	*Energy Consumption Rating*
FE	*Forwarding Engines*
GHG	*Green House Gas*
GNT	*Green Network Technologies*
GeSI	*Global e-Sustainability Initiative*
HD	*High Definition*
ICMP	*Internet Control Message Protocol*
ICT	*Information and Communication Technology*
IETF	*Internet Engineering Task Force*

IO *Input and Output*

ITU *International Telecommunication Union*

ITU-T *ITU Telecommunication Standardization Sector*

ISP *Internet Service Provider*

LSP *Label Switched Paths*

MPLS *Multi Protocol Label Switching*

MR *Model Repository*

MTBF *Mean Time Between Failures*

MTBR *Mean Time Between Repairs*

MTTF *Mean Time to Failure*

MTTR *Mean Time to Repair*

Netconf *Network Configuration Protocol*

NMS *Network Management System*

NGN *Next Generation Network*

NIC *Network Interface Card*

NPC *Normalized Power Consumption*

NSP *Network Service Providers*

OSPF *Open Shortest Path First*

OS *Operating System*

$P_B Bline$ *Power Consumption per Line of Broadband*

PBNMS *Policy-Based Network Management System*

PEP *Policy Enforcement Point*

PDL *Policy Definition Language*

PDP *Policy Decision Point*

PIB Produto Interno Bruto

PMF *Policy Management Framework*

PUE *Power Usage Effectiveness*

PR *Policy Repository*

QoE *Quality of Experience*

QoS *Quality of Service*

QoSM *Quality of Service Monitor*

R(t) *Reliability at a time interval [t_0 to t]*

REASoN *Reliability and Availability Evaluation of Sustainable Network*

RIP *Routing Information Protocol*

RPC *Remote Procedure Call*

SLA *Service Level Agreement*

SO Sistema Operacional

SNMP *Simple Network Management Protocol*

SM *Sustainability Monitor*

SSH *Secure Shell*

SustNMS *Sustainability Oriented Network Management System*

TEEER *Telecommunications Equipment Energy Efficiency Ratio*

Telcos *Telecom Operators*

TIC *Information Technology and Communication*

UAS *Unavailable Seconds*

XML *Extensible Markup Language*

LISTA DE SÍMBOLOS

c	*Fator de cobertura, a probabilidade de uma falha ser detectada e o componente em redundância ser completamente ativado*
C_m	*m-ésimo elemento contido no Conjunto-Desconexo*
d	*Destino*
$\Delta(t)$	*Tempo médio*
λ	*Taxa de falha de um dispositivo*
μ_1	*Taxa de manutenção preventiva*
μ_2	*Taxa de reparo*
o	*Origem*
P(t)	*Probabilidade no instante de tempo t*
R(t)	*Reliability*
r	*Roteador*
t	*Instante de tempo*
T_n	*n-ésimo elemento contido no Conjunto-Conexo*
w	*Quantidade total de estados no modelo de Markov*
X	*Matriz de transição*
α	*1/(tempo médio para acordar o dispositivo)*

1 INTRODUÇÃO

Nos últimos anos, as tentativas de melhorar a eficiência energética das redes de computadores, ou seja, achar o ponto de equilíbrio entre diminuir o consumo de energia e manter os níveis de qualidade de serviço (QoS - *Quality of Service*), e de melhorar a infraestrutura de serviços de telecomunicações têm se tornado um objetivo de alta prioridade no contexto das operadoras de telecomunicações (Telcos - *Telecom Operators*) e Provedores de Serviços de Internet (ISPs - *Internet Service Provider*) (BOLLA et al., 2011). Esse interesse é motivado pelo aumento dos preços da energia elétrica, o crescimento contínuo da população de usuários, a difusão de acesso à banda larga e a oferta crescente de serviços. É esperado que até 2015 o tráfego global da rede escalará exponencialmente, devido ao grande aumento do número de usuários e da largura de banda (LANGE et al., 2011). Logo, para suprir tal demanda, os Provedores de Serviço de Rede (NSPs -*Network Service Providers*) precisam ampliar sua infraestrutura de rede cada vez mais, tanto em tamanho quanto em complexidade (BOLLA; DAVOLI; CUCCHIETTI, 2011), construindo as Redes da Próxima Geração (NGN - *Next-Generation Networks*).

Tudo isso tem levado ao rápido crescimento do consumo de energia, causando problemas ambientais, como a excessiva emissão de dióxido de carbono (CO_2) (FETTWEIS; ZIMMERMANN, 2008). Segundo Wang et. al., as atividades industriais emitem mais que o dobro de CO_2 do que os processos naturais conseguem absorver (WANG et al., 2012). Estima-se que a TIC (Tecnologia da Informação e Comunicação) é responsável por cerca de 2 a 3% da emissão global total de CO_2, com a previsão

de que em 2013 seu volume seja acima de 1.2 bilhões de toneladas de CO_2 (KRAJEWSKI; JUNG, 2013) e (DESPINS et al., 2011). Segundo a Iniciativa Global de Sustentabilidade (GeSI), as indústrias compostas por TIC têm o potencial de redução de 23 a 30% das emissões globais do Gases de Efeito Estufa (GHG - *Green House Gases*). Isto se deve à existência da TIC em várias indústrias e setores da sociedade e à possibilidade de aplicar técnicas de redução do consumo de energia (KRAJEWSKI; JUNG, 2013), (DESPINS et al., 2011).

1.1 Motivação

Atualmente, as redes são projetadas e operadas de maneira mais confiável e disponível o possível de forma a garantir os Níveis de Serviços Acordados (SLA - *Service Level Agreement*), e não são considerados quesitos de eficiência energética (BOLLA et al., 2011). Desta forma, as redes são usualmente construídas com enlaces redundantes, sendo esses enlaces superdimensionados, a fim de suportar falhas e sobrecarga na rede (WANG et al., 2012). Tipicamente, os dispositivos da rede excedem mais que a metade da necessidade da rede (BOLLA et al., 2011). Sendo que em geral, as redes metropolitanas possuem 13% de redundância dos dispositivos e 100% de redundância dos enlaces e as redes centrais possuem 100% de redundância dos dispositivos e 50% de redundância dos enlaces (BOLLA et al., 2011c).

Os dispositivos redundantes são tipicamente mantidos em configuração *hot standby*, em que ficam em modo totalmente operacional, servindo de *backup* caso ocorra uma falha ou sobrecarga na rede. Por outro lado, nos períodos de baixa utilização da rede, esses equipamentos são subutilizados e acabam consumindo energia desnecessariamente. Nesse contexto, os dispositivos, na ausência de tráfego, podem ser desativados. Porém, manter esses dispositivos totalmente desativados não é viável, sendo necessário que o dispositivo possa ser reativado quando necessário. Logo, o dispositivo pode ser colocado em um estado intermediário, entre ligado e desligado,

que é chamado de “dormente” ou *cold standby*.

Existe um grande potencial para se economizar energia, por exemplo:

1 **Melhorias no *hardware* dos dispositivos**, utilizando arquiteturas e componentes com menor consumo de energia, ou mesmo componentes autônomos que auto desligam-se quando ociosos, como em (BALDI; OFEK, 2009), (BO et al., 2011), e (NISHIMURA et al., 2011).

2 **Virtualização da rede**, em que é possível a coexistência de várias redes virtuais na mesma rede física, possibilitando um melhor aproveitamento dos recursos da rede, uma vez que os recursos serão compartilhados e ficarão ociosos por menor tempo. Além disso, concentrar dispositivos virtuais em apenas um dispositivo físico diminui o consumo de energia, uma vez que o dispositivo físico é quem consome energia, como apresentado em (CHOWDHURY; BOUTABA, 2010), (NEJABATI et al., 2011), (ANHALT; DIVAKARAN; PRIMET, 2010), e (SHIMONISHI; ISHII, 2010).

3 **Gerenciamento de rede** orientado a eficiência energética, de forma a colocar em modo “dormente” os dispositivos ociosos, ou diminuir a taxa de transferência em períodos de baixa demanda, diminuindo, assim, o consumo de energia como em (DEMESTICHAS et al., 2011), (CHAUDHARI et al., 2012), (COMANICIU; MANDAYAM; POOR, 2009), (BOLLA; BRUSCHI; CARREGA, 2010), e (XIA et al., 2010).

Assim, pode-se definir que as redes sustentáveis são constituídas por mecanismos que economizam energia. Tal economia de energia tem implicações diretas na redução da emissão de CO_2, uma vez que, na maioria das vezes, a geração de energia emite CO_2. Desta forma, as pesquisas relacionadas à redução do consumo de energia da rede têm se tornado cada vez mais importante, pois, melhorar a eficiência energética da rede auxilia a redução de custos, além de combater ou prevenir os problemas ambientais.

1.2 Descrição do Problema

A aplicação das técnicas de melhorias no *hardware*, virtualização da rede e gerenciamento de rede são tipicamente acompanhadas de degradação da QoS, tais como: queda na disponibilidade, confiabilidade ou desempenho da rede (CUOMO et al., 2012). Entretanto, medir as degradações vinculadas ao exercício de funcionalidades orientadas à sustentabilidade é um desafio, uma vez que começam a ocorrer mudanças frequentes e dinâmicas no estado da rede, quando alternado entre os modos "dormente" e acordado. Logo, é importante avaliar o tempo que leva para acordar os dispositivos, pois a rede poderá estar indisponível nesse período e impactar a confiabilidade, a disponibilidade e/ou o desempenho da mesma.

Infelizmente, as abordagens existentes para calcular confiabilidade e disponibilidade (SHOOMAN, 2001; ALTIPARMAK; DENGIZ; SMITH, 2003; GREEN; HANT; LANZINGER, 2009; HE; QI, 2008; LAM; LI, 1986; LIN et al., 2010), e (YEH et al., 2010) não são totalmente adequadas para o contexto de redes de computadores sustentáveis, pois não consideram o impacto do tempo de acordar os dispositivos, e a dinamicidade decorrente de mudanças na topologia da rede. Isso ocorre pelos seguintes motivos:

I É razoável considerar que comumente, os dispositivos de redundância são mantidos em *hot standby* (totalmente ativados). Isso ocorre a fim de possibilitar a aplicação de técnicas de engenharia de tráfego, como o balanceamento de carga, ou o redirecionamento de fluxos para caminhos mais eficientes, e, assim, suportar períodos de sobrecarga na rede.

II As variações entre os estados *hot standby* e *cold standby* não são consideradas nos métodos clássicos de cálculo de confiabilidade e disponibilidade. As abordagens padrão consideram o estado da rede de forma estática, em que os elementos apenas se tornam indisponíveis em caso de falha, que é essencialmente baseado

na probabilidade de uma falha ocorrer no intervalo de tempo que o dispositivo foi ligado e seu tempo de vida estimado.

III Nas redes "não sustentáveis", um dispositivo em *cold standby* é ativado apenas na ocorrência de falha, o que ocorre em uma frequência muito baixa. Porém, nas redes sustentáveis, a ativação e desativação ocorrerão frequentemente devidas à aleatoriedade da variação da carga da rede. Essas mudanças impactam na QoS da rede, pois, em alguns períodos os recursos serão reduzidos. Desta forma, os impactos do tempo de acordar os dispositivos poderão ser significativos.

Durante o período de pesquisa deste trabalho, não foi encontrado um método que avalie dinamicamente tais efeitos degradadores da QoS, mais especificamente, na confiabilidade e disponibilidade, no contexto das redes sustentáveis. Este trabalho irá explorar a técnica de economia de energia relacionada ao gerenciamento de rede, pois, a vantagem é a tomada de decisão em relação à visão global da rede, permitindo assim alcançar um ponto ótimo máximo, e não basear-se em informações locais.

1.3 Objetivos

Este trabalho tem dois objetivos. **O primeiro objetivo** deste trabalho é prover um método capaz de calcular confiabilidade e disponibilidade considerando a dinamicidade da rede e o tempo entre as transições dos estados energéticos, ou seja, quando alguns dispositivos são colocados/tirados de modos de economia de energia. Desta forma, este trabalho propõe o método chamado de Avaliação de Confiabilidade e Disponibilidade em Redes de Computadores Sustentáveis (REASoN - *Reliability and Availability Evaluation of Sustainable Network*).

O segundo objetivo é apresentar a relação de compromisso entre economizar energia, e a confiabilidade e disponibilidade da rede, e executando-se um ambiente de teste com um sistema de gerenciamento de rede orientado à sustentabilidade

(SustNMS - *Sustainability Oriented Network Management System*) composto pelo método REASoN para calcular a confiabilidade e disponibilidade da rede. Com base nas informações calculadas pelo REASoN, e outros requisitos descritos em políticas, o sistema toma decisão de colocar dispositivos em modo “dormente” ou não.

1.4 Organização do trabalho

O trabalho está organizado da seguinte forma. O Capítulo 2 traz uma visão sobre o estado da arte na área de gerenciamento de rede orientado à sustentabilidade, apresentando a arquitetura padrão definida pelo IETF e as principais métricas relacionadas à eficiência energética. O Capítulo 3 discute a questão de QoS em redes de computadores, apresentando as definições usuais, assim como as técnicas de otimização e os métodos mais populares para realizar o cálculo de confiabilidade e disponibilidade. O Capítulo 4 descreve o método proposto por este trabalho. O Capítulo 5 apresenta um estudo de caso para redes de computadores metropolitanas e uma análise numérica, mostrando comparações entre a confiabilidade calculada pelos métodos padrão e a calculada pelo REASoN. O Capítulo 6 descreve o sistema de gerenciamento de rede orientado a sustentabilidade que implementa o REASoN, o SustNMS, apresentando suas características e arquitetura. O Capítulo 7 descreve experimentos realizados com o SustNMS, mostrando a relação de compromisso entre economizar energia e manter alta disponibilidade. Por fim, o capítulo 8 discute as contribuições deste trabalho e propõe trabalhos futuros relacionados à sustentabilidade e gerenciamento de redes.

2 GERENCIAMENTO DE REDE E A SUSTENTABILIDADE

Devido ao grande aumento do consumo de energia elétrica nas redes de computadores, torna-se importante a criação de estratégias sustentáveis para se economizar energia. Atualmente, existem várias pesquisas relacionadas a tecnologias empregadas na redução de consumo de energia de equipamentos eletroeletrônicos (BOLLA et al., 2011). O gerenciamento de rede pode ser aplicado para coordenar, analisar e aplicar mudanças para garantir QoS (Qualidade de Serviço) e economia de energia.

Para definir estratégias sustentáveis, é importante o entendimento do estado da arte sobre sistema de gerenciamento de rede, como a sustentabilidade pode ser aplicada na rede, as métricas e requisitos relacionados à QoS. Esses conceitos, definições, e descrições serão apresentadas nas sessões a seguir.

2.1 Gerenciamento de rede

Pare entender gerenciamento de rede, é importante saber que atualmente, as funcionalidades que controlam as redes de computadores estão divididas em três níveis principais: (i) o plano de dados que lida com os pacotes de dados; (ii) o plano de controle possui funções de reportar erros, configurar sistemas, alocação de recursos, e implementa algoritmos de roteamento distribuído ao longo dos elementos da rede; e (iii) o plano de gerenciamento que monitora a rede, configura o plano de dados e os

protocolos do plano de controle (GREENBERG et al., 2005).

O plano de dados é a base dos elementos de comutação em nossas redes (SALISBURY, 2012). Ele tem a responsabilidade de analisar cabeçalhos dos pacotes realizando buscas em alta velocidade. Ele gerencia QOS, realiza filtragem, encapsulamentos e trabalha com filas.

O plano de controle é o componente que estabelece como que cada roteador interage com seu vizinho quando seu estado muda (SALISBURY, 2012). O plano de controle é utilizado para 1- descoberta de recursos e inventário, 2- gerenciamento de configuração, inicialização de sistemas e atualizações, 3- controle do inventário, 4- detecção e recuperação de falhas, e 4- monitoramento de desempenho (CISCO, 2003.). Como exemplo de protocolos do plano de controle, têm-se os protocolos de roteamento OSPF (*Open Shortest Path First*), BGP (*Border Gateway Protocol*), e RIP (*Routing Information Protocol*).

O plano de gerenciamento prove a interação humana com as funcionalidades de um dispositivo de rede (OSHANA; KRAELING, 2013). Tipicamente os dispositivos de rede podem ser acessados das seguintes formas: utilizando o protocolo simples de gerenciamento de rede SNMP (*Simple Network Management Protocol*), por interface de comandos CLI (*Command Line Interface*), NETCONF (*Network Configuration Protocol*), e etc. De acordo com Griffin (GRIFFIN, 2009) as funções do plano de gerenciamento incluem tarefas de projeto, planejamento, configuração e gerenciamento da rede e gerenciamento de falhas.

O gerenciamento de rede é possivelmente implantado através de um sistema de gerenciamento de rede baseado em políticas (PBNMS - *Policy-Based Network Management System*). A vantagem desse sistema é justamente a política. Política é um conjunto de regras de descreve o comportamento do sistema e podem ser modificadas durante a execução do sistema, provendo flexibilidade quando necessário alterar o comportamento do sistema. Desta forma não é necessário para todo o sistema e o

reprogramar.

O PBNMS é uma ferramenta inteligente empregada no gerenciamento de rede, capaz de aplicar funcionalidades de QoS, segurança e eficiência energética, através de decisões autonômicas. A arquitetura genérica de um PBNMS inclui quatro módulos principais (WATERS et al., 1999):

1. O Console de Gerenciamento atua como interface para o usuário criar e ajustar políticas. A linguagem utilizada para manipular políticas é chamada Linguagem de Definição de Política (PDL - *Policy Definition Language*).

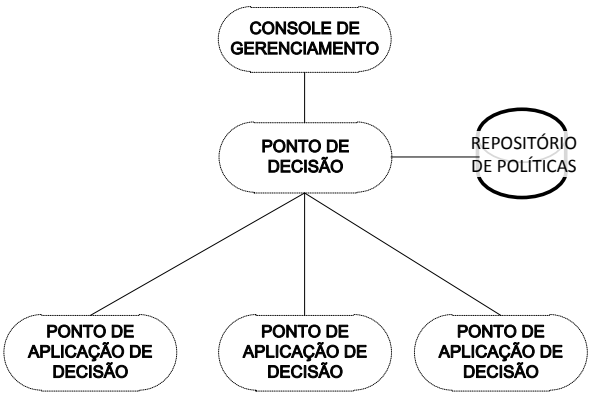
2. O Repositório de Políticas (PR - *Policy Repository*) que armazena as políticas.

3. O Ponto de Decisão de política (PDP - *Policy Decision Point*) controla as condições das políticas e verifica se alguma ação deve ser aplicada.

4. O Ponto de aplicação de decisões (PEP - *Policy Enforcement Point*) é responsável por assegurar que as instruções utilizadas pelo PDP sejam aplicadas.

A figura 1 ilustra a arquitetura de um PBNMS de acordo com as especificações do IETF (*Internet Engineering Task Force*).

Figura 1: Arquitetura genérica de um sistema a de gerenciamento de redes baseado em política - PBNMS.

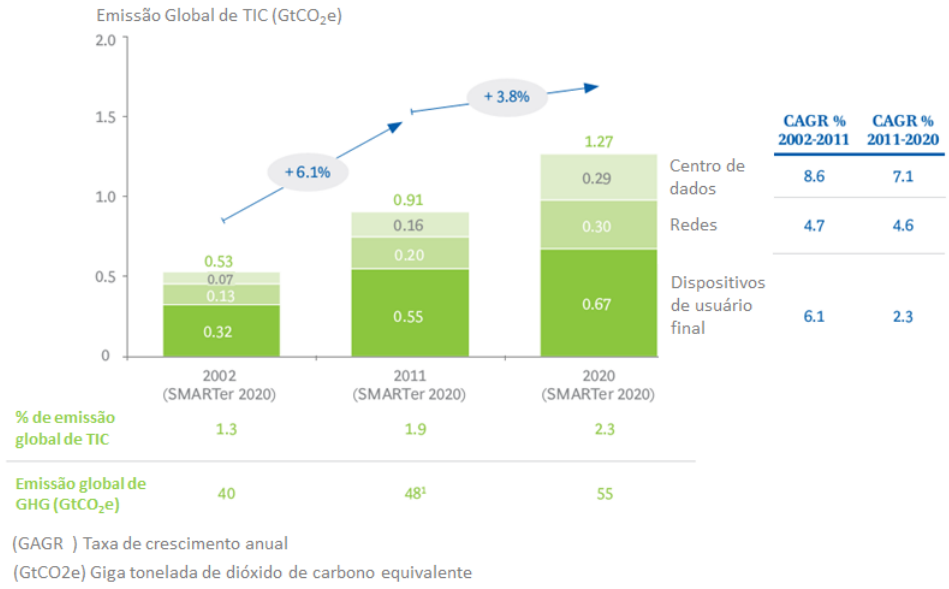


Baseada em (WATERS et al., 1999)

2.2 Redes orientadas à sustentabilidade

Sustentabilidade em redes de computadores é a racionalização do consumo de energia. A motivação de tornar as redes mais sustentáveis (energeticamente eficientes) está também relacionada ao meio ambiente, no qual existe a relação da diminuição de desperdícios, o qual impacta diretamente na emissão de CO_2 (BOLLA; DAVOLI; CUCCHIETTI, 2011). A produção de energia elétrica está atrelada a emissão de CO_2. A Figura 2 apresenta a emissão global de CO_2 relacionada a TIC (Tecnologia de Informação e Comunicação), mostrando a taxa de crescimento desde 2002, onde TIC representava 3,25% da emissão global, até as projeções para 2020, onde TIC representará 4,18% da emissão global. Logo, se pode ver que a missão de CO_2 relacionada à TIC vem se tornando cada vez mais representativa. Além disso, pode-se ver na figura que a emissão relacionada a redes de computadores ocupa uma parcela significativa, tornando importante as pesquisas relacionadas a redução de tal emissão.

Figura 2: Emissão de CO_2 relacionada a TIC.

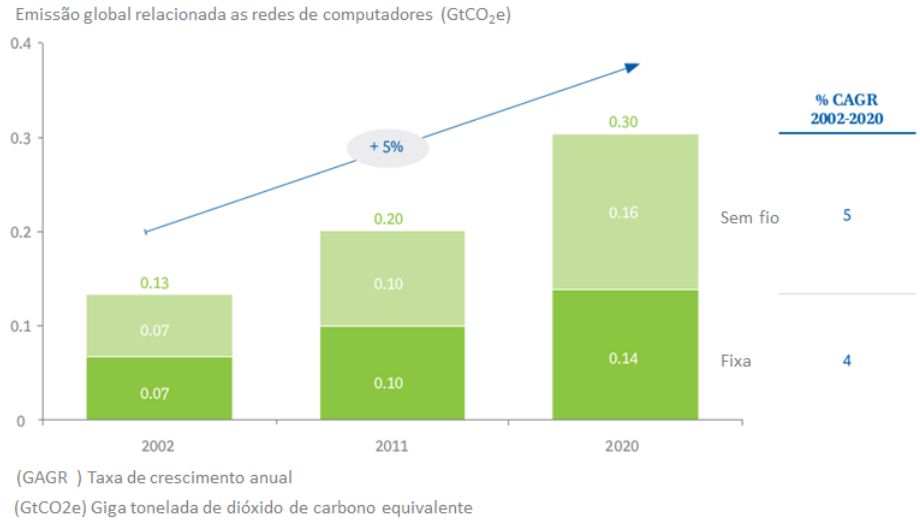


Baseada em (KRAJEWSKI; JUNG, 2013)

A Figura 3 descreve a emissão global de CO_2 relacionada a redes de computadores, apresentando que existe um equilíbrio entre as emissões provenientes das redes sem fio

e das redes fixas. Logo, as pesquisas relacionadas a ambas as áreas possuem mesma importância em relação a diminuir a emissão de CO_2.

Figura 3: Emissão de CO_2 relacionada a redes fixas e sem fio.



Baseada em (KRAJEWSKI; JUNG, 2013)

Existem três tipos de abordagens para economizar energia de acordo com Bolla et. al. (BOLLA et al., 2011):

1. **Reengenharia**: consiste em melhorar e criar novos componentes para obter um *hardware* mais sustentável.
    - Silício/*Hardware* energeticamente eficiente: Novos processadores que consomem menos energia ou demandam menos do sistema de resfriamento.
    - Redução da complexidade: Simplificação do *hardware* para demandar menor consumo de energia e menos utilização de material.
2. **Adaptação dinâmica**: Depende do controle dos recursos do equipamento (enlace, banda, buffer, etc) de acordo com a carga do tráfego da rede ou restrições de desempenho.

- Variação de desempenho: Alguns equipamentos permitem diferentes estados de consumo energético, tais como a alteração da frequência do ciclo do processador e carga máxima de uma porta.
- Lógica ociosa: Desabilitar os recursos que não são necessários durante certo período contribui para a redução do consumo energético.

3. **Dormência de forma inteligente**: consiste em colocar o dispositivo ou parte dele no estado desligado, e faze-lo entrar em um estado de consumo de energia reduzido. Nesse estado, parte das funcionalidades fica indisponível ("congelada"), a única parte que fica energizada é para manter a presença do dispositivo na rede de forma que os protocolos de rede não entendam que o dispositivo falhou.

É de extrema importância que essas funcionalidades de sustentabilidade tenham baixo impacto na QoS. As redes de computadores provêm um grande número de serviços, com diferentes tipos de requisitos de segurança, confiabilidade, disponibilidade, desempenho, entre outros, de acordo com a demanda da aplicação que faz uso desses serviços. Desta forma, a utilização de um PBNMS torna-se imprescindível devido à possibilidade desse sistema analisar e coordenar a rede, priorizando economizar energia em vez de manter alto desempenho, em alguns momentos, ou vice e versa. Além disso, é possível descrever nas políticas do sistema quais são os níveis aceitáveis de degradação da QoS.

Outro ponto importante é a aquisição de informações, dado que para que um PBNMS tome decisões, é necessário que este leia as informações dos dispositivos da rede e tire conclusões. Usualmente, as informações são acessadas por meio de:

- **Protocolo simples de gerenciamento de rede - SNMP**: Esse protocolo acessa informações da base de informações de gerenciamento (MIB - *Management Information Base*) de um dispositivo, que prove consulta de informação e

configurações. A MIB para gerenciamento de energia nos dispositivos de rede é definida pelo IETF (CLAISE; PARELLO, 2013).

- **Interface proprietária dos equipamentos**. Devido à falta de uma MIB padronizada para o gerenciamento, alguns produtores promovem acesso a essas funcionalidades por meio de linha de comandos específicos. Porém, essa não é uma abordagem escalável devido à heterogeneidade da rede.

- **NetConf**: Trata-se de um protocolo baseado em chamadas RPC (*Remote Procedure Call*) utilizando codificação XML (*Extensible Markup Language*), que foi definido pelo IETF. Esse protocolo é amplamente utilizado nos sistemas de gerenciamento de rede atuais.

Existem diversas funcionalidades de sustentabilidade conflitantes entre si e conflitantes com requisitos de QoS. Requisitos de QoS podem se conflitantes por exigiram que a rede fique ligada o tempo todo para garantir disponibilidade e/ou desempenho, ao mesmo tempo que se deseja realizar economia de energia elétrica. Desta forma, pode-se concluir que a utilização de um sistema de gerenciamento de redes baseado em políticas para gerenciar tanto funcionalidades de sustentabilidade, quanto de QoS é uma abordagem viável para tornar as redes de computadores mais sustentáveis, uma vez que os PNMS possuem vários mecanismos para monitoramento e configuração da rede.

2.3 Métricas de sustentabilidade

A identificação e a definição de métricas de sustentabilidade para avaliar o quão sustentável é uma rede de computadores, tem sido o foco de várias pesquisas relacionadas à Tecnologia de Informação e Comunicação (ICT - *Information and Communication Technology*) (CUOMO et al., 2012; AVALLONE; VENTRE, 2012). Essas métricas são, também, chamadas de métricas verdes e podem ser organizadas

segundo critérios. De acordo com Bianzino (BIANZINO; RAJU; ROSSI, 2011) as métricas podem ser classificadas em dois tipos:

1. **Nível de equipamento**: É a proporção da quantidade de energia consumida por um equipamento da rede, pela energia consumida na rede inteira.

2. **Nível das instalações**: É a proporção da quantidade de energia de todos os equipamentos da infraestrutura de TIC (*Information Technology and Communication*), incluindo rede, servidores, impressoras, etc, pela quantidade de energia consumida por toda infraestrutura da empresa.

Devido à imaturidade da área de sustentabilidade, há falta de padronização de métricas de sustentabilidade de equipamentos eletro-eletrônicos e suas instalações, o que dificulta a identificação de informações disponíveis em cada equipamento para a avaliação das métricas. A Tabela 1 apresenta um conjunto de métricas que foram identificadas por Wang (WANG et al., 2012).

A coluna "cálculo" da Tabela 1 apresenta as equações para medir as métricas e a coluna "contexto" representa para qual contexto cada métrica se aplica. As métricas apresentadas na tabela podem ser agrupadas de acordo com a classificação apresentada anteriormente por Bianzino (BIANZINO; RAJU; ROSSI, 2011):

1. **Nível de Equipamento**: Nesse nível, foram especificadas as seguintes métricas: Taxa de Consumo de Energia (ECR - *Energy Consumption Rating*), Taxa de Consumo de Energia por Consumidor (CCR - *Consumer Consumption Rating*), Taxa de Eficiência Energética por Equipamento de Telecomunicações (TEEER - *Telecommunications Equipment Energy Efficiency Ratio*), Consumo de energia normalizado (NPC - *Normalized Power Consumption*), e Consumo de Energia por Linha de Banda Larga (P_{BBline} - *Power Consumption per Line of Broadband*).

