HELEN PETERS DE ASSUNÇÃO

Gerenciamento de Serviços em Redes de Sensores Sem Fio Autonômicas

Belo Horizonte – MG

Agosto / 2007

HELEN PETERS DE ASSUNÇÃO

Gerenciamento de Serviços em Redes de Sensores Sem Fio Autonômicas

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção de título de Mestre em Engenharia Elétrica.

Orientadora:
Linnyer Beatrys Ruiz

UNIVERSIDADE FEDERAL DE MINAS GERAIS
ESCOLA DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Belo Horizonte – MG

Agosto / 2007

Agradecimentos

A Deus. “Porque dele, e por meio dele, e para ele são todas as coisas. A ele, pois, a glória eternamente. Amém!” Romanos 11:36.

À Professora Linnyer Beatrys Ruiz, orientadora e amiga, pelas críticas e sugestões relevantes feitas durante a orientação. Seus conselhos tanto profissionais quanto pessoais foram muito valiosos.

Aos meus professores do curso de mestrado pelos ensinamentos e dedicação na tarefa de formar mestres.

Aos colegas dos projetos Manna e Sensornet pela ajuda científica e pela sincera amizade.

Aos amigos do mestrado, em especial: Júlio, Cadu e Jeferson. Estes anos foram muito mais produtivos e divertidos com vocês por perto.

À UFMG pela oportunidade de desenvolver esta dissertação de mestrado.

Ao CNPq pela bolsa concedida durante os anos do curso.

À minha família pelo estímulo, apoio incondicional e grande carinho com que sempre me ajudaram. Agradeço principalmente pelas orações que me sustentaram nos momentos de maior dificuldade.

A todos os professores, funcionários e alunos do Programa de Pós-Graduação em Engenharia Elétrica, e todos aqueles que, direta ou indiretamente, contribuíram para a realização desta dissertação.

Resumo

Uma Rede de Sensores Sem Fio (RSSF) é uma ferramenta de sensoriamento distribuído de fenômenos, processamento e disseminação de dados coletados e informações processadas para um ou mais observadores. As RSSFs podem prover três tipos básicos de serviços: coleta, processamento e disseminação de dados. Uma falha em qualquer um desses serviços pode comprometer a qualidade do serviço provido. Em RSSFs falhas não são exceções, uma vez que problemas de exaustão de energia, perda do alcance de comunicação ou danos físicos são incidentes usuais. Considerando a necessidade de gerenciar as RSSFs de maneira eficiente, melhorando a qualidade e disponibilidade da informação, e que no futuro o gerenciamento dessas redes deverá ser integrado, este trabalho propõe o uso da Biblioteca de Infra-estrutura de Tecnologia da Informação (Information Technology Infrastructure Library - ITIL), um padrão "de facto" no gerenciamento de serviços de TI, no projeto de uma RSSF que implemente o serviço de auto-cura. Também é proposta uma abordagem de gerenciamento para uma rede auto-gerenciável que se adapta dinamicamente a fim de manter a disponibilidade do serviço e promover produtividade dos recursos. Esta abordagem tem por finalidade empregar o serviço de auto-cura em RSSF, permitindo à rede descobrir, examinar, diagnosticar e reagir a disfunções. Os resultados mostram que as funções e serviços de gerenciamento propostos aplicados a uma RSSF que implementa o serviço de auto-cura aumentam a longevidade e disponibilidade da rede.

Abstract

A Wireless Sensor Network is a distributed sensing tool, that disseminates collected data and processed information to one or more external entities called observers. WSNs can provide three types of basic services: sensing, processing and disseminating. A shortcoming of any of these services can disturb the network goals. However, failures are not exceptions in WSNs, since problems as energy exhaustion, communication range loss or physical damages are usual incidents. Considering the need of managing WSNs in an efficient manner, improving information quality and availability, and taking into account that in the future the management of these networks will be integrated, this work proposes the use of the Information Technology Infrastructure Library (ITIL), a de facto standard for IT service management, in the design of self-healing WSNs. We also propose a service management approach for a self-managed network that dynamically adapts itself in order to maintain the service availability and to promote the resources productivity. This approach aims to employ the self-healing service in WSNs, allowing them to discover, examine, diagnose and react to dysfunctions. Results show that service management applied to a self-healing WSN extends the longevity and availability of the network.

Lista de Figuras

2.1 Componentes de um nó sensor. 5

2.2 Nós sensores de diversas plataformas. 7

2.3 Gerente autonômico. 12

2.4 Arquitetura de um sistema autonômico. 13

3.1 Gerenciamento de Serviços de TI. 20

3.2 Gerenciamento de entrega suporte a serviços de TI. 21

3.3 Gerenciamento de entrega de serviços de TI. 23

4.1 Arquitetura de gerenciamento MANNA [RUIZ, 2003]. 32

4.2 Arquitetura baseada em gerente e agente. 37

5.1 Arquitetura para gerenciamento de serviços. 41

5.2 Touchpoint para nós sensores. 45

5.3 Níveis de gerenciamento. 47

5.4 Interação entre os gerentes autonômicos da RSSF. 49

5.5 Elemento autonômico x Suporte a serviços. 50

5.6 Responsabilidades do Gerenciamento do Nível do Serviço em RSSFs. 52

5.7 Responsabilidades do Gerenciamento de Disponibilidade em RSSF. 53

5.8 Responsabilidades do Gerenciamento de Continuidade em RSSF. 54

5.9 Responsabilidades do Gerenciamento de Capacidade em RSSF. 56

5.10 Responsabilidades do Gerenciamento de Finanças em RSSF. 57

6.1 Mensagens trocadas entre touchpoints e gerentes autonômicos. 61

6.2 Comparativo do Decaimento de Energia - Cenário 1a e 1b. 68

6.3 Comparativo do Decaimento de Energia - Cenário 2a e 2b. 68

6.4 Comparativo do Decaimento de Energia - Cenário 3a e 3b. 69

6.5 Comparativo do Decaimento de Energia - Cenário 4a e 4b. 69

6.6 Consumo de energia dos nós comuns e líderes. 70

6.7 Cenários 1a e 1b - Fluxo dos dados sensoriados na rede. 74

6.8 Cenários 2a e 2b - Fluxo dos dados sensoriados na rede. 74

6.9 Cenários 3a e 3b - Fluxo dos dados sensoriados na rede. 75

6.10 Cenários 4a e 4b - Fluxo dos dados sensoriados na rede. 76

6.11 Mensagens Perdidas. 77

Lista de Tabelas

2.1 Características de nós sensores reais. 6

6.1 Caracterização dos cenários simulados. 65

6.2 Caracterização das simulações executadas. 66

6.3 Valores sensoriados e seus limiares. 66

6.4 Consumo de energia dos nós comuns e líderes. 71

6.5 Fluxo dos dados sensoriados na rede. 73

Lista de Siglas

AC	Autonomic Computing
CA	Computação Autonômica
CMDB	Configuration Management DataBase
DSL	Definitive Software Library
DHS	Definitive Hardware Storage
ITIL	IT Infrastruture Library
ITSM	IT Service Management
MAPE	Monitor-Analyze-Plan-Execute Control Loop
QoS	Quality of Service
RfC	Request for Change
RSSF	Redes de Sensores Sem Fio
SLA	Service Level Agreement
TI	Tecnologia da Informação

Sumário

1 Introdução **1**

1.1 Objetivos 2

1.2 Contribuições 2

1.3 Organização 3

2 Redes de Sensores Sem Fio Autonômicas **4**

2.1 Redes de Sensores Sem Fio 4

2.1.1 Características e Restrições das RSSFs 6

2.1.2 Classificação de RSSFs 9

2.2 Computação Autonômica 11

2.2.1 Arquitetura de Sistemas Autonômicos 12

2.3 Redes de Sensores Sem Fio Autonômicas 14

2.3.1 Funções de Auto-Gerenciamento em RSSFs 15

2.3.2 Gerenciamento de Falhas em RSSFs 16

2.4 Considerações Finais 18

3 A Biblioteca ITIL **19**

3.1 Suporte a Serviços de TI 20

3.2 Entrega de Serviços de TI 23

3.3 Considerações Finais 25

4 Trabalhos Relacionados **26**

4.1 Auto-Cura em Sistemas Autonômicos 26

4.2 Redes Autonômicas 28
4.3 Redes de Sensores Sem Fio Autonômicas 31
4.4 Gerenciamento Baseado em Políticas para Computação Autonômica 34
4.5 Gerenciamento Tradicional 34
4.5.1 SNMP 37
4.5.2 TMN 38
4.5.3 PBNM 38
4.6 Considerações Finais 39

5 Gerenciamento de Serviços em RSSFs Autonômicas 40

5.1 Recursos Gerenciáveis das RSSFs 41
5.2 Touchpoint para Nós Sensores 44
5.3 Gerentes autonômicos para RSSFs 46
5.3.1 Gerenciamento do Nível do Serviço em RSSFs 51
5.3.2 Gerenciamento de Disponibilidade em RSSFs 52
5.3.3 Gerenciamento de Continuidade em RSSFs 53
5.3.4 Gerenciamento de Capacidade em RSSFs 55
5.3.5 Gerenciamento de Finanças em RSSFs 55
5.3.6 Base de Conhecimento 56
5.4 Considerações Finais 58

6 Projeto de Simulação 59

6.1 Estudo de Caso 60
6.2 Experimentos 65
6.3 Resultados 67
6.3.1 Consumo de Energia 67
6.3.2 Fluxo dos Dados Sensoriados 72

6.3.3 Perda de Mensagens 77

6.4 Considerações Finais 78

7 Conclusão **80**

Referências Bibliográficas **83**

1 Introdução

Uma Rede de Sensores Sem Fio (RSSF) é uma ferramenta de sensoriamento distribuído e que neste trabalho será considerada como um sistema de Tecnologia de Informação (TI), já que trata da utilização de tecnologia para criar, processar, armazenar, transmitir e utilizar informação. Sendo um sistema de TI, as RSSFs estão fundamentadas nos seguintes componentes: hardware e software dos nós sensores, comunicação entre os elementos da rede e gerenciamento dos dados e ou informações produzidas pela rede. Na maioria dos casos, essas redes são formadas por centenas a milhares de elementos (nós sensores) dotados de capacidade de coleta, processamento e disseminação de dados. Os elementos percebem o ambiente, monitoram diferentes parâmetros e coletam dados de acordo com o propósito da aplicação.

O projeto de sistemas autonômicos para RSSFs deve considerar que este tipo de rede possui características que as diferenciam das redes tradicionais, tais como restrições severas de comunicação, energia e processamento, deposição em áreas inóspitas e em ambientes sem intervenção humana. Por estas razões, qualquer operação de hardware ou software deve ser eficiente no que diz respeito ao consumo de energia, incluindo as operações de gerenciamento.

Diante da necessidade de se promover a qualidade do serviço e a produtividade das RSSFs, bem como facilitar sua integração com outros sistemas de TI e de gerenciamento, este trabalho propõe o uso de um conjunto de práticas de gerenciamento da infra-estrutura de TI definido na Biblioteca de Infra-estrutura de Tecnologia da Informação (*Information Technology Infrastructure Library* - ITIL) [OFFICE OF GOVERNMENT COMMERCE, 2002] e do paradigma de computação autonômica [KEPHART; CHESS, 2003] no projeto e desenvolvimento de uma RSSF que implemente o serviço de auto-cura.

A Biblioteca ITIL é um conjunto de práticas que busca promover qualidade dos serviços de tecnologia da informação (ver Capítulo 3). A computação autonômica é uma tecnologia que permite que sistemas se auto-gerenciem e melhorem as suas próprias operações com um mínimo de intervenção humana (ver Seção 2.2).

1.1 Objetivos

O objetivo deste trabalho é propor uma abordagem de auto-gerenciamento para uma RSSF que implemente a função de auto-cura, a fim de manter a disponibilidade de seus serviços de coleta, processamento e disseminação de dados. Na abordagem proposta a função de auto-cura será responsável por detectar, identificar e prevenir falhas, além de propor ajustes para a infra-estrutura da rede, mantendo a disponibilidade dos serviços provisionados. Para alcançar este objetivo são utilizadas práticas de gerenciamento de serviços da Biblioteca ITIL e o paradigma da computação autonômica. Estas tecnologias de gerenciamento de TI são aplicadas às RSSFs considerando que estas redes são sistemas de TI, uma vez que elas produzem, processam, armazenam e transportam dados.

1.2 Contribuições

As principais contribuições apresentadas deste trabalho de dissertação são apresentadas a seguir:

- A construção de uma solução integrada para RSSFs utilizando tecnologias recentes tais como computação autonômica, ITIL, considerando os diferentes serviços que ela provê.
- A construção de uma solução aberta que permite a integração com outras soluções de gerenciamento de outras redes no futuro. Dado que o desenvolvimento de soluções isoladas já é conhecido ser ineficiente no gerenciamento de redes, a ITIL é uma tecnologia que permite especificar serviços e funções de gerenciamento em nível de negócios e que isto é fundamental ao se considerar a integração das RSSF com outras redes e com a Internet.
- A aplicação da computação autonômica e da ITIL como tecnologias delineadoras do que permitirá, no futuro, a integração das soluções individuais para RSSFs que vem sendo propostas. É uma tendência mundial a busca por soluções integrais para RSSF.
- Um estudo de caso para uma aplicação real de monitoração de cargas refrigeradas que implementa estratégias simples e efetivas no aumento da produtividade de uma RSSF mantendo a qualidade do serviço.

Durante o desenvolvimento da dissertação, alguns trabalhos foram aceitos para publicação em periódicos [ASSUNÇÃO; RUIZ; LOUREIRO, 2006; RUIZ et al., 2005b], conferências

internacionais [RUIZ et al., 2005c; RUIZ et al., 2005a], um capítulo de livro [BRAGA et al., 2006] e conferências nacionais [ASSUNÇÃO; RUIZ, 2007; RUIZ et al., 2005d]. Resultados não relacionados diretamente ao tema desta dissertação foram publicados em eventos nacionais [LOPES et al., 2006; ZORKOT; ASSUNÇÃO; RUIZ, 2006].

1.3 Organização

Este trabalho está organizado como a seguir. O Capítulo 2 apresenta as Redes de Sensores Sem Fio Autonômicas que gerenciam a si próprias sem intervenção humana direta. Este capítulo também apresenta os principais serviços de gerenciamento propostos para uma RSSF autonômica. O Capítulo 3 faz uma breve apresentação dos processos de Suporte a Serviços e Entrega de Serviços da Biblioteca ITIL. O Capítulo 4 tem por finalidade apresentar alguns trabalhos relacionados em RSSFs autonômicas. O Capítulo 5 apresenta uma das importantes contribuições deste trabalho, a definição de uma abordagem de auto-gerenciamento de RSSFs, com foco em auto-cura, desenvolvida com base nos conceitos da computação autonômica e da biblioteca ITIL. O Capítulo 6 apresenta um estudo de caso para uma aplicação de monitoração de cargas refrigeradas utilizando a abordagem proposta. Este capítulo também apresenta os experimentos realizados bem como os resultados obtidos. Concluindo o trabalho, o Capítulo 7 apresenta os comentários finais e os trabalhos futuros.

2 *Redes de Sensores Sem Fio Autonômicas*

Este capítulo apresenta os principais conceitos envolvidos neste trabalho: as redes de sensores sem fio e a computação autonômica. Primeiramente, na Seção 2.1, são apresentadas as RSSFs, suas principais características e limitações. Estas redes possuem diversas particularidades que as diferenciam dos outros tipos de rede e por este motivo, soluções de gerenciamento tradicional (ver Seção 4.5), em geral, não são diretamente aplicáveis às RSSFs. Considerando o desafio de se desenvolver sistemas de gerenciamento, eficientes em energia em ambientes onde existem severas restrições de recursos foi proposto em [RUIZ, 2003] que as RSSFs devem ser autonômicas, isto é devem se auto-gerenciar sem intervenção humana.

A Seção 2.2 apresenta o paradigma da Computação Autonômica, as características que definem sistemas autonômicos e uma arquitetura para estes sistemas e a Seção 2.3 apresenta as Redes de Sensores Sem Fio Autonômicas, e seus principais serviços de auto-gerenciamento. Em particular esta seção apresenta as características da auto-cura, um dos focos deste trabalho que trata do auto-gerenciamento de RSSFs capazes de detectar e identificar falhas, e capazes de se recuperar destas falhas a fim de provisionar serviços de qualidade.

2.1 Redes de Sensores Sem Fio

Uma RSSF é uma ferramenta de sensoriamento distribuído de fenômenos, processamento e disseminação de dados coletados e informações processadas para um ou mais observadores.

Em geral, os elementos da rede, chamados nós sensores, são projetados com pequenas dimensões (desde poucos mm^3 até alguns cm^3) e com a expectativa de apresentarem um baixo custo, o que permitiria o uso de centenas a milhares em diferentes aplicações. Os nós sensores são compostos pelas unidades de computação, de comunicação sem fio, de sensoriamento e de energia (ver Figura 2.1). A unidade computacional, composta por processador e memória está relacionada com as atividades de processamento e armazenamento do nó. O processador

utilizado em um nó sensor geralmente opera em baixa freqüência, possui baixo consumo de energia e baixa capacidade de armazenamento (ver Tabela 2.1). A unidade de comunicação corresponde ao sistema de transmissão e recepção, amplificador e antena do nó. Em geral, os nós sensores realizam comunicação por rádio freqüência, existindo também tecnologia de nós sensores que utilizam comunicação óptica ou por infravermelho. O consumo de energia pelo transceptor é dependente da arquitetura do nó sensor - o nó Mica2 da CrossBow [CROSSBOW, 2003], por exemplo, utiliza uma corrente de 10mA para recepção e 27mA para transmissão de dados, enquanto o nó MicaZ, do mesmo fabricante, utiliza 19.7mA para recepção e 17.4mA para transmissão. Tipicamente a transmissão de dados consome mais energia que a sua recepção. A unidade de sensoriamento é composta por um ou mais sensores, dependendo do tipo de aplicação. Por fim, a unidade de energia, composta por algum tipo de fonte de energia como por exemplo baterias ou células solares, é a responsável pelo funcionamento de todos os módulos do nó sensor. A Tabela 2.1 também apresenta a quantidade de energia consumida a cada unidade de tempo para alguns nós sensores em modo de operação ativo.

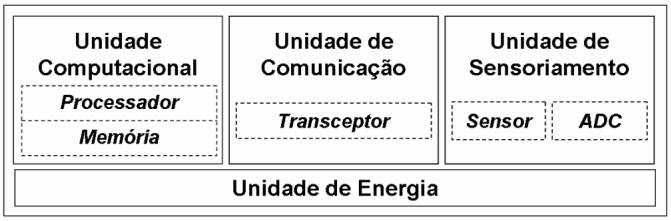


Figura 2.1: Componentes de um nó sensor.

Os nós sensores percebem o ambiente, monitoram diferentes parâmetros e coletam dados de acordo com o propósito da aplicação. Uma RSSF pode monitorar temperatura, umidade, movimento de veículos, iluminação, nível de ruído, aceleração, campo magnético, dentre outros. A escolha do tipo do sensor, bem como da arquitetura do nó, depende do tipo de aplicação em que a RSSF será empregada. A Figura 2.2 apresenta algumas plataformas de nós sensores [HOLLAR, 2000; DELIN et al., 2003; WARNEKE et al., 2001; CROSSBOW, 2003; SENSICAST, 2006; BTnodes, 2006; CROSSBOW, 2003; RABAEY et al., 2002].

A não utilização de cabos de energia e cabos de rede permite que as RSSFs possam ser distribuídas sobre o ambiente sem que haja preocupação com instalação, manutenção ou rompimento da infra-estrutura de sensoriamento. O conceito de sensoriamento associado a conexão sem fio permite que estes nós sejam utilizados em diversas aplicações como por exemplo monitoração ambiental, aplicações médicas, monitoração de áreas de desastres e risco, aplicações militares e aplicações relacionadas a segurança e controle industrial. Como exemplos de aplicações reais em que RSSFs foram utilizadas podemos citar: uma aplicação de monitoração de micro-

Nó Sensor	CPU	Consumo de Energia Máximo	Memória	Largura de Banda
BT Node (2001)	Atmel Mega 128L 8 MHz	285 mW	4 KB SRAM, 128 KB Flash, 4 KB EEPROM	Bluetooth
Imote 1.0 (2003)	ARM 7TDMI 12 MHz	195 mW	64 KB RAM, 512 KB Flash	Bluetooth
Mica2 (2001)	Atmega 128 16 MHz	165 mW	4 KB RAM, 128 KB Flash	250 kbps
MicaZ (2004)	Atmega 128 16 MHz	140.91 mW	4 KB RAM, 128 KB Flash	250 kbps
Rene (1999)	Atmel 8535 8 MHz	60 mW	512 B RAM, 8 KB Flash	10 kbps
Telos (2004)	TI MSP430 8 MHz	41 mW	10 KB RAM, 1 MB Flash	250 kbps
Tmote Sky (2005)	TI MSP430 8 MHz	69 mW	10 KB SRAM, 48 KB Flash	250 kbps
Wins Rockwell (2002)	StrongArM 1100 133 MHz	360 mW	1 MB SRAM, 4 Mb Flash	100 kpbs

Tabela 2.1: Características de nós sensores reais.

clima [MAINWARING et al., 2002], uma aplicação militar de detecção e classificação de alvos [ARORA et al., 2004] e aplicações médicas para monitoração de sinais vitais [LORINCZ et al., 2004; BAIRD et al., 2006].

Espera-se que em pouco tempo as RSSFs se tornem ubíquas e que passem a interagir com o ambiente e outros tipos de rede para permitir monitoração de fenômenos em larga escala. A fim de que isto ocorra, o custo, tamanho e consumo de energia dos nós deve diminuir drasticamente, enquanto a "inteligência"da rede deve ser melhorada. Alguns grupos de pesquisa estão dedicados no desenvolvimento e avanço das RSSFs em diversas áreas como por exemplo: projeto de circuitos RF [BERKELEY WIRELESS RESEARCH CENTER, 2006], projeto de antenas [MENDES et al., 2004], dispositivos *ultra low-power* [HEMPSTEAD et al., 2005], gerenciamento das RSSFs [SENSORNET, 2004] e aplicações utilizando RSSFs [NASA JET PROPULSION LAB, 2001].

2.1.1 Características e Restrições das RSSFs

Aspectos de escalabilidade, tolerância a falhas, topologia da rede, restrições de hardware e consumo de energia, além de impactar o projeto das RSSFs, as diferenciam das outras redes ad hoc [AKYILDIZ et al., 2002b]. A seguir são apresentadas as principais características e restrições das RSSFs.

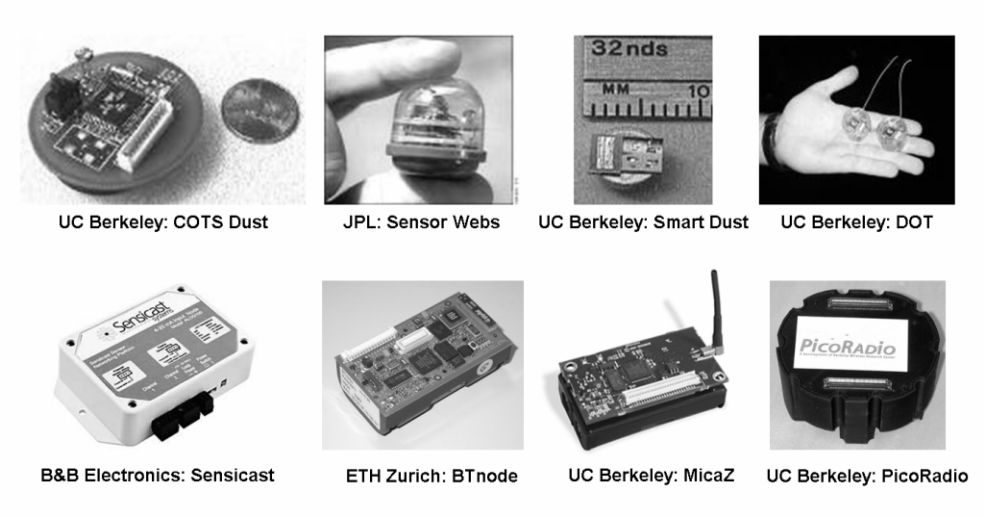


Figura 2.2: Nós sensores de diversas plataformas.

Energia é a um fator crítico. Por ser um dispositivo de pequena dimensão, os nós sensores são equipados com uma fonte de alimentação limitada. Em algumas aplicações, a recarga da bateria pode ser impraticável, e por este motivo, o tempo de vida do nó sensor tem uma dependência muito grande com o tempo de vida da bateria [RUIZ, 2003]. Por exemplo, no caso de uma RSSF com um protocolo *multihop*, se alguns nós falham podem acontecer mudanças significativas na topologia , tornando necessário reorganizar a rede. Para diminuir este tipo de falha as soluções propostas para redes de sensores devem ser eficientes no que diz respeito ao consumo de energia. A eficiência em energia é uma métrica de QoS importante para o gerenciamento das RSSF, já que influencia diretamente o tempo de vida da rede [AKYILDIZ et al., 2002a]. As tarefas de sensoriamento, comunicação e processamento de dados são as responsáveis pelo consumo de energia nos nós, sendo que tipicamente a energia consumida com tarefas de comunicação (transmissão e recepção de dados) é bem maior do que a consumida pelas outras duas tarefas. O consumo com comunicação depende de aspectos como a distância entre os nós envolvidos na recepção e transmissão de dados, potência de transmissão ou recepção, quantidade e freqüência de transmissão ou recepção de dados [SINHA; CHANDRAKASAN, 2001].

Restrições de hardware. O hardware do nó sensor é composto por unidades de sensoriamento, processamento, comunicação, memória e energia. Além disso, podem existir outros componentes adicionais dependendo da aplicação, tais como módulos geradores de energia e módulos para dar suporte a mobilidade. Dois fatores principais restringem a capacidade do hardware do nó sensor: limitação de energia e o tamanho dos nós sensores. Os componentes do nó devem fazer uso dos recursos de energia de maneira otimizada a fim de prolongar o tempo de vida da rede, já que o acesso aos nós sensores para recarga ou troca

da bateria, em grande parte das aplicações, é inviável [RUIZ, 2003].

A unidade de processamento de capacidade limitada geralmente é associada a uma pequena unidade de armazenamento, o que limita o nó sensor a realizar tarefas mais simples, mas não impede a rede de realizar tarefas colaborativas. Já a comunicação por rádio-freqüência tem como limitação a perda do sinal transmitido, que cresce exponencialmente com o aumento da distância entre nós e com a proximidade das antenas do solo. Apesar do avanço no desenvolvimento de processadores e memória com maior capacidade e de tamanho reduzido, estes recursos ainda são escassos, como por exemplo, os recursos do nó sensor MicaZ da Crossbow Technology Inc. [CROSSBOW, 2003] que utiliza duas pilhas AA como fonte de energia e possui um processador Atmel Atmega128 com 128 KB de memória de programa, 4 KB de memória RAM e 512 KB de memória flash. Neste nó a transmissão/recepção de dados a 250 kbps através do transceptor CC2420 da Chipcon (compatível com o IEEE 802.15.4) e a unidade de sensoriamento possui seis canais de entrada, cada um com um conversor analógico-digital (ADC) de 10 bits. Detalhes sobre nós de outras plataformas são apresentados na Tabela 2.1.

Nós sensores tendem a falhar. Em uma RSSF falhas são possíveis e aceitáveis e a rede deve saber lidar com elas de maneira automática e natural [AKYILDIZ et al., 2002a]. Sensores podem falhar por diversos motivos como falta de energia, falta de alcance de comunicação ou até mesmo algum dano físico. Devido às grandes quantidades de nós depositados na região a ser monitorada, a falha de alguns nós não deve atrapalhar o funcionamento do resto da rede e deve ter efeitos mínimos na tarefa para a qual a rede foi projetada. Para isto, a rede deve ser tolerante a falhas, apresentando a habilidade de manter as suas funcionalidades sem interrupção do serviço em caso de falhas dos nós.

Diferentes níveis de tolerância à falha vão possibilitar a existência de diferentes algoritmos de controle da rede, adequando algoritmos de acordo com a aplicação.

Quantidade e densidade de nós sensores. A quantidade de nós sensores depositados em uma área a ser monitorada pode ser da ordem de centenas ou milhares. De acordo com o propósito da aplicação, o número de nós pode alcançar um valor extremo de milhões de nós. Os protocolos, arquiteturas e algoritmos projetados para RSSFs devem levar em conta esta característica, considerando questões de escalabilidade e densidade. De acordo com [BULUSU et al., 2001] a densidade pode ser calculada por:

$$\mu(R) = (N\pi R^2)/A \tag{2.1}$$

onde N é o número de nós sensores em uma região de área A, R é o alcance de transmissão

do rádio e μ(R) representa o número de nós dentro do raio de transmissão de cada nó da região.

Outra maneira de indicar a densidade de nós é considerar o número de nós em uma região. Neste caso, de acordo com cada aplicação, determina-se o valor que corresponde a uma baixa ou alta densidade.

Topologia varia dinamicamente. Realizar a manutenção da topologia de uma RSSF é uma tarefa difícil, já que falhas ocorrem freqüentemente quando os nós saem de serviço por perda do alcance de comunicação e esgotamento de energia [RAJARAMAN, 2002]. Além disso, considerando uma RSSF com alta densidade de nós, problemas como colisão de mensagens e congestionamento de tráfego podem gerar atrasos. Outro fator que impacta a topologia das RSSFs é a forma de deposição dos nós. Se a deposição for planejada e fixa, a probabilidade de ocorrerem falhas é bem menor do que das RSSFs com deposição ad hoc ou que considerem mobilidade dos nós sensores.

Além dos fatores mencionados acima, outros aspectos como identificador global (nós sensores podem não ter uma identificação global, devido a grande quantidade de nós e da sobrecarga que essa identificação causaria) e de tipo de comunicação (tipicamente nas RSSFs a comunicação acontece em broadcast) diferenciam as RSSF das redes ad hoc [AKYILDIZ et al., 2002b].

2.1.2 Classificação de RSSFs

As características específicas das RSSF permitem que elas sejam classificadas segundo aspectos de configuração, sensoriamento, comunicação e processamento [RUIZ, 2003]. Esta classificação depende do objetivo e área de aplicação da RSSF, já que estes dois tem influência sobre fatores como a escolha da arquitetura, quantidade, distribuição, protocolos e serviços dos nós que compõem a rede.

Considerando a configuração, uma RSSF pode ser composta de nós que apresentam o mesmo tipo de hardware (homogênea) ou por nós com diferentes capacidades de hardware (heterogênea). O impacto de utilizar uma rede heterogênea é a realização de tarefas mais ou menos elaboradas dependendo do hardware do nó. Quanto a sua organização, uma RSSF pode ser organizada em grupos (hierárquica) ou não (plana). No caso das RSSFs hierárquicas, cada grupo apresenta pelo menos um líder (*cluster-head*), sendo necessário definir algoritmos de

organização e eleição de líderes. Além disso, uma RSSF pode ser estacionária ou móvel, dependendo se os nós são dotados de mobilidade. A deposição em RSSFs pode ser planejada ou ad hoc, sendo que nesta última os nós podem ser lançados na região a ser monitorada sem uma localização pré-determinada. De acordo com a concentração de nós por unidade de área, uma RSSF pode ser classificada como balanceada, densa ou esparsa.

Considerando a tarefa de sensoriamento, a coleta de dados em RSSFs pode ser contínua, periódica (nós sensores coletam dados em intervalos regulares), dirigida a eventos (os nós sensores coletam dados quando ocorrem eventos de interesse), sob-demanda (os nós sensores coletam dados quando solicitado pelo observador) ou híbrida (coleta contínua, programada, dirigida a eventos ou sob-demanda coexistem). Além disso, dependendo do tipo de sensor pode-se classificar o sensoriamento em homogêneo, se apenas um tipo de sensores é utilizado ou heterogêneo, no caso em que diversos tipos de sensores são acoplados no nó.

Quanto a comunicação, a disseminação dos dados sensoriados por uma RSSF pode ser contínua, programada (nós sensores disseminam dados em intervalos regulares), dirigida a eventos (os nós sensores disseminam dados quando ocorrem eventos de interesse), sob-demanda (os nós sensores disseminam dados quando solicitado pelo observador) ou híbrida (disseminação contínua, programada, dirigida a eventos ou sob-demanda coexistem). Quanto ao tipo de conexão de uma RSSF, esta pode ser simétrica, isto é as conexões existentes entre os nós sensores têm o mesmo alcance, ou assimétrica caso contrário. Dependendo do tipo de transceptor, o tipo de transmissão pode ser simplex (apenas transmite), half-duplex (transmite ou recebe em um determinado instante) ou full-duplex (transmite e recebe simultaneamente).

O processamento em RSSFs envolve procedimentos relacionados à infra-estrutura da rede (algoritmos de controle de acesso ao meio, roteamento, eleição de líderes, descoberta de localização e criptografia) e ao tratamento dos dados sensoriados (fusão de dados, correlação, compressão). Além disso, o processamento pode ser centralizado (os nós enviam dados para um único nó que realiza o todo o processamento dos dados da rede), distribuído (os nós realizam processamento distribuído dos dados coletados) ou híbrido (parte do processamento é realizado de forma distribuída e parte centralizada).

Construir e desenvolver sistemas de gerenciamento, eficientes em energia em ambientes onde existem severas restrições de recursos não é uma tarefa trivial. Considerando essas características, foi proposto em [RUIZ, 2003] que as RSSFs devem ser autonômicas, isto é, devem gerenciar a si próprias sem intervenção humana. Levando isto em consideração, o trabalho proposto nesta dissertação trata do auto-gerenciamento de RSSFs capazes de detectar e identificar

falhas, e capazes de se recuperar destas falhas a fim de provisionar serviços de qualidade.

2.2 Computação Autonômica

A computação autonômica é uma tecnologia que define sistemas que se auto-gerenciam e melhoram as suas próprias operações com um mínimo de intervenção humana. Esta tecnologia lançada pela IBM foi inspirada no sistema nervoso autonômico do corpo humano que age de forma automática e involuntária. Da mesma forma como o corpo gerencia a respiração, a digestão e o sistema imunológico, sistemas de computação autonômica terão a capacidade de gerenciarem, repararem e protegerem a si próprios. Os sistemas autonômicos realizarão manutenção e ajuste de sua operação frente a mudança em componentes, carga de trabalho, demanda e falhas de hardware e software e implementarão funções preditivas e proativas que antecipam condições variantes e problemas.

A necessidade de adoção da computação autonômica cresce, já que cada vez mais os sistemas de TI se tornam mais complexos, mais críticos e mais caros para serem gerenciados. Por esta razão algumas empresas entraram para o mercado das soluções autonômicas, dentre elas a IBM que lançou a computação autonômica [KEPHART; CHESS, 2003; IBM, 2005a], a Microsoft com a proposta de sistemas dinâmicos [Microsoft Corporation, 2005], a Sun com o software de gerenciamento *N1* [SUN MICROSYSTEMS, 2006], e Hewlett-Packard com sua iniciativa *Adaptive Enterprise* [HEWLETT-PACKARD, 2003].

Sistemas autonômicos devem possuir a capacidade de se auto-conhecer, isto é, o sistema deve conhecer seus componentes, estado, capacidade, conexões com outros sistemas e a extensão dos seus recursos. A partir deste conhecimento, o sistema deve ser capaz de se configurar e reconfigurar sob condições variáveis. Além disso, um sistema autonômico deve ser capaz de otimizar seus trabalhos através da monitoração das suas partes constituintes e realização de um ajuste fino do fluxo de trabalho para atingir objetivos pré-determinados. Para isto, o sistema pode antecipar os recursos necessários de maneira ótima enquanto mantém sua complexidade escondida. Além de se reconfigurar e otimizar seu funcionamento, um sistema autonômico deve se recuperar de eventos extraordinários que causem disfunção de seus componentes, através da descoberta de problemas, ou potenciais problemas e criar alternativas de uso de recursos ou reconfiguração do sistema para manter o seu funcionamento. Um sistema autonômico deve se auto-proteger através de tarefas de detecção, identificação e proteção contra ataques a fim de manter segurança e integridade. Por fim, um sistema autonômico deve conhecer o ambiente e o contexto que o envolve e atuar de acordo, a fim de melhor interagir com sistemas vizinhos.

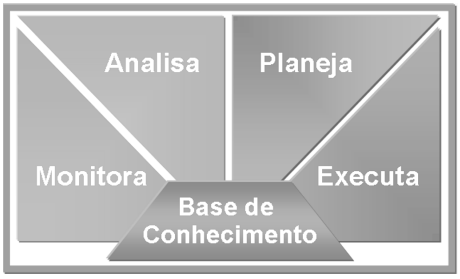


Figura 2.3: Gerente autonômico.

2.2.1 Arquitetura de Sistemas Autonômicos

Um sistema autonômico é composto por elementos autonômicos que se relacionam entre si. Cada um destes elementos possui recursos gerenciáveis, isto é, hardware ou software que constroem a infra-estrutura de TI, e gerentes autonômicos que supervisionam e controlam estes recursos utilizando uma interface padrão. O gerente autonômico (ver Figura 2.3) provê serviços de auto-gerenciamento utilizando funções de monitoração, planejamento, análise e execução (MAPE). Estas funções fazem parte de um ciclo de controle realizado pelo gerente autonômico [IBM, 2005b]:

Monitoração. Esta função envolve a monitoração do sistema (hardware e software) e do ambiente a fim de extrair o comportamento de cada um desses componentes. Esta função coleta detalhes dos recursos gerenciáveis e os correlaciona em sintomas que podem ser analisados. Estes detalhes podem incluir informação de topologia, métricas, configurações, estado, capacidade e *throughput* oferecidos.

Análise. Esta função provê mecanismos para observar e analisar as informações monitoradas a fim de determinar se alterações precisam ser realizadas. A função de análise pode modelar comportamentos complexos para fazer uso de técnicas de predição, permitindo ao gerente autonômico aprender com o ambiente de TI e prever comportamentos futuros.

Planejamento. Esta função cria ou seleciona procedimentos para executar a alteração necessária no recurso gerenciável. Um plano de mudanças, que representa um conjunto de mudanças para o recurso gerenciável, é gerado e passado para a função de execução.

Execução. Esta função provê o mecanismo para agendar e realizar as mudanças necessárias ao sistema. Ela é a responsável por realizar o procedimento gerado pela função de planejamento e por atualizar o conhecimento que é usado pelo gerente autonômico.

Estas quatro funções consomem e produzem conhecimento. A base de conhecimento é alimentada com informações sobre o sistema a medida que o gerente autonômico aprende sobre

as características dos recursos gerenciáveis. O conhecimento é compartilhado por estas quatro funções, levando o sistema a tomar melhores decisões.

A IBM lançou uma arquitetura de sistema autonômico [IBM, 2005b], organizada em camadas e blocos como apresentado na Figura 2.4. A camada inferior contém os componentes do sistema, ou recursos gerenciáveis, que constituem a infra-estrutura de TI (hardware ou software). A próxima camada incorpora uma interface padronizada de gerenciamento para acesso e controle dos recursos gerenciáveis, disponibilizada através de *touchpoints*. A terceira camada é composta por gerentes autonômicos que trabalham com os recursos gerenciáveis através dos *touchpoints* a fim de executar tarefas de auto-gerenciamento. A quarta camada contém gerentes autonômicos que regem outros gerentes autonômicos. Esta camada coordena os gerentes tornando possível a otimização de utilização de recursos em um conjunto de recursos, baseado em políticas. A última camada provê uma interface de gerenciamento do sistema para um profissional de TI, permitindo a gerência de soluções ao invés de gerência de componentes.

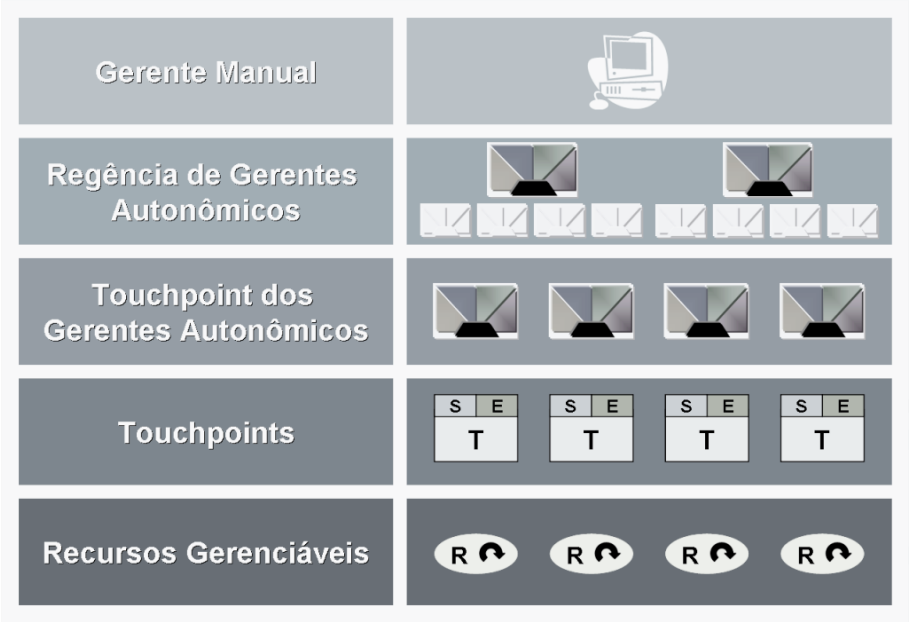


Figura 2.4: Arquitetura de um sistema autonômico.

Os cinco blocos que compõem a arquitetura apresentada na Figura 2.4 são descritos a seguir:

Gerente autonômico: um componente que gerencia outros componentes de software ou hardware usando um ciclo de controle, que inclui as funções de monitoração, análise, planejamento e execução.

Base de conhecimento: a implementação de um dicionário, banco de dados ou outro tipo de repositório que provê acesso ao conhecimento de acordo com as interfaces previstas na arquite-

tura.

Touchpoint: uma interface de padrão de gerenciamento para acesso e controle de recursos gerenciáveis. Um *touchpoint* implementa o comportamento sensor e atuador para o elemento gerenciável e mapeia as interfaces sensoras e atuadoras nas interfaces reais.

Gerente Manual: uma implementação de uma interface de usuário que permite que um administrador realize algumas funções de gerenciamento manualmente.

Enterprise service bus: um implementação que permite a integração dos outros blocos que compõem a arquitetura direcionando as interações entre estes blocos.

Utilizando esta arquitetura, as capacidades de auto-gerenciamento em um sistema são realizadas tomando ações apropriadas de acordo com uma ou mais situações percebidas do ambiente através de interfaces padronizadas e abertas.

2.3 Redes de Sensores Sem Fio Autonômicas

A computação autonômica é geralmente apresentada no contexto dos grandes sistemas computacionais, tipicamente executando software complexo em plataformas de hardware com poucas restrições de recursos, mas pode ser estendida para sistemas distribuídos e com limitações severas de energia, memória e processamento. Este é o caso das Redes de Sensores Sem Fio Autonômicas. Nestas redes as tarefas de gerenciamento dos elementos autonômicos são tipicamente menos complexas devido à restrições dos recursos de energia, memória, processamento, armazenamento e comunicação. Em geral, o comportamento dos gerentes autonômicos é regido por um conjunto de políticas e regras.

Uma RSSF autonômica é uma rede composta por elementos sensores autonômicos que se relacionam entre si [BRAGA, 2006]. Cada um desses elementos possui pelo menos um gerente autonômico que se supervisiona e controla os recursos gerenciáveis dos nós sensores, a saber: sensores, memória, processador, bateria e transceptor. É possível monitorar e configurar parâmetros que controlam o funcionamento destes recursos, sendo esta a principal tarefa dos gerentes autonômicos das RSSFs. Podem ser configurados, por exemplo, a potência de transmissão e recepção do transceptor, intervalos de sensoriamento e disseminação de dados coletados e modo de atividade do processador e do rádio (ativo, *idle*, *sleep*). Os gerentes autonômicos para RSSFs devem ser capazes de monitorar estes parâmetros e controlar os nós sensores (utilizando as funções de monitoração, análise, planejamento e execução) de forma a manter a qualidade do serviço provisionado.

Os gerentes de uma RSSF autonômica são responsáveis por configurar e reconfigurar os recursos gerenciáveis da rede, realizando ajustes dinâmicos para melhor lidar com o ambiente ou melhorar sua produtividade e a utilização de seus recursos [IBM, 2005b]. A reconfiguração pode acontecer de forma proativa ou reativa: a RSSF se reconfigura tentando otimizar seu funcionamento (proativa) ou a RSSF se reconfigura tentando se recuperar de problemas (reativa). Em ambas situações o gerente autonômico deve implementar uma função de coleta de detalhes necessários sobre a RSSF; uma função para analisar estes detalhes a fim de determinar se alguma mudança deve ser realizada; uma função para criar um plano, ou seqüência de ações, que especificam as mudanças necessárias; e uma função para executar este plano. No caso da reconfiguração proativa é necessário monitorar os recursos do nó, determinar qual melhor forma de utilização destes recursos e definir mudanças e pequenos ajustes baseados em objetivos pré-determinados (otimização). No caso da reconfiguração de forma reativa, o objetivo é identificar problemas e determinar formas alternativas de utilizar recursos e reconfigurar o sistema para mantê-lo funcionando. Para realizar estas tarefas a RSSF precisa conhecer seu ambiente e seu contexto de atividades, e principalmente se conhecer (seus componentes, limites operacionais, comportamento).

2.3.1 Funções de Auto-Gerenciamento em RSSFs

O objetivo do gerenciamento das RSSFs é promover a produtividade dos recursos com a finalidade de realizar os objetivos propostos pela aplicação e com qualidade de serviço. As tarefas de gerenciamento de RSSFs autonômicas devem considerar os seguinte aspectos [RUIZ, 2003]:

- *Auto-cura*. É o serviço de gerenciamento responsável por descobrir, diagnosticar e reagir a falhas da rede. Os componentes detectam operações impróprias e falhas e iniciam ações corretivas baseados em políticas a fim de recuperar a rede ou um nó. A recuperação automática de problemas melhora a disponibilidade do serviço.

- *Auto-otimização*. É o serviço de gerenciamento que maximiza a alocação e utilização de recursos, e garante qualidade de serviço ótima, baseado em políticas. A automação de tarefas complexas e o ajuste de componentes em resposta às cargas de trabalho variáveis permite a entrega de um serviço de alto nível.

- *Auto-configuração*. É o serviço de gerenciamento que muda parâmetros de configuração para adaptarem-se dinamicamente às condições variáveis e estados da rede. Este serviço se configura e reconfigura sob condições variáveis e até imprevisíveis. A configuração da rede deve ocorrer automaticamente, assim como ajustes dinâmicos à configuração corrente para melhor tratar

mudanças no ambiente.

- *Auto-proteção*. É o serviço de gerenciamento que detecta e protege a entidade contra ameaças (internas ou externas, acidentais ou maliciosas). Quando um ataque acontece, estes serviços executam rotinas de detecção a fim de alcançar segurança.

- *Auto-serviço*. É o serviço de gerenciamento que permite a provisão de serviços de sensoriamento, processamento e disseminação, antecipando recursos necessários e ao mesmo tempo mantendo a complexidade escondida, a fim de aproximar o negócio da aplicação e objetivos de serviço.

- *Auto-consciência*. É o serviço de gerenciamento que permite a entidade conhecer seu ambiente e seu contexto de atividades e atuar de acordo. Ele encontra e gera regras para melhor interagir com entidades vizinhas.

- *Auto-conhecimento*. É o serviço de gerenciamento que qualifica uma entidade que "conhece a si própria". Por exemplo, uma entidade que se governa deve conhecer quais são seus componentes, estado atual, capacidade e todas conexões com outras entidades. Ela precisa conhecer a extensão que seus recursos podem ser emprestados e compartilhados.

- *Auto-manutenção.* É o serviço que permite que uma entidade monitore seus componentes e faça pequenos ajustes de seus componentes para alcançar objetivos pré-determinados.

2.3.2 Gerenciamento de Falhas em RSSFs

As RSSFs devem ser robustas e sobreviver a despeito da ocorrência de falhas nos nós individuais, na rede ou falhas que ocasionam conectividade intermitente. Falhas não são exceções em RSSFs e acontecem como uma determinada freqüência. Esta é uma das razões que tornam o gerenciamento de RSSFs diferente do gerenciamento das redes tradicionais. Falhas acontecem freqüentemente devido a esgotamento de energia, interrupção de conectividade, variações ambientais, danos físicos, perda do alcance de comunicação, entre outros. As RSSFs devem ser tolerantes a falhas e robustas, buscando manter seu funcionamento independente de falhas individuais dos nós, da rede ou até dos serviços provisionados. O gerenciamento de falhas deve prover funcionalidades básicas tais como auto-cura, auto-diagnóstico e auto-manutenção.

Neste trabalho são considerados três tipos de falhas em RSSF: falhas nos nós sensores, falhas na rede e falhas no serviço provisionado.

Falhas em nível de elemento de rede. Este tipo de falha atinge o nó sensor e seus componentes. Pode ser causada, por exemplo, por fatores como esgotamento de energia e danos

físicos. Este tipo de falha depende do hardware do nó sensor e do software que é executado em sua unidade lógica.

Falhas em nível de rede. No caso das falhas que se estendem por toda a rede a sua ocorrência deve-se a problemas de comunicação, como por exemplo a perda de conectividade entre os nós, colisão resultando em perda de pacotes e congestionamento causando atraso na entrega de pacotes.

Falhas em nível de serviço. Este tipo de falha acontece quando o nível do serviço provido é insuficiente ou em caso de indisponibilidade do serviço.

Em particular, a detecção de falhas em nível de serviço, depende dos objetivos e o nível de serviço a ser prestado, que podem ser definidos em um contrato de Acordo do Nível de Serviço (*Service Level Agreement* - SLA). Em geral, o conteúdo do SLA varia de acordo com a natureza dos serviços a serem prestados e das características da rede, isto é, deve levar em conta aspectos de configuração, comunicação, performance e segurança, dentre outros. Para isto é necessário conhecer o tipo de infra-estrutura de rede a ser utilizado e entender os componentes físicos desta rede para que seja possível definir níveis de performance esperados. Além da performance, o SLA deve tratar de questões de disponibilidade da rede, *throughput*, perda de dados, latência e segurança.

No caso específico das RSSF, um SLA se baseia nos requisitos de QoS que estão diretamente relacionados a uma aplicação específica, tais como área de cobertura, número de nós ativos, tipo de sensoriamento e disseminação, e os requisitos de QoS relacionados à rede que serve a aplicação, tais como largura de banda, perda de pacotes e *throughput*. Além disso, a escolha da arquitetura dos nós sensores (CPU, memória disponível, bateria, interface de rede, dentre outros), infra-estrutura da rede e algoritmos tem impacto sobre os requisitos de QoS.

A fim de garantir produtividade das RSSFs e garantir a qualidade do serviço provido é preciso considerar na escolha da solução de gerenciamento as características específicas das RSSFs tais como restrições severas de comunicação, energia e processamento, deposição em áreas inóspitas e em ambientes sem intervenção humana. As operações de gerenciamento nas RSSFs devem ser desacompanhadas, e no caso específico das operações para detecção e recuperação de falhas, as RSSFs devem implementar serviços de auto-diagnóstico, auto-cura e auto-manutenção.

Este trabalho utiliza alguns dos conceitos da biblioteca ITIL (ver Capítulo 3) para empregar serviços de auto-gerenciamento em RSSFs autonômicas. Para isto, as funções de monitoração,

análise, planejamento e execução são implementadas para os gerentes autonômicos, definidas sob o paradigma da biblioteca ITIL no Capítulo 5.

2.4 Considerações Finais

Este capítulo apresentou os principais conceitos relacionados às RSSFs autonômicas. Primeiramente foram discutidas as principais características e restrições das RSSFs, seus componentes e uma classificação para este tipo de rede. Em seguida foi apresentada a arquitetura de sistemas autonômicos - seus elementos e as funções do gerente autonômico. Após introduzir estes conceitos, este capítulo apresenta as RSSFs autonômicas e suas principais funções de gerenciamento. Em especial é discutido o gerenciamento de falhas em RSSFs, um dos focos deste trabalho. O capítulo a seguir descreve a biblioteca ITIL, que neste trabalho é considerada uma tecnologia que permite a integração dos diferentes serviços de uma RSSF.

3 A Biblioteca ITIL

Gerenciar serviços de TI é estar ciente do que é a infra-estrutura de TI e realizar a supervisão e controle dos componentes desta infra-estrutura, de forma a provisionar serviços de qualidade de modo eficaz e eficiente. O gerenciamento de serviços de TI (*IT Service Management* - ITSM) inclui a adoção de um conjunto de práticas bem definidas para facilitar a entrega de serviços de TI de qualidade. O conceito de gerenciamento de serviços pode ser alcançado adotando as práticas propostas na Biblioteca de Infra-estrutura de TI (*IT Infrastructure Library* - ITIL).

A biblioteca ITIL é o modelo de referência para gerenciamento de infra-estrutura de Tecnologia da Informação (TI) mais aceito mundialmente [OFFICE OF GOVERNMENT COMMERCE, 2002]. Ela provê um conjunto de práticas para o gerenciamento de serviços de TI, a fim de alcançar eficiência e eficácia no uso de sistemas de informação [HOCHSTEIN; ZARNEKOW; BRENNER, 2005].

A biblioteca ITIL está documentada em aproximadamente quarenta livros que descrevem os principais processos de gerenciamento de serviços de TI, isto é, um conjunto de ações e atividades definidos para atingir o objetivo de gerenciar a infra-estrutura de TI . As duas áreas mais importantes do ITIL são o Suporte a Serviços de TI [OFFICE OF GOVERNMENT COMMERCE, 2000] e a Entrega de Serviços de TI [OFFICE OF GOVERNMENT COMMERCE, 2001]. A Figura 3.1 apresenta os principais serviços de gerenciamento do suporte e entrega de serviços de TI. Juntas, estas duas áreas são responsáveis pela provisão e gerenciamento de serviços de TI.

Este trabalho propõe que a ITIL possa ser empregada como uma tecnologia que permita a integração dos diferentes serviços de uma RSSF, bem como a integração do gerenciamento desta rede com outras, como é de se esperar para o futuro. A motivação para utilizar algumas das práticas da ITIL no auto-gerenciamento das RSSFs a fim de integrar serviços e integrar o gerenciamento de diferentes redes, vem da utilização de práticas abertas e de amplo uso pela comunidade de TI. Mesmo considerando as severas restrições que as RSSFs apresentam, a utilização da ITIL neste trabalho é feita a partir de estratégias simples, mas que efetivamente

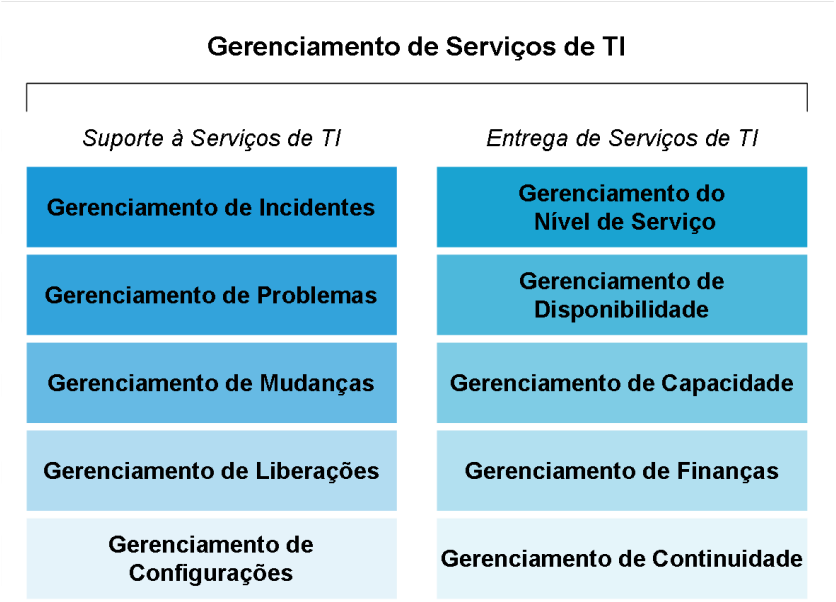


Figura 3.1: Gerenciamento de Serviços de TI.

aumentam sua produtividade mantendo a qualidade do serviço.