Tabela 1: Métricas de sustentabilidade

Métrica	Nome completo	Criador	Contexto	Cálculo	Unidade	Comentário
ECR	*Energy Consumption Rating*	ECRInitiative (ISIA, Juniper)	ISP e ICT	$ECR = \frac{Consumodeenergia}{Capacidadedosistema}$	Watts/Gbps	Energia normalizada pela capacidade
CCR	*Consumer Consumption Rating*	Juniper	Dispositivo de rede do consumidor	$CCR = \frac{E}{\sum(Aj)}$	rad	"E"= consumo de energia do dispositivo. "A"= provisão de energia. "j"= dispositivo.
TEER	*Telecommunications Energy Efficiency*	Verizon NEBS	Switches e roteadores	$TEER = log(\frac{0.35xP_{MAX}+0.40xP_{50}+0.25xP_sleep}{Vazo}$	log(Watt/Gbps)	P_{MAX} e P_{50} é o consumo de energia 100 e 50%, que são multiplicados pela carga.
NPC	*Normalized Power Consumption*	ETSI	Equipamentos de banda larga	$NPC = \frac{100xP_{BBline}}{BitRatexDistncia}$	Watts/(Mbps x km)	Consumo de energia normalizado pela distância da banda larga
P_{BBline}	*Power Consumption per Line of Broadband*	ETSI	Equipamentos de banda larga	$NPC = \frac{P_{BiBeg}}{Numerodesub-linhas}$	Watts/ usuários por linha	Consumo de energia total dos equipamentos. Sub-linhas é numero de usuários suportados.
PUE	*Power Usage Effectiveness*	Green Grid	Centro de Dados	$PUE = \frac{Totaldeenergiadasinstalaes}{Energiadeumequipamento}$	proporção	Variando de 1 até ∞
DCiE	*Data Center infrastructure Efficiency*	Green Grid	Centro de Dados	$DCiE = \frac{1}{PUE} = \frac{Totaldeenergiadasinstalaes}{Energiadeumequipamento} x100\%$	proporção	Variando de 0 até 100%
DCP	*Data Center Productivity*	Green Grid	Centro de Dados	$DCP = \frac{Trabalhotil}{Energiadeumequipamento}$	proporção	Variando de 1 até ∞

Baseada em (WANG et al., 2012)

2. **Nível das instalações**: Nesse nível, foram especificadas as seguintes métricas: Eficiência do Uso de Energia (PUE - *Power Usage Effectiveness*), Eficiência da Infraestrutura do centro de dados (DCiE - *Data Center infrastructure Efficiency*), e Produtividade do Centro de Dados (DCP - *Data Center Productivity*).

As métricas provêm um conjunto de informações úteis para o gerente possa analisar a rede. As métricas de sustentabilidade são importantes para o gerenciamento de redes porque os valores medidos servem para avaliar a situação da rede e ver o quão sustentável ela está, além de servirem como base para prover uma configuração mais eficiente energeticamente.

2.4 Considerações finais do capítulo

Esse capítulo apresentou diversas técnicas para tornar as redes de computadores mais sustentáveis como reengenharia, adaptação dinâmica e dormência de forma inteligente. Além disso, descreveu uma das principais abordagens para se economizar energia na rede, que é através do gerenciamento de rede orientado à sustentabilidade. Esse capítulo, também, apresentou as principais métricas e requisitos de sustentabilidade que auxiliam o gerenciamento de rede.

Os requisitos de uma rede sustentável definem limites e condições para existir redes consideradas sustentáveis. As definições desses limites e condições dependem de um conjunto complexo de elementos que mudam seus status dinamicamente, e isso requerer a análise de cada possível cenário para que se possam identificar os aspectos que devem ser levados em consideração. Por exemplo, é preciso saber o consumo da rede sem que tenha sido aplicada alguma técnica para economizar energia, assim se terá o consumo total. Então, aplicam-se técnicas realizando teste e verificando em quais situações se economizam mais energia. Entretanto, menos consumo de energia significa equipamentos com capacidade limitada ou inoperante, o que pode implicar

em menor confiabilidade ou desempenho.

Muitos trabalhos têm apresentado pesquisas sobre a relação de compromisso entre economia de energia e seu impacto no desempenho da rede. No entanto, também é necessário analisar o impacto de economizar energia na disponibilidade e confiabilidade na rede, o que implica em determinar o nível de vulnerabilidade da rede a falhas. Desta forma, os capítulos a seguir irão introduzir conceitos e apresentar resultados sobre os impactos das tecnologias orientadas à sustentabilidade na confiabilidade e disponibilidade da rede.

3 QUALIDADE DE SERVIÇO: CONFIABILIDADE E DISPONIBILIDADE

Segundo Marsic (MARSIC, 2013), QoS é a habilidade do elemento de rede (por exemplo, uma aplicação, um *host* ou um roteador) prover algum nível de garantia de entrega consistente dos dados na rede. As principais preocupações que as aplicações impõem na rede estão relacionadas ao desempenho, como atraso, perda de pacotes e etc. Porém, existem outros requisitos, como confiabilidade, disponibilidade e segurança que podem ser, também, muito importantes em função do uso da rede (e.g. redes bancárias tem como requisito crítico a segurança; redes médicas além de segurança tem como fato crítico a confiabilidade). Quando um ou mais enlaces ou nós intermediários falham, a rede pode ficar indisponível para prover uma conexão entre a origem e o destino, até que as falhas sejam reparadas. Algumas aplicações demandam alta confiabilidade e tipicamente alta disponibilidade, o que pode ser obtido através da existência de múltiplos caminhos redundantes entre um par de nós.

Alguns trabalhos investigam a relação entre economia de energia e seu impacto no desempenho da rede. Outros trabalhos avaliam a relação de economia de energia e seu impacto na confiabilidade e disponibilidade da rede; o que é o foco principal dessa dissertação. Desta forma, este capítulo apresenta as definições comumente encontradas na literatura sobre QoS, dando especial atenção à confiabilidade e disponibilidade. Esse capítulo também apresenta as técnicas mais comuns para o cálculo de confiabilidade e disponibilidade aplicado no contexto de redes de computadores sustentáveis.

3.1 Desempenho

O operador de uma rede pode garantir limites de desempenho para uma conexão por três formas (MARSIC, 2013):

1. **Priorização de serviços**, onde o tráfego da rede é classificado, implicando que pacotes marcados com baixa prioridade são descartados em caso de sobrecarga e de alta prioridade sempre tentam ser encaminhados;

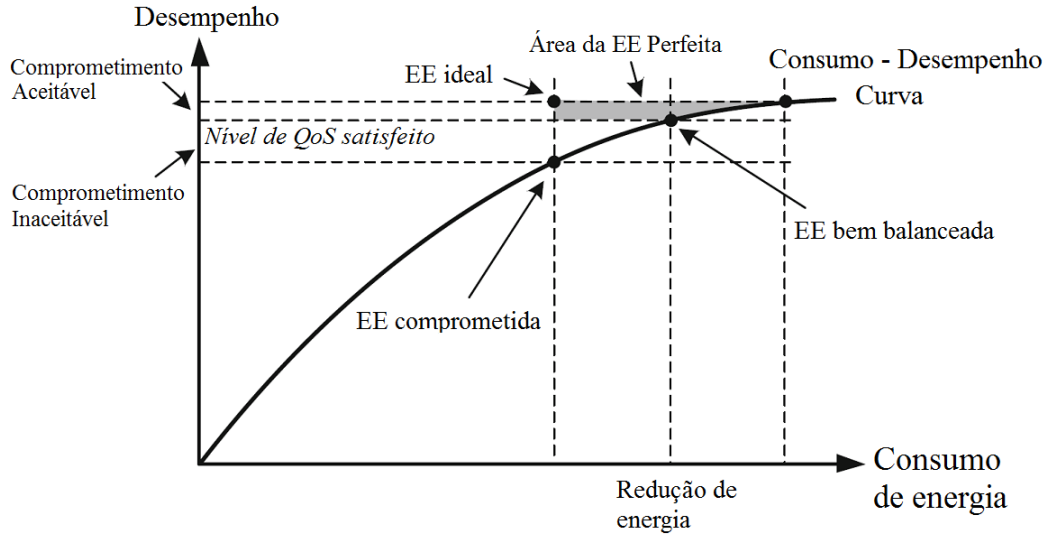
2. **Reserva de recursos dinamicamente**, os recursos da rede são distribuídos de acordo com a requisição de QoS de uma aplicação e o objetivo da política de alocação de banda.

3. **Provisionamento de recursos**, ou aumento da capacidade física da rede, como a adição de novos equipamentos.

Atualmente os itens 1 e 2 têm sido mais comumente empregados, mesmo que o item 3 seja uma solução mais simples. Isto se dá porque a solução 3 envolve elevados custos econômicos. Desta forma, QoS não expande a largura de banda, mas em vez disso, permite a divisão dos recursos da rede de uma maneira não equivalente, favorecendo alguns e penalizando outros em vez de dividir os recursos igualmente entre as aplicações (BARREIROS; LUNDQVIST, 2010). Assim, o início para um projeto de QoS para rede passa por definir critérios de priorização de tráfego, latência, disponibilidade ou banda entre diferentes conexões ou fluxo e como esses critérios irão variar no tempo.

A relação de compromisso entre desempenho e economia de energia apresenta-se como uma equação complexa com muitas variáveis, uma vez que economizar energia basicamente implica em reduzir a capacidade da rede. A decisão de quando diminuir a capacidade da rede e possivelmente penalizar o desempenho não é uma tarefa fácil. A Figura 4 apresenta uma curva que compara o **desempenho e o consumo de energia**.

Figura 4: Curva típica da relação de compromisso entre desempenho e consumo de energia em redes, apresentando a Eficiência Energética (EE).



Baseada em (WANG et al., 2012)

A Figura 4 mostra, na parte sombreada, o ponto mais energeticamente eficiente (EE) no qual níveis aceitáveis de redução de QoS são permitidos enquanto se economiza energia. Essa curva foi levantada para redes sem fio, e serve como base para ilustrar a relação de compromisso entre economizar energia e a perda de desempenho em redes.

O trabalho de Bolla (BOLLA; BRUSCHI; CARREGA, 2010) aplica técnica de economia de energia através de colocar dispositivos em modo “dormente” apenas quando o dispositivos não estiverem sendo utilizados. Esse trabalho analisa o impacto do gerenciamento de energia nas métricas de desempenho da rede. As simulações mostram que é possível economizar energia impactando sem comprometer o desempenho da rede. Os resultados apresentam 40% de economia de energia com 0% de perda de pacotes e aumento, em média, de 0.52 μs no tempo de resposta da rede.

O trabalho de Januário (JANUARIO et al., 2013) apresenta um sistema de gerenciamento de rede orientado à sustentabilidade. Neste trabalho são testadas duas políticas. A primeira política descreve a regra que dispositivos redundantes que estiverem sem carga serão colocados em modo “dormente”. A segunda política

também coloca dispositivos sem carga em modo “dormente”, porém se houver perda de pacotes na rede acima de 0.05%, alguns dispositivos em modo “dormente” serão acordados. Os resultados mostram que a primeira política economiza 40% da energia consumida, e a segunda política economiza 30%, porém mantendo o nível de QoS.

Todos esses trabalhos mostram a existência da relação de compromisso entre desempenho e economia de energia, que deve ser levada em consideração na tomada de decisão de colocar ou tirar um dispositivo em modo “dormente”.

3.2 Conceituação sobre Confiabilidade e Disponibilidade

A confiabilidade e a disponibilidade são métricas relacionadas à probabilidade de falha (*failure*) na rede. Dessa forma, é importante primeiramente entender os conceitos sobre falhas, pois, até que ocorra uma falha, previamente ocorreu um defeito (*defect*), depois um erro (*error*), e por fim uma falha. (STANISAVLJEVIC, 2011) define esses conceitos da seguinte forma:

- Um **defeito** em um sistema eletrônico é a diferença inesperada entre o hardware implementado e seu projeto inicial. Defeitos podem ser relacionados a mau contato, oxidação, ou defeito do material, como quebra ou impurezas. Podem, também, existir defeitos de software causados pela implementação errada de um processo ou método.

- Um **erro** é a produção de um sinal errado. Um erro é um efeito causado por um ou mais defeitos. Erros podem ser classificados como permanente, no qual a causa é irreversível; intermitente, onde ocorre de forma ocasional, não é contínuo ou permanente, porém precede um erro permanente; e transiente, que é um mau funcionamento temporário, causado por condições do ambiente e não é permanente.

- Quando uma **falha** ocorre, implica que um sistema se torna incapaz de prover seus serviços específicos. Um sistema tolerante a falha é capaz de voltar a funcionar corretamente.

Todo sistema está sujeito a falhar, porém uma boa prática é que os erros ocorram de forma prevista e controlada. As duas principais formas de se analisar a probabilidade de ocorrência de falhas é calculando a confiabilidade e a disponibilidade do sistema. Desta forma, a seguir, definem-se confiabilidade e disponibilidade, discute-se em que caso é mais indicado o cálculo de um, ou do outro, e comenta-se sua relação com as redes de computadores.

A **Confiabilidade** (R(t) - *Reliability*) de um sistema é *"uma função do tempo definida como uma probabilidade condicional que um sistema funcione em perfeitas condições em um intervalo de tempo* $[t_0; t]$*, dado que o sistema estava funcionando corretamente no tempo* t_0*"* (JOHNSON, 1989). A confiabilidade é a probabilidade que um sistema opere corretamente em um intervalo de tempo.

Um sistema com alta confiabilidade não é necessariamente tolerante à falha. Isso significa que um sistema ser confiável não implica em ser tolerante a um ou mais erros de software ou hardware. Além disso, um sistema tolerante a falhas não significa que o sistema é altamente confiável. Por exemplo, se a frequência da ocorrência de um problema é bem alta, a confiabilidade desse sistema será baixa, mas se esse sistema for tolerante a falhas, ele irá continuar operando normalmente. Logo, tolerância à falha e confiabilidade não são diretamente relacionados.

A **Disponibilidade** (A(t) - *Availability*) de um sistema é *"uma função do tempo, definida como a probabilidade de um sistema estar operando corretamente e estar disponível para realizar suas funcionalidades no instante de tempo t"* (JOHNSON, 1989).

Um sistema pode ser altamente disponível e mesmo assim apresentar, frequentemente, períodos de inoperáveis, de forma que a duração desses períodos de inacessibilidade seja extremamente pequena. A(t) depende do quão rápido o sistema pode ser reparado, e isso implica que um sistema com alta disponibilidade pode conter imperceptíveis períodos de inacessibilidade (também conhecido como *"down time"* [1]). A disponibilidade por apresentar vários períodos de falha e reparo, pode ser calculada até o estado de convergência (*stead state*), que representa a disponibilidade média de um sistema (SHOOMAN, 2001).

A disponibilidade difere da confiabilidade, sendo que a confiabilidade R (t) depende de um *intervalo* de tempo, e a disponibilidade A(t) é obtida em um instante de tempo específico. Além disso, R(t) é frequentemente utilizada para caracterizar sistemas que não aceitam a ocorrência de falhas, ou então, sistemas que são impossíveis de serem reparados. Por outro lado, A(t) é mais utilizada para projetos que têm o propósito de prover serviços o mais ininterrupto o possível (JOHNSON, 1989).

No caso de redes de computadores, R(t) e A(t) podem ser relacionadas de duas formas (SHOOMAN, 2001):

- Dois-pontos-terminais[2] (*two-terminal*) que representa a confiabilidade e disponibilidade da conexão entre apenas dois nós da rede;
- Todos-pontos-terminais (*all-terminal*) representa a confiabilidade e disponibilidade da conexão entre todos os nós da rede, desta forma todos os nós estão conectados entre si.

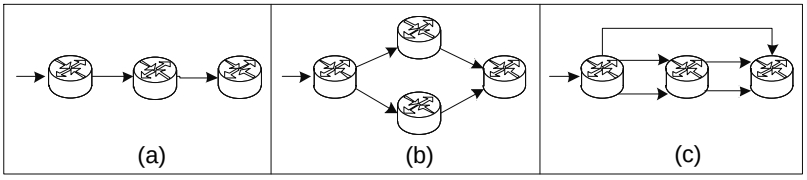
[1] Tempo que o sistema fica indisponível

[2] Os pontos terminais, são os pontos das extremidades da rede, que podem ser estações ou dispositivos da rede.

3.3 Redundância e a relação com Desempenho e Eficiência Energética

A confiabilidade e a disponibilidade de um sistema são tipicamente aprimoradas através da inclusão de redundâncias. Em redes de computadores, o aprimoramento da confiabilidade ou da disponibilidade é alcançado através da inclusão de múltiplos dispositivos ou conexões redundantes como na Figura 5.

Figura 5: Rede (a) sem redundância, (b) com redundância de dispositivo, (c) com redundância dos caminhos



Baseada em (SHOOMAN, 2001)

A Figura 5 (b), através do diagrama de blocos, apresenta múltiplos dispositivos redundantes entre um hospedeiro e um roteador final (origem $< - >$ destino). Já a Figura 5 (c) apresenta múltiplos caminhos redundantes. Desta forma, a falha de caminhos e/ou dispositivos não necessariamente interrompe a conectividade entre dois pontos da rede (ou da rede como um todo), pelo fato que múltiplos caminhos aumentam a chance da rede continuar operando, podendo implicar em uma alta na confiabilidade e/ou na disponibilidade.

A redundância pode ser obtida empregando-se três tipos de configurações: em (i) *hot standby*, em (ii) *cold standby*, e (iii) em *warm standby*. No caso (i), todos os componentes, em redundância, são mantidos ativos e totalmente operacionais. No caso (ii), os componentes em redundância são mantidos inativos até que ocorra uma falha e seja necessário ativá-los. No caso do (iii), os componentes em redundância são mantidos parcialmente inativos, e o tempo para ativar o dispositivo é menor que o tempo gasto no estado de *cold standby*.

Uma premissa básica dos conceitos de confiabilidade e disponibilidade é que

apenas os dispositivos ativos estão sujeitos a falhas e os inativos não. Desta forma, em (ii) modo *cold standby*, os componentes inativos não estão sujeitos a falhas até que sejam ativados (SHOOMAN, 2001). Já o caso do modo *warm standby* por estar parcialmente ativo, comumente é aplicado uma distribuição de falha diferente, que leva em consideração apenas os componentes ativos.

Com o passar do tempo, a chance de falhas de todos os sistemas tende a aumentar devido a sua utilização. Por exemplo, um sistema em *hot standby*, onde todos os componentes estão utilizados e sujeitos a falhas desde iniciou sua operação, possivelmente pode apresentar maior chance de falhar que um sistema em *warm* e *cold standby* . Isto é porque, um sistema em*warm* e *cold standby* apresentam alguns componentes desativados por um tempo, e logo, a confiabilidade e disponibilidade pode ser maior.

Existe também outro fator que influencia se um sistema em *hot*, *warm*, e *cold standby* irá ter confiabilidade ou disponibilidades diferentes. É o fator de cobertura "c", que se relaciona com à probabilidade da falha ser identificada e o componente redundante ser ativado completamente. Por exemplo, se "c" de um componente em *cold standby* é 100%, esse sistema terá a R(t) e A(t) maior que um sistema em *hot standby*, pois, no primeiro, alguns componentes ficam por maior tempo inativo, sofrendo menor chance de falhar. Porém, se "c" for baixo, o sistema em *cold standby* poderá ter R(t) e A(t) menor que um em *hot standby*, pois, desta forma existe a possibilidade de um dispositivo redundante não substituir um dispositivo que falhou.

A decisão de escolher uma das três configurações de *standby* sempre envolve uma análise probabilística, a qual, por sua vez, depende da propriedade dos mecanismos de ativação do *standby*, ou seja, o fator de cobertura "c". A decisão também pode envolver a análise de parâmetros relacionados à QoS, dado que no modo *hot standby* a rede fica em uma configuração com mais recursos disponíveis, o que pode auxiliar no melhor gerenciamento de rede, pois, com todos os dispositivos ligados, o gerente da

rede pode realizar balanceamento de carga ou forçar rotas de acordo com requisitos de QoS

Tipicamente, devido a requisitos de QoS, as redes são mantidas com configuração de *hot standby*. Porém, com o intuito de economizar energia, algumas soluções tem configurado a rede em *cold standby* ou *warm standby*. No *cold* os dispositivos ficam inoperantes, com partes desligadas para economizar energia. Já no *warm* os dispositivos ficam apenas com as partes não utilizadas desligadas, mantendo-se em estado totalmente operacional. Desta forma, podemos ordenar que a configuração que mais se pode economizar energia é o *cold* e a que se mais se gasta é o *hot*. Normalmente os dispositivos em estado inativos, que estão economizando energia, permanecem em estado "dormente". Logo, pode-se relacionar que um dispositivo em *cold standby* está em estado dormente.

A escolha entre utilizar uma das configurações depende do que se quer priorizar na rede. Por exemplo, se a prioridade for economizar energia, o *cold standby* é o modo mais indicado; ou se é mais interessante priorizar o desempenho da rede, então é mais indicado utilizar a configuração *hot standby*. Podem existir também outras situações, onde é melhor priorizar outros parâmetros de sustentabilidade, além de economia de energia, como tempo de vida. Nesse caso, a melhor opção é utilizar a configuração em *cold standby*, uma vez que os dispositivos menos utilizados tendem a durar mais por sofrerem menos desgastes.

3.4 Cálculo de Confiabilidade e Disponibilidade

Nessa seção são apresentados os métodos mais comuns para análise de R(t) e A(t): (1) Modelo de Markov e (2) Conjuntos-Conexos e Conjuntos-Desconexos. A discussão apresentada aqui servirá como base para o entendimento da proposta dessa dissertação, na qual ambos os métodos são estendidos e combinados.

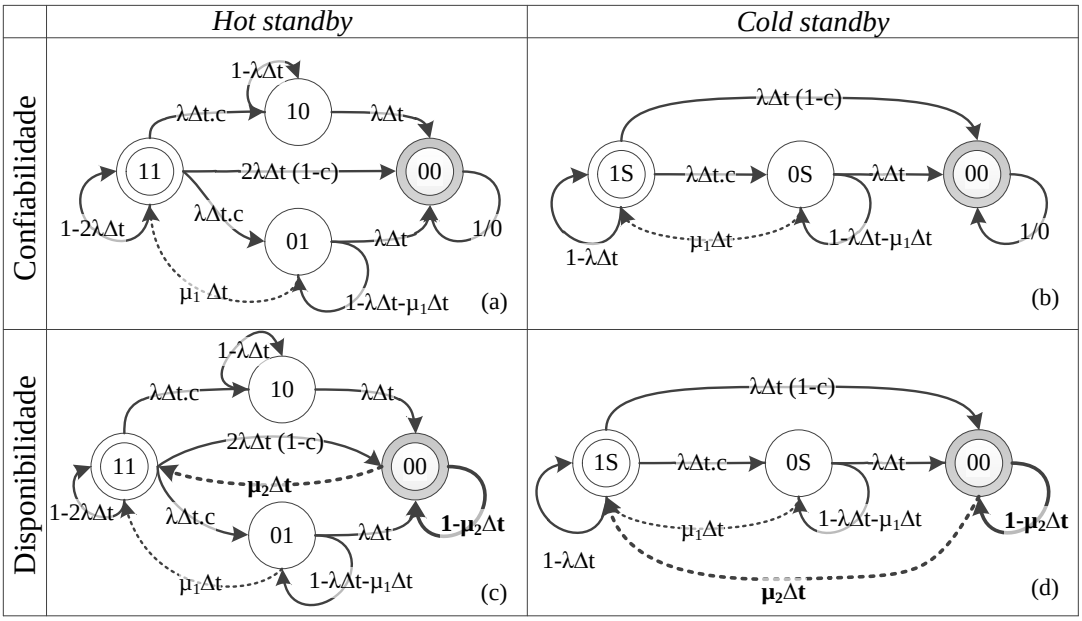
3.4.1 Modelo de Markov

O Modelo de Markov é utilizado para descrever os estados de um sistema e as transições entre esses estados. Os estados podem ser divididos em operacional, erro e falha. Após um ou mais erros, o sistema que estava operacional pode ir para o estado de falha. A premissa básica do Modelo de Markov é que não existe memória, logo, as probabilidades das transições são determinadas e baseadas apenas no estado presente, sem serem baseadas em históricos parciais ou completos (JOHNSON, 1989). Na avaliação da confiabilidade ou disponibilidade baseada no Modelo de Markov, as probabilidades de transições entre os estados são especificadas por uma distribuição exponencial da taxa de falha e do tempo de reparo ou manutenção do dispositivo da rede. Os estados do sistema e suas transições podem ser representados por um diagrama de transição de estados, que é um grafo direcionado. Esse tipo de diagrama contém informações suficientes para desenvolver equações de estados (TRIVEDI; SELVAMUTHU, 2008). O Modelo padrão de Markov para R(t) e A(t) está ilustrado na Figura 6, que, no caso, representa a modelagem de um roteador com duas conexões redundantes e as variações entre configurações de redundância *cold standby* e*hot standby*.

A Figura 6 ilustra o diagrama de transição de estados do Modelo de Markov para calcular confiabilidade e disponibilidade de um sistema em *hot standby* ou *cold satandby*. A modelagem de *hot standby* apresenta estados que representam todas as combinações das conexões dos roteadores estarem funcionando ou não. Como existem apenas duas conexões neste exemplo, podem-se ver os estados 10 e 01, que significa que pelo menos uma conexão está funcionando nesses estados, e logo o roteador está operando. Porém, o estado 00 representa uma falha, e o roteador parou de funcionar.

Na Figura 6, a modelagem de *cold standby* pressupõem que o componente redundante não está ativado, e só será ativado na ocorrência de um erro. Logo, para essa modelagem, não é feita as combinações realizadas no *hot standby*, pois o

Figura 6: Diagrama de transição de estado do Modelo de Markov para confiabilidade e disponibilidade para configurações de redundância em *hot standby* (a) e (c) e *cold standby* (b) e (d).



Baseada em (JOHNSON, 1989)

componente desativado não está sujeito à falha enquanto estiver desativado. Outro ponto importante na Figura 6, é que o modelo de disponibilidade difere do de confiabilidade por considerar a taxa de reparo. A Figura 6 é composta pelas seguintes variáveis:

- Δt representa o agregado do tempo decorrido desde que o sistema inicializou em t_0.
- "c" é o fator de cobertura, que representa a probabilidade de uma falha ser detectada e o componente em redundância ser completamente ativado.
- $(s_i s_j)$ representam o i-ésimo e j-ésimo estados das conexões i e j. Os estados podem assumir os valores: (1) totalmente operacional, (0) erro, (S) cold standby e (00) falha.

A probabilidade de o sistema estar totalmente operacional é obtida a partir do Modelo de Markov, calculando-se a soma das probabilidades de todos os estados

operacionais, ($P(t)_{11}$ + $P(t)_{10}$ + $P(t)_{01}$) no modo *hot standby*, ou $P(t)_{1S}$ + $P(t)_{0S}$) no modo *cold standby*; ou a partir do complemento da soma das probabilidades de todos os estados não operacionais, por exemplo: $(1 - P(t)_{00})$. As equações que calculam a confiabilidade e/ou a disponibilidade são: 3.1, 3.2, 3.3, e 3.4. Essas equações são derivadas dos Modelos de Markov da Figura 6. Em todas as equações, λ representa a taxa de falha do dispositivo e μ_1 a taxa de manutenção preventiva do dispositivo, que está relacionada ao tempo gasto para consertar erros, durante a operação do sistema (o sistema como um todo não falhou).

A Equação 3.1 representa a confiabilidade de um roteador com duas conexões redundantes configuradas em *hot standby*. Essa equação foi obtida a partir do diagrama de transição de estado do Modelo de Markov para confiabilidade ilustrada na Figura 6 (a).

$$\begin{aligned}
P(t+\Delta t)_{11} &= P(t_{01} * \mu_1 \Delta t + P(t_{11} * (1 - 2\lambda\Delta t) \\
P(t+\Delta t)_{10} &= P(t)_{11} * \lambda\Delta t * c + P(t_{10} * (1 - \lambda\Delta t) \\
P(t+\Delta t)_{01} &= P(t)_{11} * \lambda\Delta t * c + P(t_{01} * (1 - \lambda\Delta t - \mu_1\Delta t) \\
P(t+\Delta t)_{00} &= P(t)_{11} * 2\lambda\Delta t * (1 - c) + P(t)_{10} * \lambda\Delta t + P(t)_{01} * \lambda\Delta t + P(t)_{00} * 1 \\
\mathbf{R}(t+\Delta t) &= P(t+\Delta t)_{11} + P(t+\Delta t)_{10} + P(t+\Delta t)_{01} \\
\mathbf{R}(t+\Delta t) &\equiv 1 - P(t+\Delta t)_{00}
\end{aligned} \tag{3.1}$$

A Equação 3.2 representa a confiabilidade de um roteador com duas conexões redundantes em configuração de *cold standby*, como no diagrama de estado do Modelo

de Markov na Figura 6 (b).

$$
\begin{aligned}
P(t+\Delta t)_{1S} &= P(t)_{0S} * \mu_1 \Delta t + P(t)_{1S} * (1 - \lambda \Delta t) \\
P(t+\Delta t)_{0S} &= P(t)_{1S} * \lambda \Delta t * c + P(t)_{0S} * (1 - \lambda \Delta t - \mu_1 \Delta t) \\
P(t+\Delta t)_{00} &= P(t)_{1S} * 2\lambda \Delta t * (1 - c) + P(t)_{0S} * (1 - \lambda \Delta t) \\
\mathbf{R}(t+\Delta t) &= P(t+\Delta t)_1 + P(t+\Delta t)_{01} \\
\mathbf{R}(t+\Delta t) &\equiv 1 - P(t+\Delta t)_{00}
\end{aligned}
\tag{3.2}
$$

Nas equações 3.1 e 3.2, descritas acima, são expostas as fórmulas para o cálculo de R(t), no caso exemplificado na Figura 6 (a) e (b). A seguir serão apresentadas as equações para o cálculo de A(t), para os casos exemplificados na Figura 6 (c) e (d). A principal diferença apresentada entre as equações de disponibilidade e confiabilidade é que A(t) inclui taxa de reparo (μ_2), que está diretamente relacionada ao tempo gasto para reparar um sistema que falhou, e R(t) não possui μ_2, pois não retorna do estado de falha. A taxa de reparo μ_2 difere da taxa de manutenção preventiva μ_1. No caso da preventiva, μ_1, o sistema continua operando mesmo que tenha ocorrido um erro, já no caso de reparo, μ_2, o sistema parou de operar devido à falha e terá de ser reparado para voltar a funcionar.