3.1 Suporte a Serviços de TI

O Suporte a Serviços é a área que permite que os serviços de TI sejam provisionados de forma eficaz, estável e flexível. Está área de gerenciamento trata da identificação e registro da configuração da infra-estrutura de TI, e do controle de incidentes, problemas e mudanças. O suporte a serviços inclui o Gerenciamento de Incidentes, Gerenciamento de Problemas, Gerenciamento de Mudanças, Gerenciamento de Liberações e Gerenciamento de Configuração. A Figura 3.2 apresenta a inter-relação entre estes processos que são descritos a seguir.

Gerenciamento de Configurações

O gerenciamento de configurações tem por objetivo prover informações sobre a infra-estrutura de TI e permitir controle da infra-estrutura a partir da monitoração e manutenção de informações sobre todos os recursos necessários para entregar serviços, estado e histórico dos itens de configuração e o relacionamento entre itens de configuração. As principais tarefas desse processo incluem identificação e nomeação dos itens de configuração, gerenciamento da informação, verificação, controle e contabilização de estado.

Além disso o gerenciamento de configurações é responsável por manter o Banco de Dados de Gerenciamento de Configuração (*Configuration Management Database - CMDB*), um banco de dados com detalhes sobre os itens de configuração (IC's) e detalhes sobre o relacionamento

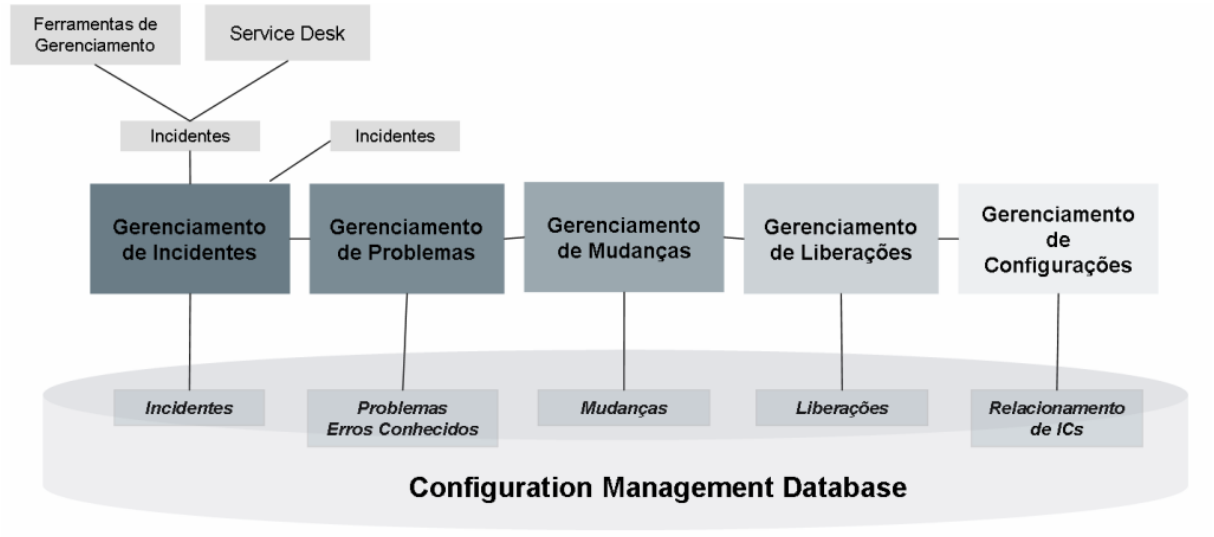


Figura 3.2: Gerenciamento de entrega suporte a serviços de TI.

entre os itens de configuração (ver Figura 3.2).

Gerenciamento de Incidentes

O gerenciamento de incidentes tem por finalidade restaurar o funcionamento dos serviços o mais rápido possível. Além disso, minimizar o impacto desses incidentes nas operações do negócio e garantir que o melhor nível de qualidade de serviços disponibilidade sejam mantidos, de acordo com o Acordo de Nível de Serviço (Service Level Agreement - SLA). Um incidente é um evento que não faz parte da operação padrão do serviço e que pode causar uma interrupção do serviço ou redução da qualidade do serviço. Suas principais tarefas são a identificação e registro de incidentes, classificação de incidentes e suporte inicial, controle do incidente, investigação, diagnose e resolução do incidente e recuperação do sistema.

Gerenciamento de Problemas

O gerenciamento de problemas objetiva estabilizar os serviços de TI através da minimização das conseqüências de incidentes, remoção da causa dos incidentes, prevenção de incidentes e problemas e prevenção da recorrência de incidentes. As principais tarefas desta disciplina incluem controle de problemas (identificação, classificação, alocação de recursos, investigação e diagnose, estabelecimento de erro conhecido), controle de erros (identificação, registro, validação e registro de erros), levantamento de Requisições de Mudanças (*Request for Change* - RfC), prevenção pró-ativa, identificação de tendências e gerenciamento da informação.

A partir dos detalhes dos incidentes, bem como detalhes de configuração e soluções provisórias já definidas, o gerenciamento de problemas definirá erros conhecidos (*know errors*), RfCs, atualizará os registros de problemas com soluções temporárias ou permanentes e respon-

derá ao gerenciamento de incidentes com o plano a ser seguido para solucionar o incidente levantado.

Gerenciamento de Mudanças

O gerenciamento de mudanças tem por finalidade coordenar e planejar mudanças de forma eficiente, a um baixo custo e com o mínimo risco para a infra-estrutura existente e para a nova infra-estrutura. As principais tarefas incluem filtragem de mudanças, gerenciamento do processo de mudança, gerenciamento de mudanças, avaliação das mudanças por uma equipe (*Change Advisory Board - CAB*) e gerenciamento de informações.

Através das RfCs e consultas ao CMDB o gerenciamento de mudanças irá definir o impacto e a prioridade das mudanças, e determinará a adição, modificação ou remoção de hardware, software, aplicações e sistemas, entre outros.

Gerenciamento de Liberações

O gerenciamento de liberações tem por objetivo implementar as mudanças propostas, garantindo que apenas versões autorizadas, testadas e corretas de software e hardware sejam utilizados nas mudanças. Suas principais tarefas são definir políticas de liberações, controlar a biblioteca de software definitivo (*Definitive Software Library - DSL*) e o armazenamento de hardware definitivo (*Definitive Hardware Storage - DHS*). O gerenciamento de liberações é responsável por checar e testar as versões de software e hardware em um ambiente controlado de testes, e armazenar na DSL e DHS e monitorar o processo de provisionamento de hardware e software.

O trabalho proposto nesta dissertação utiliza os processos do Suporte a Serviços para realizar a auto-cura em uma RSSF. São atribuídas as atividades de cada um dos processos de gerenciamento do suporte a serviço às funções realizadas pelo laço de controle MAPE (monitoração, análise, planejamento e execução) do gerente autonômico (ver Seção 2.2.1). A função de monitoração fica responsável pelas atividades de detecção de incidentes na RSSF; a função de análise realiza a detecção de problemas; mudanças à infra-estrutura da RSSF são propostas pela função de planejamento; e por fim, as alterações propostas são realizadas pela função de execução. O Capítulo 5, que apresenta a solução de auto-gerenciamento proposta neste trabalho, descreve com detalhes as tarefas atribuídas a cada função do MAPE relacionadas ao suporte a serviços de TI.

3.2 Entrega de Serviços de TI

Entrega de Serviços é a área de gerenciamento de serviços de TI que trata de como provisionar serviços de qualidade, tal como definido no SLA. A Entrega de Serviços de TI inclui o Gerenciamento de Disponibilidade, Gerenciamento de Continuidade de Serviços de TI, Gerenciamento de Capacidade, Gerenciamento de Finanças e Gerenciamento de Nível de Serviços. A Figura 3.3 apresenta a inter-relação entre estes processos, que são descritos a seguir.

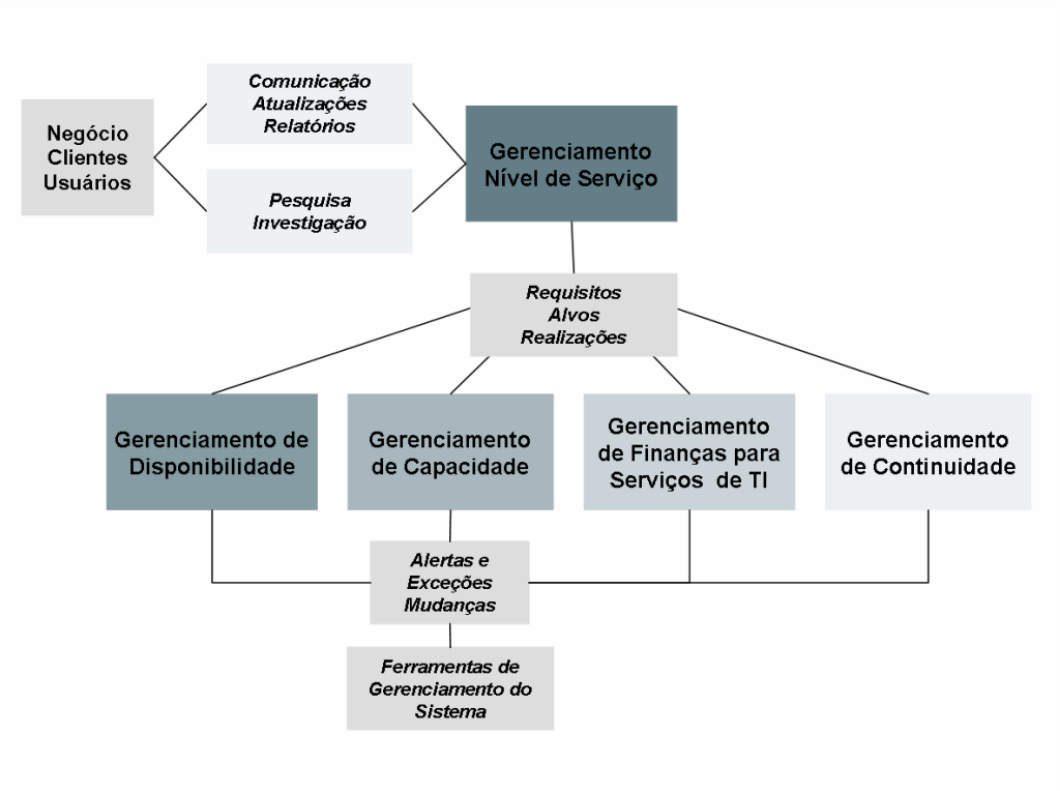


Figura 3.3: Gerenciamento de entrega de serviços de TI.

Gerenciamento de Disponibilidade

O gerenciamento de disponibilidade tem por objetivo predizer, planejar e gerenciar a disponibilidade dos serviços, considerando aspectos de confiabilidade, manutenibilidade, elasticidade e redundância. Este objetivo é alcançado através da determinação de requisitos de disponibilidade do negócio adequação destes requisitos à capacidade da infra-estrutura de TI. A medição e monitoração da disponibilidade dos serviços de TI é a atividade que garante que os níveis de disponibilidade são atendidos. O gerenciamento de disponibilidade deve procurar otimizar a disponibilidade da infra-estrutura de TI e o suporte da organização a fim de prover melhoras na entrega de serviços ao negócio e à usuários.

Gerenciamento de Continuidade de Serviços de TI

O gerenciamento de continuidade dos serviços de TI trata do planejamento de itens de configuração alternativos para recuperação de problemas. As atividades de análise de risco, opções de pesquisa, alternativas de planejamento e documentação fazem parte do gerenciamento de continuidade. A partir destas atividades são desenvolvidos planos de redução de riscos e recuperação.

Gerenciamento de Capacidade

O Gerenciamento de Capacidade tem por objetivo identificar e especificar a demanda e necessidades de clientes, traduzir estas necessidades em recursos necessários e garantir a performance dos serviços. Ele permite que uma organização gerencie recursos em tempos de crise e saiba antecipar a necessidade de capacidade adicional. O processo descreve os procedimentos necessários para o planejamento, a implementação e a execução desse processo.

Gerenciamento Financeiro para Serviços de TI

O gerenciamento financeiro envolve a identificação, cálculo e gerenciamento do custo da entrega dos serviços de TI. Este processo trata da administração dos recursos monetários da organização e dá suporte à empresa no planejamento e execução dos objetivos de negócios. Dentro de uma organização de TI este processo é visível em três áreas principais: elaboração do orçamento, contabilidade e cobrança de serviços de TI.

Gerenciamento do Nível de Serviços

O objetivo do processo de Gerenciamento do Nível de Serviços é manter e melhorar a qualidade dos serviços de TI. Isso ocorre através de um ciclo constante de ajustes, monitoramentos e relatórios de resultados de serviços e TI. O processo também permite que se desenvolva um relacionamento mais forte entre o pessoal de TI e seus clientes.

Os processos da área de entrega de serviços de TI são utilizados nesta dissertação na definição das responsabilidades de gerentes autonômicos para RSSFs. O comportamento destes gerentes deverá considerar aspectos de gerenciamento de disponibilidade, capacidade, nível de serviço e continuidade.

As disciplinas do suporte a serviço e entrega de serviços interagem para garantir que a infraestrutura da RSSF entregue um serviço de alto nível. No próximo capítulo, o serviço de autocura é definido utilizando conceitos de gerenciamento de serviços da biblioteca ITIL e modelado como tarefas de monitoração, análise, planejamento e execução de gerentes autonômicos.

3.3 Considerações Finais

Este capítulo apresentou a biblioteca ITIL, um conjunto de práticas bem definidas para facilitar a entrega de serviços de TI de qualidade. Foram descritas suas principais áreas: suporte à serviços e entrega de serviços. Também foi apresentada a motivação para utilizar a ITIL no auto-gerenciamento das RSSFs: a integração de serviços e do gerenciamento de RSSFs utilizando práticas abertas e de amplo uso pela comunidade de TI. O Capítulo 5 apresenta a descrição de como conceitos de suporte e entrega de serviços de TI podem ser aplicados as RSSFs autonômicas.

4 Trabalhos Relacionados

É notável o progresso da área de RSSFs, contudo, do ponto de vista científico, essas redes apresentam uma grande variedade de novos problemas ainda não estudados ou ainda incipientes. Este é o caso quando se considera as RSSFs como sistemas de TI, ou mesmo, quando se pretende definir essas redes como sistemas autonômicos. Este trabalho procura ser uma contribuição para área quando lida com o desafio de aplicar as práticas da biblioteca ITIL e de computação autonômica em RSSFs. Como não foram encontrados na literatura trabalhos relacionados diretamente ao uso de ITIL em RSSFs, este capítulo apresenta alguns trabalhos iniciais que vêm sendo propostos para sistemas autonômicos e redes autonômicas. A Seção 4.1 apresenta alguns trabalhos que tratam de aspectos de auto-cura em sistemas autonômicos. A Seção 4.2 descreve trabalhos em redes autonômicas tratando especialmente de redes auto-configuráveis, auto-organizáveis e auto-reparáveis. A Seção 4.3 apresenta trabalhos que incluem aspectos de auto-gerenciamento em RSSFs, e que demonstram a viabilidade da utilização de computação autonômica em RSSFs. A Seção 4.4 apresenta a PMAC, uma arquitetura autonômica de gerenciamento de redes baseada em políticas. O gerenciamento baseado em políticas é importante no contexto das redes autonômicas, já que o funcionamento dos serviços é regido por políticas. Por fim, a Seção 4.5 apresenta uma descrição do gerenciamento tradicional para redes de telecomunicações.

4.1 Auto-Cura em Sistemas Autonômicos

Recentemente a IBM lançou o *Toolkit* de Computação Autonômica [IBM, 2006], que é uma coleção de componentes, ferramentas, relatórios e documentação projetados para o desenvolvimento de comportamento autonômico em sistemas de TI. No que diz respeito a determinação de problemas, considera-se sistemas que implementem a funcionalidade de auto-cura. Nestes sistemas, os gerentes autonômicos iniciam ações corretivas em resposta a problemas que venham a afetar sua operação normal. Estas ações podem ser definidas como políticas de negócio, mas o problema deve ser identificado inicialmente a partir de uma base de dados. Como o

conhecimento fornecido ao gerente pode estar em formatos incompatíveis, a arquitetura de determinação de problemas padronizou uma Base Comum de Eventos (*Common Base Event*). São oferecidas duas implementações do gerente autonômico: o "*Log and Trace Analyzer*"(LTA) e a "*Autonomic Management Engine* (AME). O gerente autonômico LTA é capaz de analisar dados recebidos no formato da Base Comum de Eventos, a fim de facilitar a automação de correlação de eventos e ajuda na depuração do problema. O gerente autonômico AME é capaz de monitorar recursos de um sistema, enviar eventos agregados e executar ações corretivas para problemas. Este gerente provê serviços de instalação, ativação e terminação do uso dos recursos, e pesquisa do estado do recurso. Os modelos de recursos contém métricas específicas, eventos, limites e parâmetros, que são usados para determinar se os recursos estão sendo utilizados de acordo com as especificações e atuam na correção de falhas. Esta coleção de componentes foi projetada para sistemas grandes e não se aplica diretamente às RSSFs.

O uso da computação autonômica para análise de sintomas é descrito em [PERAZOLO, 2005a]. Este trabalho define um modelo de sintomas e seus componentes (artefatos e relacionamentos), além de apresentar os papéis de um sintoma na arquitetura de computação autonômica. São abordadas as questões de representação e identificação de sintomas e as vantagens para a adoção de uma representação padronizada e da análise de sintomas como parte da estratégia de sistemas de gerenciamento, quais sejam, o desenvolvimento de um ambiente comum em que sintomas possam ser reutilizados e expandidos de forma padronizada e a facilidade de interoperação entre gerentes autonômicos. Em um segundo trabalho [PERAZOLO, 2005b] o autor apresenta a descrição e definição de sintomas, além de recomendar ações corretivas no caso de ocorrência de sintomas em diversos cenários de TI, tais como segurança, suporte a serviços, disponibilidade de serviços e continuidade de serviços de TI. De acordo com o autor, a análise de sintomas ajuda a definir um conjunto de combinações de eventos e estados que um sistema pode assumir. O conhecimento e o reuso de padrões de análise assumem um papel importante na diminuição de tempo e recursos necessários para disponibilizar funcionalidades autonômicas.

O projeto Unity [CHESS et al., 2004] trata de um protótipo de sistema autonômico para demonstração e validação de aspectos de auto-gerenciamento em sistemas de computação. O trabalho trata de como melhorar aspectos de auto-configuração e auto-otimização incorporando a noção de disponibilidade de componentes nas políticas e modelos do sistema. O sistema inicialmente se configura utilizando o mínimo de informação humana explícita e durante sua operação, ele realoca e reconfigura seus recursos a fim de otimizar seu comportamento de acordo com políticas. Um componente do sistema (repositório de políticas) é implementado como um grupo de serviços sincronizados que realiza auto-cura; outros componentes do sistema, garantem que se um membro do grupo falha, outra instância do serviço toma seu lugar e

entra em serviço. Os quatro principais aspectos examinados no protótipo são: a arquitetura do sistema, o papel de funções de utilidade na tomada de decisões dentro do sistema, a maneira que o sistema utiliza objetivos para dirigir a sua auto-configuração e padrões de projeto que permitam embutir auto-cura no sistema.

O projeto de Computação Orientada a Recuperação (*Recovery Oriented Computing* - ROC) [PATTERSON et al., 2002] considera falhas de hardware, software e humanas como fatos normais e inevitáveis à operação de sistemas computacionais, e não como problemas para serem resolvidos. A recuperação pode ser realizada de diversas maneiras, desde mecanismos simples como reiniciar hardware ou software, até esquemas complexos como contenção de falhas combinada com redundância de dados e hardware, ou recuperação completa do estado do sistema através de *backups* e *logs*. Concentrando esforços em reduzir o Tempo Médio de Reparo (*Mean Time to Repair*) ao invés do Tempo Médio de Falhas (*Mean Time to Failure*), a ROC reduz o tempo de recuperação e oferece maior disponibilidade. Para alcançar a ROC são utilizadas algumas técnicas, como isolamento e redundância, suporte a operações de reversão (*undo operations*), suporte ao diagnóstico integrado, mecanismos de verificação e recuperação *online*, projeto para alta modularidade e capacidade de mensurar e reiniciar o sistema, e por fim avaliação da disponibilidade e confiabilidade do sistema utilizando *benchmarks*.

Os trabalhos apresentados nesta seção foram desenvolvidos considerando sistemas de TI complexos, com capacidade de processamento e de armazenamento superior a das RSSFs. Por este motivo, não é viável o uso das ferramentas propostas nas RSSFs. Além disso, as tecnologias utilizadas no desenvolvimento desses projetos tais como Java e WebServices, ainda não são suportadas com as RSSFs existentes.

4.2 Redes Autonômicas

A convergência de infra-estruturas de redes e serviços tem mudado a visão tradicional das redes, de simples nós homogêneos interconectados e administrados manualmente para uma infra-estrutura complexa que engloba diferentes tecnologias, como por exemplo redes cabeada ou sem fio, móvel ou fixa, estática ou ad hoc; nós com diferentes recursos, tamanhos, capacidades e restrições; diversos tipos de serviço; e objetivos e interesses que “competem”entre si. Para lidar com o desafio de gerenciar as novas infra-estruturas de rede, foram propostas as “Redes Autonômicas”, uma instância da computação autonômica. O objetivo é criar redes que se auto-gerenciem, a fim de afastar as atividades de operação dos administradores de redes, e atribuir tarefas de definição de políticas de alto-nível que regem o funcionamento dos serviços.

A seguir são apresentados soluções autonômicas para redes tradicionais, e que, por não considerar as restrições e características das RSSFs, podem não ser adequadas para este tipo de rede.

A arquitetura e conceitos definidos para sistemas autonômicos aplicados às redes devem permitir a formação de redes flexíveis e dinâmicas em larga escala, em que as funcionalidades de cada nó constituinte da rede também colabore nas tarefas de gerenciamento. Para isto é necessário que o controle das funções e operações dos nós sejam realizadas automaticamente (auto-controle); além disso é necessário que a rede seja capaz de alterar sua operação (configuração, estados e funções) para cooperar com variações do contexto em que ela se encontra; e por fim as redes devem ser conscientes do seu contexto de operação e do seu estado interno para controlar sua operação de acordo com um objetivo. Considerando estas características, o trabalho [SCHMID; SIFALAKIS; HUTCHISON, 2006] propõe que as redes autonômicas sejam automáticas, adaptativas e auto-conscientes. Segundo o autor estas três características são essenciais aos sistemas autonômicos. O autor também apresenta três formas de testar se as funcionalidades autonômicas estão atuando corretamente: teste de cada uma das funções do laço de controle MAPE (ver Seção 2.2.1) separadamente; teste de caixa preta que avalia a disponibilidade do sistema e nível do serviço; e teste *on-line* que analisa um conjunto de métricas em tempo de execução.

O trabalho [BAZAN; JASEEMUDDIN, 2006] apresenta o paradigma da comunicação autonômica e seus princípios. Além disso, o trabalho trata de aspectos de projeto no roteamento de redes autonômicas. Os autores levantam a questão de que não existem algoritmos de roteamento adequados para todos os tipos de ambiente e requisitos das redes autonômicas, e por isso novos protocolos devem ser desenvolvidos. É proposto que para se projetar um protocolo de roteamento para redes autonômicas é necessário levar em consideração as seguintes questões: auto-organização, otimização de performance, gerenciamento de mobilidade, requisitos do serviço, escalabilidade, recuperação de falhas, interoperabilidade, múltiplos contextos, sobrecarga, diversidade dos paradigmas de comunicação, segurança e confiabilidade.

O trabalho [MELCHER; MITCHELL, 2004] trata dos serviços de rede auto-configuráveis. Os autores apresentam as tecnologias de configuração de serviços de redes atuais e identificam as capacidades atuais e limitações para o desenvolvimento de auto-configuração dinâmica destes serviços. São definidos requisitos para a extensão destas tecnologias a fim de lidar com estas limitações e prover serviços autonômicos para diversos tipos de ambientes de rede. Por fim, os autores demonstram o desenvolvimento de aplicações autonômicas utilizando o *Toolkit* de Computação Autonômica da IBM [IBM, 2006].

O trabalho [STERRITT, 2004] considera aspectos de auto-cura em redes autonômicas, em particular, técnicas para correlação de eventos e identificação de falhas. Os autores propuseram um ambiente que realiza a determinação e descoberta de regras para identificação de falhas em sistemas de telecomunicações utilizando eventos de alarme. Duas ferramentas foram descritas: a acCAT (*Autonomic Computing Correlator Analysis Tool*), usada para o suporte e validação de regras e a ferramenta HACKER, para o suporte de descoberta visual de regras, captura de conhecimento de "experts"e apresentação visual das regras descobertas. Estas ferramentas ajudam a identificar e documentar regras, e a informação armazenada é considerada uma forma de raciocínio baseado em casos.

O trabalho proposto em [LITTMAN et al., 2004] formula a recuperação de falhas em redes autonômicas como um problema de aprendizado por reforço. O sistema implementado é capaz de aprender como recuperar a conectividade da rede após uma falha de forma eficiente, utilizando um esquema de recuperação de falhas sensível ao custo (*cost-sensitive fault remediation*). Foram considerados problemas de manutenção e troca de disco em um servidor web em uma aplicação experimental para a recuperação da aplicação. Um agente responsável por tomar as decisões no sistema executa testes a um determinado custo e recupera o funcionamento do sistema em caso de falhas, sendo que esta recuperação também possui um custo. Para isto, o agente tenta encontrar uma política de custo mínimo para restaurar sistema. Segundo os autores, esta abordagem baseada em aprendizado para o diagnóstico e recuperação complementa os sistemas existentes provendo auto-cura, descoberta de relacionamento entre falhas, e ações de recuperação ótimas.

Outras áreas relacionadas às redes autonômicas são as redes ativas e programáveis [TENNENHOUSE; WETHERALL, 1996; TENNENHOUSE et al., 1997]. Estas redes são uma nova abordagem para arquiteturas de rede em que dispositivos de rede executam algum tipo de processamento diferenciado nas mensagens que passam por eles. Executar processamento diferenciado dá a estes dispositivos a capacidade de adaptação e reconfiguração. Redes autonômicas devem ser auto-adaptativas, reconfigurando seu funcionamento e operações de acordo com políticas. As plataformas ativas e programáveis fornecem meios para a adaptação genérica e reconfiguração. Estes sistemas isoladamente não podem ser considerados autonômicos, mas apresentam características essenciais às redes autonômicas.

4.3 Redes de Sensores Sem Fio Autonômicas

A computação autonômica é geralmente apresentada no contexto dos grandes sistemas computacionais, tipicamente executando software complexo em plataformas de hardware com poucas restrições de recursos. Os trabalhos apresentados a seguir propõem que a computação autonômica pode ser estendida para sistemas distribuídos e com limitações severas de energia, memória e processamento.