A Equação 3.3 representa a disponibilidade de um roteador com duas conexões redundantes em configuração de *hot standby*, como descrito no diagrama de estado do

Modelo de Markov da Figura 6 (c).

$$
\begin{aligned}
P(t+\Delta t)_{11} &= P(t)_{01} * \mu_1 \Delta t + P(t)_{00} * \mu_2 \Delta t + P(t)_{11} * (1 - 2\lambda\Delta t) \\
P(t+\Delta t)_{10} &= P(t)_{11} * \lambda\Delta t * c + P(t)_{10} * (1 - \lambda\Delta t) \\
P(t+\Delta t)_{01} &= P(t)_{11} * \lambda\Delta t * c + P(t)_{01} * (1 - \lambda\Delta t - \mu_1\Delta t) \\
P(t+\Delta t)_{00} &= P(t)_{11} * \lambda\Delta t * (1 - c) + P(t)_{10} * \lambda\Delta t + P(t)_{01} * \lambda\Delta t + P(t)_{00} * (1 - \mu_2\Delta t) \\
\mathbf{A}(t+\Delta t) &= P(t+\Delta t)_{11} + P(t+\Delta t)_{10} + P(t+\Delta t)_{01} \\
\mathbf{A}(t+\Delta t) &\equiv 1 - P(t)_{00}
\end{aligned}
\tag{3.3}
$$

A Equação 3.4 representa a disponibilidade de um roteador com duas conexões redundantes em configuração de *cold standby*, como descrito no diagrama de estado do Modelo de Markov da Figura 6 (d).

$$
\begin{aligned}
P(t+\Delta t)_{1S} &= P(t)_{0S} * \mu_1 \Delta t + P(t)_{00} * \mu_2 \Delta t + P(t)_{11} * (1 - 2\lambda\Delta t) \\
P(t+\Delta t)_{0S} &= P(t)_{1S} * \lambda\Delta t * c + P(t)_{0S} * (1 - \lambda\Delta t - \mu_1\Delta t) \\
P(t+\Delta t)_{00} &= P(t)_{1S} * \lambda\Delta t * (1 - c) + P(t)_{0S} * \lambda\Delta t + P(t)_{00} * (1 - \mu_2\Delta t) \\
\mathbf{A}(t+\Delta t) &= P(t+\Delta t)_{1} + P(t+\Delta t)_{01} \\
\mathbf{A}(t+\Delta t) &\equiv 1 - P(t)_{00}
\end{aligned}
\tag{3.4}
$$

As soluções para disponibilidade, apresentadas nas equações 3.3 e 3.4, possuem um decaimento exponencial e transitório do termo e um ponto constante de convergência. Depois de algumas falhas e alguns ciclos de reparos, o termo transitório acaba e, então, a disponibilidade pode ser representada simplesmente pelo ponto constante de convergência. Para o caso de uma taxa de falha (λ) e de reparo (μ_2) constante, o ponto constante de convergência da disponibilidade (ou disponibilidade assintótica) é dado pela Equação 3.5, que também pode ser escrita relacionada ao

tempo que o sistema esteve operacional (*UP*) e não operacional (*DOWN*).

$$A(t)_{ss} = \frac{MTTF}{MTTF + MTTR} = \frac{UP}{UP + DOWN} \tag{3.5}$$

Na Equação 3.5, o Tempo Médio para Falhar (MTTF - *Mean Time to Failure*) e o Tempo Médio Entre Falhas (MTBF - *Mean Time Between Failure*) é dado pela expressão $1/\lambda$, e o Tempo Médio para Reparar (MTTR - M*Mean Time to Repair*) e o Tempo Médio Entre Reparos (MTBR - *Mean Time Between Repairs*) é dado pela expressão $1/\mu_2$. A solução mais utilizada no mercado pelos NSPs (*Network Service Providers*), para o cálculo de disponibilidade, é a Equação 3.5, uma vez que é muito mais fácil de ser calculada. (CISCO, 2003).

As equações 3.1, 3.2, 3.3, e 3.4, que foram descritas anteriormente resultam no cálculo da R(t) ou A(t) respectivamente. Essas equações podem, também, ser escritas no formato de matriz, empregando-se, por exemplo, a Equação 3.2 (confiabilidade e configuração em *cold standby*) e obtendo a matriz expressa na Equação 3.6.

$$\begin{bmatrix} P(t+\Delta t)_{1S} \\ P(t+\Delta t)_{01} \\ P(t+\Delta t)_{00} \end{bmatrix} = \begin{bmatrix} (1-\lambda\Delta t) & \mu_1\Delta t & 0 \\ \lambda\Delta tc & (1-\lambda\Delta t-\mu_1\Delta t) & 0 \\ \lambda\Delta t*(1-c) & (1-\lambda\Delta t) & 1 \end{bmatrix} \times \begin{bmatrix} P(t)_1 \\ P(t)_{01} \\ P(t)_{00} \end{bmatrix} \tag{3.6}$$

A Equação 3.6 pode, também, ser escrita em modo simplificado, como a seguir na Equação 3.7.

$$P(t+\Delta t) = \mathbf{X}P(\Delta t), \tag{3.7}$$

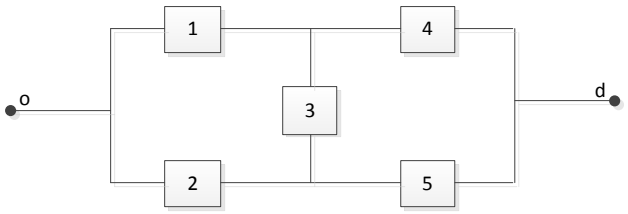
Na Equação 3.7, $P(t+\Delta t)$ é o vetor de probabilidade e $\mathbf{X}$ é a matriz de transição dos estados do Modelo de Markov da Figura 6 (b). Assumindo um valor inicial do vetor de probabilidade dos estados $P(0)$, o valor de $P(\Delta t)$ pode ser obtido pela expressão: $P(\Delta t) = \mathbf{X}P(0)$. De forma similar, o valor do vetor de probabilidade dos estados no tempo $2\Delta t$ pode ser escrito da seguinte forma: $P(2\Delta t) = \mathbf{X}P(\Delta t) = \mathbf{X}^2P(0)$. De forma

geral, a solução então, é dada como $P(n\Delta t) = \mathbf{X}^n P(0)$, onde $n\Delta t$ é o instante de tempo requerido para o cálculo da probabilidade (JOHNSON, 1989).

3.4.2 Conjuntos-Conexos e Conjuntos-Desconexos

A Figura 7 mostra o diagrama de blocos como uma representação gráfica da relação entre o funcionamento da rede e o funcionamentos dos seus componentes (roteador ou comutador). A rede é representada em uma estrutura em que os componentes ficam conectados em série, paralelo ou em malha. O método dos Conjuntos-Conexos (*Tie-Sets*) e dos Conjuntos-Desconexos (*Cut-Sets*) pode ser utilizado para avaliar a confiabilidade ou disponibilidade da rede. Esse método calcula a confiabilidade e disponibilidade da rede e assume que a confiabilidade e disponibilidade individual dos componentes (por exemplo, os roteadores) já são conhecidas *a priori* (LI; ZHAO, 2005). Desta forma, a confiabilidade individual de cada roteador pode ser, por exemplo, fornecida pela a avaliação realizada pelo Modelo de Markov apresentado anteriormente, ou simplesmente especificada na descrição do componente.

Figura 7: Diagrama de bloco da estrutura de uma rede de computadores.



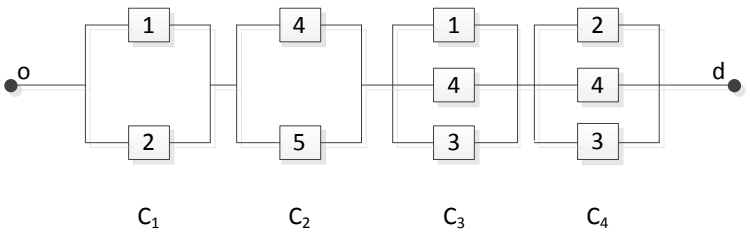
Baseada em (LI; ZHAO, 2005)

Esse método calcula de maneira combinatória a confiabilidade ou disponibilidade de dois-pontos-terminais, por exemplo, entre origem “o” e destino “d” da Figura 7. O método também pode calcular para todos-pontos-terminais, no caso da Figura 7 existe apenas um ponto terminal, porém, pode existir mais origens e destinos na rede.

O método dos Conjuntos-Conexos e Conjuntos-Desconexos reduz a complexidade do cálculo da confiabilidade e disponibilidade da rede, quando comparado com o Modelo de Markov, porque concentra a análise nos Conjuntos-Conexos mínimos ou nos Conjuntos-Desconexos mínimos (SHOOMAN, 2001). O termo mínimo implica que o nó ou a aresta não é percorrido mais de uma vez, ou seja, são combinações que não são sub-conjunto de outras combinações. A redução da complexidade acontece, porque o Modelo de Markov, diferentemente desse método, analisa todas as combinações possíveis, tendo assim, um conjunto maior de dados para serem analisados.

O Conjunto-Desconexo é o conjunto de componente que quando falha causa falha em todo o sistema, porém quando nenhum dos componentes falha, o sistema como um todo não falha. Os componentes em um Conjunto-Desconexo estão conectados em paralelo. Além disso, se existir mais de um Conjunto-Desconexo, o sistema irá falhar se todos os componentes em algum dos Conjuntos-Desconexos falharem. Desta forma, todos os Conjuntos-Desconexos estão conectados em série, como os Conjuntos-Desconexos mínimos do diagrama de bloco da Figura 7 representados na Figura 8. Por exemplo, os Conjuntos-Desconexos mínimos da Figura 7 são {1,2}, {4,5}, {1,4,3} e {2,4,3}.

Figura 8: Conjuntos-Desconexos mínimos.

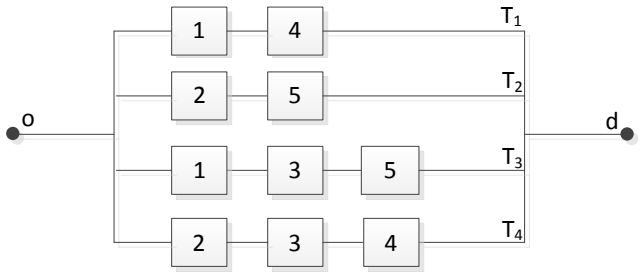


Baseada em (LI; ZHAO, 2005)

Um Conjunto-Conexo é composto pelos caminhos mínimos da rede, e logo representa os dispositivos da rede conectados de série. Esse conjuntos é o agrupamento

de arestas que formam um caminho entre a origem “o” e o destino “d”, como na Figura 9. Consequentemente, um Conjunto-Conexo falha quando qualquer um dos componentes que o compõem tenha falhado e a probabilidade de falha pode ser avaliada utilizando o princípio dos sistemas em série (LI; ZHAO, 2005). Para o sistema falhar, todos os Conjuntos-Conexos têm que ter falhado porque eles estão em paralelo. Os Conjuntos-Conexos mínimos do diagrama de bloco da Figura 7 estão representados na Figura 9. Por exemplo, os Conjuntos-Conexos mínimos da Figura 7 são {1,4}, {2,5}, {1,3,5} e {2,3,4}. Desta forma, a disponibilidade ou confiabilidade do sistema apresentado na Figura 9 é a união das probabilidades de estar funcionado dos Conjuntos-Conexos.

Figura 9: Conjuntos-Conexos mínimos.



Baseada em (LI; ZHAO, 2005)

Para calcular a confiabilidade ou disponibilidade, o método dos Conjuntos-Conexos e Conjuntos-Desconexos enumera todas as combinações de todos os nós, entre “o” e “d”, estarem em estado operacional (*UP*) ou em estado não operacional (*DOWN*), como evento mutuamente exclusivo. A união da probabilidade dos **n** elementos contidos nos Conjuntos-Conexos, ou o complemento da união da probabilidade dos **m** elementos contidos nos Conjuntos-Desconexos, resulta na Confiabilidade ou Disponibilidade da rede (SHOOMAN, 2001). Desta forma, o cálculo dos Conjuntos-Conexos é dado pelo *princípio da inclusão e exclusão*, como

descrito nas equações 3.8 e 3.9 a seguir.

$$T_n = P(Roteador_1 \cap Roteador_2 \cap ... \cap Roteador_m) \tag{3.8}$$

$$\begin{aligned} R(t) \text{ or } A(t)_s t &= P(T_1 \cup T_2 \cup ... \cup T_n) \\ &\equiv \sum_{i=1}^{n} P(T_i) - \sum_{i<j} P(T_i \cap T_j) \\ &\quad + \sum_{i<j<k} P(T_i \cap T_j \cap T_k) - ... \\ &\quad + (-1)^{n-1} P(T_1 \cap T_2 \cap ... \cap T_n) \end{aligned} \tag{3.9}$$

Na Equação 3.8, o T_n significa o n-ésimo elemento contido no Conjunto-Conexo, que significa a combinação de nós que conectam "o" e "d". A probabilidade do T_n estar funcionando é dada pela intersecção da probabilidade de cada nó, que compõe o T_n, estar funcionando. E $Router_i$ representa a confiabilidade ou disponibilidade individual dos componentes da rede.

A Equação 3.9, a partir do *princípio da inclusão e exclusão*, calcula a R(t) ou A(t) da rede. Este cálculo é feito em várias etapas: primeiro, soma-se a probabilidade individual de cada evento T_n; depois, subtraem-se a soma da probabilidade de todas as intersecções das combinações dos eventos tomada em pares; em seguida, adiciona-se novamente a soma da probabilidade de todas as intersecções das combinações dos eventos tomadas três a três; depois, subtraem-se as combinações tomadas de quatro a quatro. O processo de adição e subtração da soma das intersecções das k combinações é repetido até que a probabilidade das n combinações é atingida. R(t) ou A(t) da rede pode ser, também, obtida pelo complemento da probabilidade resultante dos Conjuntos-Desconexos, como descrito nas equações 3.10 e 3.11.

$$C_m = P(Roteador_1 \cap Roteador_2 \cap ... \cap Roteador_m) \tag{3.10}$$

$$
\begin{aligned}
R(t)_{od} \text{ or } A(t)_{od} &= 1 - [P(C_1 \cup C_2 \cup ... \cup C_m)] \\
&\equiv 1 - [\textstyle\sum_{i=1}^{m} P(C_i) - \sum_{i<j} P(C_i \cap C_j) \\
&\quad + \textstyle\sum_{i<j<k} P(C_i \cap C_j \cap C_k) - ... \\
&\quad + (-1)^{\text{m-1}} P(C_1 \cap C_2 \cap ... \cap C_m)]
\end{aligned}
\tag{3.11}
$$

As equações 3.10 e 3.11 são calculadas de forma similar a descrita nos Conjuntos-Conexos, utilizando o *princípio da inclusão e exclusão*.

Como exemplo do cálculo da confiabilidade do sistema da Figura 7, tomando que a confiabilidade individual dos roteadores já foram calculas utilizando Modelo de Markov, resultando em 0.85 para os roteadores de 1, 2, 4 e 3, e 0.95 para o roteador 3. Logo, a confiabilidade da rede pode ser obtida pela união da probabilidade dos Conjuntos-Conexos da Figura 9:

$$
\begin{aligned}
T_1 &= P(Roteador_1 \cap Roteador_4) \\
T_2 &= P(Roteador_2 \cap Roteador_5) \\
T_3 &= P(Roteador_1 \cap Roteador_3 \cap Roteador_5) \\
T_4 &= P(Roteador_2 \cap Roteador_3 \cap Roteador_4)
\end{aligned}
$$

Substituindo o valor da confiabilidade individual de cada roteador, temos:

$$
\begin{aligned}
T_1 &= P(0.85 * 0.85) &&= 0.7225 \\
T_2 &= P(0.85 * 0.85) &&= 0.7225 \\
T_3 &= P(0.85 * 0.95 * 0.85) &&= 0.6863 \\
T_4 &= P(0.85 * 0.95 * 0.85) &&= 0.6863
\end{aligned}
$$

Logo, através do *princípio da inclusão e exclusão*, e utilizando a Equação 3.9, o

cálculo fica da seguinte maneira:

$$\begin{aligned} R(t) &= (T_1 + T_2 + T_3 + T_4) \\ &\quad -(T_1 * T_2 + T_1 * T_3 + T_1 * T_4 + T_2 * T_3 + T_2 * T_4 + T_3 * T_4) \\ &\quad +(T_1 * T_2 * T_3 + T_1 * T_2 * T_4 + T_2 * T_3 * T_4) \\ &\quad -(T_1 * T_2 * T_3 * T_4) \end{aligned}$$

$$R(t) = (2.8176) - (2,9764) + (1,0388) - (0,2458) = 0.6341$$

Se em vez de utilizar a confiabilidade individual dos roteadores, for utilizada a disponibilidade individual de cada roteador, o resultado do método Conjuntos-Conexos e Conjuntos-Desconexo será a disponibilidade da rede, como apresentado por Green et. al. (GREEN; HANT; LANZINGER, 2009). Logo, neste caso, o resultado depende se no Modelo de Markov foi calculada a confiabilidade ou disponibilidade individual dos roteadores.

As formulas apresentadas anteriormente são para calcular a confiabilidade ou a disponibilidade entre dois-pontos-terminais na rede, ou seja, entre a origem "o" e o destino "d". Porém, é possível calcular para todos-pontos-terminais, no qual é necessário primeiro calcular a probabilidade de dois-pontos-terminais para todos os nós da rede tomados na combinação de 2 em 2 (origem e destino), como apresentado acima. Depois, é calculada a união de todos os resultados do cálculo de dois-pontos-terminais, como na Equação 3.12.

$$\begin{aligned} R(t)_{rede} &= \sum_{j=1}^{(roteadores,2)} R(t)_{od_j} \\ A(t)_{rede} &= \sum_{j=1}^{(roteadores,2)} A(t)_{od_j} \end{aligned} \tag{3.12}$$

3.5 Disponibilidade e confiabilidade como métricas e o impacto do tempo de inatividade

A confiabilidade e a disponibilidade podem ser medidas para diferentes fins. Se forem medidas orientadas para o cliente, ou seja, mostrando a disponibilidade e/ou confiabilidade experimentada na perspectiva do cliente, terá incluso na medição todas as camadas e partes da rede que conecta dois pontos. Por exemplo, se um enlace falha, o tráfego é redirecionado para outro enlace redundante, logo, na perspectiva do cliente a conexão está disponível e nenhum tempo de indisponibilidade é calculado (THULIN, 2004). Essa medida é também chamada de Qualidade de Experiência do usuário (QoE - *Quality of Experience*) (CABRAL, 2007).

A outra forma de medir a disponibilidade é de uma perspectiva do gerenciamento de rede. O objetivo é garantir QoS na rede, através da administração dos recursos da rede e da manutenção da rede de forma eficiente. Por exemplo, se um enlace falha e o tráfego é redirecionado para outro enlace, na perspectiva do sistema de gerenciamento de rede é importante saber quais conexões e componentes na rede apresentam menor disponibilidade e confiabilidade. Pois, desta forma, o sistema pode redirecionar recursos para encontrar os problemas dos componentes que apresentam menor disponibilidade e confiabilidade, e então, ele poderia repara-los, pois tais problemas poderão ser uma ameaça futura à qualidade do serviço da rede.

A confiabilidade e a disponibilidade são medidas em forma de porcentagem, representando a probabilidade de o sistema estar funcionando, ou estar disponível, respectivamente. Assim, comumente elas são expressas no formato de cinco noves 99,999%, esse formato é uma generalização que tem sido utilizada vastamente pelo mercado, pelos no contexto das redes de metropolitanas e de núcleo (THULIN, 2004). Cinco noves correspondem a 5 minutos de inatividade por ano, e seis noves a 32 segundos de inatividade por ano. A Tabela 2 explica a relação entre a porcentagem

e os minutos de inatividade da rede por ano. É importante notar que mudar de quatro noves para cinco requer passar de 52 minutos para 5 minutos de inatividade por ano. Lembrando que existe uma relação de compromisso entre manter alta a confiabilidade e disponibilidade e os custos para atingir a quantidade de noves requerida. A quantidade de noves depende de quais tipos de serviços estão sendo transmitidos na rede.

Tabela 2: Tempo de inatividade relacionado a probabilidade de disponibilidade

Disponibilidade	Tempo de inatividade (*downtime* por ano)
0.999999	32s
0.99999	5min 15s
0.9999	52min 36s
0.999	8h 46min
0.99	3 dias 15h e 40min

Baseada em (THULIN, 2004)

Uma pesquisa sobre as tendências sobre a alta disponibilidade e confiabilidade, ou seja, quantos noves as empresas estão interessadas, foi realizada pela revista SearchCIO-Midmarket TechTarget (GUGLIELMO, 2010). A pesquisa revela que uma entre 10 empresas afirmam que elas necessitam de mais de cinco noves de disponibilidade. Além disso, para realmente entender o que significa os seis noves, é importante saber o custo dos períodos de inatividade.

De acordo com Marcus et. al. (MARCUS; STERN, 2000), a maneira mais intuitiva de se medir o custo do tempo de inatividade não é provavelmente o que mais se custa: a perda de produtividade do usuário. O custo atual do tempo de inatividade depende de que tipo de trabalho o faz em relação ao sistema afetado. Se os usuários são programadores, então o custo não será além do custo atrelado ao tempo que os programadores ficaram ociosos. Porém, no caso de uma fabrica de software este custo pode ser significativo. Supondo que um desenvolvedor pode ter um salário entre R$ 400,00 à R$ 1000,00 por dia. É razoável assumir que um grupo de 50 desenvolvedores ociosos poderá custar R$ 20.000,00 ou mais por semana. Entretanto, esse custo de R$ 20.000,00 não é mais que o custo atrelado aos desenvolvedores ficar ociosos. Não foi

levado em consideração o tempo requerido em horas extras para repor o tempo perdido e assegurar que as entregas não serão atrasadas. Além disso, esse cálculo não leva em consideração fatores como cansaço por trabalhar mais que uma jornada de trabalho e o impacto no psicológico do funcionário.

Esses custos, também variam em relação ao tipo de trabalho que está sendo realizado. Para ações de compra e venda na bolsa de valor, o custo pode ser cerca de R$ 1 milhão a cada 5 minutos de inatividade. Esse valor de R$ 1 milhão está relacionado a todas as ações de compra e venda são rentáveis. Além disso, esse cálculo não leva em consideração a possibilidade que uma ação de compra e venda poderia estar sendo efetuada para salvar os investimentos de uma empresa, e caso não seja efetuado a empresa poderia quebrar.

De acordo com Mackay (MACKAY, 2013), foi estimado em 2011 que os custos relacionados ao tempo de inatividade referente a TI é de R$ 53 bilhões. A Tabela 3 apresenta a média de custos relacionados ao tempo de inatividade para vários tipos de serviços (essa referência é utilizada em vários trabalhos atuais). O tempo de inatividade varia de empresa para empresas (MACKAY, 2013) como em 2013: o Google, que suporta 40% do tráfego global da Internet, apresentou menos de 5 minutos; o Windows Azure da Microsoft apresentou em torno de 20 horas; os Serviços Web da Amazon menos de 3 horas, a Amazon.com 49 minutos e a NASDAQ 3 horas.

Por fim, o tempo de inatividade custa dinheiro, porém, ele pode ter um custo maior, o custo da empresa inteira, devido a fatores como: impactos na satisfação do cliente, reputação, preço de mercado (vender em períodos diferentes), etc (MARCUS; STERN, 2000).

Tabela 3: As direções dos custos relacionados ao tempo de inatividade

Indústria	Custo de tempo de inatividade por hora
Transações bancárias	R$ 12,96 milhões
Energia	R$ 5,2 milhões
Cartão de crédito	R$ 5,16 milhões
Telecomunicações	R$ 4 milhões
Manufatura	R$ 3,2 milhões
Instituições financeira	R$ 2,8 milhões
Varejo	R$ 2,2 milhões
Farmaceutica	R$ 2 milhões
Química	R$ 1,4 milhões
Saúde	R$ 1,2 milhões
Midia	R$ 680.000
Passagem aérea	R$ 180.000

Baseada em (MARCUS; STERN, 2000) apud *Network Computing, the Meta Group, and Planning Research*

3.6 Norma G.826 de Qualidade e Disponibilidade da ITU-T

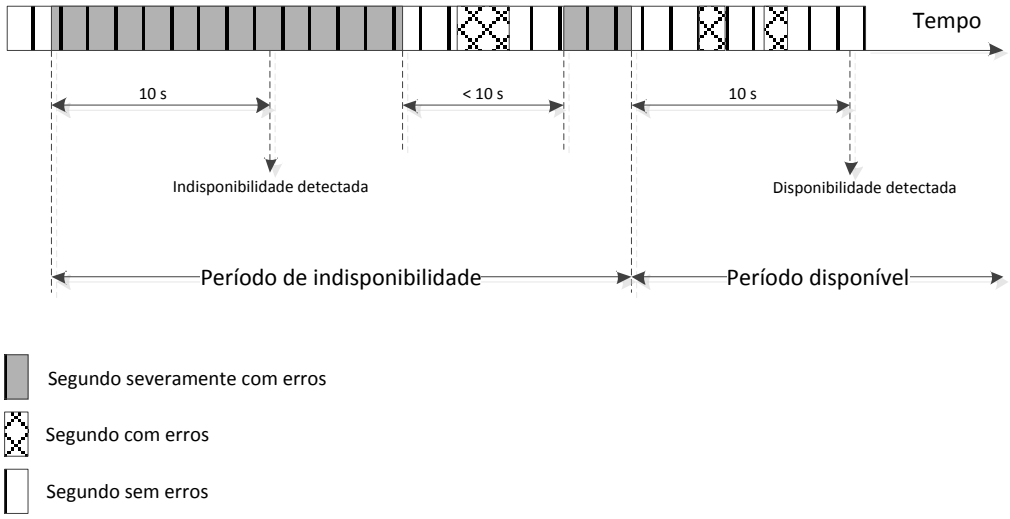
A União Internacional de Telecomunicações (ITU - *International Telecommunication Union*) é uma organização dentro das Nações Unidas, com sede em Genebra, Suíça. A ITU é composta por representantes de órgãos legislativos de países membros, assim como representantes de setores privados de países membros que trabalham na esfera de atividades da ITU. O principal objetivo da ITU é desenvolver normas e procedimentos, e trabalhar para melhorar a infraestrutura de telecomunicações. A ITU-T (ITU *Telecommunication Standardization Sector*) é o setor criação de padrões para telecomunicações da ITU.

ITU-T definiu uma norma que especifica avaliação de desempenho e medidas de disponibilidade, chamada de G.826: "*End-to-end error performance parameters and objectives for international, constant bit rate digital paths and connections*". A norma G.826 é uma recomendação que define parâmetros e objetivos para calcular disponibilidade de enlaces internacionais. Essa norma independe de a rede física trabalhar com caminhos ou conexões. Para caminhos digitais que operam abaixo de

uma taxa pré-estabelecida, a recomendação baseia-se em conceito de blocos, utilizando detecção de erros inerente do caminho que está sendo testado. Para conexões digitais que operam abaixo de uma taxa pré- estabelecida, essa recomendação baseia-se em erro de bit e medida de taxa de erro de bit (ITU-T, 2002). A Figura 10 da um exemplo de determinação dos períodos de indisponibilidade. Os parâmetros que utilizados para determinar os períodos de indisponibilidade são:

- **Bloco**: O bloco é definido como o conjunto de bits consecutivos, que estão associados a um caminho; cada bit pertence a apenas um bloco.
- **Bloco com erro**: Um bloco no qual um ou mais bits apresentam erros.
- **Segundo com erro**: Período de um segundo que um ou mais blocos apresentam erros, ou pelo menos um defeito.
- **Segundo severamente com erro**: Período de um segundo que apresenta mais de 30% de blocos com erros, ou pelo menos um defeito.

Figura 10: Exemplo de determinação dos períodos de indisponibilidade da norma G.826



Baseada em (ITU-T, 2002)

Na Figura 10, o período de indisponibilidade inicia quando acontece 10 eventos consecutivos de segundos com erros. Esses 10 segundos são considerados como parte do período de indisponibilidade. Um novo período de disponibilidade inicia-se quando acontecem 10 eventos consecutivos de segundos sem erros. Esses 10 segundos são considerados como parte do período de disponibilidade (ITU-T, 2002).

3.7 Considerações finais do capítulo

As técnicas de redundância, como redundância em *cold standby*, permite economizar energia, aumentar o tempo de vida de alguns dispositivos e a princípio melhorar a disponibilidade ou confiabilidade da rede (SHOOMAN, 2001), (JANUARIO et al., 2013), por manter alguns componentes inativos durante algum período. Por outro lado, *hot standby* pode contribuir para melhorar desempenho na rede, dado que é possível aplicar técnicas de engenharia de tráfego na rede.

As técnicas para calcular e aprimorar a confiabilidade e a disponibilidade da rede são fundamentais para a avaliação da probabilidade de falha da rede, auxiliando no provisionamento das redes de computadores. Todas as técnicas possuem vantagens e desvantagens entre si, sendo estas complementares em muitos casos ((SHOOMAN, 2001; ALTIPARMAK; DENGIZ; SMITH, 2003; GREEN; HANT; LANZINGER, 2009; HE; QI, 2008; LAM; LI, 1986; LIN et al., 2010), e (YEH et al., 2010).

Das técnicas de calcular, o Modelo de Markov é um método bastante flexível e parametrizável, permitindo uma análise aprofundada do comportamento dos componentes individuais da rede. Porém, avaliar o comportamento de uma rede muito grande, composta por centenas ou milhares de nós, pode ser muito custoso computacionalmente. No caso do método dos Conjuntos-Conexos e Conjuntos-Desconexos, é analisado um número menor de informações e, logo, a complexidade do cálculo pode ser reduzida. Porém, esse método é menos flexível, de forma a ser

difícil a customização de parâmetros e uma análise mais aprofundada da rede.