A arquitetura Manna para gerenciamento de RSSF foi proposta em [RUIZ, 2003]. A principal contribuição deste trabalho é a definição de um esquema para se construir soluções de gerenciamento a partir de serviços e funções e da utilização de modelos, sendo que as funções de gerenciamento representam a menor parte funcional de um serviço de gerenciamento. Para isto foram definidas uma arquitetura de informação, uma arquitetura funcional, uma arquitetura física e uma organização baseada em três dimensões de gerenciamento (ver Figura 4.1). A arquitetura de informação define dois tipos de informação para RSSFs, estática (representada através de classes de objetos) e dinâmica (representada através de modelos de rede). A arquitetura funcional é proposta com objetivo de planejar os locais na rede onde as entidades de gerenciamento (gerentes e agentes) podem ser executadas e por quais serviços e funções de gerenciamento cada uma delas será responsável. A arquitetura física descreve as interfaces que podem ser utilizadas para troca de informação entre as entidades de gerenciamento. A organização tridimensional é composta pelas dimensões: áreas funcionais, níveis de gerenciamento e funcionalidades de RSSFs. Duas das dimensões, áreas funcionais e níveis de gerenciamento, usadas no gerenciamento tradicional, foram redefinidas sob a perspectiva das RSSFs, e foi definida a terceira dimensão, funcionalidades de RSSFs, que lida com aspectos de configuração, sensoriamento, processamento, comunicação e manutenção de RSSFs. A interseção dos três planos da organização tridimensional para o gerenciamento de RSSFs define uma célula que contém um conjunto de funções de gerenciamento. A coordenação entre os três planos pode ser baseada em políticas. O uso desta organização é útil na definição de serviços, funções e informação de gerenciamento, assim como no planejamento e definição de aplicações. Esta arquitetura é baseada no paradigma de computação autonômica, e permite a definição de funções e serviços de auto-gerenciamento. Os experimentos realizados utilizando esta arquitetura mostraram que a solução é viável e que melhora a performance das RSSFs.

O trabalho proposto em [MARSH et al., 2004] demonstra que o comportamento autonômico pode ser incorporado em nós sensores. É discutido como o uso de técnicas autonômicas podem trazer benefícios para as redes de sensores sem fio sem sobrecarregá-las com um custo adicional de computação. Os autores defendem que isto pode ser alcançado utilizando sis-

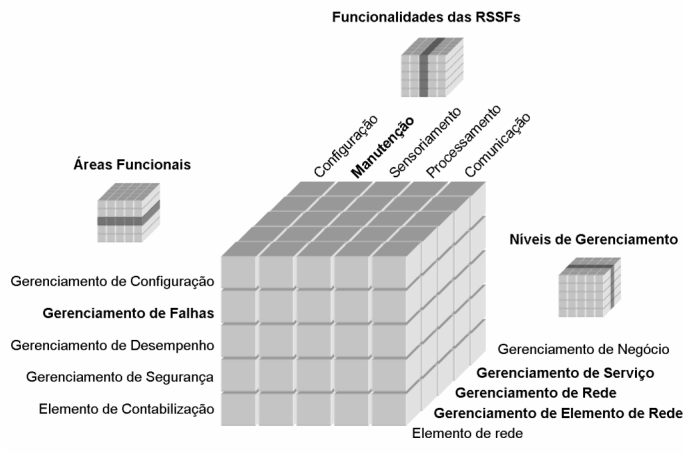


Figura 4.1: Arquitetura de gerenciamento MANNA [RUIZ, 2003].

temas multi-agentes, considerando que estes agentes facilitam a tomada de decisão dentro da rede. Por exemplo, no caso da falha de um nó, os agentes da região vizinha ao nó podem determinar o melhor método para recuperação baseado nos dados locais. O trabalho também apresenta um experimento, utilizando agentes distribuídos, a fim de provar a viabilidade das RSSFs autonômicas. Neste experimento, foram construídos agentes para nós sensores reais, utilizando funcionalidades de auto-gerenciamento. Foram utilizados três nós sensores da família Mica Motes, sendo que cada nó foi programado com um agente diferente. O agente percebe seu ambiente através dos sensores acoplados, executa um processamento simples e toma decisão para alterar seu comportamento baseado nos resultados obtidos do processamento e de políticas pré-definidas. Os resultados demonstram que o consumo de energia foi minimizado utilizando a abordagem autonômica, além de provar a viabilidade de aplicar propriedades de auto-gerenciamento e a utilização de agentes em um cenário restrito como o das RSSFs.

A arquitetura Impala, um *middleware* para RSSF é apresentada em [LIU; MARTONOSI, 2003]. Esta arquitetura permite que uma atualização de software seja recebida pelo transceptor do nó e aplicada à execução no sistema dinamicamente. A adaptação de aplicações permite melhora de performance, consumo de energia eficiente e confiabilidade no software do sistema. Outra motivação para a adaptatividade decorre do fato que nós sensores apresentam restrições severas de processamento e energia; operações de ajuste da configuração do nó podem resolver este problema. Por fim, considerando que redes de sensores são dispositivos em que falhas são prováveis de acontecer, a adaptação para lidar com dispositivos que falham se torna outra funcionalidade desejada. O *middleware* atua como um gerente de eventos e de dispositivos para cada nó sensor do sistema. Foram realizados experimentos simulados utilizando cinqüenta nós e três algoritmos de roteamento diferentes. Os resultados mostraram que a arquitetura resulta em baixa sobrecarrega dos nós, reprogramação eficiente da rede, e que pode oferecer melhoras

efetivas na performance, consumo de energia e robustez das RSSFs. Além disso, esta arquitetura pode ser utilizada para embutir comportamento autonômico em RSSFs.

O trabalho [RUIZ et al., 2005b] trata do desafio de prover uma solução de auto-gerenciamento para uma RSSF que monitora temperatura e avalia risco de incêndio. As funcionalidades autonômicas de auto-organização, auto-configuração, auto-serviço e auto-manutenção são o foco deste trabalho. Em particular, os autores propõem o uso de negociação de serviços para demonstrar o conceito de auto-serviço. A aplicação de monitoração ambiental apresentada considera a utilização de RSSFs heterogêneas na realização de gerenciamento distribuído. A solução de auto-gerenciamento utiliza um diagrama de Voronoi para identificar nós redundantes, calcula o risco de incêndio utilizando o índice de Angstrom, realiza negociação de serviços baseado em uma máquina de estados, e reconfigura o hardware dos nós da rede utilizando funções de gerenciamento dinâmico de potência. As políticas de auto-gerenciamento foram definidas usando a linguagem Ponder. Foi realizado um experimento, utilizando a ferramenta Mannasim [LOPES et al., 2006], com cento e quarenta e quatro nós comuns e doze nós líderes em um cenário que implementa funcionalidades de gerenciamento autonômicas e um que não realiza nenhuma forma de gerenciamento. Os resultados de simulação mostram que a solução de gerenciamento promove a produtividade dos recursos da rede, economiza energia, aumenta o tempo de vida da rede, permite a notificação precoce de incêndios e melhora a qualidade da informação extraída da rede, garantindo a qualidade dos serviços providos.

O trabalho [BRAGA, 2006] apresenta um modelo de um Elemento Sensor Autonômico (ESA) para RSSFs. O ESA é a menor parte de uma RSSF autonômica, e é composto por um gerente autonômico responsável por realizar tarefas de monitoração, análise, planejamento e execução, e um ou mais elementos gerenciados. O ESA foi projetado para ser executado em ambientes com restrições severas de hardware, software, comunicação e energia. O modelo foi simulado utilizando uma rede plana com cem ou duzentos nós em cenários em que o ESA foi implementado de forma centralizada, distribuída e um cenário sem o ESA. Os resultados obtidos mostram que a utilização do ESA reduziu o consumo com energia, diminui a quantidade de pacotes perdidos e melhora a produtividade e precisão na entrega de informações. Uma instância simplificada do modelo foi criada para a arquitetura dos nós Mica Motes a fim de comprovar a viabilidade de embutir o ESA na memória dos nós.

4.4 Gerenciamento Baseado em Políticas para Computação Autonômica

Esta seção descreve brevemente o Gerenciamento Baseado em Políticas para Computação Autonômica (PMAC) lançada pela IBM. A PMAC [IBM, 2007] é uma infra-estrutura que utiliza políticas para simplificar o gerenciamento e automação de sistemas autonômicos. No PMAC, cada regra possui 4 componentes: condições, ações, prioridade e papel. As condições associadas a regra de uma política especificam quando ela é aplicável. Caso ela seja aplicável, as ações associadas à política são executadas. A prioridade indica a importância relativa da política associada, determinando qual política será aplicada quando existem diversas políticas aplicáveis com ações que podem vir a entrar em conflito. O último componente, o papel, define em que contexto a política será considerada relevante.

A plataforma PMAC suporta o modelo de sistema adotado pela arquitetura de computação autonômica da IBM [IBM, 2005b]. No PMAC, o gerente autonômico tem seu comportamento guiado por políticas que permitem a monitoração, análise e planejamento da infra-estrutura de TI de acordo com as políticas definidas para os recursos gerenciados. O PMAC visa ser uma plataforma genérica e independente, suportando formatos e arquiteturas existentes, e flexível de forma que funções para aplicações específicas possam ser adicionadas. Como resultado, a plataforma é baseada em um formato aberto, o XML, e em tecnologias padrão como o J2EE. No nível mais alto, a plataforma provê quatro componentes principais: a ferramenta de definição de políticas (PDT), interface através da qual o administrador pode criar e modificar políticas; o repositório do editor políticas (PES), um repositório centralizado para o armazenamento das políticas; gerente autonômico, que obtém as políticas do PES, registra os recursos gerenciáveis, e avalia políticas planejando suas ações; e por fim os recursos gerenciáveis que tem sua configuração alterada de acordo com as orientações do gerente autonômico.

Como mencionado anteriormente, esta plataforma é baseada em XML e J2EE. As RSSFs atuais ainda não provêm suporte a J2EE e por isto esta plataforma não é aplicável diretamente a estas redes.

4.5 Gerenciamento Tradicional

O objetivo do gerenciamento das RSSFs é promover a produtividade dos recursos a fim de realizar os objetivos propostos pela aplicação e com qualidade de serviço. O gerenciamento de RSSFs é uma função complexa, dado que várias características deste tipo de rede, como o

grande número de nós, alta densidade, difícil acesso para realização de tarefas operacionais, dificultem, embora não inviabilizem, o gerenciamento das RSSFs. Soluções tradicionais de gerenciamento, desenvolvidas para serem aplicadas em vários contextos não consideram estas características e restrições específicas das RSSFs, e por este motivo não são aplicadas diretamente à este tipo de rede [RUIZ, 2003]. Ainda assim, vários conceitos importantes definidos para redes tradicionais podem ser adaptados para o contexto das RSSFs.

O gerenciamento de redes tradicionais é uma aplicação de processamento de informação que envolve a troca de informação para monitorar e controlar vários recursos físicos e lógicos da rede. Fazem parte do gerenciamento funções como supervisão e monitoração dos seus equipamentos e recursos, medição da utilização dos recursos, configuração dos equipamentos para funcionamento, disponibilidade de recursos, manutenção dos equipamentos e integridade de dados dentre outros.

A fim de resolver os problemas relativos à configuração de uma rede, as falhas que possam ocorrer nos componentes, aos níveis de desempenho que a rede apresenta, a segurança que esta apresenta e a contabilização de sua utilização foram propostas pelo ITU-T [INTERNATIONAL TELECOMMUNICATION UNION (ITU), 1996] as Áreas Funcionais de Gerência. Estas áreas funcionais são descritas a seguir:

- Gerência de Desempenho: Provê funções que, através da coleta de dados estatísticos, permitem controlar, monitorar, relatar, corrigir e analisar o comportamento e eficácia da rede, elementos de rede ou equipamentos, bem como auxiliar no planejamento e análise dos mesmos. Responsável pela avaliação de desempenho, gerência de tráfego e funções de qualidade de serviço (QoS).

- Gerência de Falhas: Provê funções que possibilitam a detecção, isolação e a correção de uma operação anormal da rede de telecomunicações. Responsável pela supervisão de alarmes, localização de falhas, funções de testes e gerência de anormalidades.

- Gerência de Configuração: Provê funções que habilitam o usuário a criar e modificar o modelo de gerenciamento de recursos físicos e lógicos da rede de telecomunicações.

- Gerência de Contabilização: Inclui as funções para informar aos usuários os custos dos recursos consumidos, estabelecendo métricas, quotas e podendo gerar tarifas. Responsável pelas funções de faturamento, funções de tarifação dentre outras.

- Gerência de Segurança: Trata da proteção das informações estratégicas, procurando agregar aos dispositivos de acesso ao sistema, controles de acesso aos usuários e notificando possíveis problemas de segurança.

Além das Áreas Funcionais, o ITU-T adotou um modelo em camadas funcionais para o gerenciamento com o objetivo de dividir a complexidade do ambiente a ser gerenciado em partes mais compreensíveis:

- Camada de Gerenciamento de Negócios: composta por sistemas necessários para o gerenciamento do empreendimento como um todo. A gerência de negócio trata das questões relativas às finanças, aos interesses dos acionistas, dos clientes, dos empregados e da sociedade. A partir destes objetivos é que surgem os requisitos para o gerenciamento dos serviços que a empresa fornece aos seus clientes, incluindo gerência de ordens de serviço, qualidade de serviço, manipulação de problemas, cobrança, desenvolvimento de serviços, etc.

- Camada de Gerenciamento de Serviços: composta por sistemas destinados à operação, administração e manutenção de serviços, abrangendo cadastro de usuários, relacionamento com usuários, provisionamento e manutenção de serviços, informações de faturamento, entre outros.

- Camada de Gerenciamento de Rede: é a primeira camada que relaciona os elementos de rede individuais, possibilitando a visão da rede como um todo. Composta por sistemas destinados à operação, administração, provisionamento e manutenção de rede, tais como detecção e isolamento de falhas.

- Camada de Gerenciamento de Elemento de Rede: composta por sistemas diretamente relacionados às atividades de gerência individuais dos elementos de rede.

- Camada de Elemento de Rede: corresponde aos componentes da rede de telecomunicações que necessitam ser gerenciados, e que possuem funções de gerenciamento.

No gerenciamento tradicional, um sistema possui gerentes e agentes (ver Figura 4.2). Os gerentes coletam informações de gerência dos agentes, através de operações de gerência e notificações. Os agentes fornecem uma visão orientada a objetos dos equipamentos gerenciados que são denominados de forma genérica como objetos gerenciados.

O gerente é a parte da aplicação distribuída que emite operações e recebe notificações, enquanto que o papel do agente é responder às operações de gerenciamento emitidas pelo gerente, além de fornecer ao gerente uma visão destes objetos, emitindo notificações que espelhem o comportamento dos mesmos. Um agente pode estar envolvido numa troca de informação com vários gerentes. Da mesma forma, um gerente pode estar envolvido numa troca de informação

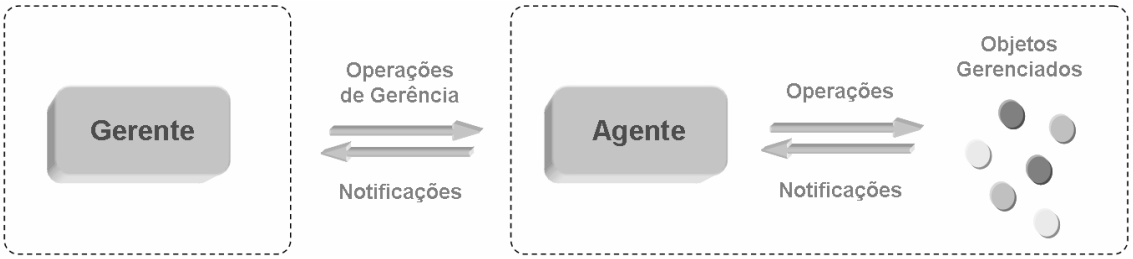


Figura 4.2: Arquitetura baseada em gerente e agente.

com vários agentes. Em um ambiente complexo, podem haver vários gerentes interagindo com agentes associados, e a sincronização das operações de gerenciamento pode ser necessária.

A informação trocada entre gerente e agente é modelada em termos de objetos gerenciados. Um objeto gerenciado é uma abstração de um recurso gerenciado e representa suas propriedades, podendo também representar um relacionamento entre recursos ou uma combinação de recursos. Um objeto gerenciado é definido pelos seus atributos visíveis, pelas operações de gerenciamento que lhe podem ser aplicadas, pelo comportamento apresentado em resposta a estímulos internos ou externos, e pelas notificações emitidas por ele.

Dentro de um sistema de gerência, o conjunto de todos os objetos e suas propriedades (atributos, operações, notificações e comportamento) constituem a Base de Informação de Gerência (*Management Information Base* - MIB) do sistema. Uma MIB é então, uma coleção de classes de objetos gerenciados, suas relações de herança, e as regras para nomeação das instância de objetos das classes.

A seguir são descritas as arquiteturas de gerenciamento SNMP (*Simple Network Management Protocol*), TMN (*Telecommunication Management Network*) e PBNM (*Policy Based Network Management*).

4.5.1 SNMP

O protocolo SNMP (*Simple Network Management Protocol*) [STALLINGS, 1993] é um protocolo de gerência definido em nível de aplicação, e é utilizado para obter informações de agentes espalhados em uma rede baseada na pilha de protocolos TCP/IP. Os dados são obtidos através de requisições de um gerente a um ou mais agentes utilizando os serviços do protocolo de transporte UDP (*User Datagram Protocol*) para enviar e receber suas mensagens através da rede. O gerenciamento da rede através do SNMP permite o acompanhamento simples e fácil do estado, em tempo real, da rede, podendo ser utilizado para gerenciar diferentes tipos de sistemas. Neste protocolo as informações são trocadas entre a MIB e a aplicação de gerência através de

comandos baseados no mecanismo de busca/alteração. No mecanismo de busca/alteração estão disponíveis as operações de alteração de um valor de um objeto, de obtenção dos valores de um objeto e suas variações. A utilização de um número limitado de operações simples torna o protocolo de fácil implementação, estável e flexível. Como conseqüência reduz o tráfego de mensagens de gerenciamento através da rede e permite a introdução de novas características. O funcionamento do SNMP é baseado em dois dispositivos o agente e o gerente. Cada recurso gerenciado é visto como um conjunto de variáveis que representam informações referentes ao seu estado atual, estas informações ficam disponíveis ao gerente através de consulta e podem ser alteradas por ele.

4.5.2 TMN

A TMN (*Telecommunication Management Network*) [PRAS; BEIJNUM; SPRENKELS, 1999] é uma estrutura organizada para interconexão entre sistemas de gerência e equipamentos de telecomunicações. Essa ligação visa a troca de informações de gerenciamento através de interfaces padronizadas, incluindo a definição de protocolos e mensagens. Os padrões estabelecidos pela TMN referem-se genericamente a um modelo de arquitetura funcional e especificações de interfaces/protocolos criadas pelo ITU e outros organismos internacionais de padronização. O termo TMN trata do suporte à gerência da rede e dos serviços de telecomunicações nas atividades de planejamento, provisionamento, instalação, manutenção, operação e administração. A TMN fornece funções de gerenciamento para redes e serviços de telecomunicações, sendo que os sistemas que compõem a TMN se comunicam entre si, com os equipamentos da rede e com sistemas de outras TMN. A TMN considera as redes e os serviços de telecomunicações como um conjunto de sistemas cooperativos para gerenciá-los de forma harmônica e integrada. O modelo TMN clássico divide-se em três arquiteturas que devem ser consideradas no projeto de uma plataforma de gerência: funcional, de informação e física. Esse conjunto de componentes básicos serve para a construção da gerência das redes como também o relacionamento entre estes componentes.

4.5.3 PBNM

O PBNM (*Policy Based Network Management*) [SLOMAN, 1994] é baseado na idéia de que cada recurso ou processo da rede tem um papel e regras específicas de procedimento. A agregação de um conjunto de ações em um nível de abstração maior é denominada política. As políticas, associadas à PBNM, definem um método eficaz de expressar o comportamento desejado de recursos. Tais políticas são armazenadas em um repositório e transferidas para o

sistema de gerenciamento PBNM para serem aplicadas na rede. O PBNM traduz, em momentos pré-definidos, as políticas descritas em alto nível em ações de configuração dos dispositivos, utilizando os diversos protocolos de configuração existentes na rede (por exemplo Telnet/CLI, SNMP ou HTTP). Nesse contexto, o administrador de rede expressa nas políticas os objetivos e metas e o sistema PBNM se encarrega de configurar os dispositivos de forma automatizada, liberando os administradores de uma intervenção manual. A arquitetura PBNM mais aceita atualmente é composta pelos seguintes elementos: console de gerenciamento, repositório de políticas, ponto de decisão de política e pontos de aplicação de política.

4.6 Considerações Finais

Este capítulo apresentou uma série de trabalhos relacionados com esta dissertação. Em especial os trabalhos tratam de aspectos de auto-cura em sistemas autonômicos, redes autonômicas e aspectos de auto-gerenciamento em RSSFs. Em sua maioria os trabalhos que tratam de computação autonômica tratam de sistemas ou redes tradicionais, que não consideram as características e restrições das RSSFs. Por este motivo muitos destes trabalhos não são diretamente aplicáveis as RSSFs. Dentre os trabalhos que tratam de computação autonômica para RSSFs com aspectos de auto-cura em sistemas autonômicos não foram encontrados na literatura trabalhos que utilizem a ITIL como solução integradora dos serviços e gerenciamento de RSSFs. Este capítulo também apresentou a plataforma PMAC lançada pela IBM, que aplica a computação autonômica ao gerenciamento baseado em políticas. Por fim, este capítulo apresentou uma descrição do gerenciamento tradicional para redes de telecomunicações, suas áreas funcionais e camadas, e de tecnologias SNMP, TMN e PBMN. As soluções de gerenciamento tradicional, em geral, não são diretamente aplicáveis às RSSFs devido às restrições e características particulares que diferenciam as RSSFs dos outros tipos de rede (ver Seção 2.1).

O capítulo a seguir apresenta a principal colaboração deste trabalho: uma abordagem de gerenciamento de serviços em RSSFs autonômicas. Esta abordagem utiliza a biblioteca ITIL como metodologia para integrar serviços e gerenciamento das RSSFs.

5 *Gerenciamento de Serviços em RSSFs Autonômicas*

O gerenciamento de serviços diz respeito a supervisão e controle das funcionalidades associadas ao objetivos da aplicação. Os serviços de uma RSSF, por exemplo a coleta de dados no ambiente, processamento de dados, comunicação, devem ter como prioridade o atendimento ao nível de qualidade de serviço determinado. O gerenciamento do desempenho dos serviços é importante para garantir a qualidade do serviço entregue pela rede.

A disponibilidade do serviço em RSSF pode ser afetada por vários tipos de problemas relacionados aos recursos da rede (nós indisponíveis ou fora do alcance de comunicação), aos recursos de comunicação (perda da conexão), aos recursos de hardware (esgotamento de energia ou memória), e às informações produzidas (transação não realizada, problema de consistência, erro de processamento). Considerando que falhas não são exceções em RSSFs, este trabalho propõe uma abordagem de auto-gerenciamento para RSSFs que detecta e identifica falhas, e propõe alterações na infra-estrutura da rede, a fim de manter a disponibilidade do serviço.

Baseado na organização tridimensional para o gerenciamento de RSSFs proposta na arquitetura de gerenciamento Manna (áreas funcionais, níveis de gerenciamento e funcionalidades) [RUIZ, 2003] apresentada na Seção 4.3, a abordagem de gerenciamento de serviços para RSSFs proposta neste trabalho utiliza a área funcional de gerenciamento de falhas, os níveis de gerenciamento de elemento de rede, rede e serviço, e a funcionalidade de manutenção de RSSFs para implementar o serviço de auto-cura (ver Figura 4.1). O serviço de auto-cura é definido utilizando conceitos de gerenciamento de serviços da biblioteca ITIL (ver Capítulo 3) e modeladas como tarefas de monitoração, análise, planejamento e execução de gerentes autonômicos (ver Seção 2.2).

A organização da rede autonômica proposta é apresentada na Figura 5.1. Na organização proposta, os nós sensores e seus componentes (hardware e software) correspondem aos recursos a serem gerenciados pelos gerentes autonômicos. Uma interface de gerenciamento, o *touchpoint* dos nós sensores, foi definida para tornar possível a monitoração e atuação sobre os nós da rede.

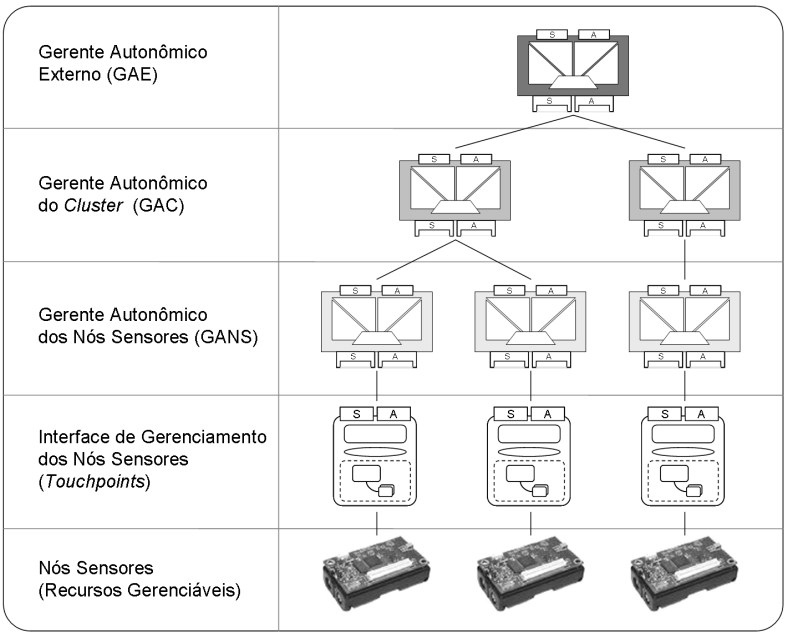


Figura 5.1: Arquitetura para gerenciamento de serviços.

Três tipos de gerentes autonômicos foram definidos: um gerente autonômico instalado em cada nó da rede (GANS), que controla a utilização dos recursos do nó, e identifica e recupera o nó de eventuais falhas em seus componentes; um gerente autonômico de *cluster* (GAC), que rege o funcionamento dos gerentes autonômicos dos nós pertencentes ao grupo (este gerente é utilizado no caso da rede ser hierárquica); e por fim um gerente externo a rede (GAE), que tem conhecimento maior do serviço provisionado e por isso pode ajustar os objetivos e comportamento da rede.

As seções a seguir descrevem cada componente da solução proposta. Os recursos gerenciáveis das RSSFs são apresentados na Seção 5.1. Em seguida é apresentado um *touchpoint* para o nó sensor na Seção 5.2. Os gerentes autonômicos definidos neste trabalho são descritos na seção 5.3 e por fim, a base de conhecimento é apresentada na Seção 5.3.6.

5.1 Recursos Gerenciáveis das RSSFs

Os recursos gerenciáveis das RSSFs correspondem ao nó sensor e seus componentes de hardware (sensor, processador, memória, bateria e rádio), software embutido no nó (parâmetros configuráveis da aplicação), conexão, serviços, etc. Alterações em parâmetros do nó sensor são guiadas por políticas a fim de atender requisitos de qualidade do serviço provisionado, que geralmente, estão diretamente relacionados a uma aplicação específica [IBM, 2005b].

A seguir é apresentada uma lista com alguns dos parâmetros configuráveis do nó sensor e da rede:

- estado de funcionamento do processador, rádio, sensor (ativo, *idle* ou *sleep*);
- potência de transmissão/recepção do rádio (controle de potência);
- estado operacional do nó (em operação ou fora de serviço);
- papel do nó na rede (nó comum, nó líder, nó roteador);
- número de nós por área monitorada (controle de densidade, área de cobertura);
- número de nós ativos (produzindo);
- *throughput* de informação (taxa de produção de informação);
- intervalos de sensoriamento e disseminação;
- tipo de algoritmo utilizado no processamento dos dados coletados;
- esquema de roteamento (controle da topologia e conectividade);
- tamanho da fila de pacotes (controle de congestionamento);
- comunicação *multihop* ou *singlehop*.

O planejamento da rede (configuração estática - em tempo de planejamento) e a definição de políticas para a manutenção dos serviços da rede (configuração dinâmica - em tempo de execução) baseiam-se na definição destes parâmetros.

A configuração estática da rede trata da configuração inicial da rede ou de aspectos que uma vez definidos não podem ser alterados. Por exemplo, o tipo de sensoriamento da rede deve ser definido inicialmente, mas pode ser alterado durante o tempo de vida da rede; já a escolha da arquitetura dos nós não pode ser alterada neste período de tempo. Alguns requisitos utilizados para a configuração estática são: número de nós, tipo e capacidade do nó (CPU, memória disponível, bateria, interface de rede, presença de atuadores), configuração da rede (RSSF plana ou hierárquica, RSSF homogênea ou heterogênea, RSSF móvel ou estática), tipo de deposição e distribuição (planejada ou ad hoc, uniforme ou irregular), densidade inicial dos nós (balanceada, densa ou esparsa), tipo de sensoriamento inicial (sensoriamento homogêneo ou heterogêneo, coleta contínua, periódica, sob demanda ou dirigida a eventos), mecanismo para disseminação de informação inicial (contínua, periódica, sob demanda ou dirigida a eventos),

tipo de conexão (simétrica ou assimétrica), tipo do canal (transmissão *simplex*, *half-duplex* ou *full-duplex*), tipo inicial de esquema de roteamento, tipo de sincronização, parâmetros de tráfego e infra-estrutura inicial da rede.

A configuração dinâmica da rede trata da monitoração e controle em "tempo de execução"de diversos aspectos: densidade da rede, atraso na entrega de pacotes, consumo de energia, potência de transmissão, topologia, cobertura, mobilidade, sensoriamento (tipo e intervalo de sensoriamento), disseminação (tipo e intervalo de disseminação), precisão (processamento - tipo de algoritmo utilizado).

A configuração estática e dinâmica da rede afetam diretamente os requisitos de QoS de uma RSSF. A seguir são listados alguns destes requisitos:

Área de cobertura do rádio ou sensor. O alcance do rádio e do sensor utilizado ajudam a definir a quantidade de nós a serem utilizados e a área que eles conseguem cobrir. No SLA para RSSFs (ver Seção 2.3.2) deve ser definida a porcentagem da área monitorada que deve estar coberta, isto é, permitir que um observador receba informações coletadas de um determinado número de nós que representem a visão da área monitorada.

Número de nós ativos. Pode ser necessário especificar o número mínimo de nós ativos coletando dados num determinado momento. Por exemplo, no caso de uma aplicação que deve entregar a média dos nós da rede com uma determinada freqüência a um observador, a garantia de um determinado número de nós ativos decorre na entrega de um resultado de correlação mais confiável.

Throughput de dados sensoriados. A produção de informação nas RSSFs depende do tipo de aplicação da rede e do comportamento da aplicação sob determinadas condições. Os dados produzidos estão distribuídos na rede e a informação que se deseja extrair da rede pode ser fornecida por um único nó, ou pode ser necessária a colaboração de diversos nós.

Tempo de vida da rede. Esta métrica determina por quanto tempo a rede deve estar em funcionamento. Dependendo do tipo de aplicação o tempo de vida da rede é um fator que determina a maior parte das escolhas do projeto da rede, como por exemplo o tipo de sensoriamento, tipo de disseminação, arquitetura dos nós, topologia e tipo de comunicação da rede.

Integridade, confiabilidade e privacidade da informação sensoreada. Estas métricas estão relacionadas à exatidão, proteção e o controle do acesso aos dados depois de serem transmitidos ou processados.

Atraso, latência. Estas métricas tratam do tempo gasto entre a transmissão e a recepção de mensagens. O tempo que a informação leva para chegar ao destino é uma métrica importante para garantir o tempo de validade de uma determinada informação.

Perda de pacotes (individual/coletiva). A perda de pacotes é a métrica que mede a quantidade de dados descartados quando uma rede está sobrecarregada ou por algum outro motivo não pode receber dados em um determinado momento. Para identificar problemas de perda de pacotes em RSSFs é importante separar a perda de pacotes individual (de um único nó) da coletiva (de um grupo de nós ou de todos os nós da rede).

Disponibilidade do serviço (*uptime*). Esta métrica trata da quantidade de tempo que um serviço está em funcionamento e disponível.

Tempo de resposta A métrica que trata do atraso entre uma requisição e sua resposta é o tempo de resposta. Esta métrica é importante em RSSFs configuradas com sensoriamento e disseminação sob demanda.

Tempo médio para notificação de eventos. Esta é uma métrica relevante para as RSSFs dirigidas à eventos. Nesta métrica é considerado o tempo médio entre a detecção de eventos e a notificação destes eventos a um observador.

Tempo médio para notificação/recuperação de problemas. Estas métricas tratam do tempo para se tomar uma ação para resolução de problemas. A partir da detecção de um problema, pode ser medido o tempo médio para notificar um observador do problema, ou o tempo médio para realizar a recuperação do nó ou da rede.

5.2 Touchpoint para Nós Sensores

Um *touchpoint* é uma interface de gerenciamento que encapsula um ou mais recursos. Quando existem múltiplos recursos, um recurso principal deve ser especificado. Este recurso, no caso das RSSFs, é o nó sensor, e os recursos secundários são os componentes do nó (software embutido no nó e componentes de hardware - sensor, processador, memória, bateria e rádio). Para que o gerentes autonômicos tenham a capacidade de atuar sobre os recursos gerenciáveis de uma RSSF é necessário definir uma interface de padrão de gerenciamento para acesso e controle destes recursos. Para isto este trabalho define um *touchpoint* para nós sensores sem fio (ver Figura 5.2). Através de um conjunto de propriedades, operações, eventos e dados deve ser possível determinar o comportamento do recurso gerenciável num determinado instante de tempo.

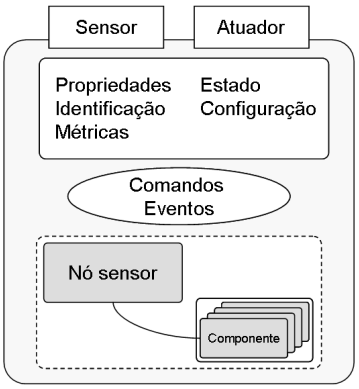


Figura 5.2: Touchpoint para nós sensores.

Um *touchpoint* deve disponibilizar o estado operacional do recurso gerenciável relacionado a ele, bem como disponibilizar as propriedades do recurso. No caso do *touchpoint* dos nós sensores, estas propriedades podem ser a energia residual do nó sensor, estado operacional do processador ou do transceptor, intervalo de sensoriamento, intervalo de disseminação e potência de transmissão, por exemplo. Além disso, este *touchpoint* deve permitir a alteração da configuração do nó sensor quando requisitado por um gerente autonômico.

O *touchpoint* para nós sensores deve implementar o comportamento sensor e atuador. A interface sensora[1] permite que o gerente autonômico obtenha informação do *touchpoint*, através das operações *request-response* e *send-notification*. A interface atuadora permite que o gerente autonômico gerencie o *touchpoint* utilizando os comandos *perform-operation* e *solicit-response*. Estes comandos são descritas a seguir.

- *Request-Response*: o *touchpoint* dos nós sensores deve responder a requisições de gerentes autonômicos autorizados, quando solicitado (abordagem sob-demanda).
- *Send-Notification*: o *touchpoint* dos nós sensores envia notificações quando do acontecimento de determinados eventos, como por exemplo quando o estado operacional do nó muda de disponível para indisponível (abordagem dirigida em eventos).
- *Perform-Operation*: o *touchpoint* dos nós sensores realiza operações para alterar a configuração do nó sensor.
- *Solicit-Response*: o *touchpoint* solicita informações ao gerente autonômico, como por exemplo métricas para a definição de eventos a serem notificados.

Além destes comandos, o *touchpoint* armazena informação sobre detalhes dos recursos gerenciáveis, tais como identificação, métricas, configurações e estado. Os gerentes autonômicos

[1]Esta interface sensora não corresponde a unidade de sensoriamento do nó sensor.

interagem com os *touchpoints* através destas funções, e devem ser capazes de identificar *touchpoints*. Por este motivo, devem ser criadas formas de identificar globalmente os *touchpoints* e recursos gerenciáveis. Neste trabalho, assumimos que cada nó possui uma identificação única na rede, e que o *touchpoint* associado, por conseqüência também possa ser identificado globalmente.

5.3 Gerentes autonômicos para RSSFs

Os gerentes autonômicos supervisionam e controlam o comportamento dos nós sensores. Neste trabalho, estes gerentes são responsáveis por identificar falhas e propor ajustes à infraestrutura da rede, a fim de manter a qualidade dos serviços provisionados pela RSSF. A capacidade de ajustar soluções de acordo com o contexto de funcionamento e utilização dos recursos da rede, bem como o tipo, extensão e gravidade de falhas, é o que torna a abordagem proposta neste trabalho em uma solução autonômica, já que as alterações são baseadas no conhecimento extraído do nó e da rede no decorrer do tempo. Para realizar o auto-gerenciamento das RSSFs foram definidos três tipos de gerentes:

Gerente Autonômico dos Nós Sensores (GANS). O gerente autonômico localizado em cada nó sensor atua em nível de gerenciamento de elemento de rede. Este gerente é responsável por monitorar o funcionamento do nó e seus recursos e componentes, otimizar o funcionamento e detectar comportamento anômalo, analisar eventos e fazer ajustes à configuração do nó para diminuir o risco de problemas e em caso de problemas, tentar restaurar o funcionamento do nó o mais rápido possível.

Gerente Autonômico de *Cluster* (GAC). O gerente autonômico está localizado nos nós líderes, no caso da rede ser hierárquica. Este gerente atua em nível de gerenciamento de rede e é responsável por manter a disponibilidade do serviço dentro do seu grupo. Para isto, ele realiza um mapeamento das políticas que representam o SLA para os nós participantes do grupo, distribui estas políticas, garante que os níveis de serviços estão sendo cumpridos dentro do grupo, e caso contrário, faz ajustes a fim de atender estes níveis. Além destas tarefas, este gerente acumula as tarefas definidas para o gerente autonômico dos nós comuns, a fim de gerenciar seus próprios recursos.

Gerente Autonômico Externo (GAE). Um gerente autonômico instanciado fora da rede e que pode atuar em nível de gerenciamento de serviço e de rede é responsável por monitorar a qualidade e disponibilidade do serviço, utilizando políticas do SLA (estas políticas serão

distribuídas aos nós líderes), e se necessário, renegociar o SLA. Este gerente tem visão global da rede e pode propor ajustes a rede caso o nível do serviço não seja satisfatório. No caso da rede ser plana, este gerente acumula as funções do gerente autonômico instanciado nos nós líderes.

Os gerentes autonômicos da RSSF tem seu comportamento guiado por políticas do acordo de nível de serviço (Service Level Agreement - SLA), isto é, requisitos de QoS que a RSSF deve atender modelados como políticas de gerenciamento do serviço. Estas políticas são armazenadas em um repositório centralizado, e são repassadas aos gerentes da rede, que as armazenam em repositórios locais (bases de conhecimento). Gerentes autonômicos regidos por estas políticas passam a controlar os recursos gerenciáveis, através das interfaces de sensoriamento e atuação disponibilizadas pelos *touchpoints*, realizando as funções de monitoração, análise, planejamento e execução.

As políticas que determinam o comportamento dos gerentes autonômicos devem definir as ações a serem tomadas quando do acontecimento de determinados eventos, além de definir o nível de acesso que o gerente possui sobre cada recurso gerenciável e sobre outros gerentes. A definição dos níveis de controle de acesso de cada gerente, neste trabalho é baseada nos níveis de gerenciamento da arquitetura Manna, a saber: **Gerenciamento de negócio**, camada responsável pelo gerenciamento de todo o sistema e onde são definidos os objetivos de negócio; **Gerenciamento de serviço**, camada responsável pelos aspectos de serviço; **Gerenciamento de rede**, camada responsável por gerenciar a interação entre elementos de rede; e **Gerenciamento de elemento de rede**, camada responsável por gerenciar o elemento de rede individualmente (ver Seção 4.5). Estes níveis de gerenciamento foram definidos na Arquitetura de Camadas Lógicas (*Logical Layered Architecture* - LLA) em [INTERNATIONAL TELECOMMUNICATION UNION (ITU), 1996] (ver Figura 5.3).

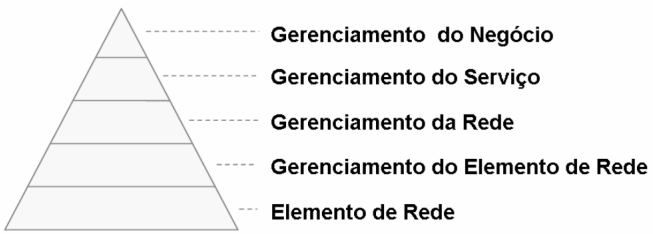


Figura 5.3: Níveis de gerenciamento.

Considerando que as camadas superiores têm maior conhecimento sobre o sistema e por conseqüência tem atuação mais eficaz e eficiente nas atividades de gerenciamento, este trabalho propõe que os gerentes autonômicos destas camadas tenham prioridade (capacidade de acessar

informações e de atuar) sobre os gerentes autonômicos das camadas inferiores. Por exemplo, os gerentes autonômicos que lidam com aspectos de serviço podem sugerir ou interferir nas ações tomadas pelos gerentes autonômicos dos níveis de rede e elemento de rede, podendo também atuar diretamente sobre o elemento gerenciado, através dos *touchpoints* dos nós sensores.

A interação entre agentes e gerentes, e gerentes e gerentes pode acontecer de três maneiras (ver Figura 5.4):

- Gerente de Agentes (*Manager of Agents* - MoA): um gerente autonômico supervisiona e controla recursos gerenciáveis através dos *touchpoints*.
- Gerentes de Gerente (*Manager of Managers* - MoM): existe uma hierarquia de gerentes autonômicos, na qual determinado(s) gerente(s) supervisionam e controlam outros gerentes autonômicos.
- Gerentes para Gerentes (*Manager to Managers* - M2M): supervisão e controle distribuídos entre os gerentes autonômicos.

Neste trabalho é considerado que os gerentes autonômicos que atuam diretamente sobre os recursos gerenciáveis utilizam a abordagem MoA; os gerentes autonômicos pertencentes a um mesmo nível de gerenciamento interagem utilizando a abordagem M2M; e por fim, os gerentes autonômicos de camadas superiores interagem com os gerentes de camadas inferiores utilizando a abordagem MoM. Por exemplo, um gerente autonômico instanciado para tratar de aspectos da camada de gerenciamento de rede atua de forma MoA quando age diretamente sobre o recurso gerenciável; quando interage com outros gerentes da camada de gerenciamento de rede atua de forma M2M; e quando se relaciona com gerentes autonômicos da camada de gerenciamento de elemento de rede, atua de forma MoM.

Os gerentes autonômicos possuem uma interface de sensoriamento e atuação que se comunica com os *touchpoints* para o gerenciamento dos recursos gerenciáveis. Para isto estes gerentes implementam os comandos *Request-Response*, em que realizam requisições de informações aos *touchpoints* e *Perform-Operation*, e realizam operações para alterar a configuração do nó sensor. A comunicação entre gerentes acontece utilizando as seguintes operações:

- *Request-Response*: o gerente autonômico deve responder a requisições de outros gerentes autonômicos quando solicitado (abordagem sob-demanda).
- *Send-Notification*: gerente autonômico envia notificações quando do acontecimento de determinados eventos, como por exemplo quando da detecção de alguma falha (abordagem dirigida em eventos).

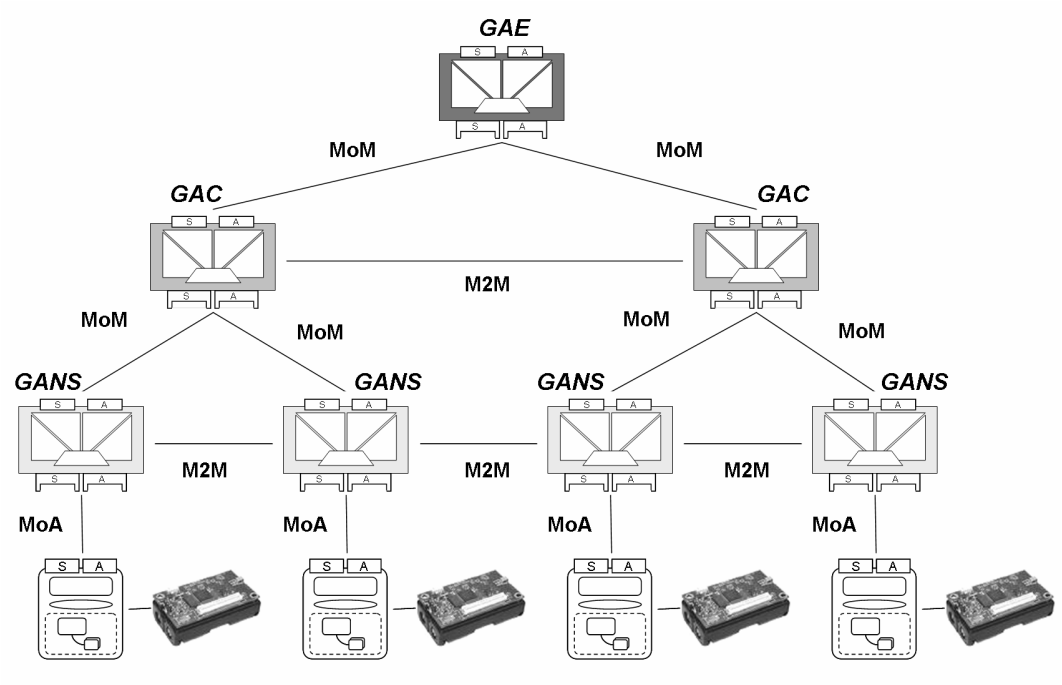


Figura 5.4: Interação entre os gerentes autonômicos da RSSF.

- *Send-Policy*: o gerente autonômico envia políticas para alterar o comportamento de outros gerentes.
- *Solicit-Response*: o gerente autonômico solicita informações a outro gerente autonômico.

Cada gerente autonômico supervisiona alguns recursos gerenciáveis através da função de monitoração. A partir de consultas a bases de conhecimento, a função de análise busca políticas para definir se mudanças são necessárias por causa da ocorrência de incidentes, e a função de planejamento ajusta uma solução existente de acordo com a extensão e gravidade do problema que se deseja corrigir. Uma vez definidos os parâmetros de mudança a função de execução aplica as mudanças e atualiza as bases de conhecimento. No componente lógico dos nós da RSSF são instanciados os gerentes autonômicos e os repositórios de políticas (Base de Conhecimento - BC). A fim de criar uma rede que implemente a função de auto-cura foram utilizados conceitos da área de Entrega de Serviço da Biblioteca ITIL na definição das responsabilidades dos gerentes autonômicos da organização proposta, a saber:

Gerenciamento do Nível de Serviços da RSSF: é o responsável por garantir que os níveis de serviço definidos no SLA 2.3.2 estão sendo cumpridos na rede e redefinir o SLA quando necessário, através de um gerente manual ou políticas de atualização do acordo.

Gerenciamento Autonômico de Disponibilidade: é o responsável por planejar e gerenciar a disponibilidade dos serviços, através da medição e monitoração da disponibilidade dos

serviços da rede.

Gerenciamento Autonômico de Continuidade: é o responsável por fazer análise de risco da rede, isto é, tentar identificar possíveis falhas e então criar um plano de redução de risco ou um plano de recuperação.

Gerente Autonômico de Capacidade: é o responsável por monitorar os recursos do nó, identificar demandas e em caso de capacidade insuficiente (atual ou futura) realocar recursos e antecipar novos recursos. Para isto, é necessário definir um modelo de uso dos recursos a fim de determinar se estes estão ou não atendendo os requisitos definidos.

Gerente Autonômico de "Finanças": é o responsável por gerenciar o custo da entrega dos serviços de TI. O custo pode ser calculado em função do consumo de energia ou outra métrica que seja relevante para aplicação. No caso desta dissertação o custo se refere ao consumo de energia da rede.

Cada uma destas responsabilidades de gerenciamento utiliza conceitos da área de Suporte a Serviços da biblioteca ITIL na realização das funções de monitoração, análise, planejamento e execução, considerando o serviço de auto-cura (ver Figura 5.5).

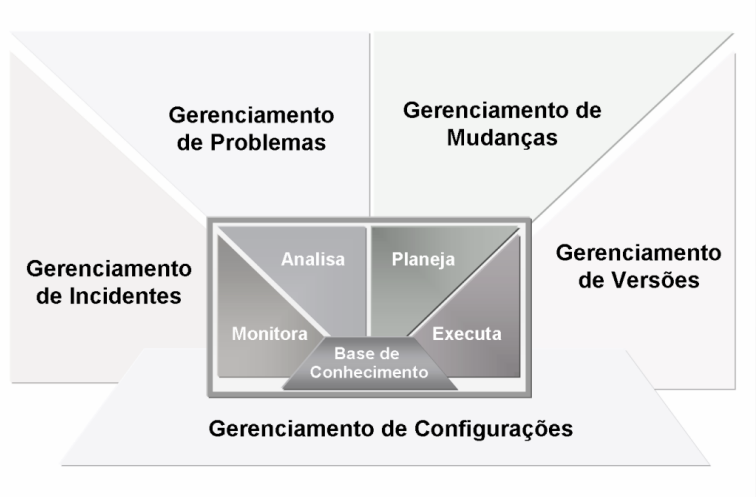


Figura 5.5: Elemento autonômico x Suporte a serviços.

- A função de **monitoração** detectará eventos que não fazem parte da operação padrão do serviço (incidentes) e que podem causar a interrupção do serviço ou diminuir sua qualidade.
- A função de **análise** determinará se mudanças precisam ser realizadas através da análise da causa dos incidentes e diagnose de erros, e levantará requisições de mudanças (Request for Change - RfC), se necessário.

- A função de **planejamento** irá propor mudanças na infra-estrutura da rede a fim de resolver problemas na rede.
- A função de **execução** do elemento autonômico realizará a mudança proposta nos elementos da rede, e avaliará o impacto da mudança sobre a rede.

As informações utilizadas e geradas pelo gerente autonômico nestas quatro funções são armazenados em uma base de conhecimento. No caso das RSSFs autonômicas, a descoberta de novos elementos na rede e a atualização de informações das bases de conhecimento é realizada pela função de aprendizado.

Os gerentes definidos contribuirão para manter a disponibilidade do serviço de disseminação da rede. As seções a seguir descrevem detalhadamente as responsabilidades dos gerentes definidos neste trabalho.

5.3.1 Gerenciamento do Nível do Serviço em RSSFs

O gerenciamento do nível do serviço em RSSFs deve garantir que as políticas do SLA estão sendo cumpridas na rede. Este gerente autonômico deve possuir uma visão geral do serviço da rede. Ele é instanciado fora da rede, no GAE, e trabalha em nível de gerenciamento de serviços, e por isto tem capacidade para influenciar as ações de todos outros gerentes da rede.

O gerenciamento do nível do serviço é responsável por monitorar a qualidade e disponibilidade do serviço. Para realizar esta função ele deve apresentar a capacidade de monitorar as métricas de QoS especificadas no SLA, utilizando para isto informações obtidas junto aos *touchpoints* dos nós sensores, informações das bases de conhecimento dos elementos autonômicos (EA) da rede (EAs dos nós comuns e dos nós líderes, no caso da rede ser hierárquica), e informações da sua base de conhecimento. De posse destas informações, o gerente autonômico externo (GAE) deve ser capaz de determinar se o nível do serviço acordado está sendo cumprido. Para isto este gerente compara cada uma das métricas e informações coletadas com as métricas definidas nas políticas de SLA. Caso o serviço provisionado esteja em um nível aceitável, este gerente pode utilizar as informações obtidas para tentar melhorar o funcionamento do serviço de forma proativa, tentando identificar tendências do nível do serviço e ajustando a rede para melhorar o nível do serviço. Caso o nível do serviço não seja suficiente, o GAE precisa identificar se o serviço foi degradado de maneira provisória ou permanente. Quando o problema é provisório, o gerente autonômico do nível de serviço propõe um plano de mudanças na infra-estrutura da rede ou no comportamento de um ou mais nós sensores que corrija o funcionamento do serviço. No caso de um problema permanente, como por exemplo, perda de muitos nós por esgotamento

de energia, este gerente tenta renegociar o SLA, propondo novas métricas de operação que possam ser cumpridas. Por fim, o gerente deve ser capaz de realizar o plano de ação definido (mudanças a serem aplicadas na rede ou redefinição do SLA).

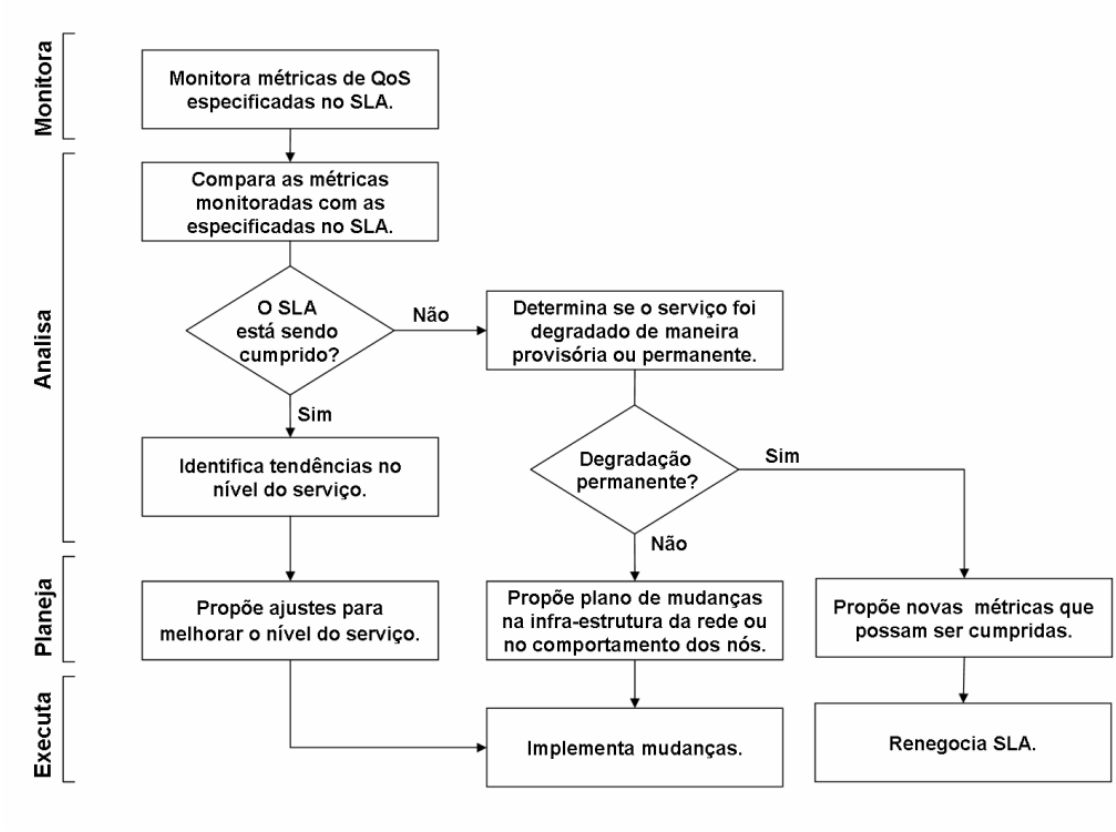


Figura 5.6: Responsabilidades do Gerenciamento do Nível do Serviço em RSSFs.

5.3.2 Gerenciamento de Disponibilidade em RSSFs

O gerenciamento de disponibilidade em RSSFs trata da disponibilidade dos serviços, através da medição e monitoração da disponibilidade dos serviços da rede. A monitoração da disponibilidade do serviço também pode ser realizada pelo gerenciamento do nível do serviço, se existir um requisito de disponibilidade no SLA. Neste trabalho foi definido que a responsabilidade do gerenciamento de disponibilidade é a manutenção da disponibilidade de serviços em grupos de nós, no caso das redes hierárquicas. Este gerente deve ser instanciado em nós que atuam como líderes de *cluster*, o GAC. Para realizar determinar o nível de disponibilidade, os gerentes utilizam a equação 5.1:

$$Disponibilidade = \frac{(T_{disponível} - T_{indisponível}) * 100}{T_{disponível}} \quad (5.1)$$

onde $T_{disponível}$ representa o tempo que o serviço deve permanecer disponível, definido no SLA, e $T_{indisponível}$ corresponde ao tempo que o serviço ficou fora de operação.