Por fim, é importante saber quais são os objetivos dos serviços que irão ser transmitidos na rede para poder avaliar a confiabilidade e disponibilidade da rede. Pois, cada tipo de serviço possui seus próprios requisitos e alguns são mais sensíveis a períodos de indisponibilidade na rede. Por exemplo, os serviços de transações bancárias, no qual reflete a perda de R$ 1 milhão a cada 5 minutos de inatividade, e 5 minutos de inatividade significa ter a disponibilidade com mais de seis casas decimais. Logo, essa análise é imprescindível para uma empresa, pois, o tempo de inatividade custa dinheiro, e além disso pode ter um ainda custo maior, o custo de a empresa inteira falir dado que ela ficará sem prover seu serviço.

4 MODELAGEM PROPOSTA PARA CALCULAR CONFIABILIDADE E DISPONIBILIDADE

Este capítulo descreve a proposta de um método para Avaliação de Confiabilidade e/ou Disponibilidade em Redes de Computadores Sustentáveis chamado REASoN (*Reliability and Availability Evaluation of Sustainable Network*). Esse método pode ser utilizado para analisar o impacto de colocar e retirar um dispositivo da rede em estado "dormente" em termos de confiabilidade e disponibilidade da rede. Mais especificamente, o REASoN auxilia na avaliação do impacto das dinâmicas e frequentes transições entre os estados *hot standby* (totalmente operacional) e *cold standby* (descritos no capitulo 3). Para tanto, é considerado no cálculo do método o tempo atrelado à ação de mudança de estado dos dispositivos.

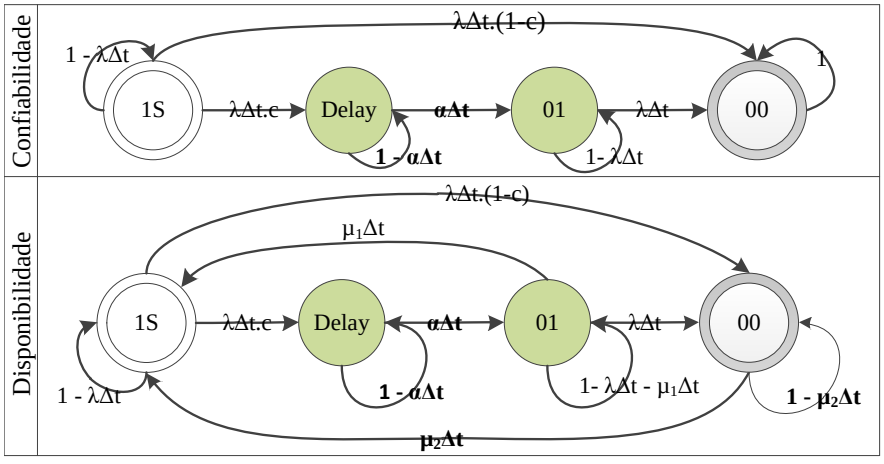
4.1 REASON - Avaliação de confiabilidade e disponibilidade em redes de computadores sustentáveis

O REASoN calcula a confiabilidade e/ou disponibilidade da rede em duas etapas, como descrito no Capíulo 3. Primeiramente, é calculada a R(t) ou a A(t) individual de cada dispositivo da rede (roteador ou comutador), utilizando uma versão extendida do Modelo de Markov. Depois disso, é calculada a R(t) ou a A(t) da rede, ou seja, quando esses dispositivos estão conectados entre si, utilizando uma versão estendida do método dos Conjuntos-Conexos e Conjuntos-Desconexos. Como o método dos Conjuntos-Conexos e Conjuntos-Desconexos necessita que a confiabilidade individual

dos roteadores seja definida *a priori*, esse método tem como entrada o resultado do cálculo do Modelo de Markov.

O REASoN realiza o cálculo da primeira etapa utilizando uma modelagem estendida do Modelo de Markov apresentado anteriormente no capítulo 3. Nessa extensão é considerado no cálculo o tempo médio para acordar um dispositivo em *cold standby* (estado "dormente" onde se economiza energia). Como dito anteriormente, essa etapa calcula apenas a R(t) ou A(t) individual de cada dispositivo. A modelagem estendida do Modelo de Markov é apresentada na Figura 11, a qual representa a modelagem de R(t) ou A(t) para um roteador com duas conexões que levam para um mesmo destino final, ou seja, redundantes. A extensão proposta está relacionada à modelagem do *cold standby*, na qual foi incluído um estado a mais, chamado de *Delay*, que representa um situação em que o sistema fica inoperante durante o tempo médio para ativar um dispositivo em estado "dormente".

Figura 11: Modelo de Markov estendido para considerar o tempo médio de acordar um dispositivo (α).



Na Figura 11, o 1 representa uma conexão ativa, o S, uma conexão em *cold standby*, o 0, um erro e o 00, uma falha. O *Delay* representa a situação em que ocorreu um erro e o sistema encontra-se no estado 01′, o 1′ representa uma conexão sendo ativada e temporariamente inoperante, logo, nesse estado o roteador não está disponível. A transição do estado *Delay* não está relacionada a probabilidade de

falha, mas está relacionada probabilidade de transição para um estado operacional, em relação ao tempo médio para a acordar uma conexão "dormente". Essa transição é como se fosse a transição da manutenção preventiva descrita do Capítulo 3. O estado $01'$ (*Delay*) e o estado 01 não podem ser combinados em apenas um estado, uma vez que o primeiro representa um estado indisponível e o outro um estado disponível. Além disso, se o tempo médio para acordar um dispositivo for incluído no estado 01, esse tempo irá alterar a probabilidade de falha do estado. Logo, devido a esses motivos, esses estados são modelados de forma separada.

Ainda na Figura 11, o α significa 1/(tempo médio para acordar o dispositivo). O tempo médio para acordar o dispositivo está relacionado ao tempo que a rede leva para se estabilizar e o tempo que o dispositivo leva "para ser ativado". O tempo médio de ativação do dispositivo depende especificamente da implementação do estado "dormente" (atividade do SO - Sistema Operacional, alocação de memória, operações relacionadas à troca de estado, atividade de subsistemas, etc). O "c" é o fator de cobertura, que representa a probabilidade que um erro tem de ser identificado, juntamente com a probabilidade de ativar o dispositivo redundante. O μ_1 representa o tempo médio de manutenção preventiva, que ocorre durante o funcionamento do sistema, sem que este tenha falhado. O μ_2 representa o tempo médio para reparar um sistema que falhou e parou de operar.

A principal diferença entre a modelagem padrão do Modelo de Markov e a modelagem apresentada pelo REASoN para redundância em *cold standby* é que o REASoN inclui um estado, que representa o tempo médio que um dispositivo leva para ser ativado. É importante ressaltar que esse estado representa um "pênalti". Esse estado de pênalti apenas está relacionado ao tempo médio de atraso para ativar o dispositivo redundante, não interferindo na probabilidade de não ativação. Esse "estado de pênalti" interfere na probabilidade do sistema estar operando, o que degrada a confiabilidade e a disponibilidade.

A equação 4.1 representa o diagrama de estado da modelagem de confiabilidade da Figura 11.

$$\begin{aligned}
P(t+\Delta t)_{1S} &= P(t)_{1S} * (1-\lambda\Delta t) \\
P(t+\Delta t)_{Delay} &= P(t)_{1S} * \lambda\Delta t * c + P(t)_{Delay} * (1-\alpha\Delta t) \\
P(t+\Delta t)_{0S} &= P(t)_{Delay} * \alpha\Delta t + P(t)_{0S} * (1-\lambda\Delta t) \\
P(t+\Delta t)_{00} &= P(t)_{0S} * \lambda\Delta t + P(t)_{00} \\
\mathbf{R}(t+\Delta t) &= P(t+\Delta t)_{1S} + P(t+\Delta t)_{0S} \\
\mathbf{R}(t+\Delta t) &\equiv 1 - (P(t+\Delta t)_{00} + P(t+\Delta t)_{Delay})
\end{aligned} \tag{4.1}$$

A equação 4.2 representa o diagrama de estado da modelagem de disponibilidade da Figura 11.

$$\begin{aligned}
P(t+\Delta t)_{1S} &= P(t)_{0S} * \mu_1\Delta t + P(t)_{00} * \mu_2\Delta t \\
&\quad + P(t)_{1S} * (1-\lambda\Delta t) \\
P(t+\Delta t)_{Delay} &= P(t)_{1S} * \lambda\Delta t * c + P(t)_{Delay} * (1-\alpha\Delta t) \\
P(t+\Delta t)_{0S} &= P(t)_{Delay} * \alpha\Delta t + P(t)_{0S} * (1-\lambda\Delta t + \mu_2\Delta t)) \ . \\
P(t+\Delta t)_{00} &= P(t)_{0S} * \lambda\Delta t + P(t)_{00} * (1-\mu_1\Delta t) \\
\mathbf{A}(t+\Delta t) &= P(t+\Delta t)_{1S} + P(t+\Delta t)_{0S} \\
\mathbf{A}(t+\Delta t) &\equiv 1 - (P(t+\Delta t)_{00} + P(t+\Delta t)_{Delay})
\end{aligned} \tag{4.2}$$

É importante notar que nas equações 4.1 e 4.2, a probabilidade do estado de pênalti (estado de atraso) não é adicionada à probabilidade geral do sistema estar funcionado, pois será considerada um estado de erro. É importante ressaltar aqui, que um estado de erro não é um estado de falha, mas um ou mais erros podem levar ao estado de falha. Essas equações são resolvidas de forma interativa como descrito no **Algoritmo 1**. No algoritmo é considerada a premissa básica de confiabilidade e disponibilidade, que considera que todos os componentes estão funcionando corretamente no estado inicial, ou seja, é assumido que no instante t = 0, o primeiro estado tem probabilidade

igual a 1, enquanto todos os outros estados têm probabilidades iguais a 0. O tempo é, então, incrementado iterando as expressões dadas pelo Modelo de Markov, como as equações 4.1 e 4.2.

O REASoN realiza o cálculo da segunda etapa, utilizando uma extensão do método dos Conjunto-Conexo e Conjunto-Desconexo apresentado anteriormente no capitulo 3. Porém, o REASoN, diferentemente do método padrão, considera que os dispositivos que estão em sobrecarga (acima de um determinado limite) ou em estado "dormente" dentro dos Conjuntos-Desconexos, dado que estão indisponíveis temporariamente. Como visto anteriormente, o método dos Conjunto-Conexo e Conjunto-Desconexo necessita que a probabilidade individual dos roteadores sejam fornecidas *a priori*, logo, no REASoN, esse método utiliza as probabilidades calculas para os roteadores utilizando o Modelo de Markov. Esse processo é descrito no **Algoritmo 2**.

O Algorítmo 2 calcula a confiabilidade ou a disponibilidade utilizando os Conjuntos-Conexos ou os Conjuntos-Desconexos da rede e o Modelo de Markov. Primeiramente é calculada a confiabilidade individual de cada roteador utilizando Modelo de Markov (para aprimorar a complexidade do cálculo, essa avaliação pode ser feita previamente e os valores apenas consultados em alguma estrutura de dados). Depois desse passo, o algoritmo realiza a interseção dos valores das probabilidades dos roteadores contidos dentro de um Conjunto-Conexo ou um Conjunto-Desconexo para todos os Conjuntos-Conexos ou Conjuntos-Desconexos, respectivamente. Então, por fim, o algoritmo calcula a confiabilidade ou disponibilidade através da união de todos dos Conjuntos-Conexos, ou pelo complemento da união dos Conjuntos-Desconexos.

A complexidade assintótica do Algoritmo 1, que contém dois laços aninhados é $\Theta(w * T)$. Sendo w a quantidade de estados do Modelo de Markov e T o tempo corrido durante execução do sistema. A complexidade assintótica do Algoritmo 2 é $\Theta(r * n * T)$. Sendo r a quantidade de elementos no Conjunto dos Caminhos Conexos, v a quantidade de elementos no Conjunto dos Caminhos Desconexos, e T o tempo corrido

Algoritmo 1: Processo iterativo para calcular R(t) e A(t) individual dos roteadores, segundo o Modelo de Markov.

Saída: R(t) ou A(t) individual dos dispositivos da rede

t é o tempo atual;
n é tempo máximo a ser analisado;
i é o estado atual;
w é a quantidade total de estados;
P(*t*) *é a probabilidade de cada estado*;

```
if Confiabilidade then
    for t ← 0 to n − 1 do
        if t = 0 then
            P(0)_0 ← 1
            P(0)_1...w ← 0
            R(t) = P(0)_i ∗ ... ∗ P(0)_w
        else
            for i ← 0 to w − 1 do
                R(t) + = P(t − 1)_i ∗ P(t)_i
                i + = 1
            end
        end
        t + = Δt
    end
else
    for t ← 0 to n − 1 do
        if t = 0 then
            P(0)_0 ← 1
            P(0)_1...w ← 0
            A(t) = P(0)_i ∗ ... ∗ P(0)_w
        else
            for i ← 0 to w − 1 do
                A(t) + = P(t − 1)_i ∗ P(t)_i
                i + = 1
            end
        end
        t + = Δt
    end
end
```

Algoritmo 2: Processo iterativo para calcular R(t) e A(t) da rede utilizando o método do Conjunto-Conexo e do Conjunto-Desconexo.

Saída: R(t)ou A(t) de (dois ou todos)-pontos-terminais da rede

t é o tempo atual;
T_n é o n-ésimo elemento do Conjunto-Conexo T;
C_n é o n-ésimo elemento do Conjunto-Desconexo C;
r representa um roteador de T_n;
v representa um roteador de C_n;

```
if T then
    for T_n in T do
        for r in T_n do
            T_n[r] = Algoritmo1(r, t, λ, α, c, μ)  //Probabilidade individual do roteador
        end
        T[T_n] = Equação 3.8 (T_n)  //Intersecção dos roteadores
    end
    R(t) ou A(t) = Equação 3.9 (Tie-Sets)  //Princípio da inclusão e exclusão
else
    for C_n in C do
        for v in C_n do
            C_n[v] = Algoritmo1(v, t, λ, α, c, μ)  //Probabilidade individual do roteador
        end
        C[C_n] = Equação 3.10 (C_n)  //Intersecção dos roteadores
    end
    R(t) ou A(t) = Equação 3.11 (Cut-Sets)  //Princípio da inclusão e exclusão
end
```

durante execução do sistema.

4.2 Considerações finais do capítulo

As redes de computadores sustentáveis trazem uma nova complexidade, dado que durante alguns intervalos de tempo, partes da rede poderão ter suas capacidades reduzidas, a fim de economizar energia na rede, colocando, por exemplo, um ou mais dispositivos no estado "dormente". Essa prática pode trazer impactos na rede, sejam, por exemplo, na sua confiabilidade ou disponibilidade, o que dificilmente são analisados a priori. Esses impactos decorrem, principalmente, do tempo médio necessário para ativar um dispositivo caso isso seja necessário fazer no caso da ocorrência de falhas ou de aumento de tráfego da rede.

Dentro desse contexto, esse capítulo descreveu um método capaz de avaliar os impactos inerentes na confiabilidade e/ou disponibilidade de uma rede em operação quando os dispositivos são colocados/tirados do estado de "dormente". Esse método foi denominado REASoN, e os seus resultados servirão como parâmetro empregados na tomada de decisão na rede, uma vez que os dispositivos considerados críticos, ou seja, que degradam muita a confiabilidade e/ou disponibilidade da rede quando alterados, devem ser evitados.

O REASoN é um método hierárquico, no qual primeiramente utiliza cadeia de Markov estendida para o cálculo da confiabilidade individual de cada dispositivo considerando o tempo médio para ativar o dispositivo quando colocado em estado "dormente". E após esse cálculo, é utilizado o método Conjunto-Conexo e Conjunto-Desconexo para o cálculo da confiabilidade e/ou disponibilidade da rede.

No próximo capítulo será apresentada uma análise numérica do método, onde são apresentados resultados de cálculos numéricos de uma topologia de rede em anel. Serão avaliados os métodos padrão e proposto, a fim de promover uma comparação

entre eles.

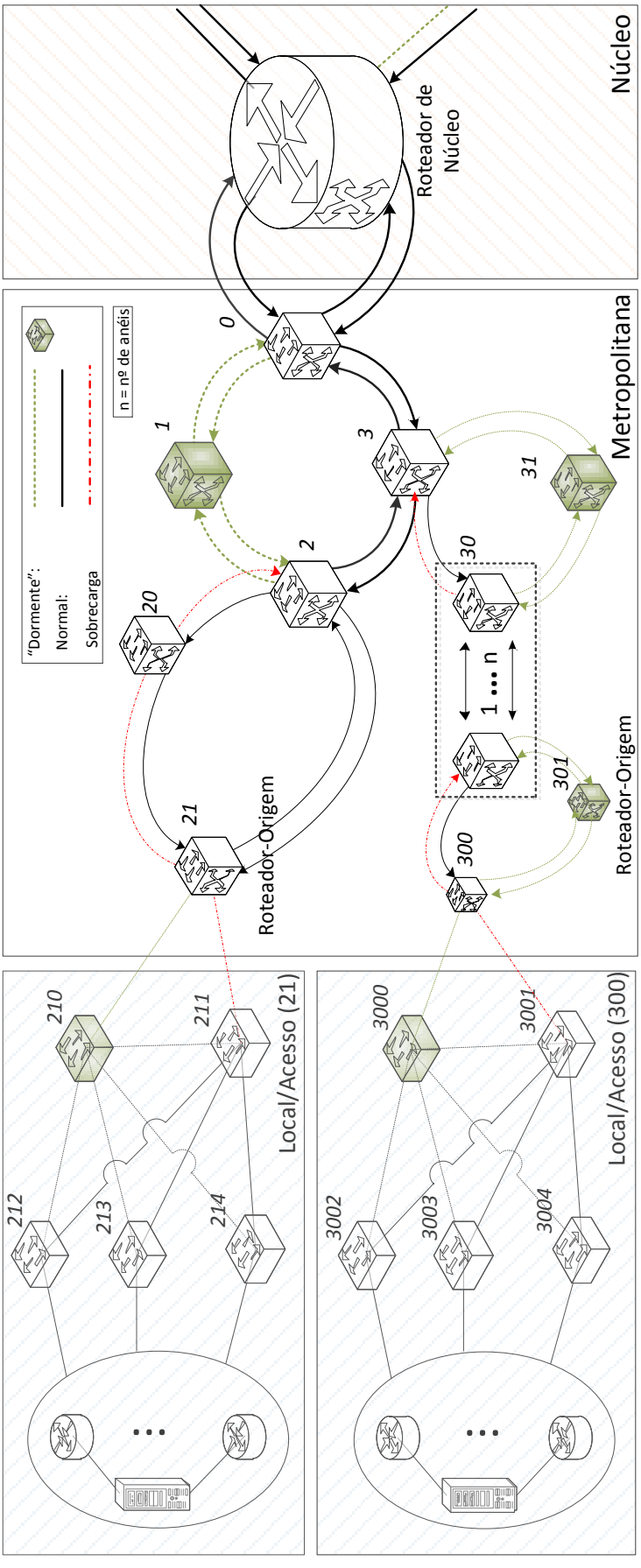
5 ANÁLISE NUMÉRICA DA MODELAGEM PROPOSTA PARA CONFIABILIDADE

A fim de avaliar o método proposto, esse capítulo apresenta a implementação de um cálculo numérico iterativo, seguindo as especificações apresentadas no capítulo 4. Os resultados obtidos pelo REASoN são comparados com os resultados utilizando o método padrão de calcular confiabilidade, conforme descritos no Capítulo 3. Todos os cenários foram baseados na topologia de uma rede metropolitana, representada na Figura 12, que está diretamente relacionada ao contexto de NSPs (*Network Service Providers*). A topologia da rede representada é aumentada adicionando mais anéis na rede, onde cada anel é composto por 3 roteadores. O número de anéis na rede pode variar de 1 até n.

A Figura 12 apresenta uma topologia de rede em anel. Vale a pena ressaltar aqui que o método proposto não depende da topologia de rede, e essa topologia é utilizada, meramente como prova de conceito, de forma a mostrar eficácia do método, e não a sua abrangência. A decisão de escolher uma topologia em anel foi tomada por se tratar de uma topologia altamente utilizada no contexto de redes metropolitanas e ser considerada de alta confiabilidade (CISCO, 2007). O critério de decisão de colocar os dispositivos em estado "dormente" (roteadores escuros) ou os manter acordados (roteadores claros) não é o foco deste capítulo.

A confiabilidade dos roteadores e da rede é calculada em todos os cenários. Os roteadores possuem conexões redundantes para entrada e saída de dados. A confiabilidade da rede é de dois-pontos-terminais, sendo a "origem" o roteador 21 e o

Figura 12: Topologia de rede local, metropolitana e de núcleo.



"destino" o roteador-final da rede de núcleo, como mostrado na Figura 12. O intervalo de tempo Δt utilizado é de 1 segundo, o MTTF variou entre 60 mil, 80 mil, e 100 mil horas (valores que são comumente encontrados nas fichas técnicas comerciais de roteadores de alto desempenho como em (CISCO, 2009) e o fator de cobertura "c" igual a 1. As experiências avaliam e comparam três cenários distintos, como descritos a seguir:

Cenário (i) é composto por apenas roteadores em *hot standby*, ou seja, todos os dispositivos estão acordados e nunca entram em estado "dormente".

Cenário (ii) é composto por alguns roteadores em *cold standby*, que representam os roteadores que estão no estado "dormente", e alguns roteadores em *hot standby* a fim de manter a conectividade da rede. O cálculo utilizado nesse cenário é realizado pelo método padrão e não leva em consideração o tempo de ativar um dispositivo.

Cenário (iii) tem a configuração similar ao segundo, contendo alguns dispositivos no estado "dormente" e outros ativos. Porém, esse cenário tem a confiabilidade calculada pelo método proposto, o REASoN, considerando o tempo médio para acordar os dispositivos e estabilizar a rede entre 10 e 25 minutos, valores encontrados nas fichas técnicas comerciais de roteadores de alto desempenho em (CISCO, 2007.).

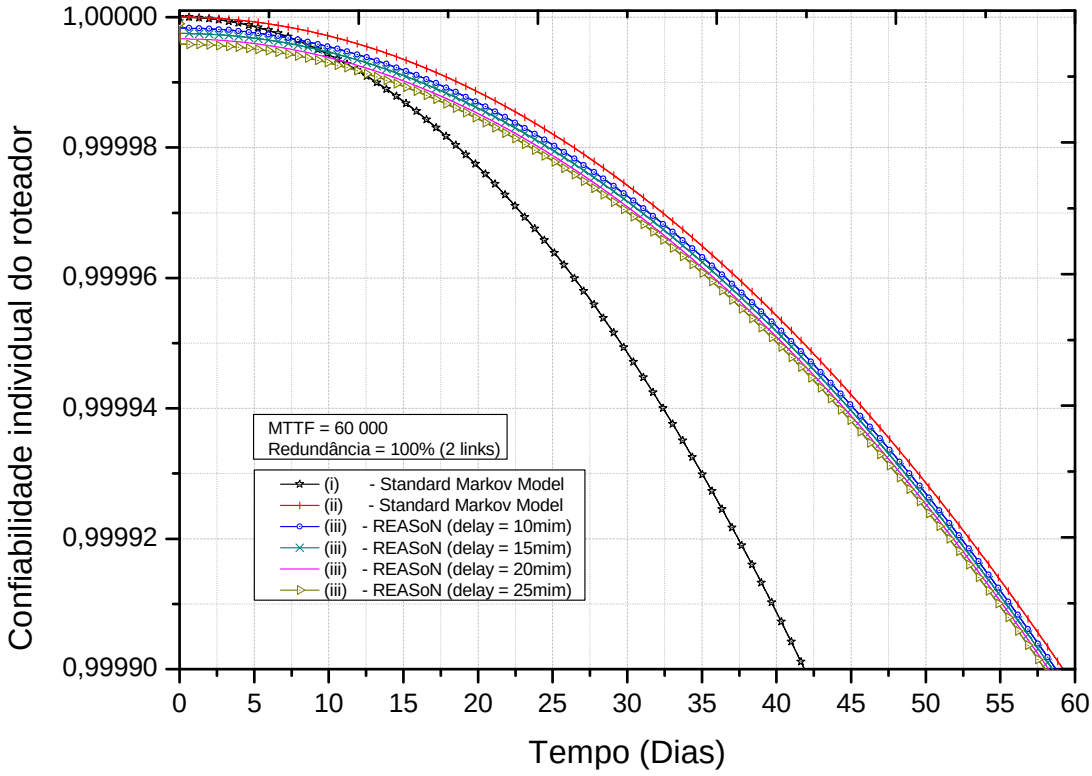
Primeiramente é avaliada a confiabilidade individual de cada roteador utilizando o Modelo de Markov padrão e o proposto pelo REASoN. Depois, utilizando os resultados objetivos pelos Modelos de Markov, é calculada a confiabilidade da rede utilizando o método Conjuntos-Conexos e Conjuntos-Desconexos.

5.1 Confiabilidade individual dos roteadores

A curva da Figura 13 representa a confiabilidade individual do roteador 21 em diferentes configurações de redundância: cenário (i) *hot standby*, cenário (ii) *cold standby* avaliado pelo método padrão de Markov e Conjunto-Conexo e Conjunto-

Desconexo, e cenário (iii) *cold standby* avaliado pelo REASoN. Como mencionado anteriormente, os dois primeiros cenários, (i) e (ii), tiveram a confiabilidade calculada através dos métodos padrão, e o cenário (iii) calculado pelo REASoN.

Figura 13: Confiabilidade individual do roteador 21 com redundância em: cenário (i) *hot standby*; cenário (ii) *cold standby* avaliado pelo método padrão de Markov e Conjunto-Conexo e Conjunto-Desconexo; e cenário (iii) *cold standby* avaliado pelo REASoN.



Pode-se ver na Figura 13 que a confiabilidade calculada no cenário (ii) é maior que a calculada nos outros dois cenários. Esse comportamento deve-se principalmente à configuração de redundância de *cold standby*, na qual alguns componentes permanecem por mais tempo em estado inativo e, logo, não estão sujeitos à falha até que sejam ativados. Ao contrário da configuração *hot standby*, em que todos os componentes já iniciam totalmente ativos e estão sujeitos à falha desde o início, fazendo com que a curva da probabilidade tenha um decaimento mais acentuado. Neste contexto, as chances de um componente em redundância *cold standby* falhar são menores que em redundância *hot standby* conforme descrito no capítulo 3).

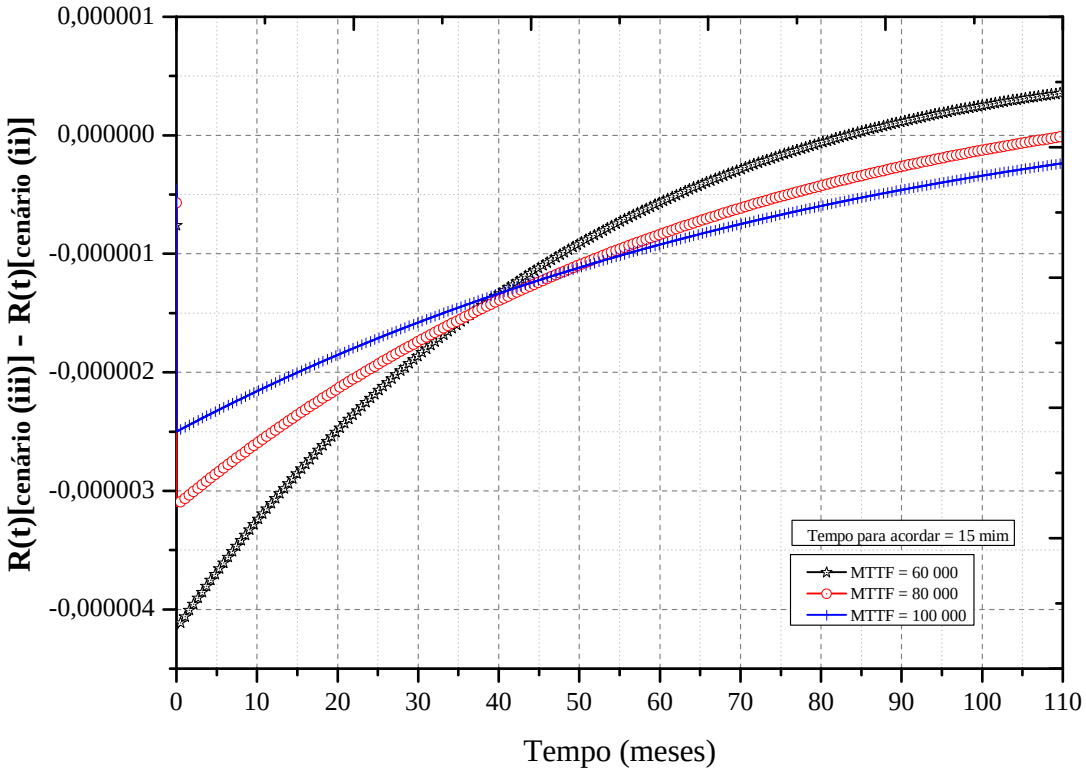
Por outro lado, o tempo de ativar um dispositivo pode impactar diretamente a confiabilidade do mesmo em um determinado intervalo de tempo. Ainda na Figura 13, pode-se visualizar que a curva referente à confiabilidade calculada para o cenário (iii) possui um comportamento distinto das outras. Este comportamento deve-se ao fato que, neste cenário, a confiabilidade foi calculada pelo REASoN, considerada o tempo que um dispositivo demora a ser ativado. Desta forma, esse tempo de atraso é inserido como um "pênalti" na confiabilidade do dispositivo. Esse "pênalti" é visto principalmente durante os primeiros dias. O evento que leva a tal "pênalti" é a ocorrência de uma falha, juntamente com a necessidade do dispositivo em estado "dormente" ser ativado. Porém, nesse intervalo, existe um tempo de indisponibilidade até que o dispositivo em estado "dormente" esteja totalmente ativado e operacional. Desta forma, a Figura 13 mostra que a confiabilidade, quando considerado o tempo para ativar um dispositivo em estado "dormente", é menor que a confiabilidade de todos os dispositivos acordados (cenário (i)), pelo menos durante o intervalo de tempo necessário para ativar o dispositivo em estado "dormente".