Caso o nível de disponibilidade não seja suficiente, o GAC precisa identificar se é possível prover este nível utilizando os nós do seu grupo. Se for possível, este gerente reconfigura os parâmetros dos nós do seu grupo a fim de prover a disponibilidade necessária. Caso não seja possível prover o nível de disponibilidade determinado para o grupo, o líder tenta repassar parte das tarefas executadas pelo seu grupo para nós de outros grupos, negociando recursos com outros gerentes de disponibilidade. Se não houver nenhuma maneira de cumprir o nível de disponibilidade do serviço, este gerente deve notificar o gerente autonômico externo (GAE), que então tenta renegociar o SLA, propondo uma nova métrica de disponibilidade de serviço.

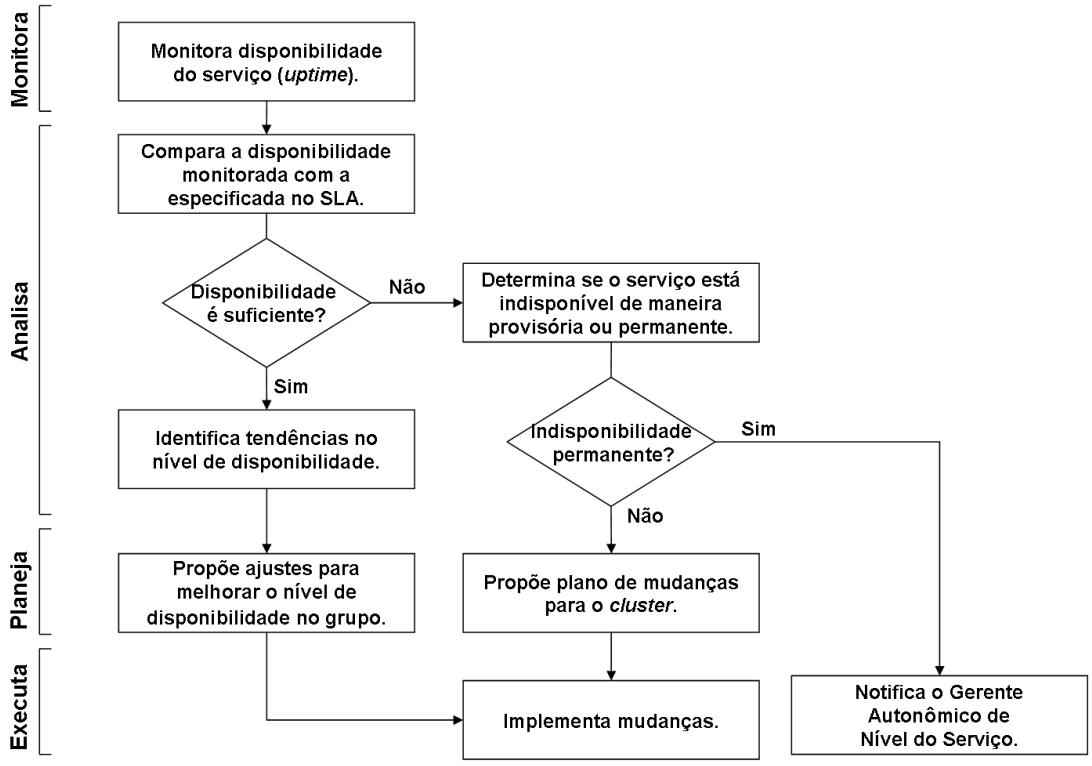


Figura 5.7: Responsabilidades do Gerenciamento de Disponibilidade em RSSF.

5.3.3 Gerenciamento de Continuidade em RSSFs

O gerenciamento de Continuidade em RSSFs deve tentar identificar possíveis falhas na rede e tentar recuperar a rede. Gerentes autonômicos responsáveis por estas tarefas devem ser instanciados em cada nó sensor da rede, podendo atuar em nível de gerenciamento de elemento de rede (GANS) ou em nível de rede (GAC). Os gerentes responsáveis pela continuidade da rede

devem monitorar os nós da rede a fim de identificar possíveis perdas de dados, falhas de conectividade, falhas do serviço por demanda excessiva e falhas de segurança. Para identificar estas falhas estes gerentes se utilizam de um modelo de falhas, onde são definidos tipo, localização, duração, extensão de falhas do nó e da rede. Além do modelo de falhas, este gerente deve ter acesso a planos para recuperação da rede, e ser capaz de "customizar"estes planos de acordo com a natureza da falha.

Através da função de monitoração, são monitorados os recursos gerenciáveis utilizando *touchpoints*. Assim que a função de análise detecta uma falha, é determinada a natureza da falha (baseado no modelo de falhas) e, utilizando a função de planejamento, é definido um plano de recuperação que pode ser executado de forma gradual ou imediata, dependendo da gravidade da falha. A recuperação gradual tenta recuperar o funcionamento do nó ou da rede fazendo alterações na configuração dos recursos gerenciáveis. Em caso de falhas graves, medidas como ativação de nós redundantes, ou desativação de nós podem ser executadas. Estes gerentes devem ser capaz de atuar sobre os nós e seus recursos para executar o plano de recuperação definido.

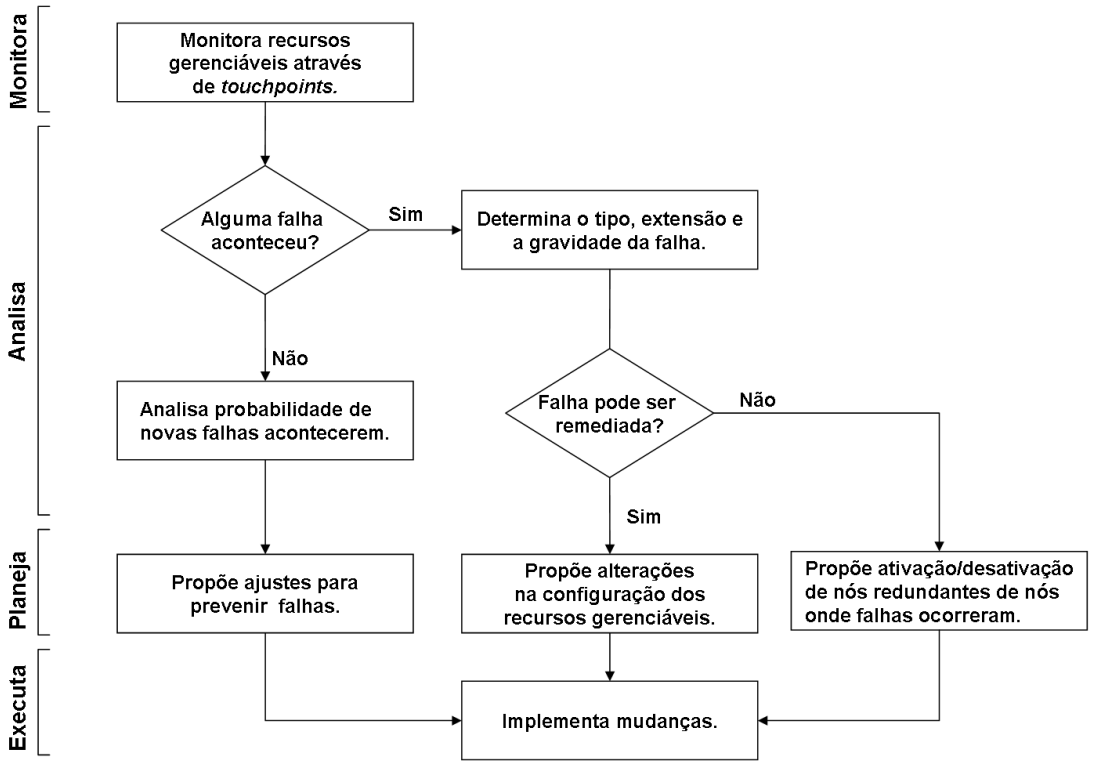


Figura 5.8: Responsabilidades do Gerenciamento de Continuidade em RSSF.

5.3.4 Gerenciamento de Capacidade em RSSFs

O gerenciamento de capacidade em RSSFs deve garantir que a rede possua recursos suficientes a fim de entregar um serviço de qualidade. Gerentes autonômicos incumbidos desta tarefa devem ser instanciados em cada nó sensor da rede, podendo atuar em nível de gerenciamento de elemento de rede (GANS) ou em nível de rede (GAC).

O gerenciamento de capacidade é o responsável por identificar demandas e em caso de capacidade insuficiente realocar recursos. Os gerentes responsáveis pela capacidade (GANS e GAC) devem ser capazes de monitorar os recursos do nó (energia residual, dados coletados, etc) através dos *touchpoints* de cada nó. Para isto estes gerentes utilizam a operação *Request-Response* para obter as informações necessárias do recurso gerenciável. Além disso, devem ser capazes de analisar estas informações e compará-las com um modelo de uso dos recursos, a fim de determinar se existe a demanda por mais recursos. Este modelo deve descrever os limites de funcionamento dos recursos do nó para que seja possível identificar uma demanda. Atuando de forma proativa, estes gerentes podem tentar identificar com antecedência a necessidade de novos recursos, utilizando, por exemplo, algoritmos de predição simples. Uma vez analisados os recursos, pode ser definido se um recurso foi esgotado ou se existem recursos "desperdiçados"(não utilizados em toda sua capacidade). Caso a capacidade dos seus recursos esteja esgotada, os gerentes tentam diminuir o consumo de recursos gradativamente de forma que o serviço não seja prejudicado. Caso existam recursos sub-utilizados, os gerentes notificam outros gerentes de que podem disponibilizar parte de seus recursos. A fim de executar estas ações o gerenciamento de capacidade em RSSFs deve ser capaz de reconfigurar parâmetros do nó para atender o plano de mudanças e ser capaz de enviar mensagens de notificação para outros gerentes avisando da disponibilidade de recursos.

5.3.5 Gerenciamento de Finanças em RSSFs

O gerenciamento de finanças no ITIL trata da supervisão e controle dos recursos monetários. No caso das RSSFs autonômicas o gerenciamento de finanças é responsável por outro tipo de custo: custo com energia, largura de banda, ou outra métrica de custo relevante para a RSSF e sua aplicação. O gerenciamento de finanças em RSSFs deve determinar o custo por atividade e controlar estes custos na provisão dos serviços. Esta responsabilidade de gerenciamento deve estar presente em cada nó sensor da rede, podendo atuar em nível de gerenciamento de elemento de rede (GANS) ou em nível de rede (GAC).

Os gerentes que implementam a responsabilidade de finanças (GANS e GAC) devem ser

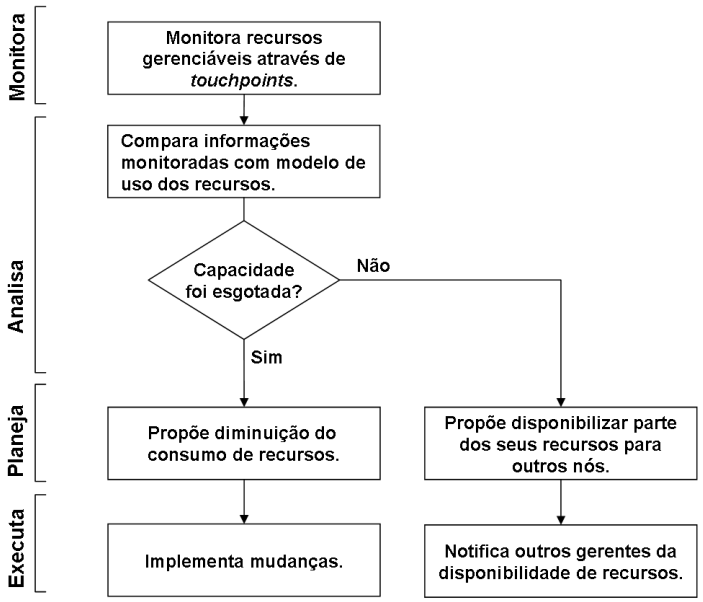


Figura 5.9: Responsabilidades do Gerenciamento de Capacidade em RSSF.

capaz de monitorar os recursos do nó (energia residual, largura de banda, etc) através dos *touchpoints* de cada nó. A obtenção destas informações é realizada através da operação *Request-Response*. Além disso, estes gerentes devem ser capazes de analisar estas informações a fim de determinar se o consumo previsto para cada atividade foi ultrapassado ou se não houve gastos adicionais. Para isto é necessário que este gerente possa predizer um custo fixo para realização de tarefas periódicas e estimar um custo variável no caso de eventos não-periódicos. Por exemplo, o gasto com energia em uma tarefa de sensoriamento e disseminação programados (considerando um custo com processamento constante) pode ser considerado fixo. No caso de eventos de prioridade, em que a RSSF passa a atuar sob-demanda, o consumo com energia é variável. Uma vez realizadas as tarefas de monitoração e análise, é necessário atualizar periodicamente as informações sobre custo de energia e outros tipos de custo na base de conhecimento, e se necessário realizar ajustes no orçamento pré-definido.

As responsabilidades de gerenciamento que constroem o sistema proposto podem não ser individualmente autonômicas, mas o sistema completo pode se comportar de forma autonômica coletivamente, se considerarmos uma perspectiva de gerenciamento de alto nível.

5.3.6 Base de Conhecimento

Em sistemas autonômicos, conhecimento é descrito como a informação que o gerente autonômico pode utilizar [MILLER, 2005], isto é, qualquer forma de dados estruturados ou

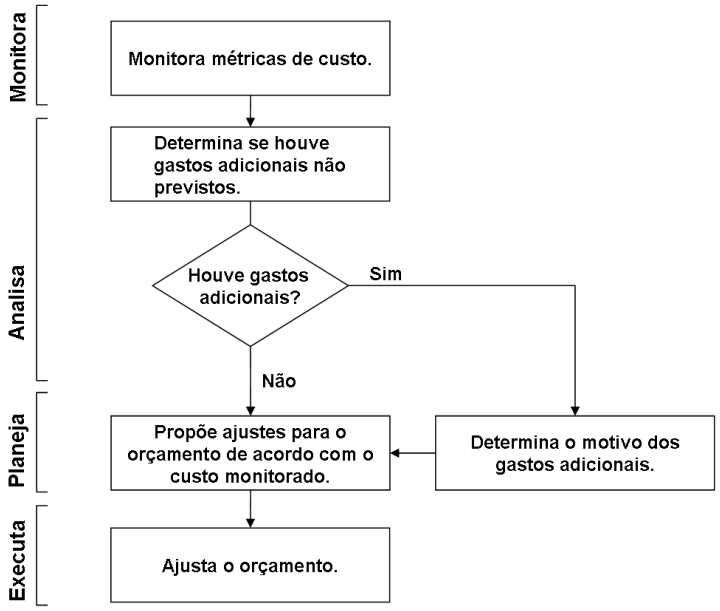


Figura 5.10: Responsabilidades do Gerenciamento de Finanças em RSSF.

informação que possa ser utilizada na execução de processos (dados sobre o estado de um *touchpoint* ou de um processo, sobre mudanças aplicada ao sistema, etc).

A base de conhecimento é conectada as quatro funções do gerente autonômico. Esta conexão significa que o conhecimento armazenado sobre a topologia da RSSF, sobre as políticas que regem o comportamento dos gerentes autonômicos, sobre dados monitorados, sobre incidentes detectados, entre outros, é usado em todas atividades do gerente autonômico. Este conhecimento pode ser utilizado na detecção de novos incidentes, reconhecimento de problemas, definição de métricas, reformulação de políticas, entre outros.

A base de conhecimento neste trabalho armazena informações sobre:

- políticas que regem o comportamento dos gerentes autonômicos;
- identificação, estado, propriedades, configurações e métricas disponibilizadas pelos *touchpoints*;
- dados monitorados;
- incidentes e problemas passados;
- soluções para plano de mudanças.

É importante salientar que as ações do gerente autonômico são baseadas nas políticas e

informações armazenadas na base de conhecimento, por isto é necessário que as informações sejam atualizadas periodicamente.

5.4 Considerações Finais

Este capítulo descreveu a principal contribuição deste trabalho: uma abordagem para o gerenciamento de RSSFs baseado na computação autonômica e na ITIL. Foram definidos os conceitos de recursos gerenciáveis, *touchpoints* e gerentes autonômicos no contexto das RSSFs e apresentadas as principais responsabilidades de cada um destes gerentes. As áreas de suporte e entrega de serviços da ITIL foram utilizadas na definição das responsabilidades dos gerentes autonômicos, considerando aspectos de capacidade, continuidade, disponibilidade, custo e nível de serviço. Através das tarefas de monitoração, análise, planejamento e execução estes gerentes contribuem para promover a produtividade dos recursos da rede e disponibilidade do serviço. O capítulo a seguir apresenta um estudo de caso que utiliza a abordagem proposta para o gerenciamento de uma aplicação de monitoração de cargas refrigeradas.

6 *Projeto de Simulação*

Este capítulo apresenta um estudo de caso utilizando a abordagem de auto-gerenciamento proposta. Neste estudo de caso, incidentes foram instanciados como problemas de comunicação. O sistema deverá detectar qual a causa raiz deste problema - no caso específico deste estudo de caso serão considerados problemas causados por congestionamento do tráfego de mensagens ou falta de energia. Após detectar operações impróprias ou problemas em componentes o sistema deverá se recuperar automaticamente dos impedimentos, mantendo a disponibilidade e continuidade do serviço.

A fim de avaliar a abordagem de gerenciamento proposta neste trabalho foram realizados experimentos simulados utilizando os gerentes autonômicos GANS, GAC e GAE (ver Seção 5.3) para o gerenciamento de uma rede hierárquica, e experimentos com uma rede que não apresenta nenhum serviço de gerenciamento. É esperado que a abordagem com gerenciamento aumente a disponibilidade da rede e promova continuidade do serviço. Os experimentos visam apresentar o impacto da utilização de funções de auto-gerenciamento, além de mostrar a viabilidade da solução proposta para as RSSFs.

A abordagem de auto-gerenciamento proposta neste trabalho, foi implementada neste estudo de caso para uma aplicação de monitoração de cargas refrigeradas. Esta é uma aplicação relevante no contexto brasileiro, já que a cadeia produtiva apresenta problemas de logística de transportes, dentre eles as condições precárias das vias de transporte, a deficiente infra-estrutura portuária para carga refrigerada e a falta de disponibilidade de contêineres. São problemas que interferem no custo de produção e na qualidade dos alimentos. Por causa desta deficiência, o Ministério da Agricultura, Pecuária e Abastecimento propôs em 2004 um planejamento para atender as novas demandas do agronegócio a partir das mudanças pelas quais o setor passará nos próximos 10 anos. Um dos objetivos deste planejamento é garantir a fiscalização sanitária à todos os agroprodutos em regime de trânsito internacional, a fim de assegurar o padrão de qualidade dos agroprodutos brasileiros, buscando uma maior competitividade internacional [MORAIS, 2007]. O estudo de caso apresentado neste trabalho pretende ser uma contribuição neste contexto no que diz respeito a fiscalização sanitária, através da monitoração

de cargas refrigeradas em contêineres.

Entendendo ser de grande importância o cuidado em transportar produtos no segmento de cargas refrigeradas, este estudo de caso trata da monitoração de aspectos relevantes à fiscalização sanitária, a saber: temperatura, umidade relativa do ar, luminosidade e vibração. Estes parâmetros permitem determinar se os produtos foram mantidos sob refrigeração constante e eficaz, tanto no transporte como durante o armazenamento, para que suas propriedades e características comerciais sejam preservadas. Se algumas destas etapas falhar ou for insuficiente, a qualidade do produto estará comprometida. Além disso, através da monitoração destes parâmetros também é possível determinar se os produtos refrigerados ou congelados foram mantidos sob temperatura e umidade especificadas em legislação, se o contêiner que transporta a carga foi aberto durante o transporte ou se, por exemplo, a carga foi danificada por algum tipo de queda durante o embarque e desembarque. Esta monitoração permite a criação de um relatório que comprove a qualidade dos produtos durante o transporte, geração de alertas em caso de problemas ou até determinar o descarte das cargas refrigeradas transportadas por desacordo com as normas sanitárias.

As seções a seguir descrevem detalhadamente a aplicação de monitoração de cargas refrigeradas, apresentam o projeto das RSSFs utilizadas neste estudo de caso e os requisitos que elas devem atender.

6.1 Estudo de Caso

Conforme mencionado anteriormente, este estudo de caso trata de uma aplicação para a monitoração da temperatura, umidade relativa do ar, luminosidade e vibração de cargas refrigeradas em contêineres. Decisões de arquitetura foram feitas considerando o tipo de aplicação e seus requisitos de QoS. No escopo deste trabalho, foi desenvolvida uma solução de gerenciamento através da implementação da função de auto-cura. Para isto foram definidas as tarefas do laço MAPE que controlam o funcionamento dos gerentes autonômicos propostos neste trabalho. Estes gerentes atuam na rede alterando o comportamento dos nós de acordo com as condições do contexto de operação e do ambiente, alterando o comportamento do nó para aumentar ou reduzir as atividades de sensoriamento, processamento e disseminação, controlar o consumo de energia e a produção da rede. Estas são tarefas importantes utilizadas pelos serviços de gerenciamento no projeto de uma RSSF auto-gerenciada.

A RSSF simulada deve atender aos seguintes requisitos propostos:

- *Throughput* de dados sensoriados: um observador externo à rede deve receber informações periódicas sobre os contêineres monitorados.
- Eventos de alta prioridade: os nós da rede devem informar o observador sobre eventos em que os parâmetros sensoriados excedam determinados limiares logo que a detecção seja feita.

Os gerentes autonômicos GANS, GAC e GAE devem colaborar entre si com o objetivo de atender estes requisitos e prover a qualidade do serviço. Os recursos gerenciáveis que servirão como fonte de informação para os gerentes autonômicos, e que eventualmente sofrerão alterações de suas configurações para garantir o cumprimento destes requisitos são o estado operacional do nó (*sleep/wake up*), o número de nós ativos, os intervalos de sensoriamento e de disseminação, e a energia residual dos nós sensores .

O acesso aos recursos gerenciáveis acontece por meio da interface de sensoriamento e atuação dos *touchpoints* definidas na Seção 5.2. A comunicação entre o *touchpoint* e os gerentes autonômicos foi implementada de duas maneiras diferentes, dependendo da localização do gerente. Na comunicação entre o *touchpoint* e o Gerente Autonômico dos Nós Sensores (GANS), que é instanciado no mesmo nó que o *touchpoint*, são realizadas operações de *Get* e *Set* (para as operações *Request-Response* e *Perform-Operation*) ou são disparados eventos (para as operações *Send-Notification* e *Solicit-Response*). Já no caso da comunicação com o Gerente Autonômico de *Cluster* (GAC) e o Gerente Autonômico Externo (GAE) são trocadas mensagens entre os gerentes e o *touchpoint* (*Request-Response Message*, *Send-Notification Message*, *Perform-Operation Message* e *Solicit-Response Message*).

TIPO	ID	TEMPO	OPERAÇÃO	PROPRIEDADE A SER ALTERADA	NOVO VALOR
int operationType Perform-Operation Request-Response Send-Notification Solicit-Response	int NodeId	float Timestamp	int actionType Get / Set	int property Dissemination interval Sensing interval Sleep / Wakeup	double data

Figura 6.1: Mensagens trocadas entre touchpoints e gerentes autonômicos.

Os gerentes autonômicos propostos tem seu comportamento guiado por políticas que tentam garantir o cumprimento dos requisitos definidos. As políticas relacionadas a cada responsabilidade de gerenciamento são descritas a seguir:

Disponibilidade: Se nós comuns têm problemas de comunicação, por excesso de mensagens trafegando na rede, os nós afetados diminuem sua taxa de sensoriamento e entrega de

mensagens a fim de diminuir colisões de mensagens na rede.

Continuidade: Se nós comuns têm problemas de comunicação, por excesso de mensagens trafegando na rede, gerentes de *cluster* podem colocar nós de seu grupo fora de serviço temporariamente a fim de diminuir o tráfego na rede.

Capacidade: Se não existem alertas sendo gerados e nenhum incidente relacionado a aplicação, os nós da rede tem suas atividades de sensoriamento e disseminação reduzidas a fim de poupar energia.

Finanças: Se um nó está com baixo nível de energia residual, reduzir as atividades de sensoriamento e disseminação do nó, mas sem comprometer os alertas da aplicação que devem continuar sendo gerados.

Com base nestas regras foram definidas funções de monitoração, análise, planejamento e execução para gerentes GANS e o GAC da rede. Neste estudo de caso o GAE monitora o nível do serviço, mas não atua sobre a rede.

As atribuições do GANS neste estudo de caso são:

Monitoração. Este gerente é responsável por determinar se alertas de temperatura, umidade, luminosidade ou vibração ocorreram; detectar se mensagens foram perdidas; e monitorar a energia residual do nó. Este gerente detecta a perda de mensagens, conferindo periodicamente se os nós da rede receberam mensagem de *ACK* para cada mensagem de dado sensoriado enviada. Com relação a energia residual do nó, no momento que a energia passa a ser menor que um limiar (2 Joules) determinado um incidente é detectado e o nó comum passa a disseminar apenas as mensagens consideradas de alta prioridade. Além disso, o gerente é capaz de detectar incidentes de aumento anormal de produção, que ocorre quando algum evento de interesse é sensoriado.

Análise. A diagnose do erro é realizada utilizando as informações armazenadas nas bases de conhecimento dos nós. Caso ocorra algum alerta relacionado a algum dos parâmetros sensoriados, tentar determinar a causa do problema, através da análise de todos parâmetros sensoriados. No caso das mensagens perdidas por nós comuns, uma das causas do problema que podem ser diagnosticadas com as informações armazenadas é o aumento anormal da produção dos nós da rede, fazendo com que os nós percam mensagens por falta de espaço na fila de pacotes.

Planejamento. Quando alertas são gerados, o gerente reconfigura o nó para informar o observador com maior freqüência dos dados coletados. No momento que a energia passa a ser menor que um limiar o nó comum diminui sua produção, mas sempre dissemina as mensagens com alertas, consideradas de alta prioridade. Além disso, assim que a função de análise determina que o problema de perda de mensagens foi causado pelo aumento da produção dos nós, um plano de mudanças será criado, tentando diminuir a produção do nó, até que mensagens não sejam mais perdidas. Para isto, os intervalos de sensoriamento e disseminação são aumentados gradativamente (de acordo com a taxa de mensagens que efetivamente foram entregues).

Execução. O plano de mudanças é executado e os intervalos de sensoriamento e disseminação são alterados. Enquanto mensagens ainda estiverem sendo perdidas ou alertas da aplicação sendo gerados o gerente autonômico continua ajustando estes parâmetros e avaliando o impacto sobre a rede.

As principais tarefas do GAC são descritas abaixo:

Monitoração. O GAC monitora a disponibilidade dos nós de seu grupo e identifica aqueles que estão fora de serviço. Este gerente também realiza a detecção de perda de mensagens através da conferência periódica de ACKs recebidos para cada mensagem enviada ao ponto de acesso. Este gerente também detecta incidentes de aumento de produção e de baixo nível de energia no grupo.

Análise. O GAC consulta sua base de conhecimento para diagnosticar se os nós que falharam por problemas de comunicação ou por falta energia, o que indica uma falha provisória, ou se tem suas baterias esgotadas, o que caracteriza uma falha permanente. No caso das mensagens perdidas por nós líderes, a base de conhecimento é consultada e analisada para diagnosticar se a falha é devida a aumento de produção, fazendo com que os nós percam mensagens por falta de espaço na fila de pacotes.