Entretanto, o decaimento da confiabilidade, com redundância em *hot standby*, é mais acentuado que em *cold standby* em longo prazo, mesmo quando contabilizando o "pênalti" na confiabilidade, como no cenário (iii). Desta forma, a confiabilidade em *hot standby*, com o passar do tempo, tende a ser menor que em *cold standby*,se torna mais confiável. Além disso, assintticamente os valores obtidos pelo REASoN e os valores obtidos pelos métodos padrão tornam-se iguais. Ou seja, após um longo período, o impacto do "pênalti" diminui e praticamente desaparece. A duração do impacto do "pênalti" depende do tempo que um dispositivo demora a ser ativado. De forma que, um tempo maior para ativar um dispositivo, implica em um maior tempo de duração do "pênalti", e também em maior amplitude do "pênalti". Por exemplo, na Figura 13, quando considerado um tempo para ativar um dispositivo igual a 10 minutos, a confiabilidade do dispositivo em configuração de *cold standby* (do cenário

(iii)) é menor que do dispositivo em *hot standby* em menos de 7 dias; já para 25 minutos de atraso, a confiabilidade no modo *cold standby* é menor em 13 dias. Ou seja, a duração e a amplitude do "pênalti" são diferentes dependendo do atraso para ativar o dispositivo que está em estado "dormente".

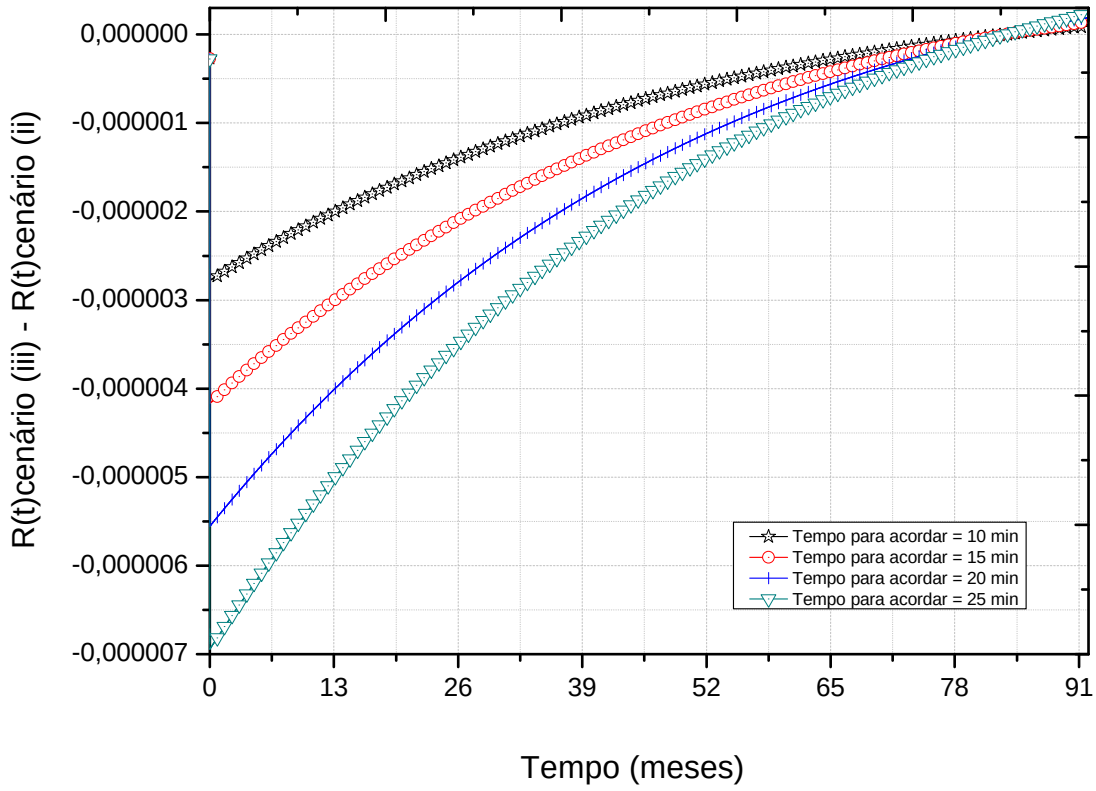
Para mostrar a real diferença entre o cálculo da confiabilidade para *cold standby* realizada pelos métodos padrão e pelo REASoN foram criadas as Figuras 14 e 15. Essas figuras mostram a curva do cálculo da diferença da confiabilidade dos dois casos, mostrando assim o tempo e a amplitude do tempo para ativar um dispositivo, tanto quando se varia o MTTF (entre 60 e 100 mil), como quando se varia o tempo para ativar um dispositivo.

Figura 14: Diferença entre o cálculo da confiabilidade para *cold standby* realizada pelos métodos padrão, cenário (ii) e pelo REASoN, cenário (iii), variando-se o MTTF.



As curvas nas Figuras 14 e 15 mostram a diferença entre o cálculo da confiabilidade do cenário (ii) e (iii), sendo realizada a subtração da confiabilidade do (iii) menos a confiabilidade do (ii). Desta forma, o valor negativo representa que o (iii)

Figura 15: Diferença entre o cálculo da confiabilidade para *cold standby* realizado pelos métodos padrão, cenário (ii) e pelo REASoN, cenário (iii), variando-se o tempo de ativação de um dispositivo.



é menor que o (ii). Como resultado, se pode ver que os valores do tempo para ativar um dispositivo e o MTTF impactam diretamente na amplitude e tempo do "pênalti", como descrito anteriormente. Na Figura 15, onde é apresentado que a amplitude do "pênalti" pode ser maior que $\approx -7 \times 10^{-6}$, uma mudança na 6ª casa decimal, da confiabilidade nos primeiros 78 meses de operação do dispositivo.

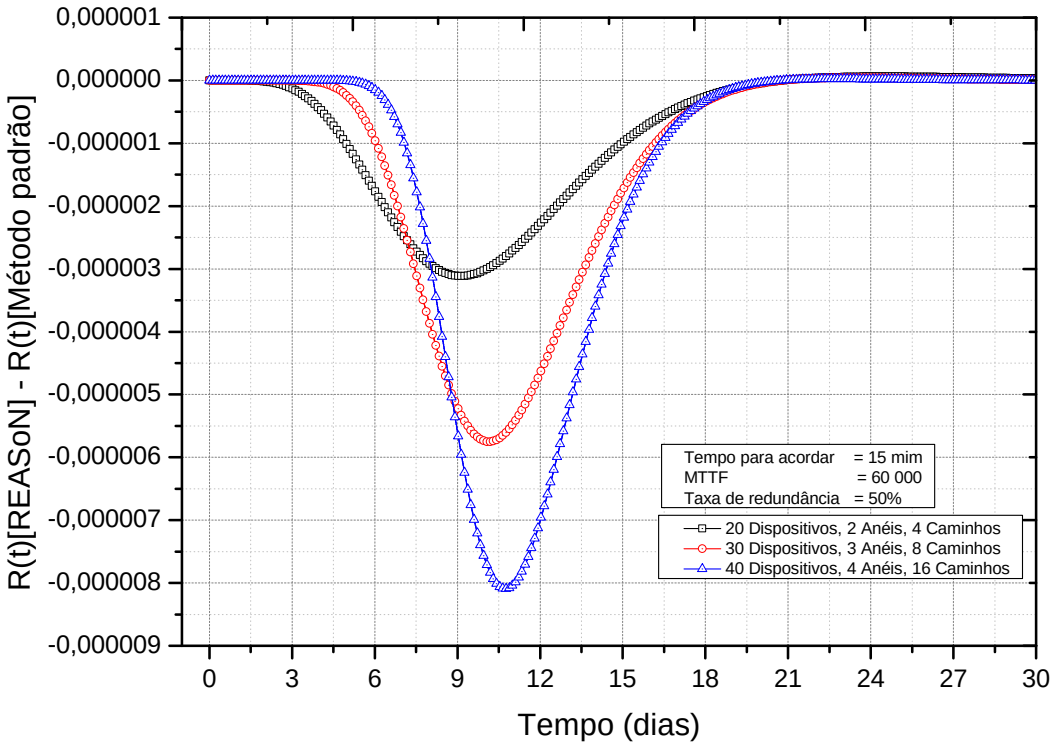
Além disso, essas figuras mostram outro comportamento, que é o aumento temporário da confiabilidade do cenário (iii) após 85 meses. Então, pode-se perceber que, na Figura 14 após 40 meses, e, na Figura 15 após 85 meses, a diferença torna-se positiva, de forma a representar que o cenário (iii) tornou-se mais confiável. Esse comportamento ocorre porque considerar no cálculo a ocorrência de atraso na ativação do dispositivo faz com que esse dispositivo seja considerado desativado por um período maior, quando comparado ao cenário em que esse tempo não é levado em consideração no cálculo. Ou seja, o tempo de ativar um dispositivo representa que esse dispositivo ainda não está ativo e, logo, não está sujeito à falha. Contudo, o tempo que o dispositivo está desativado e não sujeito à falha no cenário (iii) é considerado maior que no cenário (ii), impactando em um decaimento menos acentuado da confiabilidade no cenário (iii). A importância de analisar a duração do "pênalti", é para saber a relevância do cálculo, pois com uma duração longa do impacto, é importante analisar tal impacto.

5.2 Confiabilidade da rede

A sessão anterior apresentou como o atraso para ativar um dispositivo impacta na confiabilidade de um roteador com o passar do tempo. Essa sessão apresenta o comportamento de roteadores quando combinados e conectados de forma a compor uma rede, mostrando como o "pênalti" para ativar um dispositivo pode impactar na confiabilidade da rede. De modo geral, este capítulo de análise numérica tenta responder a questão de como a confiabilidade avaliada pelos métodos padrão difere dos resultados obtidos pelo REASoN. Para realizar tal comparação, os cenários deste

trabalho avaliam a confiabilidade da rede, variando-se o tamanho da topologia. A topologia é expandida considerando-se de 5 a 50 dispositivos ligados em rede. A topologia básica das redes é em anel, em que o aumento do número de dispositivos aumenta a quantidade de anéis de rede, sendo cada anel composto por 3 roteadores. Também, de forma a manter coerência e servir de base para comparações, é mantido o mesmo nível de redundância de 50% dos caminhos, para todas as configurações da topologia (Figura 12). Por fim, no caso dos cenários (ii) e (iii), em que alguns dispositivos são mantidos em *cold standby*, a porcentagem de dispositivos em estado "dormente' é mantida sempre em 30%.

Figura 16: Diferença entre o cálculo da confiabilidade para *cold standby* realizada pelos métodos padrão, cenário (ii) e pelo REASoN, cenário (iii).



As curvas da Figura 16 representam a diferença entre a confiabilidade de dois-pontos-terminais, sendo a origem o roteador 21 e o destino o roteador de núcleo, para redundância em *cold standby* realizada pelos métodos padrão, cenário (ii) e pelo REASoN, cenário (iii). Esta figura expõe que a confiabilidade da rede no cenário (ii) é maior que no cenário (iii) até o período de 43 meses. É mostrado, também, que

existe uma mudança do comportamento da confiabilidade calculada apenas para um roteador individual, como exemplificado na Figura 13, e a confiabilidade para a rede toda. Esse comportamento é diferente, pois quanto maior o número de dispositivos ligados na rede, maior é o impacto do tempo de ativar os dispositivos na rede, dado que mais dispositivos foram colocados no estado "dormente". Isso mostra que o "pênalti" na confiabilidade de colocar um dispositivo no estado "dormente" é maior na rede, quando vários dispositivos estão combinados conectados entre si. Essa figura evidencia, também, que ambas a amplitude e a duração do "pênalti" na confiabilidade, são afetadas pelo número de dispositivos na rede. Isso mostra que quanto maior o número de dispositivos, maior é a amplitude e menor é a duração do "pênalti". Para o caso de 40 dispositivos, a amplitude do "pênalti" é $\approx -8 \times 10^{-6}$ a duração de ≈ 20 dias. Já para o caso de 20 dispositivos, a amplitude do "pênalti" é de $\approx 3.2 \times 10^{-6}$ e a duração de ≈ 20 dias.

Os resultados mostram que para uma rede com menos de uma dúzia de dispositivos, na qual o modo *cold standby* é dinamicamente ativado e desativado, considerando no cálculo o tempo de ativar um dispositivo, é atingido um impacto considerável na sexta casa decimal. Esse "pênalti" acontece durante os primeiros dias de operação. Tal fato pode ser de grande importância na hora de tomar a decisão sobre qual dispositivo deve ser colocado ou não em *cold standby*.

5.3 Considerações finais do capítulo

As comparações contidas nesse capítulo apresentam uma avaliação da confiabilidade calculada pelos métodos padrão e pelo método proposto por esse trabalho, o REASoN, o qual considera no cálculo o tempo de ativar um dispositivo. No caso da confiabilidade individual de um roteador dotado de duas conexões redundantes e MTTF de 60 mil horas, a confiabilidade é afetada na 6ª casa decimal para os primeiros 78 meses, ao incluir no cálculo o tempo de ativar o dispositivo. Já na análise

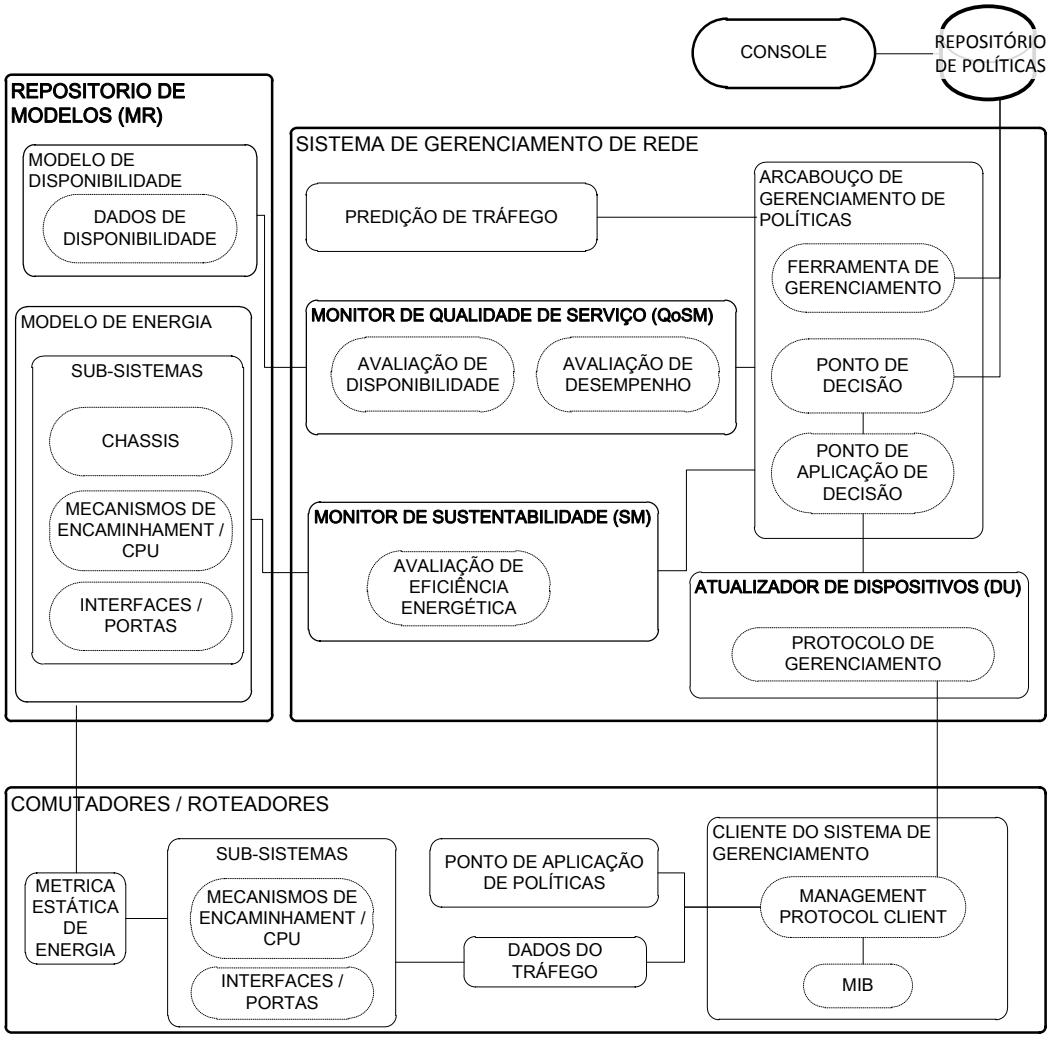
em relação a uma rede composta por dois anéis, tendo 4 caminhos redundantes e MTTF de 60 mil horas, a confiabilidade é afetada, também, na sexta casa decimal, porém a duração do pênalti é de apenas 20 dias e não 78 meses. Pode-se ver que a confiabilidade calculada para a rede difere da calculada individualmente para cada roteador, tanto na amplitude quanto na duração. Como contribuição deste trabalho é mostrado que o REASoN possui um resultado com maior precisão, dado que mede o impacto dos dispositivos que são colocados no estado “dormente”. A análise de disponibilidade é apresentada no capítulo 7.

6 SISTEMA DE GERENCIAMENTO DE REDE ORIENTADO À SUSTENTABILIDADE - SUSTNMS

Como descrito anteriormente no capítulo 2, pode-se economizar energia em redes de computadores através de reengenharia (*hardware* eficiente energeticamente), adaptação dinâmica (gerenciando a rede adaptando desempenho e lógica ociosa) e dormência de forma inteligente (modo consumo energético reduzido) (BOLLA et al., 2011). As duas últimas técnicas podem ser aplicadas por meio de um sistema de gerenciamento de redes, por um algoritmo de roteamento verde ou por protocolos quaisquer de controle. No entanto, existe uma vantagem em utilizar um sistema de gerenciamento de redes. Essa vantagem é que se tem uma visão geral (alto nível) da rede, o que possibilita ao sistema analisar a rede de modo mais abrangente, em vez de tomar apenas decisões locais, como, na maioria dos casos, ocorre nos algoritmos de roteamento por exemplo.

Desta forma, esse capítulo descreve um sistema de gerenciamento de rede orientado a políticas de sustentabilidade chamado de SustNMS (*Sustainability Oriented Network Management System*) (COSTA et al., 2012). As políticas de sustentabilidade têm como objetivo economizar energia na rede, e podem descrever limiares que ao serem ultrapassados desencadeiam ações para tirar ou colocar dispositivos em estado de "dormência", promovendo assim uma maior eficiência energética da rede considerada como um todo. A arquitetura do SustNMS é ilustrada na Figura 17. A arquitetura é baseada na arquitetura padrão de sistema de

Figura 17: Arquitetura do sistema de gerenciamento de redes orientado à políticas de sustentabilidade - SustNMS



Essa figura é baseada no trabalho de Costa et. al. (COSTA et al., 2012)

gerenciamento de rede orientado a políticas definido pelo IETF (descrito no capítulo 2). O SustNMS estende a arquitetura do IETF adicionando quatro novos módulos: repositório de modelos (MR - *Model Repository*), monitor de qualidade de serviço (QoSM - *Quality of Service Monitor*), monitor de sustentabilidade (SM - *Sustainability Monitor*), e atualizador de dispositivos (DU - *Device Updater*), conforme descrição a seguir:

- **MR (*Model Repository* ou Repositório de Modelos):** é composto por dois sub-módulos:
  - **Modelo de disponibilidade:** contêm informações sobre taxa de falha e reparo dos dispositivos da rede, como MTTF (*Mean Time to Failure*) e MTTR (*Mean Time to Repair*).
  - **O Modelo de Energia:** define um modelo para determinar o consumo de energia de um dispositivo, tomando por base os parâmetros de perfil de consumo de energia. O perfil de consumo de energia é determinado por uma função f (tráfego da rede), que estima como o consumo de energia varia de acordo com o tráfego da rede. Como exemplo de uma função f(trafego da rede) para um roteador com consumo de energia escalando de forma linear, baseado em (BOLLA et al., 2011), pode-se citar:

$$P(estado, f(trafego\ da\ rede)) = \begin{cases} P_{standby} & \text{se } estado = standby \\ f(trafego\ da\ rede) & \text{senão} \end{cases}$$

$$f(trafego\ da\ rede) = P_{ocioso} + \frac{trafego\ da\ rede}{capacidade\ maxima}(P_{total} - P_{ocioso}). \quad (6.1)$$

O estado ocioso representa o período em que o roteador está ligado, mas não está sendo utilizado, ou seja, 0% de carga. Porém, devido a existência de pacotes de controle, pode-se definir que um roteador ocioso é aquele com uma carga menor que um limiar pré-determinado, por exemplo, com

uma carga inferior a 0.1Mbps, sendo essa carga apenas de mensagens de controle.

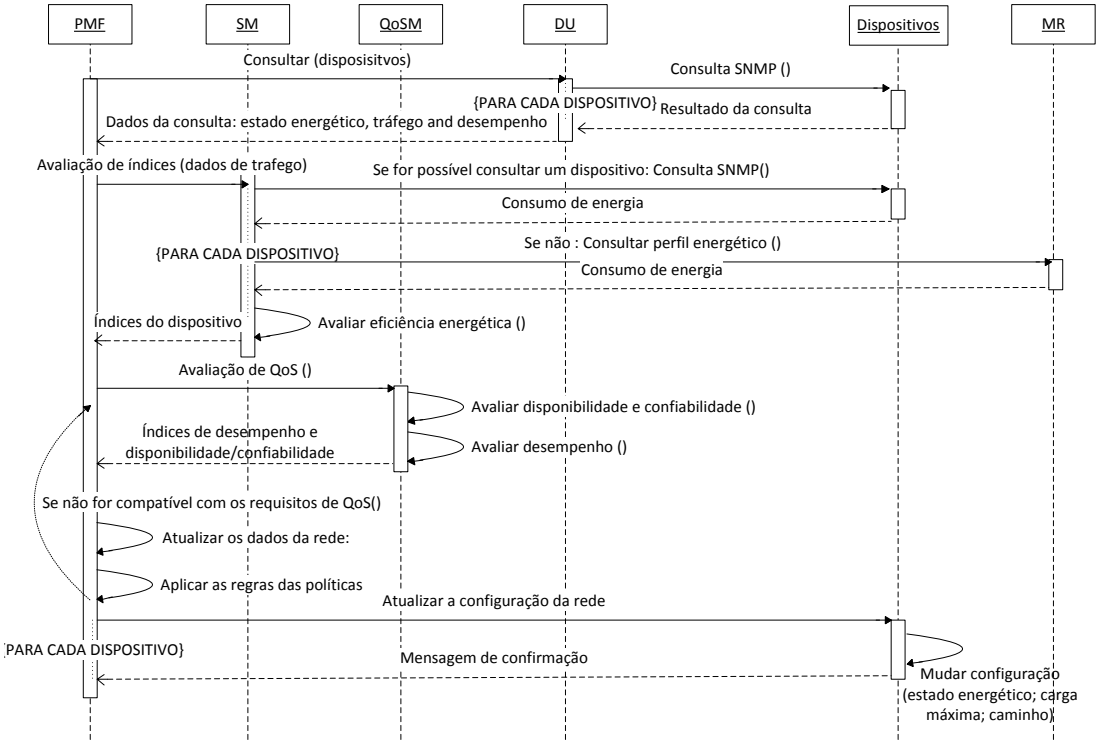
- **QoSM (*Quality of Service Monitor* ou Monitor de qualidade de serviço):** que inclui dois módulos para avaliar a rede:
  - **Avaliador de Disponibilidade:** avalia dinamicamente a disponibilidade da rede toda vez que o estado da rede muda, ou seja, quando um dispositivo muda seu estado energético (ativo para dormência ou vice-versa), ou na ocorrência de falhas. Para avaliar a disponibilidade da rede orientada à sustentabilidade é utilizado o método proposto no capítulo 4.
  - **Avaliador de Desempenho:** analisa as informações da rede e calcula a quantidade de pacotes perdidos, atrasos, e *jitter*. Os indicadores a serem utilizados dependem da abordagem de gerenciamento da rede e dos requisitos definidos nos SLAs (*Service level Agreement*).
- **SM (*Sustainability Monitor* ou Monitor de Sustentabilidade):**): é composto pelo módulo **Avaliador de Eficiência Energética**, o qual é responsável por avaliar a eficiência energética da rede.

  O **Avaliador de Eficiência Energética** mede o consumo de energia de cada dispositivo na rede. O protocolo de gerenciamento de rede pode, então, ser utilizado para coletar informações sobre o consumo de energia instantâneo do dispositivo. Se essa funcionalidade não estiver disponível no dispositivo, esse módulo pode obter junto ao MR o perfil de consumo de energia que se encaixa com a descrição do dispositivo. Se um perfil não pode ser localizado no MR, um perfil pessimista é utilizado, descrevendo-se limites pré-definidos. Assim sendo,para avaliar a taxa de eficiência energética da rede em Watts/bps são usadas informações referentes ao total de energia consumida, juntamente com o tráfego da rede.

- **DU (*Device Updater* ou Atualizador de Dispositivos):**, que coleta informações da rede e configura o modo de consumo de energia dos roteadores. O DU acessa as informações e aplica configurações na rede através de um protocolo de gerenciamento de rede SNMP (*Simple Network Management Protocol*), ou através de algum outro protocolo, como NETCONF (*Network Configuration Protocol*), ou ainda por uma interface de linha de comando, CLI (*Command Line Interface*).

A seguir, na Figura 18, é apresentado o diagrama de sequencia que descreve o comportamento do SustNMS e as interações de cada módulo.

Figura 18: Diagrama de sequência da execução do SustNMS



Essa figura é baseada no trabalho de Costa et. al. (COSTA et al., 2012)

Na Figura 18, o sistema começa coletando informações da rede em uma frequência definida pelo administrador da rede. Para cada dispositivo da rede, uma consulta SNMP baseado na MIB-II (RFC1213) é enviada, coletando informações de: tráfego da

rede, estado e desempenho. A requisição é feita pelo arcabouço de gerenciamento de políticas (PMF - *Policy Management Framework*) descrito no capítulo 2. Em seguida, os dados são armazenados e representados em um grafo da topologia da rede. O grafo é enviado para os módulos de avaliação (SM e QoSM), e para o módulo PDP (*Policy Decision Point* ou Ponto de Decisão) que está dentro do PMF.

Depois de coletar informações de todos os dispositivos que estão sendo gerenciado pelo sistema, o PMF manda requisições para o SM avaliar a eficiência energética da rede. No SM, para os dispositivos que suportam a MIB de monitoração de energia (CLAISE; PARELLO, 2013), o consumo energético instantâneo é obtido por meio dos dados no grafo. Já para os dispositivos que não suportam a MIB, o consumo energético é avaliado a partir de um perfil específico de consumo energético associado ao tipo de dispositivo considerado e que está armazenado no MR. Desta forma, a métrica de consumo de energia (ECR - *Energy Consumption Rating*) é avaliada para cada dispositivo baseado no tráfego da rede e no consumo de energia para cada nó do grafo. Então o grafo é atualizado com os índices calculados.

Imediatamente depois que a eficiência energética da rede é avaliada, o PMF faz uma requisição para o QoSM avaliar a disponibilidade e o desempenho da rede. Para avaliar a disponibilidade, o MTTF e o MTTR de cada dispositivo é obtido a partir do MR. Esses dados, juntamente com a informação do estado de cada dispositivo, são utilizados para calcular a disponibilidade. A disponibilidade da rede é armazenada em uma variável específica do grafo. Ao mesmo tempo, o desempenho é avaliado utilizando as informações coletadas pelo protocolo de gerenciamento de rede (por exemplo, SNMP), calculando estatísticas de perda de pacotes. Atrasos e variações de atraso (*jitter*) são obtidos por meio do protocolo ICMP (*Internet Control Message Protocol*). O grafo é, então, atualizando com as informações dos índices calculados.

O PMF gerencia o estado da rede, sendo que o PDP define as ações que devem ser aplicadas na rede de acordo com políticas de sustentabilidade pré-definidas pelo

administrador da rede. O PMF verifica se as ações podem se aplicadas, fazendo-se uma projeção de como a rede ficaria com as ações a serem efetivadas e verificando-se como os requisitos de QoS descritos nas políticas seriam atendidos. A partir daí, pode ser necessário reconfigurar a rede para satisfazer os requisitos de QoS.

O PEP aplica a decisão do sistema configurando a rede e utilizando o módulo DU para enviar mensagens aos dispositivos. Desta forma, as requisições são traduzidas para comandos específicos que colocam/tiram os dispositivos do estado de "dormência".

Todo o processo é repetido toda vez que o SustNMS identificar uma mudança no status da rede. Desta forma, quanto menor for a frequência de medições da rede, maior será a precisão das informações que estarão sendo analisadas. Com um intervalo muito grande entre as medições, o sistema pode não identificar mudanças na rede, e então, não aplicar a melhor decisão. Porém, para definir uma taxa adequada, é necessário levar em consideração, que aumentar a frequência de medições pode ter como resultado a sobrecarga do plano de controle da rede. Além disso, mais mensagens na rede implicam em um maior consumo de energia.

6.1 Considerações finais do capítulo

Esse capítulo descreveu o sistema de gerenciamento de redes orientado a sustentabilidade, chamado SustNMS. O propósito desse sistema é gerenciar a rede de forma a torna-la energeticamente eficiente. As ações são baseadas em decisões de alto nível (do nível de negócio) que descrevem como coordenar a relação de compromisso entre economizar energia e os potenciais impactos nas operações da rede. Foi apresentada a arquitetura do sistema que permite gerenciar, e avaliar a eficiência energética e o desempenho da rede em tempo real. Desta forma foram descrito os módulos e as relações entre eles, mostrando como o sistema opera. A descrição desse

sistema servirá como base para a análise do REASoN implementado em um sistema de gerenciamento de rede, no capítulo 7.

7 ESTUDO DE CASO: SISTEMA DE GERENCIAMENTO ORIENTADO À SUSTENTABILIDADE (SUSTNMS) UTILIZANDO O REASON

Esse capítulo descreve experimentos relacionados ao Sistema de Gerenciamento de Rede orientado à Sustentabilidade (SustNMS - *Sustainability Oriented Network Management System*), com foco em eficiência energética, que considera na tomada de decisão, a confiabilidade e a disponibilidade. O objetivo dos experimentos é mostrar a relação de compromisso entre economizar energia e manter alta confiabilidade e disponibilidade na rede, pois, como visto no Capítulo 5, colocar dispositivos no estado de "dormente" diminui a confiabilidade. Desta forma, foram executados experimentos com políticas que priorizam requisitos diferentes, como economia de energia, ou confiabilidade e disponibilidade.

Esse trabalho estende o ambiente de teste apresentado por Januário et. al. (JANUARIO et al., 2013), que implementa o sistema de gerenciamento de redes, o SustNM proposto por Costa et. al. (COSTA et al., 2012). O Capítulo 6 descreve o funcionamento e arquitetura do SustNMS e mostra que o sistema contempla um módulo para a avaliação da disponibilidade da rede. Porém, essa avaliação é calculada utilizando os métodos padrão. Este trabalho substitui a avaliação da disponibilidade e inclui a avaliação da confiabilidade calcula pelo REASoN na implementação do SustNMS realizada por Januário et. al. (JANUARIO et al., 2013). Logo, os experimentos tem por objetivo apresentar o SustNMS tomando decisões baseadas na

confiabilidade e disponibilidade calculadas pelo REASoN.

7.1 Ambiente de testes

O ambiente de testes avalia o impacto de ações de eficiência energética na disponibilidade e confiabilidade de redes sustentáveis é baseado em softwares de emulação de redes. Essa abordagem de emulação é baseada em roteadores definidos por software, utilizando o sistema operacional Linux. A utilização de roteadores baseados em Linux está aumentando (ANTONAKOPOULOS; FORTUNE; ZHANG, 2010) e,possibilita criar um ambiente de testes aplicável a um grande conjunto de tipos de dispositivos, ou seja, uma rede heterogenia. Para emular diferentes tipos de roteadores, a rede foi construída com máquinas virtuais, utilizando VMWare vSphere (ESXi 5.0) para gerenciar as máquinas virtuais. A conexão entre as máquinas virtuais é realizada pelo vSwitch (*Layer 2 forwarding, VLAN tagging*). As máquinas virtuais que emulam os clientes e servidores rodam em cima do sistema operacional Linux de distribuição Ubuntu 11.04.

Os roteadores trabalham com MPLS (*Multi Protocol Label Switching*), mais especificamente o MPLS-Linux (DUMITRASCU.; POPA., 2013), que é uma abordagem de código aberto que suporta o tunelamento MPLS, caminhos de *backup*, pilha de rótulos MPLS, pesquisa recursiva e o protocolo de reserva de recursos DiffServ (*Differentiated Services*). O MPLS é adequado para controlar e regular o tráfego da rede através de caminhos rotulados LSP (*Label Switched Paths*). Os LSPs tem algumas propriedades de QoS, suportando níveis de prioridade e precedência (CHABAREK; BARFORD, 2011). O MPLS implementa a descoberta de caminhos na rede utilizando o protocolo OSPF (*Open Shortest Path First*) (SHUAIB; SALLABI, 2005). O sistema operacional utilizado no roteador emulado, diferentemente das máquinas dos clientes e servidores, é um Linux de distribuição Debian (versão Lenny, *Kernel* Linux 2.6.27.24).

O estado “dormente” dos roteadores é emulado por meio da desabilitação das interfaces de rede (NIC - *Network Interface Card*). Desta forma, quando as interfaces são desabilitadas, o SustNMS calcula o consumo de energia utilizando o perfil para o estado “dormente”. As interfaces relacionadas ao plano de controle nunca são desabilitadas para manter a presença do dispositivo na rede e aceitar comandos para acordá-los quando necessário. Além disso, os roteadores têm os caminhos de *backup* configurados previamente, uma vez que, isso promove re-roteamento mais rápido e com menor sobrecarga no plano de controle (DONGMEI; GUANGZHI, 2008).

7.2 Implementação do SustNMS

Este trabalho estende a implementação do SustNMS do trabalho de Januário et. al. (JANUARIO et al., 2013) por meio da inclusão da implementação do REASoN. Todos os módulos do SustNMS foram implementados utilizando a linguagem de programação *Python* em Januário et. al. (JANUARIO et al., 2013). Mais especificamente, o módulo DU (Atualizador de Dispositivos) contempla a biblioteca de SNMP, pySNM (ETINGOF, 2013), que permite utilizar comandos SNMP, versão 1.1, para coletar informações da rede. O SustNMS infere informações sobre os tuneis MPLS a partir de informações de carga da rede coletadas utilizando SNMP, que acessa variáveis da MIB padrão (RFC1213), por exemplo *InOctets* e *OutOctets*.

O SustNMS aplica as configurações dos tuneis MPLS, utilizando CLI transmitidos via conexões SSH. Dessa forma é definido, nos roteadores de ingresso o rótulo que cada fluxo irá conter e a ativação dos caminhos de *backup* caso necessário. O MR (*Model Repository*) armazena os modelos de disponibilidade e perfis de energia no banco de dados MySQL. Por fim, a máquina virtual que executa o SustNMS roda sobre o sistema operacional Linux de distribuição Ubuntu 11.04.

Como dito anteriormente, o SustNMS é um sistema de gerenciamento de rede

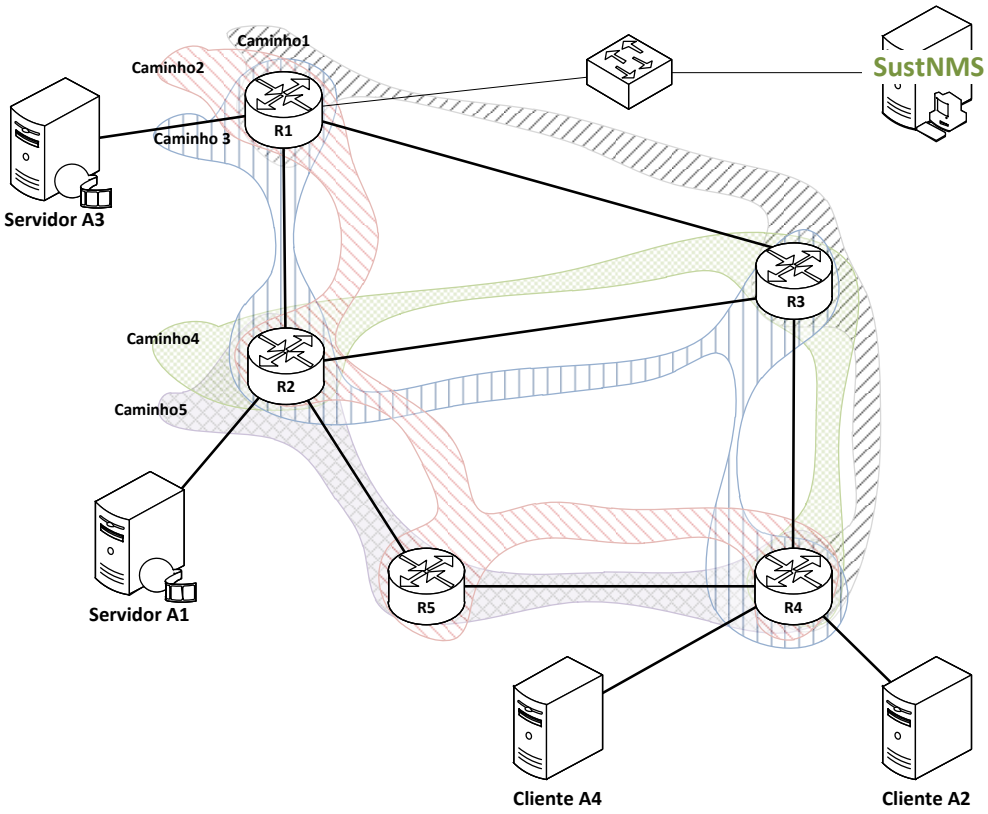
baseado em políticas. Na implementação do SustNMS foi utilizado um arcabouço de gerenciamento de política chamado Ponder2 (TWIDLE et al., 2009). A escolha desse arcabouço está atrelada ao fato de ser um arcabouço de código aberto e de fácil acesso, o que permite fazer as modificações necessárias para adapta-lo no contexto no SustNMS. Além disso, o Ponder2 é um arcabouço que permite definir políticas de rede no formato evento-condição-ação (if -> then -> else), permitindo a criação de políticas de QoS e de eficiência energética. Mais informações sobre o arcabouço Ponder2 podem ser encontradas no Apêndice A.

7.3 Topologia do Ambiente de Teste

O ambiente de testes é composto por cinco roteadores que conectam quatro pontos finais (vide Figura 19). Dois pontos finais são servidores de vídeo, e os outros dois são clientes que acessam os fluxos (*streams*) de vídeo. Nessa topologia, quando todos os dispositivos estão ligados, existe redundância de caminhos a fim de atender requisitos de QoS referentes à disponibilidade e confiabilidade. Além da rede dados existe uma rede de controle paralela. A topologia do ambiente de teste utilizado no experimento emulado é simples, porém representa um cenário de distribuição de conteúdo, que permite fazer análise e projeções para uma topologia maior.

Pode-se visualizar na Figura 19, que existem caminhos redundantes para os fluxos que advém do servidor A3, por exemplo, Caminho1: R1 -> R3 -> R4, Caminho2: R1 -> R2 -> R5 -> R4, e Caminho3: R1 -> R2 -> R3 -> R4.E, também, existem diversos fluxos que advém do servidor A1, como Caminho4: R2 -> R3 -> R4, e Caminho5: R2 -> R5 -> R4.

Figura 19: Topologia utilizada para todos os experimentos



Essa figura é baseada no trabalho de Januário et. al (JANUARIO et al., 2013)

7.4 Dados para calculo de confiabilidade e disponibilidade

Para todos os experimentos a serem realizados, os parâmetros utilizados para o cálculo da confiabilidade e disponibilidade são: MTTF igual a 60mil horas (valor utilizado na análise numérica apresentada anteriormente), tempo médio para acordar os dispositivos (α) igual a 15 minutos (representando o tempo médio para inicializar um dispositivo e o tempo para a rede estabilizar, tornando-se totalmente operacional), e fator de cobertura "c" igual a 100% ou 97%.

A confiabilidade foi calculada para o intervalo de 24hs, por representar o ciclo de um dia inteiro de perfil de tráfego na rede. Para o cálculo da disponibilidade, também é utilizado o MTTF igual a 60mil horas, o α (tempo para acordar um dispositivo) igual 15 minutos (valor considerado em todos os experimentos), e o MTTR igual a 4h30 ou 30min (valores comumente encontrados na especificação de roteadores (CISCO, 2009)). Diferentemente da confiabilidade, a disponibilidade é calculada até atingir o estado de convergência (*stead state*), e não no intervalo de 24h.

7.5 Perfil de energia dos roteadores

O consumo de energia de cada roteador é calculado tomando-se por base perfis de energia pré-definidos. Para todos os experimentos são definidos apenas dois tipos de perfis de energia dos roteadores. Um perfil representa o caso de um dispositivo, cujo comportamento visa a operação com eficiência energética, no qual o consumo de energia escala linearmente com a carga da rede, como em (ANTONAKOPOULOS; FORTUNE; ZHANG, 2010). O outro tipo de perfil de energia representa os equipamentos legados da rede, que não apresentam uma variação significativa no consumo de energia quando a carga da rede aumenta, mantendo um consumo de energia quase constante, como em (BOLLA et al., 2011).

Todas as informações utilizadas para construir o perfil de energia dos roteadores são baseadas em informações de roteadores reais. Porém, para simplificar a análise do experimento, foi pré-definida a capacidade máxima de todos os roteadores, como sendo:

- R1 = 32Mbps;
- R2 = 32Mbps;
- R3 = 40Mbps;
- R4 = 40Mbps; e
- R5 = 32Mbps.

Esses valores são baixos para facilitar a ocorrência de sobrecarga dos roteadores durante a emulação das diversas situações de trafego e falha na rede, mas não representa um limitante do experimento. Além disso, o perfil de energia também descreve como que se comporta o consumo de energia dos dispositivos que estão em estado “dormente”.

Desta forma, foram criados quatro perfis de energia, baseado nos dois tipos de perfis de energia dos roteadores descritos acima. Os roteadores R1, R2, R4 e R5 tem o consumo de energia escalando linearmente como base em (ANTONAKOPOULOS; FORTUNE; ZHANG, 2010). Já o Roteador R3 tem o consumo de energia constante, com base em (BOLLA et al., 2011). A seguir são apresentadas as equações que definem o consumo de energia em Watts:

- l é a porcentagem de utilização dos roteadores, que varia 1 a 100%;
- l_{max} representa a carga máxima dos roteadores.

$$P_{R1} = \begin{cases} 120 & \text{se em estado “dormente”} \\ 200 + l(500/l_{max}) & \text{senão} \end{cases} \tag{7.1}$$

$$P_{R2} = \begin{cases} 170 & \text{se em estado “dormente”} \\ 200 + l(1000/l_{max}) & \text{senão} \end{cases} \quad (7.2)$$

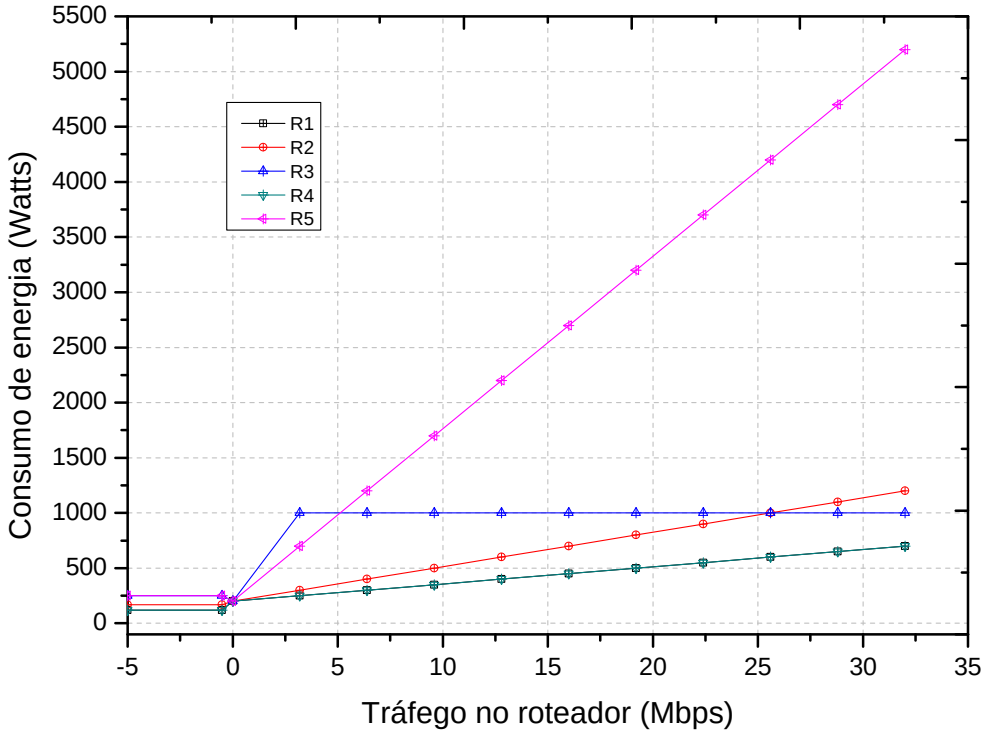
$$P_{R4} = \begin{cases} 120 & \text{se em estado “dormente”} \\ 200 + l(500/l_{max}) & \text{senão} \end{cases} \quad (7.3)$$

$$P_{R3} = \begin{cases} 250 & \text{se em estado “dormente”} \\ 1000 & \text{senão} \end{cases} \quad (7.4)$$

$$P_{R5} = \begin{cases} 220 & \text{se em estado “dormente”} \\ 300 + l(5000/l_{max}) & \text{senão} \end{cases} \quad (7.5)$$

A Figura 20 representa as curvas de consumo de energia versus carga para todos os roteadores empregados nos experimentos.

Figura 20: Consumo de energia dos roteadores de acordo com a carga.



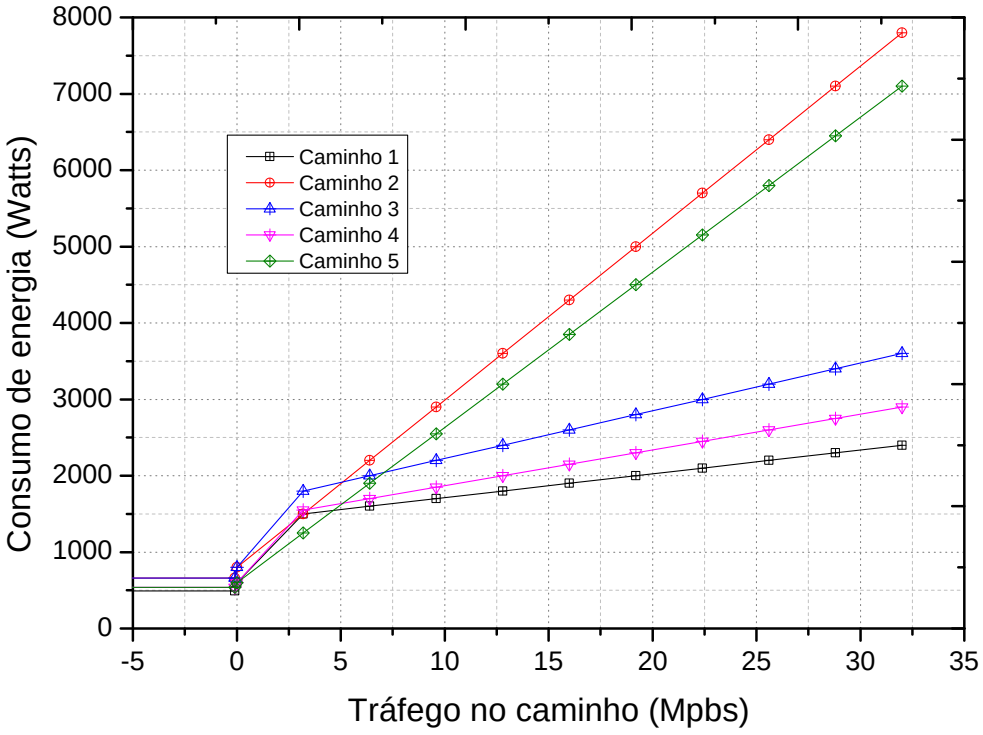
Essa figura é baseada no trabalho de Januário et. al (JANUARIO et al., 2013)

O consumo energético dos roteadores quando tais roteadores estão conectados em rede tem um perfil um pouco diferente, como mostrado na Figura 21. Pode-se ver nesta

figura, que de acordo com a carga na rede, alguns caminhos podem consumir menos energia que outros.

Por exemplo, o caminho 3 consome mais energia que os caminhos 1 e 2 quando o caminho está com menos de 8Mpbs de carga nos fluxos, porém quando a carga aumenta, o consumo inverte e o caminho 3 começa a consumir menos energia. Logo, o sistema analisa todas essas combinações de acordo com a carga dos caminhos da rede e leva em consideração esses dados para escolher o melhor caminho, por exemplo, o mais eficiente energeticamente e com maior disponibilidade.

Figura 21: Consumo de energia em função da carga nos roteadores.



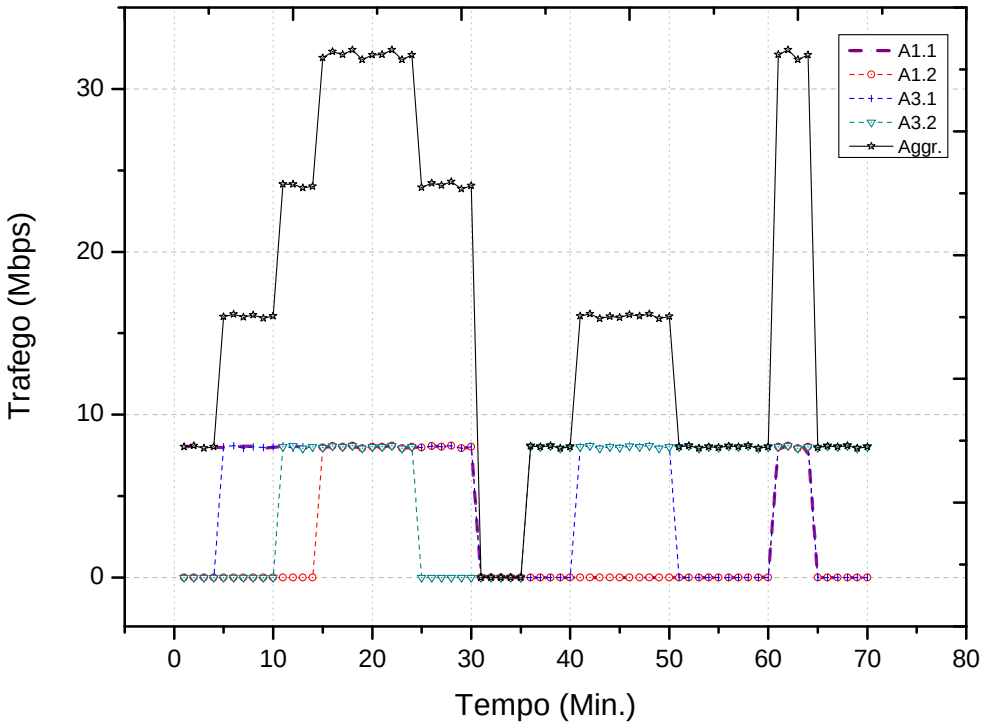
Essa figura é baseada no trabalho de Januário et. al (JANUARIO et al., 2013)

7.6 Perfil de tráfego da rede

Como prova de conceito, foi definido para a execução dos experimentos no ambiente de teste descrito, um perfil de tráfego na rede composto por fluxos de vídeos. O tráfego proveniente do servidor A3 para os Clientes A2 ou A4, pode ultrapassar

30Mbps quando todos os vídeos estão sendo distribuídos ao mesmo tempo. O perfil de tráfego consiste de quatro fluxos de vídeos que são iniciados nos servidores 1 e 3 da Figura 19. Os fluxos vídeo iniciam-se assincronamente, como visto na Figura 22.

Figura 22: Perfil do tráfego gerado no ambiente de teste.



Essa figura é baseada no trabalho de Januário et. al (JANUARIO et al., 2013)

7.7 Experimentos

O objetivo dos experimentos é mostrar os impactos das decisões de economizar energia na confiabilidade e disponibilidade da rede. Desta forma, os experimentos servirão como base para auxiliar o administrador de rede verificar as relações de compromisso entre economizar energia e diminuir a QoS da rede. Então, são descritos três experimentos com cenários diferentes, sendo cada um com uma política diferente que pode priorizar QoS, economia de energia ou nenhum dos dois. O primeiro experimento servirá como base de comparação sobre o consumo de energia, uma vez que nenhuma técnica de economia de energia é aplicada, desta forma este experimento terá consumo máximo de energia. No segundo experimento é aplicada uma técnica

para economizar energia e permite-se degradação na QoS. No terceiro experimento, tem-se economiza energia à custa de restrições de QoS, devendo-se garantir o maior nível possível de confiabilidade e disponibilidade da rede.

O capítulo 2.3 apresentou uma lista de métricas que podem ser utilizadas para avaliar a eficiência energética da rede. Neste trabalho será calculado o ECR (*Energy Consumption Rating*). O ERC é a proporção da energia total consumida pela rede em Watts pela capacidade da rede em Mbps

O primeiro experimento descreve um cenário onde a política apenas define, como premissa, o balanceamento de carga na rede. A política aplicada não requer o uso de nenhuma técnica para economizar energia, e nenhum requisito de QoS é diretamente especificado, ou seja, nada será realizado para diminuir perda de pacotes ou atrasos. Os dispositivos ociosos, isto é, sem nenhum tráfego, permanecem ativos e consumindo energia desnecessariamente. Esse cenário não está longe de um cenário real, pois ter dispositivos ociosos permite realizar engenharia de tráfego na rede e balanceamento de carga, assim como também prover mecanismos para tolerância à falha, uma vez que na ocorrência de falha o tráfego pode ser rapidamente redirecionado para outro caminho.

Os fluxos de vídeo, iniciados nos servidores 1 e 2, podem percorrer dois caminhos diferentes, e a política descreve como os caminhos escolhidos devem se comportar. Primeiramente, é escolhido o caminho mais curto, como definido pelo algoritmo OSPF (*Open Shortest Path First*). Porém, a política também descreve que após certo limite de tráfego nesse caminho, os fluxos sobressalentes devem ser redirecionados para outro caminho. Desta forma, o fluxo 1 será encaminhado pelo caminho 1 e o fluxo 2 pelo caminho 2 e assim por diante, pois os limites definidos nas políticas são exatamente o valor da carga de apenas um fluxo de vídeo.

O Algoritmo 3 apresenta as regras descritas pela política aplicada no primeiro

Algoritmo 3: Política de balanceamento de carga

```
c1 = caminho principal ou mais curto (rótulo MPLS 100)
c2 = caminho alternativo ou redundante (rótulo MPLS 200)
se novo_fluxo == verdade então
    se c1 < 8Mbps então
        marcar novo fluxo com rótulo MPLS 100;
    senão se c2 < 8Mbps então
        marcar novo fluxo com rótulo MPLS 200;
    senão se c1.carga < c2.carga então
        marcar novo fluxo com rótulo MPLS 100;
    senão se c2.carga < c1.carga então
        marcar novo fluxo com rótulo MPLS 200;
    fim
fim
```

experimento (para facilitar a visualização não será utilizada uma linguagem de política específica, mas um pseudocódigo. Porém, a sintaxe utilizada no arcabouço Ponder2 pode ser encontrada no Apêndice A).

O segundo experimento descreve um cenário em que a política contém regras para economizar energia e analisa a confiabilidade e disponibilidade da rede utilizando os métodos padrão (Modelo de Markov e Conjunto Conexo e Desconexo). Esse experimento apresenta economia de energia e os roteadores que estão ociosos são colocados no estado "dormente". Entende-se roteadores ociosos, aqueles que tiverem tráfego menor que 0.1Mbps.

Diferentemente do primeiro experimento, que realiza balanceamento de carga, esse experimento aglomera, o máximo possível, os fluxos em apenas um caminho para que os dispositivos nos outros caminhos possam ficar ociosos, e logo, entrar no estado "dormente". Ao iniciar o primeiro fluxo, o caminho mais curto é escolhido. Desta forma, todos os outros fluxos que iniciarem serão direcionados pelo mesmo caminho, até se atingir um limiar de degradação de QoS e ser necessário redirecionar todos os fluxos para o outro caminho. Devido a esse experimento economizar energia e aceitar degradação da QoS, em nenhum

momento o caminho principal e o alternativo (redundante) irão operar ao mesmo tempo, a rede sempre estará economizando energia.

O Algoritmo 4 apresenta a lógica da política que rege o segundo experimento.

Algoritmo 4: Política de economia de energia

c1 = caminho principal ou mais curto (rótulo MPLS 100)
c2 = caminho alternativo ou redundante (rótulo MPLS 200)
d1 = disponibilidade da rede com os roteadores do ***caminho principal*** *em estado "dormente", e os outros roteadores ativos*
d2 = disponibilidade da rede com os roteadores do ***caminho alternativo*** *em estado "dormente", e os outros roteadores ativos*
conf1 = confiabilidade da rede com os roteadores do ***caminho principal*** *em estado "dormente", e os outros roteadores ativos*
conf2 = confiabilidade da rede com os roteadores do ***caminho alternativo*** *em estado "dormente", e os outros roteadores ativos*

se *carga no c1 < 10Mbps* **então**
  se *carga no c1 < 10Mbps e (d1 - d2) < 0.000001 e (conf1 - conf2) < 0.000001* **então**
    Colocar os roteadores redundantes ∈ c2 em estado de "dormente";
  senão se *carga no c1 < 0.1Mbps e (d1 - d2) > 0.000001 e (conf1 - conf2) < 0.000001* **então**
    Colocar os roteadores redundantes ∈ c1 em estado de "dormente";
  fim
fim

se *novo_fluxo == verdade* **então**
  se *c1 == funcionando e c1 < 10Mbps* **então**
    marcar novo fluxo com rótulo MPLS 100;
  senão se *c1 == não funcionando ou c1 > 25Mbps* **então**
    marcar novo fluxo com rótulo MPLS 200;
  fim
fim

Para conseguir medir o impacto na disponibilidade e na confiabilidade desse experimento, a disponibilidade e a confiabilidade também serão calculadas pelo REASoN, que servirá para comparações com o terceiro experimento.

O terceiro experimento aplica a mesma política do Experimento 2 e atinge economia de energia da mesma maneira, porém a confiabilidade e disponibilidade são avaliadas utilizando o REASoN. Nesse experimento, o sistema de gerenciamento

de rede se esforça para manter a disponibilidade e confiabilidade mais alta o possível e também economiza energia. Nesse caso, os dispositivos ociosos são colocados em estado "dormente" de acordo com o impacto que tal ação pode ocasionar na confiabilidade e disponibilidade da rede.

7.8 Resultado dos experimentos

Todos os experimentos foram executados ao longo de 70 minutos e a topologia da rede variou em três formas. Os resultados da confiabilidade e disponibilidade medidos pelo SustNMS para a ocorrência de cada topologia são descritos a seguir:

1. **Topologia 1**, quando todos os fluxos de vídeo estão simultaneamente sendo distribuídos, todos os dispositivos estão "ativos" e manipulando tráfegos nos intervalos de tempo 15-25 min, e 60-65 min como mostra a Figura 22.

2. **Topologia 2**, quando apenas o fluxo de vídeo proveniente do servidor A3 está sendo distribuído. Nesse caso, o SustNMS decidiu colocar o roteador R3 no estado "dormente", pois o sistema analisou que a melhor opção, de acordo com a política aplicada, era modificar o estado R3.

3. **Topologia 3**, quando apenas o fluxo de vídeo proveniente do servidor A1 está sendo distribuído. Nesse caso o SustNMS decidiu colocar o roteador R5 no estado "dormente".

Os resultados dos experimentos realizados com essas 3 topologias são apresentados nas Tabelas 4, 5, 6 e 7.

A Tabela 4 apresenta a confiabilidade avaliada pelo REASoN e pelo método padrão utilizando o fator de cobertura de 100%, mostrando uma diferença na quinta casa decimal para a Topologia 2 da rede em relação a Topologia 1, onde todos equipamentos estão ligados. Além disso, a tabela mostra uma diferença na sexta casa

Tabela 4: Confiabilidade da rede avaliada pelo REASoN no intervalo de horas t = 0h e t = 24h, para as Topologias 1, 2 e 3.