Planejamento. O GAC propõe a desativação/ativação de nós para remediar as falhas ocorridas. Se um incidente de baixa energia residual do grupo é detectado pelo nó líder, este escolhe alguns nós do seu grupo com menor energia residual e os deixa temporariamente fora de funcionamento. Uma mudança similar é realizada quando a produção da rede é anormal - o nó líder escolhe 20% dos nós do seu grupo com maior produção e os coloca fora de funcionamento temporariamente (o nó fica em estado *sleep* por 5 segundos). Estas ações visam aumentar o tempo de vida da rede e fazer com que seja possível entregar os dados produzidos pelos nós.

Execução. O plano de mudanças é executado e os intervalos de sensoriamento e disseminação são alterados. Enquanto mensagens ainda estiverem sendo perdidas o gerente autonômico continua ajustando estes parâmetros e avaliando o impacto sobre a rede.

As bases de conhecimento estão distribuídas nos nós da rede e líderes, sendo que os nós comuns só armazenam informações sobre o seu próprio estado. O nós líderes de *cluster* armazenam as informações sobre seu próprio estado e informações de gerência da rede, uma vez que é assumido neste estudo de caso que estes nós tem hardware com maior capacidade de comunicação e armazenamento. Uma base de conhecimento instanciada na entidade externa à rede, armazena informações de gerência de toda rede e informações sobre o funcionamento dos nós da rede. As decisões tomadas pelo GACs são prioritárias as decisões dos GANSs, porque é assumido que o gerente dos nós líderes tem maior conhecimento sobre o estado da rede e sobre como os nós podem melhor colaborar para manter a qualidade, disponibilidade e continuidade do serviço.

A base de conhecimento é atualizada quando os nós recebem mensagens para de alteração dos parâmetros de configuração ou quando estes são alterados localmente. Cada mensagem de um nó desconhecido recebida pelo nó líder, faz com que uma entrada para este nó seja criada na BC. A partir daí, os nós comuns periodicamente enviam uma mensagem contendo informações locais para atualizar as BCs dos nós líderes. Além disso, as BCs do nó líder são atualizadas a cada nova mensagem de dados sensoriados recebida, atualizando informações sobre a taxa de mensagens recebidas de cada nó.

As BCs dos nós comuns armazenam as seguintes informações: identificador do nó; energia residual; intervalos de sensoriamento e disseminação do nó; quantidade de bytes enviados a seu nó líder; taxa de mensagens enviadas por segundo; número de mensagens descartadas e sua quantidade em bytes; taxa de mensagens descartadas por segundo; e prioridade das mensagens. O nó líder, além de armazenar estas informações, possui informações sobre a energia residual de cada nó pertencente a seu grupo, a quantidade de dados enviados por cada um desses nós e seu intervalo de disseminação.

A aplicação construída a partir das funções acima foi simulada e os experimentos são apresentados na próxima seção.

Cenário	Número de Contêineres	Número de Nós Comuns por Contêiner	Número Total de Nós	Auto-cura
1a	6	6	43	Sim
1b	6	6	43	Não
2a	12	6	85	Sim
2b	12	6	85	Não
3a	18	6	127	Sim
3b	18	6	127	Não
4a	6	12	79	Sim
4b	6	12	79	Não

Tabela 6.1: Caracterização dos cenários simulados.

6.2 Experimentos

Esta seção apresenta os cenários de simulação bem como os experimentos realizados utilizando a ferramenta Network Simulator 2 (NS-2) [NS-2, 2005] e o MannaSim [LOPES et al., 2006], um arcabouço constituído por um conjunto de classes base que estende o NS-2 para simulação de aplicações de RSSFs.

Os cenários de simulação foram construídos considerando uma RSSF hierárquica de acordo com os cenários apresentados na Tabela 6.1. Nestes cenários existe um determinado número de nós comuns em cada contêiner (6 ou 12 nós) e um nó líder responsável por cada contêiner. Fora destes contêineres existe um ponto de acesso que recebe os dados sensoriados. Os nós líderes são colocados no centro do contêiner e os nós comuns depositados aleatoreamente. A comunicação entre nós comuns e nós líderes acontece em um único passo, assim como a comunicação entre os nós líderes e o ponto de acesso. Utilizando esta organização, foram realizados experimentos utilizando o sistema de auto-gerenciamento de serviços para RSSFs descrito na Seção 6.1 (cenários *a*) e outro que não implementa funcionalidades de auto-gerenciamento (cenários *b*).

Os parâmetros de configuração utilizados nas simulações realizadas são apresentados na Tabela 6.2. As características dos dois tipos de nós utilizados, comuns e líderes, foram configuradas de acordo com aquelas apresentadas pelos nós reais Mica Motes [CROSSBOW, 2003] e WINS [WINS, 2002], respectivamente. Cada cenário foi executado 33 vezes e os resultados apresentados na Seção 6.3 são referentes às médias dos valores obtidos. A dimensão do cenário varia de acordo com o número de contêineres simulados (a área ocupada por um contêiner é de 12m de comprimento por 11m de largura). Neste estudo de caso foram considerados o número de contêineres em 6, 12 e 18 utilizando 6 nós por contêiner, e um cenário com 6 contêineres e 12 nós comuns por contêiner.

Configurações da Rede	**Configurações das Simulações**
Protocolo de Transporte: UDP; *Protocolo MAC:* IEEE 802.11;	*Tempo de Simulação:* 300 segundos; *Número de Simulações:* 33;
Configuração dos Nós Líderes	**Configuração dos Nós Comuns**
Alcance: 250 metros; *Consumo com Processamento:* 0.360W; *Consumo com Transmissão:* 0.6W; *Consumo com Recepção:* 0.3W; *Consumo com Sensoriamento:* - *Tipo de Disseminação:* Programada; *Tipo de Sensoriamento:* - *Capacidade da Bateria:* 100J; *Largura de Banda:* 100kbps;	*Alcance:* 40 metros; *Consumo com Processamento:* 0.024W; *Consumo com Transmissão:* 0.036W; *Consumo com Recepção:* 0.024W; *Consumo com Sensoriamento:* 0.015W; *Tipo de Disseminação:* Programada; *Tipo de Sensoriamento:* Programada; *Capacidade da Bateria:* 5J; *Largura de Banda:* 28.8kbps;

Tabela 6.2: Caracterização das simulações executadas.

Parâmetro	**Valor médio**	**Desvio Padrão**	**Limiar**
Temperatura (^{o}C)	-16	2	-18
Umidade Relativa (%)	80	10	90
Vibração - Aceleração dinâmica(m/s^2)	0,1	0,001	variação de 10% na aceleração
Luminosidade (lumens/m^2)	10	2	12

Tabela 6.3: Valores sensoriados e seus limiares.

A aplicação implementada como cenário realiza a monitoração de temperatura, umidade, luminosidade e vibração. Os valores médios sensoriados e seu desvio padrão são apresentados na Tabela 6.3. Uma mensagem é considerada de alta prioridade quando um dos valores sensoriados excede os limiares apresentados na Tabela 6.3. Os nós comuns e pontos de acesso de cada contêiner agregam as mensagens coletadas e recebidas, respectivamente, e disseminam a mensagem agregada periodicamente.

A rede trabalha por 180 segundos coletando dados a cada 0.01 segundos e disseminando com um intervalo de 1 segundo. Para simular problemas de comunicação, é induzida uma elevação do valor de um dos parâmetros sensoriados e o intervalo de sensoriamento e disseminação dos nós comuns são diminuídos para 0.001 e 0.01 segundos, fazendo com que a produção de dados disseminados aumente e mensagens sejam perdidas por falta de espaço na fila de pacotes que comporta 10 mensagens nos nós comuns e 100 mensagens nos nós líderes. Este evento tem uma duração de 60 segundos, e em seguida a rede volta a operar com intervalos de sensoriamento e disseminação definidos no início da simulação. Os valores do tamanho da fila de pacotes e dos intervalos de sensoriamento e disseminação foram escolhidos com base em [ASSUNÇÃO; RUIZ; LOUREIRO, 2006].

Os nós da rede tentam detectar incidentes a cada um segundo. Além disso, a cada um segundo, os nós comuns enviam mensagens para atualizar as informações da base de conheci-

mento do nó líder.

6.3 Resultados

Nesta seção, são apresentados os resultados das simulações executadas e suas avaliações. A fim de analisar os cenários propostos na Seção 6.2 serão consideradas as seguintes métricas:

- Energia residual: esta métrica permite avaliar como os recursos energéticos da rede foram utilizados. Será avaliado o consumo de energia total dos nós comuns e dos nós líderes. Para isto é apresentado o valor médio de energia consumida, em Joules, para os cenários simulados, bem como seu desvio padrão. Além disso, é apresentado o decaimento de energia dos nós comuns, para a análise do consumo de energia durante o problema de comunicação simulado.
- Fluxo dos dados sensoriados: através desta métrica pode-se avaliar a utilização dos recursos de comunicação da rede, detectar problemas de comunicação e a correção destes problemas a fim de analisar a continuidade do serviço da rede. Para isto é apresentado o valor médio e o desvio padrão de mensagens enviadas pelos nós comuns e recebidas pelos nós líderes, e as mensagens enviadas pelos nós líderes e recebidas pelo ponto de acesso. Além disso é apresentado o valor médio das mensagens perdidas durante a simulação. A análise destes valores permite determinar se os gerentes autonômicos foram capazes de detectar e corrigir o problema de comunicação na rede.

6.3.1 Consumo de Energia

A energia é o recurso responsável pelo funcionamento de todos os módulos do nó sensor. Em casos de falha por esgotamento de energia, a produção da rede é diminuída de maneira irreversível.

Os gráficos das Figuras 6.2, 6.3, 6.4 e 6.5 apresentam o decaimento de energia dos nós comuns durante os 300 segundos de simulação. Os nós comuns dos cenários sem gerenciamento saem de serviço por esgotamento de energia antes do fim da simulação. O decaimento de energia nestes cenários, logo no início do evento de problema de comunicação, ocorre no intervalo de tempo $t = 180$ até $t = 240$ segundos. Já os nós comuns do cenário com gerenciamento, que implementam a rede descrita na Seção 6.1, sobrevivem por toda a simulação passando pelo evento de problema de comunicação por um período de 60 segundos, quando o nó detecta incidentes e se adapta para manter o serviço disponível.

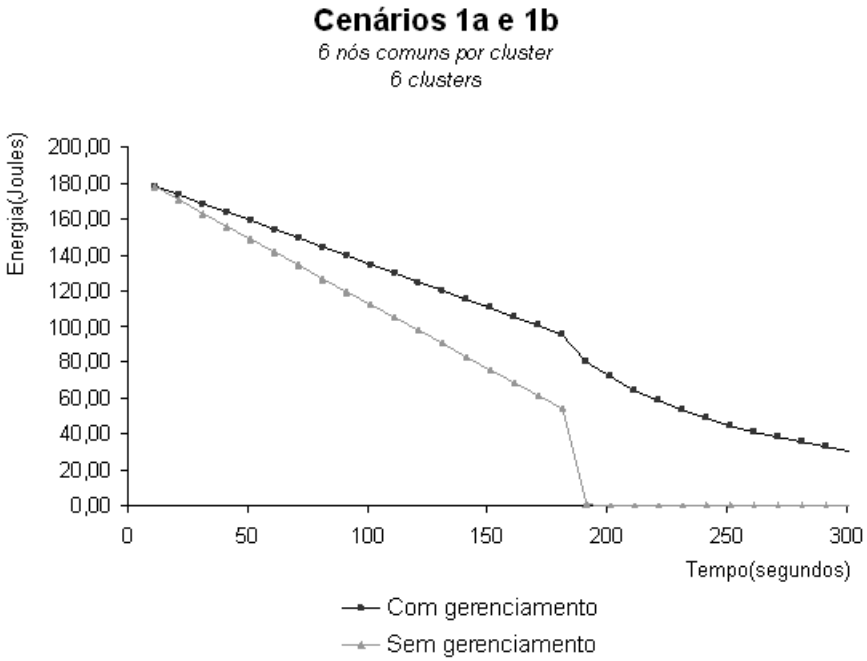


Figura 6.2: Comparativo do Decaimento de Energia - Cenário 1a e 1b.

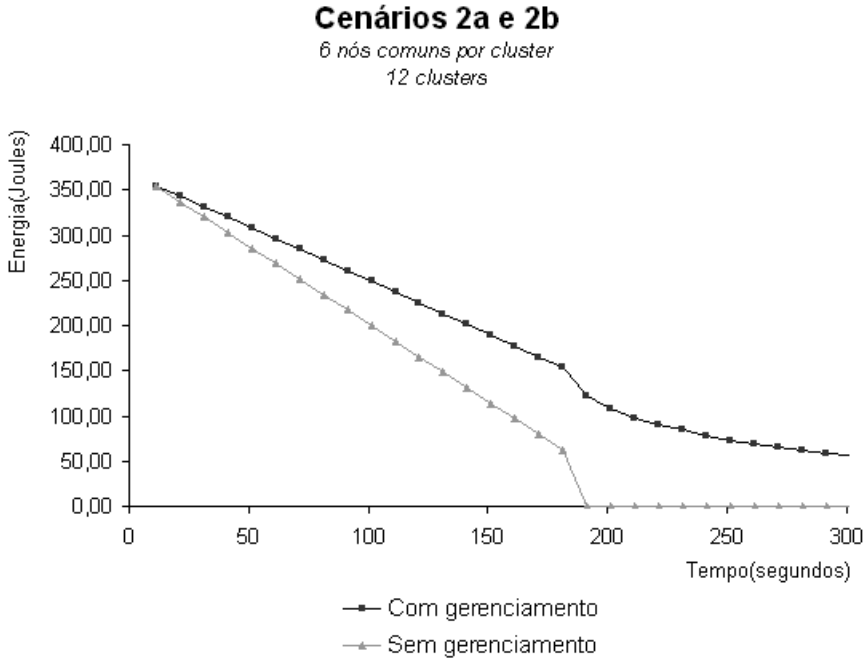


Figura 6.3: Comparativo do Decaimento de Energia - Cenário 2a e 2b.

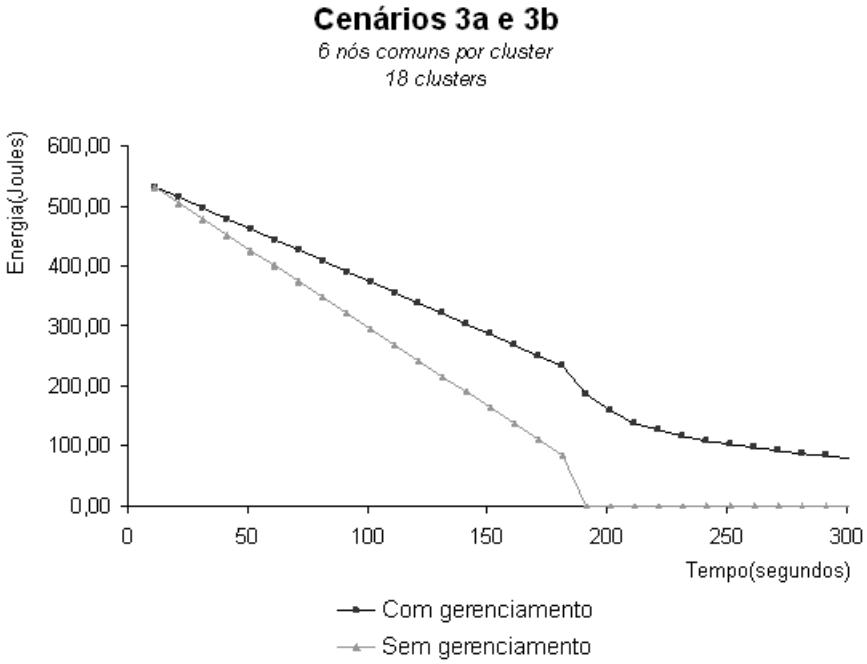


Figura 6.4: Comparativo do Decaimento de Energia - Cenário 3a e 3b.

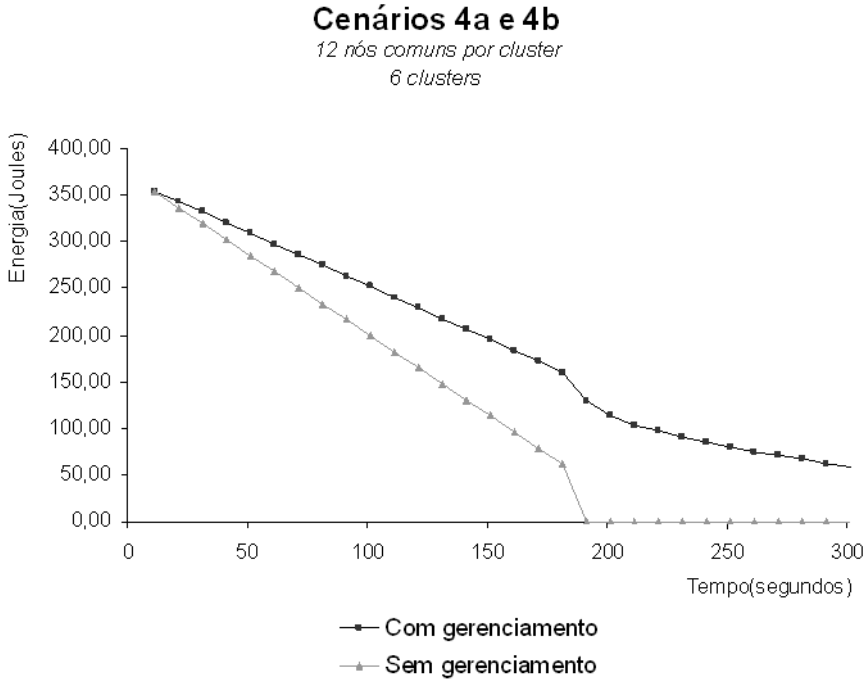
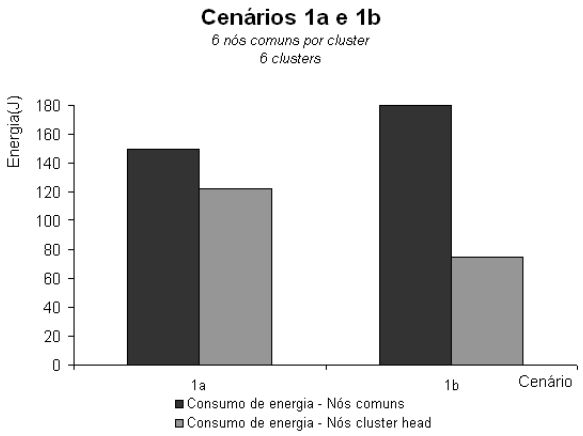
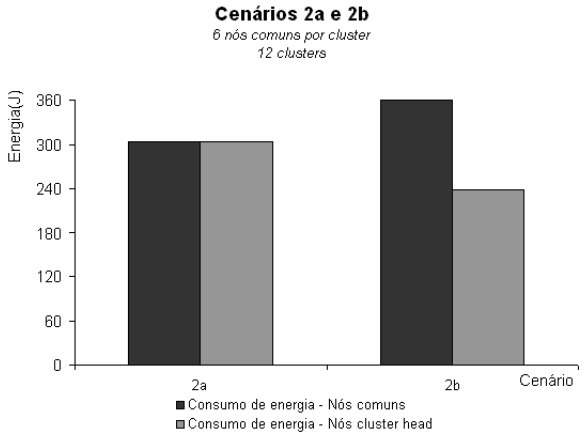


Figura 6.5: Comparativo do Decaimento de Energia - Cenário 4a e 4b.

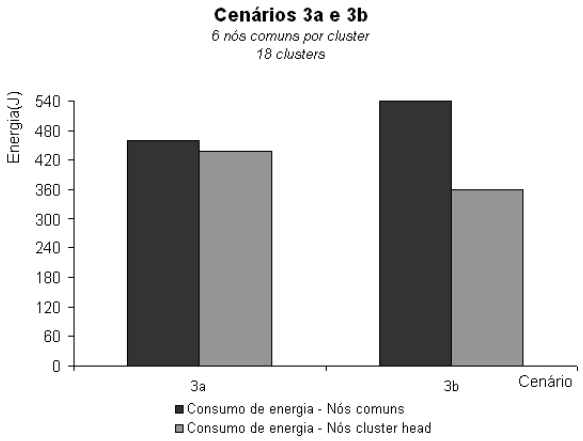
O gráfico da Figura 6.6 e a Tabela apresentam o consumo de energia dos nós comuns da rede para os cenários de simulação.



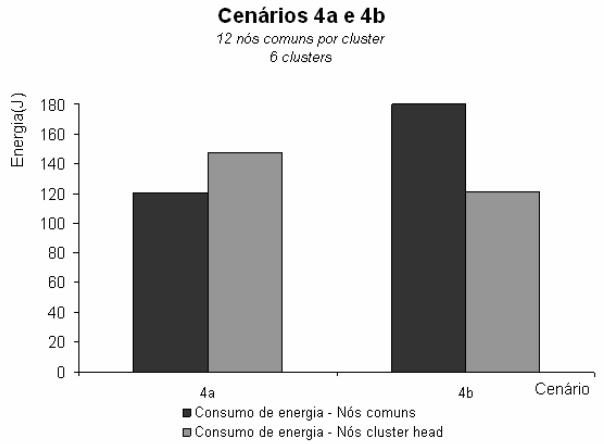
(a) Comparativo do Consumo de Energia - Cenário 1a e 1b



(b) Comparativo do Consumo de Energia - Cenário 2a e 2b.



(c) Comparativo do Consumo de Energia - Cenário 1a e 1b



(d) Comparativo do Consumo de Energia - Cenário 4a e 4b

Figura 6.6: Consumo de energia dos nós comuns e líderes.

O consumo de energia dos nós comuns dos cenários com gerenciamento é menor porque ao detectar que mensagens estão sendo perdidas a produção de cada nó rede é diminuída. É importante salientar que a rede permanece em funcionamento por mais tempo, e que mesmo assim seu consumo de energia é inferior ao dos cenários sem gerenciamento. Além disso, o fato do nó comum disseminar apenas informações prioritárias quando seu nível de energia é baixo, faz com que energia seja economizada.

No caso dos nós líderes, a energia consumida nos cenários com gerenciamento é maior, já que estes nós recebem mais mensagens dos nós comuns e por isso podem enviar mais mensagens para o ponto de acesso. A atuação dos GACs controlando o número de nós ativos ajuda a controlar o tráfego de mensagens na rede. No máximo 20% dos nós de um *cluster* são colo-

Cenário	Energia consumida pelos nós Comuns	Desvio Padrão	Energia Consumida pelos Nós Líderes	Desvio Padrão
1a	149,9469	6,13	122,294	8,71
1b	180	0,00	74,55	0,10
2a	303,78	10,54	303,073	17,63
2b	360	0,00	238,329	1,17
3a	460,2281	7,21	437,82	8,90
3b	540	0,00	360,81	2,86
4a	120,751	8,44	147,688	5,22
4b	180	0,00	120,927	0,08

Tabela 6.4: Consumo de energia dos nós comuns e líderes.

cados fora de operação, ou seja, nos cenários 1a, 2a e 3a no máximo 1 nó é posto em estado de *sleep* por *cluster* (em intervalos de 5 segundos), portanto o impacto na área de cobertura por contêiner não é grande, com a densidade de nós no pior caso diminuindo de 0,045 nos/m_2 para 0,038 nos/m_2. No caso do cenário 4a, 2 nós podem ser colocados fora de operação por 5 segundos e com isso a densidade no pior caso é diminuída de 0,091$nós/m^2$ para 0,076nos/m^2.

Análise dos Cenários

Cenários 1a e 1b: Os nós comuns do cenário 1a (ver gráfico 6.6(a)), com gerenciamento, consomem 16,70% menos energia que os do cenário 1b, enquanto os nós líderes consomem 39,04% de energia a mais. Antes do incidente de comunicação o consumo de energia dos nós comuns da rede sem gerenciamento corresponde em média a 7,27J, e durante os 10 primeiros segundos de problema de comunicação este consumo aumenta para 52,81J. Já a rede com gerenciamento que consumia em média 4,86J antes do incidente, consome nos primeiros 10 segundos do incidente 14,93J, e este valor decai se estabilizando novamente. Após alcançar o valor estável, o consumo continua diminuindo por causa da política de disseminar apenas mensagens prioritárias no caso de baixa energia residual dos nós do *cluster*. A atuação dos GANSs e dos GACs fazem com que os nós controlem a quantidade de mensagens trafegando na rede e como conseqüência, menos energia é consumida. Isto permite que a aplicação fique disponível por mais 60 segundos (até o fim da simulação) com sua energia residual final de 30,05J, de acordo com o gráfico 6.2.

Cenários 2a e 2b: Nestes cenários os nós comuns que implementam a solução de gerenciamento de serviços proposta consomem 15,62% menos energia do que a abordagem sem gerenciamento. Os nós líderes, por sua vez, consomem 21,36% a mais que os do cenário sem gerenciamento (ver gráfico 6.6(b)). O consumo de energia dos nós comuns do cenário sem os gerentes autonômicos definidos neste trabalho antes do incidente de comunicação é, em média, 17J a cada 10 segundos. Logo que o incidente se inicia (ver gráfico 6.3), estes nós consomem os

62,65J restantes e estes nós saem de operação por esgotamento de energia. Já os nós do cenário com gerenciamento que consumiam em média 11,84J a cada 10 segundos, passam a consumir nos primeiros 10 segundos do incidente 30,92J e este valor cai a medida que os GANSs ajustam os valores de intervalos de disseminação e sensoriamento, e os GACs colocam nós fora de operação.

Cenários 3a e 3b: Os nós comuns do cenário 3a, com gerenciamento, consomem 14,77% menos energia que os do cenário 3b, e os nós líderes que implementam a solução de gerenciamento consomem 17,59% de energia a mais (ver gráfico 6.6(c)). O consumo de energia, no cenário com gerenciamento, a cada 10 segundos antes do incidente é de 17,59J e durante os 10 primeiros segundos do incidente passa para 46,97J. No cenário sem gerenciamento este consumo passa de 26,21J para 85,22J durante o incidente, quando os nós comuns saem de operação por esgotamento de energia (ver gráfico 6.4). No cenário com gerenciamento os nós sobrevivem até o fim da simulação e apresentam energia residual de 79,77J após os 300 segundos de simulação.

Cenários 4a e 4b: Os nós comuns do cenário 4a, com gerenciamento, consomem 32,92% menos energia que os do cenário 4b, enquanto os nós líderes consomem 18,12% de energia a mais (ver gráfico 6.6(d)). A média do consumo de energia dos nós comuns do cenário sem os gerentes autonômicos, antes do incidente de comunicação, é 17,19J a cada 10 segundos. Assim que o incidente é gerado, estes nós consomem os 61,56J restantes e estes nós saem de operação por esgotamento de energia (ver gráfico 6.5). Já a rede com gerenciamento consome 11,42J inicialmente e nos 10 primeiros segundos do incidente consome 31,04J. A medida que o problema é reparado esta variação diminui. Os nós comuns da RSSF que implementa os GANS´s e GAC´s possui energia residual de 452,31J no fim da simulação, enquanto a rede sem gerenciamento sai de serviço nos 10 primeiros segundos do problema de comunicação.

6.3.2 Fluxo dos Dados Sensoriados

Uma variação nos parâmetros de temperatura, umidade, vibração e luminosidade foi simulada para que mensagens cujos parâmetros sensoriados excederem determinados limiares sejam enviadas logo que a detecção seja feita. Com isso a quantidade de mensagens na rede aumenta e mensagens começam a ser perdidas. A análise do fluxo dos dados sensoriados permite determinar se os gerentes autonômicos foram capazes de detectar e corrigir o problema de comunicação na rede.