Confiabilidade da Rede		
	REASoN	Método padrão
Topologia 1 Todos os dispositivos ligados	0.999999	0.999999
Topologia 2 R3 em modo “dormente”	0.999959	0.999999
Topologia 3 R5 em modo “dormente”	0.999993	0.999999

Tabela 5: Confiabilidade da rede avaliada pelo REASoN no intervalo de horas t = 0h e t = 24h, com fator de cobertura “c” = 0.97.

Confiabilidade da Rede		
	REASoN	Método padrão
Topologia 1 Todos os dispositivos ligados	0.999999	0.999999
Topologia 2 R3 em modo “dormente”	0.998511	0.999999
Topologia 3 R5 em modo “dormente”	0.999991	0.999999

decimal para a Topologia 3 em relação à Topologia 1. Por meio, da diferença da confiabilidade nas diferentes topologias da rede, pode-se observar que na rede com a Topologia 1, onde todos os dispositivos estão ligados, a confiabilidade é maior que nas outras topologias, mostrando que colocar um roteador no estado “dormente” tem um impacto significativo na confiabilidade. Esse comportamento é o mesmo observado no Capítulo 5, e a comparação aqui é meramente para validação, de forma a mostrar que a implementação do ambiente de teste está de acordo com o previsto.

A Tabela 4 apresenta que a confiabilidade é diretamente dependente da topologia da rede, isso porque, quando o roteador R5 está no estado “dormente”, a confiabilidade sofre impacto em uma casa decimal em relação à Topologia 2, onde o R3 é colocado no estado “dormente”. Essa diferença mostra que roteadores de regiões diferentes da rede têm impactos diferentes na confiabilidade da rede, quando esses roteadores são colocados em estado “dormente”. Isso se deve ao fato de um roteador poder estar em

uma posição mais crítica que o outro, isto é, um roteador pode pertencer a um conjunto maior de caminhos na rede. Logo colocar um dispositivo crítico em estado “dormente” irá ter mais impacto que colocar um roteador com o qual nenhum caminho passa por ele, por exemplo.

A Tabela 4 mostra, também, que a diferença da confiabilidade calculada pelo REASoN para as Topologias 2 e 3 é de uma casa decimal, que é significativo quando se analisa confiabilidade e disponibilidade de redes com serviços críticos. Por exemplo, os serviços de transações bancárias que a cada 5 minutos de inatividade refletem na perda de R$ 1 milhão, e para ter um período de inatividade menor que 5 minutos, é necessário ter a disponibilidade e confiabilidade maior que seis casas decimais. Esse tempo de inatividade é o período 10 segundos que tenha ocorrido algum errro, como especificado pela norma G.826 no Capítulo 3. A Tabela 5 apresenta a confiabilidade calculado com o fator de cobertura igual 97% em vez de 100%. Essa mudança no fator de cobertura tem o impacto de colocar um dispositivo em estado de “dormência” ainda maior, com queda na terceira casa decimal. Este comportamento é devido à existência da probabilidade de 3% o dispositivo em estado de “dormência” não ser ativado quando requisitado. Quando fator de cobertura é igual a 97% o tempo de indisponibilidade para a Topologia 2 chega a ser de 3 dias 15h e 40min (vide Tabela 2), o que significa em uma perda de mais de 1 bilhão de reais. A análise da confiabilidade e disponibilidade depende de que tipo de serviço esta trafegando na rede, pois para serviços mais críticos períodos de indisponibilidade pode gerar grandes prejuízos.

Tabela 6: Disponibilidade da rede avaliada pelo REASoN com fator de cobertura igual a 1.

Disponibilidade da Rede				
	REASoN		Método padrão	
	MTTR = 0.5h	MTTR = 4hs	MTTR = 0.5h	MTTR = 4hs
Topologia 1	0.999999	0.999999	0.999999	0.999999
Topologia 2	0.999916	0.999981	0.999999	0.999999
Topologia 3	0.999999	0.999999	0.999999	0.999999

Tabela 7: Disponibilidade da rede avaliada pelo REASoN com fator de cobertura igual a 0.97.

Disponibilidade da Rede				
	REASoN		Método padrão	
	MTTR = 0.5h	MTTR = 4hs	MTTR = 0.5h	MTTR = 4hs
Topologia 1	0.999999	0.999999	0.999999	0.999999
Topologia 2	0.999047	0.999254	0.999999	0.999999
Topologia 3	0.999999	0.999999	0.999999	0.999999

Pode-se ver na Tabela 6 que o MTTR impacta a disponibilidade da rede, quanto menor o tempo para se reparar maior é a disponibilidade. A tabela mostra, também, que mesmo com uma taxa de reparo relativamente baixa, 0.5h, o tempo para acordar os dispositivos continua causando um grande impacto na disponibilidade, como na quinta casa decimal para a Topologia 2. Pode-se observar na tabela que a disponibilidade apresenta uma diferença ainda maior que quando calculada a confiabilidade (Tabela 4) entre as Topologias 2 e 3, onde roteadores diferentes são colocados no estado "dormente".

A Tabela 6 apresenta a cálculo da disponibilidade da rede, no qual é considerada uma taxa de reparo (MTTR). A MTTR é o tempo médio para à rede voltar ao seu estado inicial sem erros, mais informações podem ser encontradas no Capítulo 3. A Tabela 6 apresenta o resultado da disponibilidade calculada pelo REASoN e o método padrão, utilizando os valores de MTTR com 0.5h e 4hs e fator de cobertura igual a 1. Já a Tabela 7 apresenta o resultado da disponibilidade calculada pelo REASoN e o método padrão, utilizando os valores de MTTR com 0.5h e 4hs e fator de cobertura igual a 0.97. A tabla mostra que quando o fator de cobertura é diminuído o impacto de colocar um dispositivo é maior, com uma queda na quarta casa decimal. O que de acordo com a Tabela 2 significa 52min 36s de indisponibilidade e uma perda de R$ 10 milhões em um sistema de transações bancárias. Logo, o impacto na disponibilidade pode acarretar perda financeira, sem contar com as perdas relacionadas aos impactos na satisfação do cliente e reputação da empresa, além de causar violações nos SLA (*Service Level*

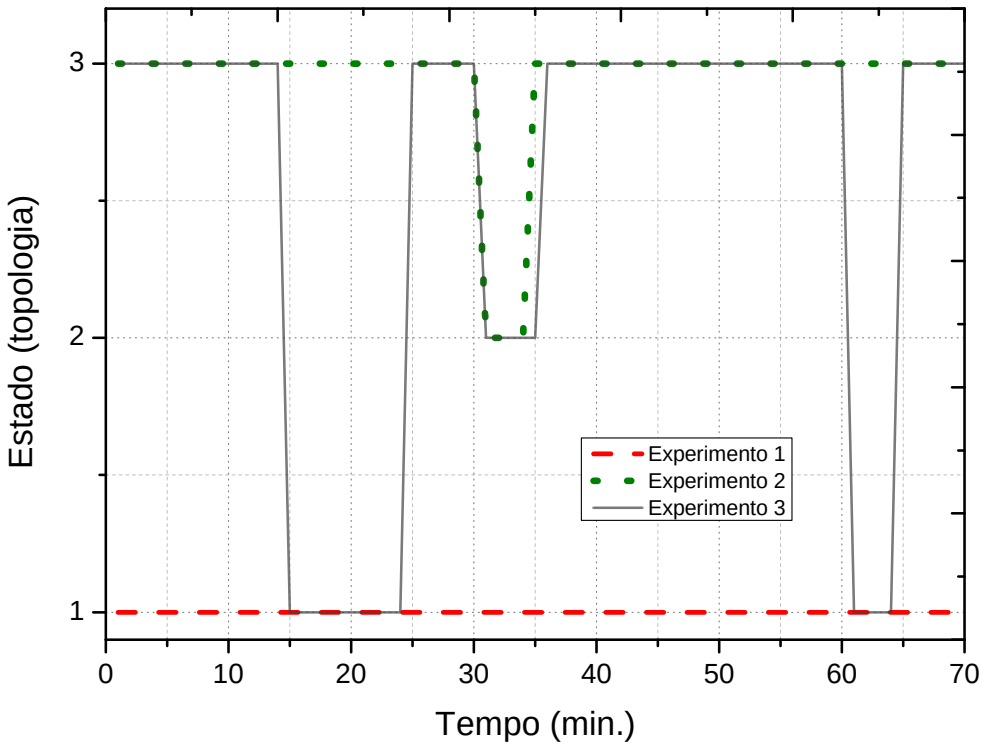
Agreement). O impacto de economizar energia pode acarretar em grandes perdas financeiras, sem contar com as perdas relacionadas ao impacto na satisfação do cliente e na reputação da empresa.

7.8.1 Análise da execução dos experimentos

O comportamento da topologia nos experimentos pode ser visualizado na Figura 23. O primeiro experimento não apresenta variação na topologia, uma vez que o estado dos roteadores nunca é alterado. O segundo experimento tem pouca variação na topologia, apenas apresentando variação quando só um fluxo de vídeo está sendo transmitindo pelo Servidor A1 (o que ocorre no intervalo entre 30 e 35 minutos). Neste caso, o SustNMS identifica que o caminho que mais economiza energia é aquele contém o roteador R3, logo, o R3 é acordado e o R5 é colocado no estado "dormente". A análise para se definir qual caminho é mais energeticamente eficiente é baseada nos perfis de tráfego apresentados anteriormente. O experimento que mais apresenta variação na topologia é o terceiro, porque ele evita o máximo possível manter a rede com disponibilidade inferior a 0.999999. Desta forma, sempre que possível o sistema muda a topologia para obter maior disponibilidade, mas também tenta economizar, causando bastantes variações. No terceiro experimento, a Topologia 2 é sempre evitada, e apenas ocorre no caso da rede estar sem carga e o roteador que mais gasta energia e impacta a disponibilidade, ser colocado em estado de "dormente", dado que o intuito sempre é economizar energia.

A Figura 23 mostra como a topologia da rede varia durante a execução dos experimentos. Para complementar as informações dessa figura, foi criada a Figura 24 que apresenta a variação da topologia e a disponibilidade da rede para a execução de todos os experimentos. Pode-se visualizar na Figura 24, que no caso do experimento 1 a disponibilidade não é alterada. Já durante a execução do experimento 2 é mostrado que em alguns momentos a disponibilidade diminui para permitir que o

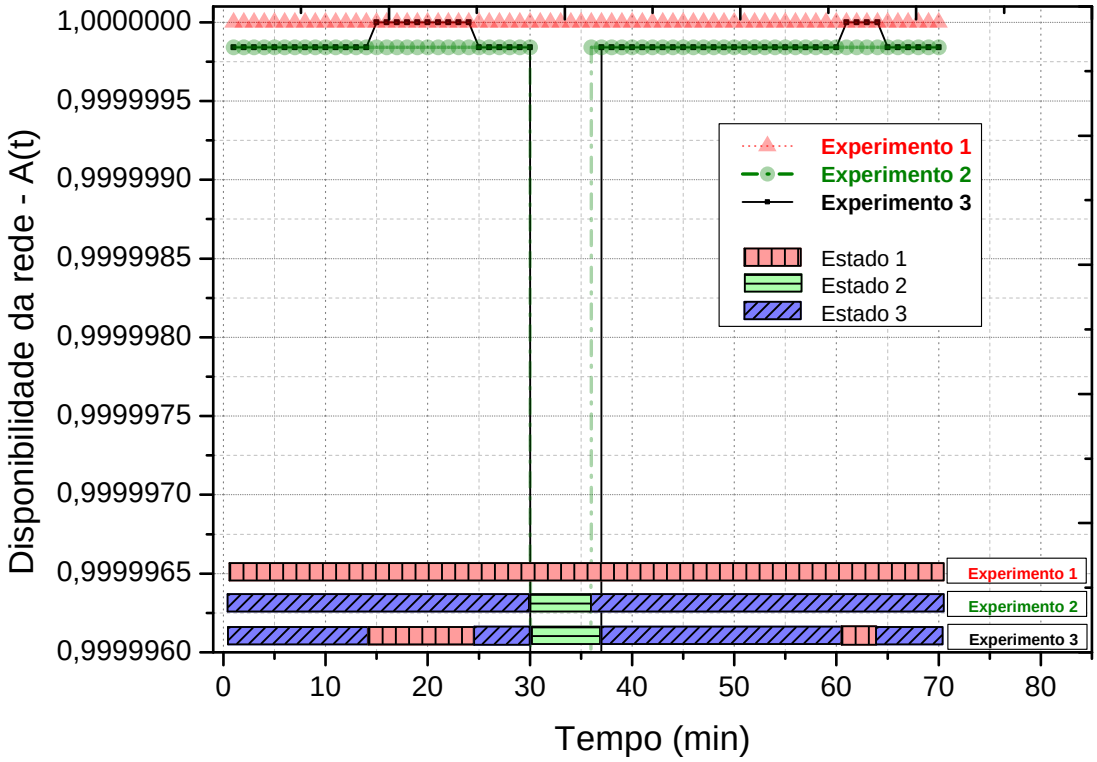
Figura 23: Variação da topologia da rede para os experimentos 1, 2 e 3.



SustNMS possa colocar algum roteador no estado “dormente” e economizar energia. Por outro lado, existem períodos no qual a disponibilidade aumenta devido a algum dispositivo ter sido acordado. Assim como no experimento 2 e 3 apresentam variações na confiabilidade, porém com maior frequência. Além disso, o experimento 3 apresenta um período maior onde a disponibilidade está com seis noves depois da casa decimal, quando comparado com o experimento 2, onde na maior parte do tempo, a disponibilidade da rede está com seis noves na casa decimal.

A Figura 24 apresenta o comportamento das decisões do SustNMS, mostrando que toda vez que um roteador tem seu estado alterado, a disponibilidade é impactada. Diferentemente do comportamento do experimento 2, no experimento 3 o SustNMS economiza energia e ao mesmo tempo mantém a disponibilidade da rede mais alta o possível. No experimento 3 o SustNMS toma decisões baseadas no cálculo do REASoN, o qual consegue identificar o impacto de colocar um dispositivo no modo “dormente”. O experimento 2 não consegue identificar o impacto de economizar

Figura 24: Variação da disponibilidade da rede durante a execução dos experimentos 1, 2 e 3, com fator de cobertura de 100%.



energia por calcular a disponibilidade a partir do método padrão. Logo, o experimento 2 é gerenciado sem se ter conhecimento sobre as quedas na disponibilidade, e apresenta disponibilidade menor que os outros experimentos.

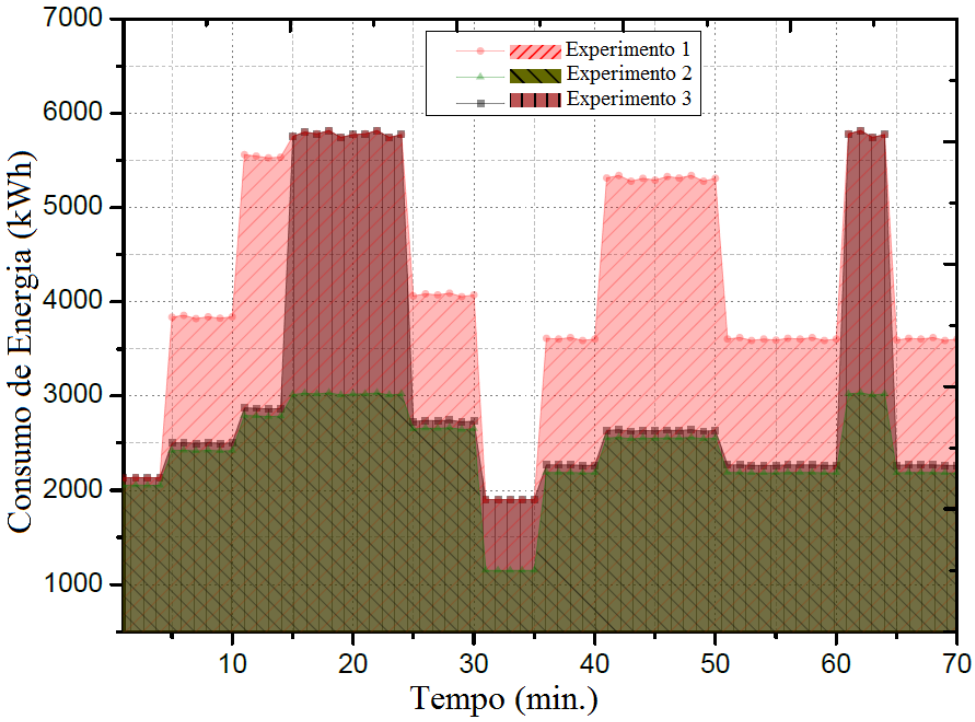
Por fim, as Figuras 23 e 24 mostram que o SustNMS pode configurar a topologia da rede de formas diferentes quando considerados requisitos diferentes. Assim, toda vez que o SustNMS tiver que escolhe qual roteador deve ser colocado em estado "dormente", é importante definir que deve ser priorizado, economia de energia, disponibilidade ou confiabilidade. A seguir é apresentado o consumo energético de cada experimento.

7.8.2 Consumo energético dos experimentos 1, 2 e 3

As decisões que tangem o comportamento da rede, apresentadas anteriormente nos experimentos, podem acarretar em consumo de energia diferente, como mostrado

na Figura 25. Desta forma, esta seção apresenta o consumo energético instantâneo da rede, quando políticas diferentes estão sendo aplicadas, nas quais, prioridades diferentes são definidas.

Figura 25: Consumo energético para a operação de todos os experimentos.



A Figura 25 apresenta o consumo de energia dos experimentos 1, 2 e 3. O experimento 1 serve como base de comparações, no qual todos os dispositivos ficam sempre ligados, representando o experimento onde o consumo de energia é máximo. Já os outros experimentos apresentam economia de energia, uma vez que definem cenários onde dispositivos são colocados em estado "dormente" como visto na Figura 23.

Nos experimentos 2 e 3 sempre que possível algum dispositivo é colocado no estado "dormente" [1]. No caso do Experimento 2, se houver a possibilidade da escolher qual dispositivo colocar um em estado "dormente", deverá ser escolhido o dispositivo que consome mais energia. O experimento 3 apresenta, também, economia de energia,

[1] É utilizado o temo "sempre que possível", pois, nunca irá ser colocado um dispositivo no modo de "dormente" que pode desconectar a rede.

porém na escolha entre dispositivos, será escolhido o dispositivo que menos degrada a confiabilidade e a disponibilidade da rede.

Tabela 8: ECR de todos os experimentos.

ECR	
Experimento 1	8.48
Experimento 2	4.82
Experimento 3	6.17

Os valores de ECR (*Energy Consumption Rating*) dos Experimentos, são apresentados na Tabela 8. Quanto menor o ECR, mais eficiente energeticamente é a rede, logo em ordem de eficiência se tem o Experimento 2, 3 e 1.

O consumo energético da rede do Experimento 2 é 43% menor que o consumo energético do Experimento 1. O Experimento 3 tem o consumo energético 27% menor que o Experimento 1. O Experimento 3 economiza menos energia que o Experimento 2, porém o terceiro mantém a rede com a disponibilidade mais alta. Em outras palavras, pode-se ver que colocar o R3 em estado “dormente” (que acontece majoritariamente no experimento 2), economiza menos energia que colocar o R5 (que acontece majoritariamente no experimento 3), porém, com o R3 ligado a disponibilidade da rede é maior (Tabela 4).

O experimento 3 consome 21% de energia a mais que o Experimento 2, mostrando que priorizar disponibilidade implica em economizar menos energia na rede. Porém, a economia de energia do experimento 3 continua representativa, e indicando que é possível economizar energia e mesmo assim manter alta disponibilidade e confiabilidade na rede. Esses resultados mostram que existe uma relação de compromisso entre economizar energia e manter mais alta o possível a disponibilidade e confiabilidade. Essa análise é de extrema importância na tomada de decisão de colocar um roteador em estado “dormente” ou não.

7.9 Considerações finais do capítulo

Esse capítulo mostrou que a decisão de um sistema sobre economizar energia depende de vários fatores, como priorizar o máximo possível à economia de energia, ou economizar menos energia, mas priorizar alta disponibilidade e confiabilidade. Foi apresentado que quando a disponibilidade não é um requisito forte, pode-se economizar até 43%, porém, quando a prioridade é disponibilidade, a possibilidade de economia diminui passando para 27%. Essas porcentagens de economia de energia apresentadas estão fortemente relacionadas com a topologia da rede, perfil do tráfego da rede, e o perfil de consumo energético dos dispositivos. Desta forma, as porcentagens obtidas ilustram a existência de uma relação de compromisso entre consumo de energia e parâmetros de QoS, sendo, portanto, importante a análise da confiabilidade e disponibilidade da rede utilizando o REASoN.

Os experimentos realizados permitem, então, identificar a existência dessa relação de compromisso entre economizar energia e manter alta confiabilidade e disponibilidade da rede, fatores que são expressos por meio de parâmetros de QoS normalmente, especificados em um SLA. Essa análise mostra-se importante quando um sistema de gerenciamento de rede orientado a eficiência energética opera segundo restrições fortes confiabilidade e a disponibilidade.

8 CONSIDERAÇÕES FINAIS

Esta dissertação conseguiu realizar os dois objetivos. Sendo que o primeiro objetivo era prover um método capaz de calcular confiabilidade e disponibilidade considerando a dinamicidade da rede e o tempo entre as transições dos estados energéticos. Então, foi proposto o método de Avaliação de Confiabilidade e Disponibilidade em Redes de Computadores Sustentáveis (REASoN - *Reliability and Availability Evaluation of Sustainable Network*). O segundo objetivo era apresentar a relação de compromisso entre economizar energia, e a confiabilidade e disponibilidade da rede. Então, foi executado testes com um sistema de gerenciamento de rede orientado à sustentabilidade (SustNMS - *Sustainability Oriented Network Management System*) composto pelo método REASoN para calcular a confiabilidade e disponibilidade da rede. Com base nas informações calculadas pelo REASoN, e requisitos de consumo de energia, o sistema tomou decisões de colocar dispositivos em modo "dormente" ou não. Foi apresentado, também, o consumo de energia quando o sistema priorizava eficiência energética, ou disponibilidade e confiabilidade, apresentando a relação de compromisso entre esses economizar energia e manter a disponibilidade e confiabilidade mais alta o possível.

8.1 Análise dos Resultados Obtidos

Como resultado, a dissertação apresenta, primeiramente, uma análise comparativa entre o REASoN e algumas das técnicas padrão para calcular confiabilidade e

disponibilidade em redes de computadores (Cadeia de Markov, e Conjunto-Conexo e Conjunto-Desconexo). Depois, a dissertação apresenta o REASoN implementado dentro de um sistema de gerenciamento de rede orientado a sustentabilidade chamado SustNMS, que utiliza o REASoN para tomada de decisão.

A primeira análise comparativa do REASoN com métodos tradicionais mostra que o REASoN consegue medir uma degradação na confiabilidade e disponibilidade que os métodos padrão não conseguem. Essa degradação está diretamente relacionada ao fato do REASoN considerar em seu cálculo o tempo médio para acordar um dispositivo em um estado de consumo energético reduzido, ou seja "dormente". Foram realizados vários testes mudando o estado e a topologia da rede e para cada experimento foi calculada a confiabilidade e a disponibilidade de todos os dispositivos e da rede como um todo. Sempre foram executados dois cálculos nos experimentos, um utilizando os métodos padrão e outro utilizando o REASoN para efetuar comparações. Os resultados mostram que a confiabilidade de um roteador dotado de duas conexões redundantes e um MTTF de 60 mil horas, calculada pelo REASoN, é menor que a confiabilidade calculada pelos métodos padrão, obtendo-se uma diferença na sexta casa decimal para os primeiros 78 meses. A dependência da amplitude e da duração do "pênalti" está diretamente relacionada ao tamanho da topologia, desta forma foram avaliados tamanhos diferentes de topologia, de 5 até 50 dispositivos. Os resultados também mostram que quanto maior o número de dispositivos ligados em rede, maior é a amplitude do "pênalti", porém, é menor a duração do mesmo. Esses impactos são significativos no contexto das redes de alto desempenho, que demandam alta confiabilidade e disponibilidade.

A segunda análise apresenta o REASoN sendo utilizado para a tomada de decisão em um sistema de gerenciamento de rede que aplica técnicas para economizar energia. A economia de energia ocorre, em condições de baixa carga na rede, quando alguns dispositivos são colocados no estado "dormente". O REASoN consegue medir o

impacto na confiabilidade e na disponibilidade da rede quando alguns dispositivos são colocados no estado “dormente”. Desta forma, o sistema consegue analisar se é vantajoso mudar ou não o estado dos dispositivos. A tomada de decisão está também alinhada com a política que está sendo empregada, ou seja, se a política prioriza economia de energia e permite qualquer tipo de degradação na disponibilidade da rede, o sistema irá se comportar de uma forma. Porém, se a política priorizar alta disponibilidade, mas mesmo assim ainda permitir economia de energia, o sistema tomará decisões de outra forma. Os resultados desse experimento mostram que quando o SustNMS está regido pela primeira política, a rede apresenta 43% de economia de energia em relação ao caso onde todos os dispositivos permanecem ligados. Porém, quando a segunda política está sendo aplicada, a rede economiza apenas 27%, mas apresenta maior disponibilidade e confiabilidade.

Esses resultados mostram que existe uma relação de compromisso entre economizar energia e manter a disponibilidade e a confiabilidade da rede mais alta o possível. Tal compromisso pode ser averiguado com a análise provida pelo REASoN, dado que os resultados mostram que os métodos padrão não capturam a diferença na disponibilidade e confiabilidade da rede quando um dispositivo está em estado “dormente”. Desta forma, o REASoN é uma ferramenta de bastante utilidade para a tomada de decisão nas redes orientadas à sustentabilidade.

Para validar os resultados dessa dissertação, é importante saber o escopo da rede, identificando quais são os objetivos dos serviços que irão ser transmitidos na rede. Pois, cada tipo de serviço possui seus próprios requisitos e alguns são mais sensíveis a períodos de inatividade na rede. Por exemplo, os serviços de transações bancárias, no qual reflete a perda de R$ 1 milhão a cada 5 minutos de inatividade. Isto implica em ter vários noves de disponibilidade, pois, ter menos de 5 minutos de inatividade significa ter a disponibilidade com mais de seis casas decimais. O resultados dos experimentos realizado com o SustNMS mostram queda

na disponibilidade e confiabilidade, avaliada pelo REASoN, na quinta casa decimal com fator de cobertura igual a 100%, representando uma mudança de 5 minutos de inatividade para 52 minutos (com apenas quatro noves). E utilizando o fator de coberto mais real, como de 97%, o impacto é ainda maior, deixando a disponibilidade com apenas 3 noves, representando 52min 36s de indisponibilidade e uma perda de R$ 10 milhões em um sistema de transações bancárias. Logo, tal mudança pode acarretar grande perda financeira, sem contar com as perdas relacionadas aos impactos na satisfação do cliente e reputação da empresa.

O trabalho realizado nessa dissertação possui duas limitações. Primeiramente, o trabalho avaliou uma rede com no máximo 50 dispositivos no Capítulo 5. Logo, não foi testado exaustivamente a escalabilidade da execução do método, ou seja, para uma rede com mais de mil dispositivos. Uma possível solução para minimizar o problema de escalabilidade seria separar a rede em domínios e analisá-los de forma hierárquica. Como segunda limitação, não foi avaliado o desempenho da rede, que juntamente com a disponibilidade e confiabilidade seria uma análise mais aprofundada da rede.

8.2 Contribuições

8.2.1 Base Teórica

Essa dissertação realizou pesquisas relacionadas ao cálculo de confiabilidade e disponibilidade nos trabalhos de (SHOOMAN, 2001; ALTIPARMAK; DENGIZ; SMITH, 2003; GREEN; HANT; LANZINGER, 2009; HE; QI, 2008; LAM; LI, 1986; LIN et al., 2010), e (YEH et al., 2010), e não encontrou um trabalho que considera no cálculo o tempo para acordar um dispositivo que está no estado “dormente”. Essa dissertação realizou, também, pesquisas relacionadas à relação de compromisso entre economia de energia e o desempenho da rede, e os trabalhos encontrados com mais relevância na área foram (MARSIC, 2013) e (BOLLA; BRUSCHI; CARREGA, 2010).

Porém, não foi encontrado um trabalho que analisa a relação de compromisso entre economia de energia e a confiabilidade e disponibilidade da rede.

8.2.2 Contribuição Prática e Inovação

Essa dissertação, propõe o método REASoN (*Reliability and Availability Evaluation of Sustainable Network*) que calcula, dinamicamente, a confiabilidade e a disponibilidade de uma rede levando em consideração, nesse cálculo, o tempo para "ativar" um dispositivo que está no estado ´´dormente". Esse método pode ser utilizado para analisar o impacto de colocar e retirar um dispositivo da rede em estado "dormente" em termos de confiabilidade e disponibilidade da rede. O REASoN calcula a confiabilidade e/ou disponibilidade da rede em duas etapas. Primeiramente, é calculada a R(t) ou a A(t) individual de cada dispositivo da rede (roteador ou comutador), que representa a probabilidade deste estar funcionando, utilizando a modelagem de Markov proposta. Depois disso, é calculada a R(t) ou a A(t) da rede, utilizando a extensão proposta para o Conjunto Conexo e Desconexo. Em ambos os casos, a extensão está relacionada incluir a consideração do tempo de "ativar" um dispositivo em estado "dormente".

8.2.3 Publicações

Publicações relacionadas diretamente com a dissertação:

- Patente:
  - MEIROSU, C; **AMARAL, M. C.**; COSTA, C. H. A.; CARVALHO, T. C. M. *Sustainablility oriented network management*.European Patent Office, The Hague. 2010. PCT/EP2011/067175.
- Journal (submetido):

- **AMARAL, M. C**.; COSTA, C. H. A.; JANUÁRIO, G.; RIEKSTIN, A.; CARVALHO, T. C. M.; MEIROSU, C. *Reliability evaluation for sustainable network: analytic and real application analyzes. Elsevier Computer Networks*.

- Conferências:

  - JANUÁRIO, G.; COSTA, C. H. A.; **AMARAL, M. C.**.; RIEKSTIN, A. C.; CARVALHO, T. C. M. ; MEIROSU, C. *Evaluation of a Policy-Based Network Management System for Energy-Efficiency*. IFIP/IEEE IM - International Symposium on Integrated Network Management, Maio/2013. Ghent, Belgica.

  - **AMARAL, M. C.**.; COSTA, C. H. A.; CARVALHO, T. C. M. ; MEIROSU, C.. *REASoN - REliability and/or Availability evaluation for Sustainable Networking*. Proceedings of the RNDM'12 - 4th International Workshop on Reliable Networks Design and Modeling. Outubro/2012. Russia, St. Petersburg.

  - COSTA, C. H. A.; **AMARAL, M. C.**, JANUÁRIO, G. C.; CARVALHO, T. C. M.; MEIROSU, C.. *SustNMS: Towards Service Oriented Policy-Based Network Management for Energy-Efficient Networks*. Proceedings of the Sustainable Internet and ICT for Sustainability (SustainIT). Outubro/2012. Pisa, Itália.

  - CARVALHO, T. C. M.; RIEKSTIN, A. C.; **AMARAL, M. C.**; COSTA, C. H. A.; JANUÀRIO, G. C.; DOMINICINI, C. K.; MEIROSU, C. *Towards Sustainable Networks Energy Efficiency Policy from Business to Device Instance Levels*. Proceedings of the 14th International Conference on Enterprise Information Systems (ICEIS). Junho/2012. Wroclaw, Polônia.

  - Capítulo de Livro:

    * CARVALHO, T. C. M.; MEIROSU, C; **AMARAL, M. C**.; JANUÁRIO, G.; RIEKSTIN, A.; COSTA, C. H. A.; MIERS, C.; GABOS, D.; CHENG, E.; FIGUEREDO, L. *Sustainability Oriented Policies applied to Network Management - A pratical view of refinement application of sustainability oriented policies.* 31º Simposio Brasileiro de Redes de Computadores e Sistemas Distribuidos (SBRC), Brasilia, Brasil, 2013.

- Pôsteres:

  – **AMARAL, M. C.**.; CARVALHO, T. C. M. *REASoN - Avaliação de Confiabilidade e Disponibilidade em redes de computadores sustentáveis.* II Workshop de Pós-Graduação da Área de Concentração Engenharia da Computação (II WPG-EC). Outubro de 2013. São Paulo, Brasil.

  – **AMARAL, M. C.**.; CARVALHO, T. C. M. *REASoN - Avaliação de Confiabilidade e Disponibilidade em redes de computadores sustentáveis.* Resultados parciais. Workshop de Pós-Graduação da Área de Concentração Engenharia da Computação (WPG-EC). Outubro de 2012. São Paulo, Brasil.

Publicações não relacionadas diretamente à dissertação:

- Conferências:

  – EVANGELISTA, P.; **AMARAL, M. C.**.; MIERS, C.; GOYA, W.; CARVALHO, T. C. M.; e SOUSA, V.. 2011. *EbitSim: An Enhanced BitTorrent Simulation Using OMNeT++ 4*. Proceedings: MASCOTS, 437-440. Singapore/Singapore: IEEE, Julho/27. doi:10.1109/MASCOTS.2011.46.

  – **AMARAL, Marcelo C**.; MIERS, C.; e CARVALHO, T. C. M. 2010. *Proposta de melhoria do Electronic Program Guide EPG para*

redes híbridas de IPTV. Proceedings JPC2010 - Jornada Peruana de Computacion. Trujilho, Peru. Outubro/2010.

Apresentação de trabalhos:

- Protótipo:
  - **AMARAL, M. C.**; COSTA, C. H. A.; CARVALHO, T. C. M. ; *Apresentação do protótipo de sistema de gerenciamento de rede orientado a sustentabilidade - SustNMS*. Apresentação realizada no centro de inovações *Ericsson Research Sweden*. Março/2012.

8.3 Trabalhos Futuros

Como trabalho futuro, será realizada a análise da capacidade de sobrevivência (*survivability*) da rede, considerando os dispositivos em estado "dormente". A Sobrevivência é a medida de tolerância a uma ou mais falhas e o sistema ainda conseguir prover serviços, mesmo que em um nível de desempenho reduzido (LIU VEENA B. MENDIRATTA, 2004). Ou seja, na análise da sobrevivência é necessário tanto calcular a probabilidade de falha, como a disponibilidade e confiabilidade, quanto analisar o desempenho do sistema quando ocorreu uma falha. Quando se analisa sobrevivência da rede são induzidos erros, para verificar o desempenho da rede após esse erro. Além disso, será realizado o teste da escalabilidade do método REASoN, avaliando a confiabilidade e disponibilidade de uma rede com mais de mil nós, incluindo a avaliação do desempenho da rede.

REFERÊNCIAS

ALTIPARMAK, F.; DENGIZ, B.; SMITH, A. Reliability estimation of computer communication networks: ANN models. In: . Washington, DC, USA: IEEE Computer Society, 2003. p. 1353 – 1358 vol.2. ISSN 1530-1346.

ANHALT, F.; DIVAKARAN, D. M.; PRIMET, P.-B. A virtual switch architecture for hosting virtual networks on the internet. In: . Texas, USA: High Performance Switching and Routing (HPSR), 2010 International Conference on, 2010. p. 26 –31.

ANTONAKOPOULOS, S.; FORTUNE, S.; ZHANG, L. Power-aware routing with rate-adaptive network elements. In: . Florida, USA: GLOBECOM Workshops (GC Wkshps), 2010 IEEE, 2010. p. 1428 –1432.

AVALLONE, S.; VENTRE, G. Energy efficient online routing of flows with additive constraints. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 56, n. 10, p. 2368–2382, jul 2012. ISSN 1389-1286. Disponível em: <http://dx.doi.org/10.1016/j.comnet.2012.03.011>.

BALDI, M.; OFEK, Y. Dynamic optical switching for a greener internet. In: . Newark, USA: Wireless and Optical Communications Conference, 2009. WOCC 2009. 18th Annual, 2009. p. 1 –3.

BARREIROS, M.; LUNDQVIST, P. *QOS-Enabled Networks: Tools and Foundations*. Wiley, 2010. (IT Pro). ISBN 9780470976746. Disponível em: <http://books.google.com.br/books?id=hgaPi-Pqa2gC>.

BIANZINO, A. P.; RAJU, A. K.; ROSSI, D. Apples-to-apples: a framework analysis for energy-efficiency in networks. *SIGMETRICS Perform. Eval. Rev.*, ACM, New York, NY, USA, v. 38, n. 3, p. 81–85, jan 2011. ISSN 0163-5999. Disponível em: <http://doi.acm.org/10.1145/1925019.1925036>.

BO, Y. et al. A green parallel forwarding and switching architecture for green network. In: . Sichuan, China: Green Computing and Communications (GreenCom), 2011 IEEE/ACM International Conference on, 2011. p. 85 –90.

BOLLA, F. D. R. et al. The potential impact of green technologies in next-generation wireline networks: Is there room for energy saving optimization? *Communications Magazine, IEEE*, v. 49, n. 8, p. 80–86, 2011. ISSN 0163-6804.

BOLLA, R.; BRUSCHI, R.; CARREGA, A. Greensim: An open source tool for evaluating the energy savings through resource dynamic adaptation. In: . Ottawa, Canada: Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010 International Symposium on, 2010. p. 89 –95.

BOLLA, R. et al. Enabling backbone networks to sleep. *Network, IEEE*, v. 25, n. 2, p. 26 –31, march-april 2011. ISSN 0890-8044.

BOLLA, R. B. R.; DAVOLI, F.; CUCCHIETTI, F. Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures. *Communications Surveys Tutorials, IEEE*, v. 13, n. 2, p. 223–244, 2011. ISSN 1553-877X.

CABRAL, R. Qoe - quality of experience - a conceptual essay. In: WANG, W. et al. (Ed.). *Integration and Innovation Orient to E-Society Volume 2*. Springer US, 2007, (IFIP International Federation for Information Processing, v. 252). p. 193–199. ISBN 978-0-387-75493-2. Disponível em: <http://dx.doi.org/10.1007/978-0-387-75494-9_24>.

CHABAREK, J.; BARFORD, P. Power-awareness extensions for network testbeds. In: . Kyoto, Japan: IEEE International Conference on Communications Workshops (ICC), 2011. p. 1 –6.

CHAUDHARI, S. et al. Green-it: An approach to energy savings using energy aware network management system. In: . Kharagpur, India: National Conference on Communications (NCC), 2012. p. 1 –5.

CHOWDHURY, N. M. K.; BOUTABA, R. A survey of network virtualization. *Computer Networks*, v. 54, n. 5, p. 862–876, abr. 2010. ISSN 1389-1286. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1389128609003387>.

CISCO. *Cisco Cable IP Solutions for High-Availability Networks*. USA, 2003. Disponível em: <http://www.cisco.com/en/US/products/hw/cable/ps2209/products_white_paper09186a00801af388.shtml>.

CISCO. *Control Plane*. USA, 2003. Disponível em: <http://www.cisco.com/en/US/docs/routers/crs/crs1/16_slot_lc/system_description/reference/guide/sysdsc8.pdf>.

CISCO. *Cisco Resilient Ethernet Protocol*. USA, 2007. Disponível em: <http://www.cisco.com/en/US/prod/collateral/switches/ps6568/ps6580/prod_white_paper0900aecd806ec6fa.pdf>.

CISCO. *Stateful Switchover*. USA, 2007. Disponível em: <http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.htmlwp1211930>.

CISCO. *Cisco ME 6500 Series Ethernet Switch*. USA, 2009. Disponível em: <http://www.cisco.com/en/US/prod/collateral/switches/ps6568/ps6845/ps6846/product_data_sheet0900aecd8040657e.html>.

CLAISE, B.; PARELLO, J. Eman: Energy-management activities at the ietf. *IEEE Internet Computing*, IEEE Computer Society, Los Alamitos, CA, USA, v. 17, n. 3, p. 80–82, 2013. ISSN 1089-7801.

COMANICIU, C.; MANDAYAM, N.; POOR, H. Radio resource management for green wireless networks. In: . Anchorage, Alaska: Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th, 2009. p. 1 –5. ISSN 1090-3038.

COSTA, C. H. A. et al. SustNMS: towards service oriented Policy-Based network management for Energy-Efficiency. In: . Pisa, Italy: Second IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT 2012) (SustainIT 2012), 2012.

CUOMO, F. et al. Network pruning for energy saving in the internet. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 56, n. 10, p. 2355–2367, jul 2012. ISSN 1389-1286. Disponível em: <http://dx.doi.org/10.1016/j.comnet.2012.03.009>.

DEMESTICHAS, P. et al. Green footprint of cognitive management technologies for future networks. In: . Adriatic Coast, Croatia: MIPRO, Proceedings of the 34th International Convention, 2011. p. 645 –649.

DESPINS, C. et al. Leveraging green communications for carbon emission reductions: Techniques, testbeds, and emerging carbon footprint standards. *Communications Magazine, IEEE*, v. 49, n. 8, p. 101 –109, aug. 2011. ISSN 0163-6804.

DONGMEI, W.; GUANGZHI, L. Efficient distributed bandwidth management for mpls fast reroute. *Networking, IEEE/ACM Transactions on*, v. 16, n. 2, p. 486 –495, april 2008. ISSN 1063-6692.

DUMITRASCU., I.; POPA., A. *MPLS for Linux Project*. 2013. Disponível em: <http://sourceforge.net/projects/mpls-linux/>.

ETINGOF, L. *SNMP library for Python*. 2013. Disponível em: <http://pysnmp.sourceforge.net/>.

FETTWEIS, G.; ZIMMERMANN, E. Ict energy consumption trends and challenges. In: . Lapland, Finland: Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2008.

GREEN, H.; HANT, J.; LANZINGER, D. Calculating network availability. In: . United States: IEEE Aerospace conference, 2009. p. 1–11. ISBN 978-1-4244-2621-8.

GREENBERG, A. et al. *Refactoring Network Control and Management: A Case for the 4D Architecture*. Pennsylvania, USA, 2005.

GRIFFIN, D. *The relationship between management and control planes for delivering quality of service in multi-service networks*. Tese (Doutorado) — University College London - UCL, 2009.

GUGLIELMO, K. *Trends in high availability and fault tolerance*. Revista SearchCIO-Midmarket TechTarget, 2010. Disponível em: <http://searchcio-midmarket.techtarget.com/podcast/Trends-in-high-availability-and-fault-tolerance>.

HE, F.; QI, H. A method of estimating network reliability using an artificial neural network. In: . Wuhan, China: IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application. PACIIA'08, 2008. v. 2, p. 57–60. ISBN 978-0-7695-3490-9.

ITU-T. *End-to-end error performance parameters and objectives for international, constant bit-rate digical paths and connections*. Suiça, 2002. Disponível em: <http://www.cisco.com/en/US/prod/collateral/switches/ps6568/ps6845/ps6846/product_data_sheet0900aecd8040657e.html>.

JANUARIO, G. et al. Evaluation of a policy-based network management system for energy-efficiency. In: . Ghent, Belgium: Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on, 2013. p. 596–602.

JOHNSON, B. W. *The Design and Analysis of Fault Tolerant Digital Systems*. USA: Addison-Wesley, 1989. ISBN 0201075709.

KEOH, S. et al. Policy-based management for body-sensor networks. In: LEONHARDT, S.; FALCK, T.; MäHöNEN, P. (Ed.). 4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007), Springer Berlin Heidelberg, 2007, (IFMBE Proceedings, v. 13). p. 92–98. ISBN 978-3-540-70993-0. Disponível em: <http://dx.doi.org/10.1007/978-3-540-70994-7_16>.

KEOH, S. L. et al. Self-managed cell: A middleware for managing body-sensor networks. In: . Philadelphia, PA: Fourth Annual International Conference on Mobile and Ubiquitous Systems: Networking Services MOBIQUITOUS, 2007. p. 1–5.

KRAJEWSKI, L. N. J.; JUNG, P. *SMARTer 2020: The Role of ICT in Driving a Sustainable Future.* Brussels, Belgium, 2013.

LAM, Y. F.; LI, V. O. K. A survey of network reliability modeling and calculations. In: . USA: IEEE Military Communications Conference - Communications-Computers: Teamed for the 90's, 1986. MILCOM 1986, 1986. v. 1, p. 1.2.1–1.2.5.

LANGE, C. et al. Energy consumption of telecommunication networks and related improvement options. *IEEE Journal of Selected Topics in Quantum Electronics*, v. 17, n. 2, p. 285–295, abr. 2011. ISSN 1077-260X.

LI, H.; ZHAO, Q. A cut/tie set method for reliability evaluation of control systems. In: *American Control Conference, 2005. Proceedings of the 2005*. [S.l.: s.n.], 2005. p. 1048–1053 vol. 2. ISSN 0743-1619.

LIN, C. et al. A mesh network reliability analysis using reliability block diagram. In: . Osaka, Japan: 8th IEEE International Conference on Industrial Informatics (INDIN), 2010. p. 975–979. ISBN 978-1-4244-7298-7.

LIU VEENA B. MENDIRATTA, K. S. T. Y. Survivability analysis of telephone access network. In: . [S.l.]: Proceedings of the 15th International Symposium on Software Reliability Engineering, IEEE Computer Society, 2004. p. 367–378.

LUPU, E. et al. Amuse: autonomic management of ubiquitous e-health systems. *Concurr. Comput. : Pract. Exper.*, John Wiley and Sons Ltd., Chichester, UK, p. 277–295, mar. 2008. ISSN 1532-0626. Disponível em: <http://dx.doi.org/10.1002/cpe.v20:3>.

LYMBEROPOULOS, L.; LUPU, E.; SLOMAN, M. An adaptive policy based management framework for differentiated services networks. In: . Monterey, USA: Policies for Distributed Systems and Networks, 2002. Proceedings. Third International Workshop on, 2002. p. 147–158.

MACKAY, M. *Downtime Report: Top Ten Outages in 2013*. Yahoo SMALL BUSINESS ADVISOR, 2013. Disponível em: <http://smallbusiness.yahoo.com/advisor/downtime-report-top-ten-outages-2013-132000202.html>.

MARCUS, E.; STERN, H. *Blueprints for high availability: designing resilient distributed systems*. John Wiley & Sons, 2000. ISBN 9780471356011. Disponível em: <http://books.google.com.br/books?id=ct1QAAAAMAAJ>.

MARSIC, I. *Computer network - performance and quality of service*. New Brunswick, New Jersey, USA: Rutgers University, 2013. Disponível em: <http://www.ece.rutgers.edu/~marsic/books/CN/book-CN_marsic.pdf>.

NEJABATI, R. et al. Optical network virtualization. In: . Bologna, Italy: Optical Network Design and Modeling (ONDM), 2011 15th International Conference on, 2011. p. 1 –5.

NISHIMURA, S. et al. Components and interconnection technologies for photonic-assisted routers toward green networks. *Selected Topics in Quantum Electronics, IEEE Journal of*, v. 17, n. 2, p. 347 –356, march-april 2011. ISSN 1077-260X.

OSHANA, R.; KRAELING, M. *Software Engineering for Embedded Systems: Methods, Practical Techniques, and Applications*. Elsevier Science, 2013. ISBN 9780124159419. Disponível em: <http://books.google.com.br/books?id=qNrl7xV2nxkC>.

RUBIO-LOYOLA, J. *Towards the Policy Refinement Problem in Policy-based Management Systems: A synthesis study*. German: VDM Publishing, 2008. ISBN 9783836486934.

SALISBURY, B. *The Control Plane, Data Plane and Forwarding Plane in Networks*. USA, 2012. Disponível em: <http://networkstatic.net/the-control-plane-data-plane-and-forwarding-plane-in-networks/>.

SCHAEFFER-FILHO, A. et al. Towards supporting interactions between self-managed cells. In: . Mass, USA: Self-Adaptive and Self-Organizing Systems, 2007. SASO '07. First International Conference on, 2007. p. 224–236.

SHIMONISHI, H.; ISHII, S. Virtualized network infrastructure using openflow. In: . Osaka, Japan: Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP, 2010. p. 74 –79.

SHOOMAN, M. L. *Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design*. 1st. ed. NJ, USA: Wiley-Interscience, 2001. ISBN 0471293423.

SHUAIB, K.; SALLABI, F. Extending ospf for large scale mpls networks. In: . St Louis, MO: Advances in Wired and Wireless Communication, 2005 IEEE/Sarnoff Symposium on, 2005. p. 13–16.

STANISAVLJEVIC, M. *Reliability of Nanoscale Circuits and Systems: Methodologies and Circuit Architectures*. Springer New York, 2011. ISBN 9781441962171. Disponível em: <http://books.google.com.br/books?id=C1NnXZXDVIcC>.

STRASSNER, J.; STRASSNER, J. *Policy-based Network Management: Solutions for the Next Generation*. Morgan Kaufmann Publishers, 2004. (Morgan Kaufmann series in networking). ISBN 9781558608597. Disponível em: <http://books.google.com.br/books?id=lid7plhI1PQC>.

THULIN, M. *Measuring Availability in Telecommunications Networks*. Dissertação (Mestrado) — KTH Royal Institute of Technology, 2004.

TRIVEDI, K. S.; SELVAMUTHU, D. Markov modeling in reliability. In: . USA: Encyclopedia of Quantitative Risk Analysis and Assessment, John Wiley & Sons, Ltd, 2008. ISBN 9780470061596.

TWIDLE, K. *Ponder2 Wiki*. Site oficial, 2011. Disponível em: <http://ponder2.net/>.

TWIDLE, K. et al. Ponder2: A policy system for autonomous pervasive environments. In: . Valencia, Spain: Autonomic and Autonomous Systems, 2009. ICAS '09. Fifth International Conference on, 2009. p. 330–335.

WANG, X. et al. A survey of green mobile networks: Opportunities and challenges. *Mob. Netw. Appl.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 17, n. 1, p. 4–20, 2012. ISSN 1383-469X. Disponível em: <http://dx.doi.org/10.1007/s11036-011-0316-4>.

WATERS, G. et al. *Policy Framework Architecture*. California, USA, 1999.

XIA, M. et al. Greening the optical backbone network: A traffic engineering approach. In: . Cape Town, South Africa: Communications (ICC), 2010 IEEE International Conference on, 2010. p. 1 –5. ISSN 1550-3607.

YEH, W. et al. A particle swarm optimization approach based on monte carlo simulation for solving the complex network reliability problem. *IEEE Transactions on Reliability*, v. 59, n. 1, p. 212–221, mar. 2010. ISSN 0018-9529.

APÊNDICE A – ARCABOUÇO PONDER2

O Ponder2 foi criado pela universidade Imperial College London. Ele é baseado em orientação a objeto e pode ser utilizado para especificação de políticas de segurança, e políticas de gerenciamento de rede e sistemas distribuídos (LYMBEROPOULOS; LUPU; SLOMAN, 2002). Ele compreende um sistema de propósito geral de gerenciamento de objetos, que troca mensagens entre os objetos (TWIDLE et al., 2009).

Trata-se de um arcabouço de código aberto, que provê acesso total ao código fonte, permitindo modificações. Ele contem um manual completo, assim com um tutorial que auxilia no aprendizado de como configurar o arcabouço. Além disso, o Ponder2 é utilizado em várias aplicações (RUBIO-LOYOLA, 2008).

A primeira versão do Ponder incluía um conjunto de ferramentas para manipulação de políticas. Ela foi projetada para gerenciamento de redes e sistemas em geral. Já a segunda versão, o Ponder2, foi significantemente reprojetada e re-implementada, para abranger mais áreas, como desde pequenos dispositivos até sistemas mais complexos (TWIDLE, 2011).

O Ponder2 é baseado no conceito de células auto gerenciáveis (SMC - Self-Managed Sell). Uma SMC é definida como um conjunto de componentes de hardware e software em um domínio administrativo capaz de trabalhar e se gerenciar de forma autônoma (TWIDLE et al., 2009).

O arcabouço combina o gerenciamento de objetos distribuídos com (TWIDLE,

2011):

- Um serviço de domínios que prove uma estrutura para gerenciar objetos de forma similar a diretórios em um sistema operacional;
- Um interpretador de políticas de obrigação, o qual interpreta regras de Evento-Condição-Ação;
- Um interpretador de comandos que aceita comandos da linguagem PonderTalk, que é uma linguagem de alto nível utilizada no Ponder2; e
- Aplicador de autorização que trabalha como políticas de autorização, além de um domínio de resolução de conflitos entre as políticas de autorização.

A.1 Células Auto-Gerenciáveis (SMC)

Uma SMC, como comentado anteriormente, é um conjunto de componentes de *hardware* e *software* em um domínio administrativo capaz de trabalhar e se gerenciar de forma autônoma. As principais funcionalidades da SMC é um transportador de eventos, um serviço de descoberta e um serviço de política (LUPU et al., 2008).

Para grandes sistemas e ambientes complexos é possível construir uma composição de várias (KEOH et al., 2007a). A interação entre as SMCs pode ser feita por um transportador de eventos ou por meio de algum outro paradigma de comunicação, como por exemplo, mensagens de rede de conexão ponto-a-ponto ou invocações remotas (LUPU et al., 2008) utilizando mensagens TCP ou UDP. Além disso, ainda é possível escalar a arquitetura da SMC considerando os recursos gerenciáveis são as próprias SMCs que como compostas por SMCs (LUPU et al., 2008).

Políticas descrevem como um sistema deve se adaptar para responder a um evento, que pode ser falhas ou mudanças de requisitos. A SMC implementa a lógica da política

de resposta à ocorrência de mudanças no estado de um objeto ou recurso gerenciado, que implica em disparar uma ação para, caso necessário, afetar o estado do sistema (KEOH et al., 2007b).

O serviço de políticas é composto por:

- Uma interface que recebe notificações;
- Uma interface que aceita requisições externas; e
- Uma interface que envia instruções para os objetos externos.

A.2 Objetos gerenciados

Qualquer módulo no Ponder2 é considerado um objeto gerenciável, sendo que os objetos podem ser eventos, políticas e domínios. Os objetos são carregados dentro de uma SMC, e então após esse passo é possível enviar mensagens para criar instancias dos objetos gerenciados. Os objetos gerenciados devem ser carregados dentro de um SMC, produzindo uma "fábrica" de objetos gerenciados (como ocorre com uma classe em Java). Essa "fábrica" de objetos recebe instruções para criar uma nova instancia de um objeto gerenciado que irá trabalhar no sistema, como se fosse um instancia de uma classe em Java (TWIDLE, 2011).

Os objetos gerenciáveis, em geral, são implementados na linguagem de programação Java, e é utilizada a notação *@annotation* no código, que indicará para o compilador Java que o objeto dessa classe será um objeto gerenciado pelo Ponder2. Desta forma, com essa notação, o Ponder2 consegue enviar mensagens para os objetos utilizando a linguagem PonderTalk (que será descrito mais a frente). Um exemplo da criação de uma classe em Java que será mapeada ao Ponder2 é dado no Algoritmo 5.

Os domínios são objetos gerenciados que atuam como *containers* para os objetos gerenciados. Eles são como diretórios de um sistema operacional convencional, porém

Algoritmo 5: Exemplo de um código em java que recebe mensagens do Ponder2.

```
#@Ponder2op("name:age:") //Notação - interpreta os comandos dos
PonderTalk
public void setInfo(String name, int age) this.name = name;
this.age = age;
```

com uma diferença importante, pois um objeto gerenciado pode pertencer a vários domínios. Os domínios são utilizados para agrupar objetos, logo, dessa maneira uma política pode ser aplicada a um conjunto de dispositivos (LYMBEROPOULOS; LUPU; SLOMAN, 2002). Os domínios são objetos gerenciados que aceitam ações de adição e remoção de objetos (KEOH et al., 2007b).

A.3 Eventos

Os eventos são notificações com atributos, que são criados por objetos gerenciados e também são objetos gerenciados. Eles podem ser gerados internos ou externamente, ou seja, capturado por um serviço de monitoramento (LYMBEROPOULOS; LUPU; SLOMAN, 2002). Os eventos disparam gatilhos para ativar políticas, e podem se integrados com um ou mais eventos por meio de um adaptador de objetos.

A.4 Serviço de Descoberta

O serviço de descoberta detecta novos dispositivos nas SMC, e baseado nas informações dos dispositivos e das políticas determina em que domínio o novo dispositivo será alocado (SCHAEFFER-FILHO et al., 2007). Quando designado para um domínio, o objeto irá receber todas as mensagens enviadas para esse domínio. O serviço de descoberta é esperado que trabalhasse com qualquer tipo de política, porque é um serviço complementar ao serviço de políticas. Um exemplo de serviço de descoberta é apresentado no Algoritmo 6.

Algoritmo 6: Exemplo de um algoritmo de serviço de descoberta.

1- O serviço de descoberta manda sua mensagem de identidade em uma frequência de x minutos.
2 - Um dispositivo novo responde para o serviço de descoberta uma mensagem que identifica ele mesmo.
3 - O serviço de descoberta faz uma requisição sobre o perfil e credenciais de autenticação para o novo dispositivo.
4 - O serviço de descoberta decide se irá aceitar a entrada do novo dispositivo e para qual papel ele será designado.
5 - Todos os dispositivos membros enviam uma mensagem de presença para o serviço de descoberta em uma frequência de x minutos.
se *dispositivo == aceito* **então**
    O dispositivo aceito é informado para os outro módulos, e é criado um componente de detecção de eventos.
fim
se *Não enviou N mensagens de presença* **então**
    É entendido que o dispositivo não faz mais parte da SMC e é gerado um evento que de saída do dispositivo.
fim

A.5 Políticas

Política é um conjunto de regras utilizadas para gerenciar e controlar um conjunto recursos e serviços de rede (STRASSNER; STRASSNER, 2004). No Ponder2 a política é criada em uma fabrica e é considerado um objeto gerenciado. Além disso, o Ponder2 define dois tipos de políticas: de autorização e de obrigação. Política de autorização define controle de acesso, permitindo ou negando troca de mensagens entre objetos, determinando quais atividades um membro de um domínio pode executar em um objeto. Políticas de obrigação especifica que ação uma a gente deve fazer dado a ocorrência de um evento e de algum limiar pré-definido ser ultrapassado.

A.6 PonderTalk

A linguagem PonderTalk, que é baseada na linguagem SmallTalk, é utilizada para configurar e controlar o Ponder2. Em comparação com o SmallTalk, o PonderTalk não possui definição de classes, possui algumas diferenças na sintaxe, e os objetos são

criados no Java, porém vários outros aspectos são bem semelhantes.

O PonderTalk é um sequencia de sentenças separadas por ponto final, sendo que cada sentença contem um objeto e um comanda a ser aplicado nesse objeto. Em resposta ao comando, o objeto retorna uma mensagem contendo outro objeto ou ele mesmo. Variáveis temporárias podem ser criadas sem declaração e para definir valores para as variáveis, a expressão ":="é utilizada. Existem três tipos de mensagens:

- Unária, um comando de apenas uma palavra que o objeto deve reconhecer;
- Binária, onde é passado um argumento, em geral um caractere especial "+"e »>"seguindo por um valor;
- Palavra-chave, uma função com um ou mais argumentos.

Em geral, tudo é processado da esquerda para a direita, porém, no caso de mensagens combinadas de vários tipos, as mensagens unárias são processadas primeiro, seguindo pelas binárias e depois palavra-chave. Uma funcionalidade poderosa do PonderTalk é o uso de blocos, que atua como uma função e é uma sessão do código como uma ou mais sentenças e argumentos. O bloco funciona com caixa preta e retorna o resultado da última sentença. Um descrição mais detalhada do PonderTalk está disponível no site oficial do projeto Ponder2 (TWIDLE, 2011).

A.7 Extras

O Ponder2 tem um shell interativo, similar ao shell Unix, que é apto em listar as estruturas dos domínios e executar as sentenças do PonderTalk. Ponder2 pode ser executado em uma máquina separada e as sentenças do PonderTalk podem ser enviadas remotamente através de uma conexão SSH.