A Tabela 6.5 apresenta o fluxo de dados sensoriados na rede. A quantidade de dados enviados pelos nós comuns nos cenários com os gerentes autonômicos GANSs e GACs são menores

Cenário	Mensagens Enviadas por Nós Comuns	Desvio Padrão	Mensagens Recebidas por Nós Líderes	Desvio Padrão	Mensagens Enviadas por Nós Líderes	Desvio Padrão	Mensagens Recebidas pelo Ponto de Acesso	Desvio Padrão
1a	13712,55	1439,48	12886,82	1509,73	889,42	4,15	887,52	4,47
1b	34320,82	146,75	10207,00	74,98	555,06	1,12	536,88	2,00
2a	17936,25	1201,27	15109,00	846,84	1726,50	36,10	1721,75	35,60
2b	44166,52	498,38	14873,48	77,14	1090,24	1,32	1081,67	2,26
3a	30919,90	1330,44	27342,00	1306,44	2683,33	235,72	2551,70	223,47
3b	60611,50	1721,00	29859,40	925,14	1629,76	1,92	1508,64	55,78
4a	18114,40	1096,89	15744,70	934,34	890,15	3,65	882,00	3,26
4b	43362,30	240,78	15110,70	380,79	545,73	0,45	534,03	1,22

Tabela 6.5: Fluxo dos dados sensoriados na rede.

que as dos cenário sem gerenciamento. No entanto a quantidade de dados recebidos pelos nós líderes do cenário com gerenciamento são maiores que a do cenário que não implementa a solução proposta. Isto significa que os gerentes autonômicos auxiliam na economia de energia, já que alteram o comportamento do nó para enviar e descartar menos dados porque tentam entregar mensagens quando a rede não está congestionada. Ainda assim o número de mensagens entregues aos nós líderes é maior se comparado ao cenário sem gerenciamento. Isto demonstra que realmente o número de mensagens perdidas na rede foi diminuído.

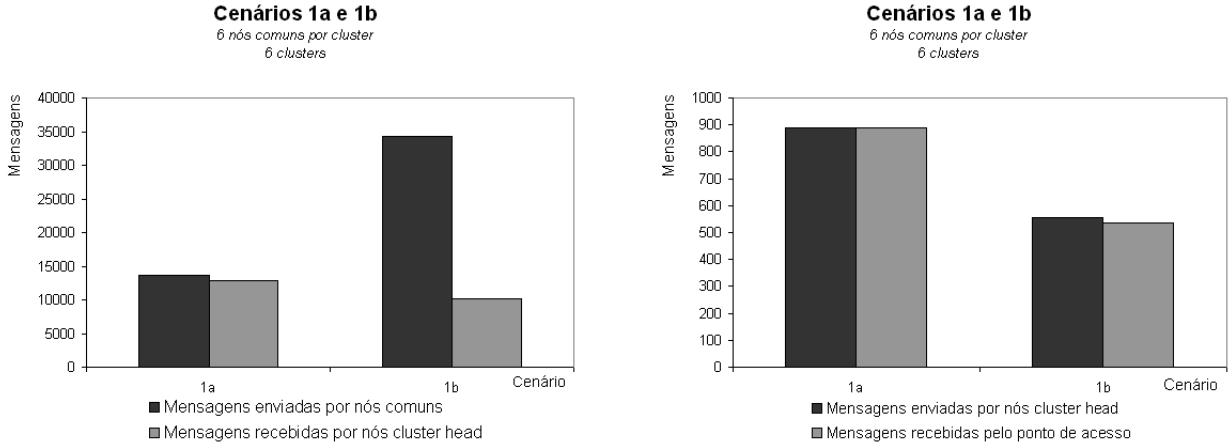
As mensagens recebidas por nós líderes são agregadas e então enviadas ao ponto de acesso. Por este motivo o número de mensagens enviadas por este nó é bem menor que o número de mensagens recebidas. A quantidade de dados enviados pelos nós líderes ao ponto de acesso no cenário com os GANSs e GACs é maior que a do cenário sem gerenciamento. Isto acontece porque o nó líder decidiu diminuir o seu intervalo de disseminação em alguns períodos para que fosse capaz de entregar as mensagens recebidas pelos nós do seu grupo.

Análise dos Cenários

Cenários 1a e 1b:

Os nós comuns do cenário sem gerenciamento enviam 250,29% mais mensagens que os nós do cenário com os GANS's e GAC's. No entanto apenas 29,74% destas mensagens efetivamente chegam ao nó líder (ver gráfico 6.7(a)). No cenário que implementa a solução proposta 93,98% das mensagens são entregues. A quantidade de mensagens entregues por este cenário é 20,80% maior que a do cenário sem gerenciamento. Isto mostra que menos mensagens foram descartadas porque o GANS ajusta a taxa de entrega do nó de forma a não congestionar a rede.

No caso dos nós líderes, o cenário sem gerenciamento envia 37,59% menos mensagens que

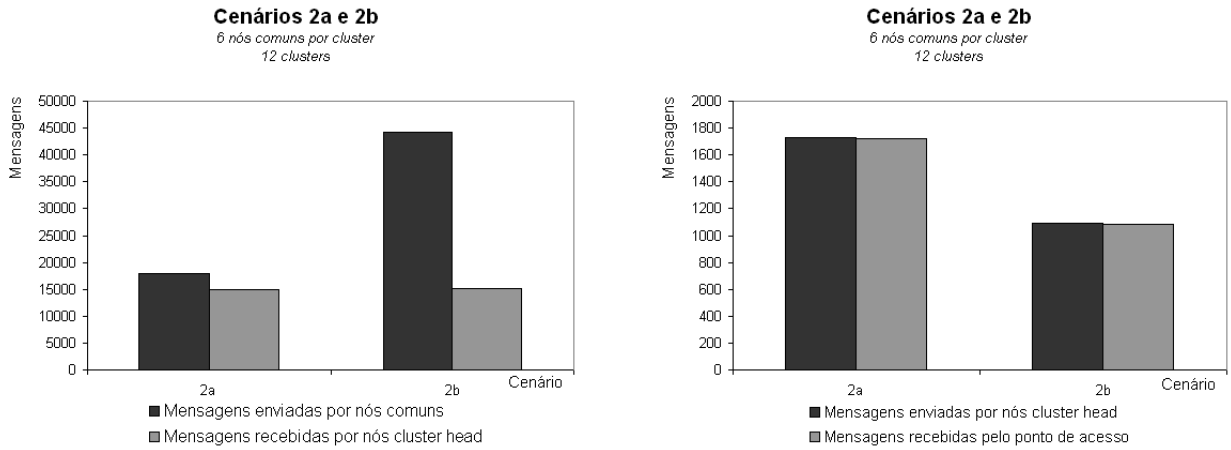


(a) Mensagens enviadas por nós comuns e recebidas pelos nós líderes.

(b) Mensagens enviadas por nós líderes e recebidas pelo ponto de acesso.

Figura 6.7: Cenários 1a e 1b - Fluxo dos dados sensoriados na rede.

os nós que implementam funcionalidades de auto-gerenciamento (ver gráfico 6.7(b)). Mesmo enviando menos mensagens a taxa de mensagens que efetivamente foi entregue ao ponto de acesso foi 96,72% enquanto no cenário com gerenciamento esta taxa chega a 99,79%. Isto mostra que os nós líderes do cenário que implementa os gerentes autonômicos definidos neste trabalho enviam mais mensagens ao ponto de acesso e ainda tem uma taxa de entrega melhor que a do outro cenário.



(a) Mensagens enviadas por nós comuns e recebidas pelos nós líderes.

(b) Mensagens enviadas por nós líderes e recebidas pelo ponto de acesso.

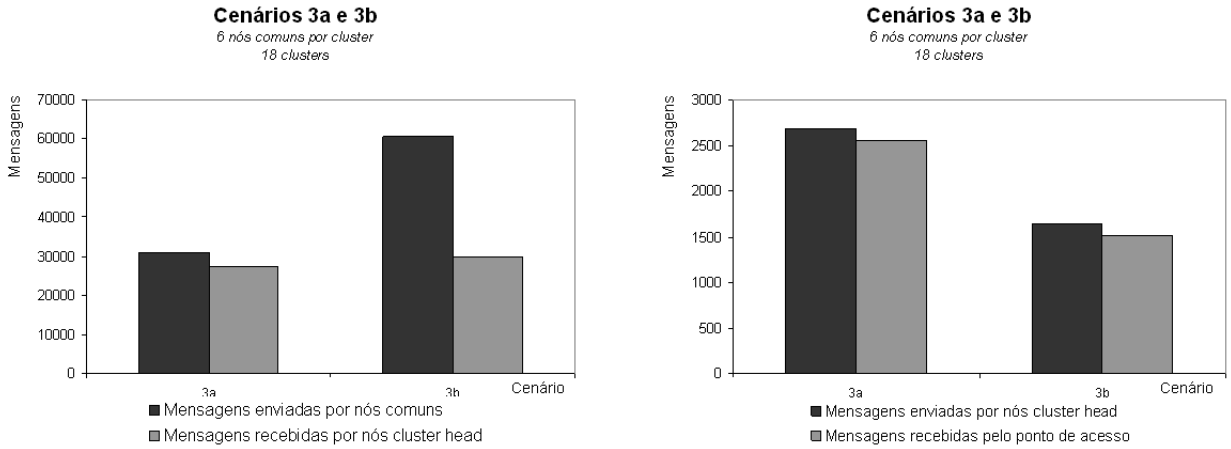
Figura 6.8: Cenários 2a e 2b - Fluxo dos dados sensoriados na rede.

Cenários 2a e 2b:

Os nós comuns do cenário que não possuem GANS enviam 246,24% mais mensagens que os nós do cenário com estes gerentes. No entanto 33,68% destas mensagens chegam ao nó líder

enquanto no cenário com gerenciamento 84,28% das mensagens são entregues (ver gráfico 6.8(a)). A quantidade de mensagens entregues por este cenário é apenas 1,59% maior que a do cenário sem gerenciamento, o que mostra que o ajuste realizado pelos GANSs permitiu que mensagens fossem entregues quando a rede não estivesse congestionada.

No caso dos nós líderes, o cenário sem gerenciamento envia 36,85% menos mensagens que os nós que implementam funcionalidades de auto-gerenciamento (ver gráfico 6.8(b)). A taxa de mensagens que efetivamente foi entregue ao ponto de acesso para os dois cenários foi bem próxima, 99,21% sem gerenciamento e 99,73% com gerenciamento. Ao comparar o cenário 2b com cenário 1b (sem gerenciamento), o aumento dos número de nós líderes tem um impacto grande na solução sem gerenciamento, enquanto nos cenários 1a e 2a (com GANS e GAC) este impacto foi pequeno, já que a taxa de entrega continua bem alta.



(a) Mensagens enviadas por nós comuns e recebidas pelos nós líderes.

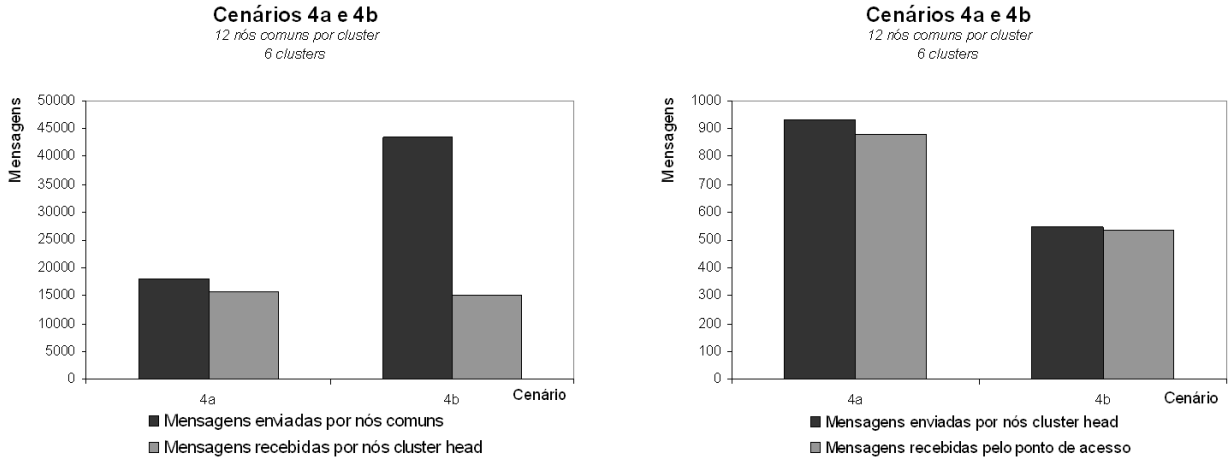
(b) Mensagens enviadas por nós líderes e recebidas pelo ponto de acesso.

Figura 6.9: Cenários 3a e 3b - Fluxo dos dados sensoriados na rede.

Cenários 3a e 3b:

No cenário sem gerenciamento, os nós comuns enviam 196,08% mais mensagens que os nós do cenário com os GANS's e GAC's. Ainda que 49,26% destas mensagens tenham sido perdidas, neste cenário a quantidade de mensagens entregues ao nó líder foi 9,21% maior que no cenário com gerenciamento (ver gráfico 6.9(a)). No cenário que implementa a solução proposta 88,43% das mensagens são entregues. O GANS continuou ajustando a taxa de entrega do nó diminuindo o número de mensagens perdidas, mas este ajuste não correspondeu a capacidade máxima de entrega que a rede suportaria, já que no cenário sem gerenciamento, apesar do descarte de mensagens ser bem maior, o que leva a um consumo de energia com envio de mensagens também maior, a taxa de mensagens entregues foi maior.

A taxa de mensagens que efetivamente foi entregue ao ponto de acesso foi 92,57% no cenário sem gerenciamento enquanto no cenário os gerentes propostos neste trabalho esta taxa chega a 95,10% (ver gráfico 6.9(b)). Isto significa que os nós líderes do cenário que implementa a solução proposta enviam mais mensagens ao ponto de acesso e ainda tem uma taxa de entrega melhor que a do outro cenário.



(a) Mensagens enviadas por nós comuns e recebidas pelos nós líderes.

(b) Mensagens enviadas por nós líderes e recebidas pelo ponto de acesso.

Figura 6.10: Cenários 4a e 4b - Fluxo dos dados sensoriados na rede.

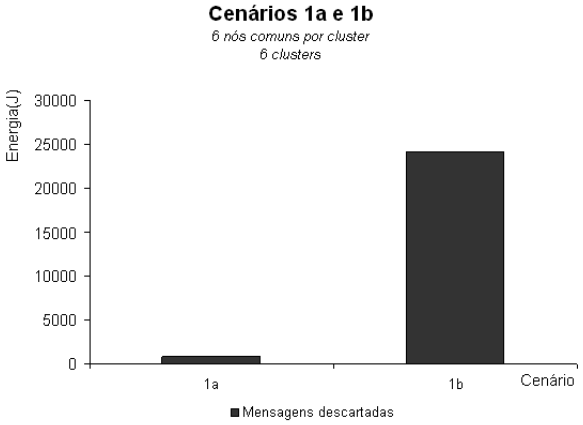
Cenários 4a e 4b:

Os nós comuns do cenário sem gerenciamento enviam 239,38% mais mensagens que os nós do cenário com os GANS's e GAC's. Destas mensagens 34,85% efetivamente chegam ao nó líder (ver gráfico 6.10(a)). No cenário que implementa a solução proposta 86,92% das mensagens são entregues. A quantidade de mensagens entregues por este cenário é 4,03% maior que a do cenário sem gerenciamento. Isto mostra que o ajuste dos intervalos de sensoriamento e disseminação propostos pelo GANS permitiu que menos mensagens fossem descartadas diminuindo o congestionamento na rede.

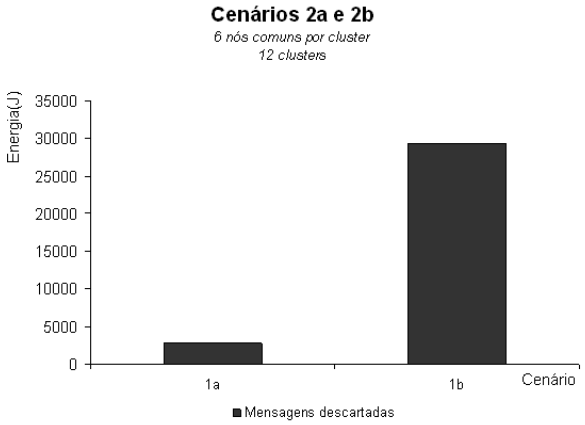
No caso dos nós líderes, o cenário com gerenciamento envia 58,41% mais mensagens que os nós que não implementam funcionalidades de auto-gerenciamento. A taxa de mensagens que efetivamente foi entregue ao ponto de acesso utilizando o GAC foi de 94,40% enquanto no cenário sem gerenciamento esta taxa corresponde a 97,86% (ver gráfico 6.10(b)). Isto mostra que o aumento do número de nós comuns no contêiner foi melhor tratado pelos GANS's do que pelos GAC's. O gerente dos nós líderes conseguiu diminuir o congestionamento enquanto o ajuste proposto pelo gerente do nó líder não foi tão eficaz. Ainda assim, considerando a solução completa o congestionamento na rede foi diminuído.

6.3.3 Perda de Mensagens

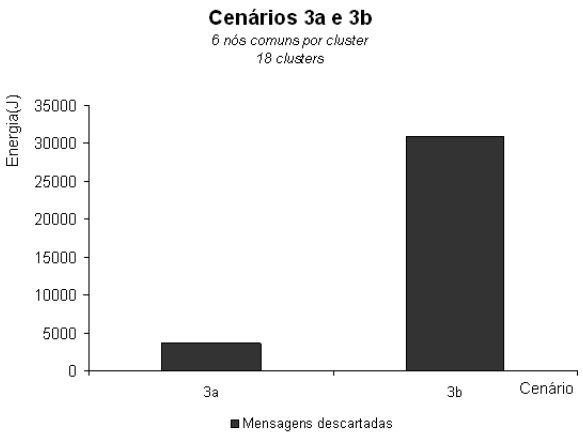
A quantidade de mensagens perdidas é apresentada na Figura 6.11. O cenário sem gerenciamento perde mais mensagens que o cenário com a solução proposta, isto porque não existe nenhuma maneira de detectar problemas de comunicação e nem como corrigi-los.



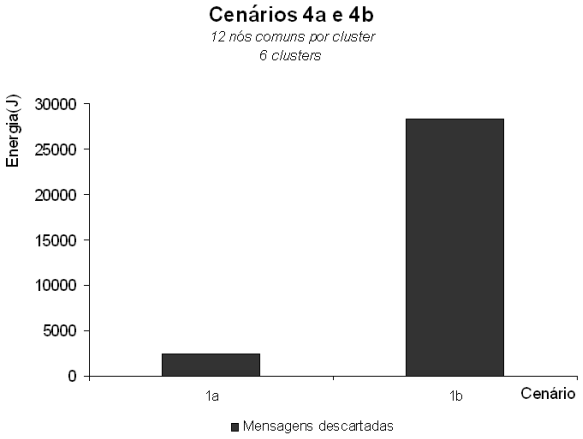
(a) Comparativo da Perda de Mensagens - Cenário 1a e 1b



(b) Comparativo da Perda de Mensagens - Cenário 2a e 2b.



(c) Comparativo da Perda de Mensagens - Cenário 3a e 3b



(d) Comparativo da Perda de Mensagens - Cenário 4a e 4b

Figura 6.11: Mensagens Perdidas.

Análise dos Cenários

Cenários 1a e 1b:

De acordo com o gráfico 6.11(a), o cenário sem gerenciamento perde 96,57% mais mensagens que o cenário com os gerentes propostos nesta dissertação. A quantidade de mensagens perdidas pelos nós líderes em ambos cenários é pouco significativa se comparado a porcentagem

de mensagens perdidas por nós comuns, a saber: no 99,92% no cenário sem gerenciamento e 99,78% no cenário que utilizam os GANS's e GAC's.

Cenários 2a e 2b:

Nestes cenários a quantidade de mensagens perdidas foi 90,34% maior para o cenário sem gerenciamento, sendo que os nós comuns são responsáveis pela perda de 99,97% das mensagens no cenário 2b e por 99,83% no cenário 2a (ver gráfico 6.11(b)) . A utilização dos gerentes propostos diminui a perda de mensagens de 29301 para 2832.

Cenários 3a e 3b:

A perda de mensagens foi diminuída em 87,98% utilizando a abordagem de gerenciamento proposta. Neste cenário, a contribuição dos nós líderes na perda de mensagens foi maior do que a dos outros cenários, correspondendo a 3,68% das mensagens descartadas (ver gráfico 6.11(c)). Este aumento é justificado pelo aumento do número de nós líderes, que disseminam mensagens com uma potência de transmissão maior e que por isso a probabilidade de haver colisão de pacotes é maior. Os nós líderes do cenário sem gerenciamento são responsáveis por 99,61% das perdas enquanto no cenário com gerenciamento este valor é de 96,45%.

Cenários 4a e 4b:

A quantidade de mensagens perdidas pelo cenário sem gerenciamento é 91,43% maior que no cenário que utiliza a solução proposta neste trabalho, de acordo com o gráfico 6.11(d). Os nós comuns do cenário 1a são responsáveis por 97,84% das mensagens perdidas. No cenário 1b os nós comuns são responsáveis por 99,96% das perdas. Como os nós líderes com os GAC's enviam mais mensagens, como esperado, estes nós também contribuem mais com a perda de mensagens.

Analisando todos estes resultados a solução proposta se mostrou eficiente na correção de problemas de comunicação da rede e promoveu uma melhor disponibilidade de serviço e longevidade da rede. Os resultados obtidos em cenários que implementam a abordagem de auto-gerenciamento mostram que esta pode ser uma ótima solução para o gerenciamento falhas, considerando que estas não são exceções em RSSFs.

6.4 Considerações Finais

Este capítulo apresentou um estudo de caso para a abordagem de gerenciamento de serviços proposta nesta dissertação. Primeiramente foi apresentada a aplicação de monitoração de cargas refrigeradas e a motivação para se estudar esta aplicação. Neste capítulo também foi descrito o

projeto da simulação, detalhes da simulação e cenários implementados. Também foram apresentados os parâmetros utilizados na configuração da rede, nós sensores e aplicação. O processo de avaliação por meio de simulação, apesar de não refletir todos detalhes de um experimento real, permite um bom nível de detalhamento e testes com uma grande quantidade de nós.

Este estudo de caso visou demonstrar a viabilidade da solução proposta para as RSSFs. As métricas de simulação consideradas foram a energia residual dos nós, a fim de avaliar como os recursos energéticos da rede foram utilizados, e o fluxo dos dados sensoriados, que permite avaliar a utilização dos recursos de comunicação da rede, detectar problemas de comunicação e a correção destes problemas.

A Seção 6.3.1 mostrou que utilizando a abordagem de gerenciamento de serviços proposta neste trabalho os nós conseguem se manter em operação por mais tempo, já que detectam incidentes e se adaptam para manter o serviço disponível. Nos casos sem gerenciamento, devido a falha por esgotamento de energia, a produção da rede é diminuída de maneira irreversível. Os resultados apresentados nas seções 6.3.2 e 6.3.3 mostram que os gerentes autonômicos auxiliam na economia de energia já que alteram o comportamento do nó para enviar e descartar menos dados porque tentam entregar mensagens quando a rede não está congestionada, diminuindo o número de mensagens perdidas na rede.

Os resultados obtidos mostram que a abordagem de gerenciamento de serviços pode ser implementada em RSSFs e que se mostrou eficaz como solução que integra serviços e gerenciamento de RSSFs promovendo melhor disponibilidade do serviço e longevidade da rede.

7 Conclusão

Este trabalho considera as RSSFs como ferramentas ou sistemas de TI, já que elas produzem, processam, armazenam e entregam seus dados. Como um sistema de TI, as RSSFs são baseadas nos seguintes componentes: hardware e software dos nós sensores, comunicação entre os elementos de rede, e informação de gerenciamento produzida pela rede. Este trabalho também propõe que as RSSFs sejam desenvolvidas como sistemas autonômicos que implementam diferentes serviços de gerenciamento, tais como: auto-organização, auto-configuração, auto-diagnóstico e auto-cura. Para tanto foi definida uma solução aberta que permite a integração com outras soluções de gerenciamento de outras redes no futuro. A biblioteca ITIL, neste contexto, é uma tecnologia que permite especificar serviços e funções de gerenciamento em nível de negócios e que isto é fundamental ao se considerar a integração das RSSFs com outras redes e com a Internet. Além disso, este trabalho trata a computação autonômica e a ITIL como tecnologias para a integração das soluções individuais para RSSFs que vem sendo propostas na literatura.

O foco desta dissertação foi a definição de uma abordagem de gerenciamento de serviços para RSSFs utilizando conceitos de gerenciamento de serviços da biblioteca ITIL modelados como tarefas de monitoração, análise, planejamento e execução de gerentes autonômicos. Uma RSSF autonômica foi modelada de forma que os nós sensores e seus componentes (hardware e software) correspondessem aos recursos gerenciados dos gerentes autonômicos. Foi definida também uma interface de gerenciamento para os nós sensores - o *touchpoint* - que permite que os gerentes autonômicos monitorem e atuem sobre os nós da rede. Foram definidos três tipos de gerentes autonômicos GANS, GAC e GAE. O GANS é instanciado em cada elemento da rede e controla a utilização dos recursos do nó, e identifica e recupera o nó de eventuais falhas em seus componentes. O GAC é instanciado nos nós líderes de redes hierárquicas e controla o funcionamento dos gerentes autonômicos dos nós pertencentes ao seu grupo. O GAE é instanciado externamente a rede, tem conhecimento sobre o serviço provisionado e pode ajustar os objetivos e comportamento da rede. Estes gerentes implementam o serviço de auto-cura baseados na área funcional de gerenciamento de falhas, nos níveis de gerenciamento de elemento de

rede, rede e serviço, e na funcionalidade de manutenção de RSSFs, definidos na organização tridimensional da arquitetura de gerenciamento Manna [RUIZ, 2003]. O GANS, GAC e GAE supervisionam recursos gerenciáveis e, a partir de consultas a bases de conhecimento, definem se mudanças são necessárias por causa da ocorrência de incidentes. Se incidentes ocorreram, os gerentes definem parâmetros de mudança, de acordo com a gravidade e extensão do problema e aplicam as mudanças à rede, atualizando em seguida as bases de conhecimento. Estas tarefas são realizadas considerando o nível de serviços de uma RSSF, disponibilidade, continuidade, capacidade e finanças - de acordo com o gerenciamento de entrega de serviços da biblioteca ITIL. Em nosso conhecimento, nenhum trabalho tem sido proposto na literatura considerando o uso da biblioteca ITIL para RSSFs autonômicas.

Esta dissertação apresentou um estudo de caso utilizando a abordagem de auto-gerenciamento proposta para uma aplicação de monitoração de cargas refrigeradas. Neste estudo de caso, incidentes foram instanciados como problemas de comunicação entre os nós da rede. Foram definidas as tarefas de monitoração, análise, planejamento e execução, considerando as responsabilidades de gerenciamento definidas na Seção 5.3. Os gerentes detectam incidentes e propõem alterações para que o nó se recupere automaticamente dos impedimentos, mantendo a disponibilidade e continuidade do serviço. Os resultados do trabalho indicam que há relevante importância no uso de ITIL no âmbito das RSSFs e apontam para benefícios ainda maiores ao se ampliar o uso de ITIL além das áreas de suporte e entrega de serviços. Estes resultados, obtidos em cenários que implementam o sistema de auto-gerenciamento, mostram que esta pode ser uma ótima solução para a integração de diferentes tarefas das redes, assim como, uma solução adequada para se gerenciar falhas, considerando que estas não são exceções em RSSFs.

O projeto e a implantação de RSSFs autonômicas representam uma nova oportunidade de pesquisa e, em especial, quando essas redes são definidas sob o paradigma de TI. Especificamente no que diz respeito a solução proposta neste trabalho, existem alguns desafios que podem ser vistos como trabalhos futuros, a saber:

- implementação de soluções autonômicas em nós sensores reais a fim de avaliar detalhes que não são considerados em simulações, apesar da possibilidade de realizar testes com uma grande quantidade de nós e um bom nível de detalhamento.

- avaliação da solução utilizando algoritmos de correlação de dados ou aprendizado na tarefa de análise dos gerentes autonômicos, para melhorar a identificação de tendências do nível do serviço e disponibilidade da rede e auxiliar a atuação proativa dos gerentes autonômicos.

- extensão da resolução de conflitos entre decisões de gerentes de diferentes níveis utilizando alguma forma de negociação ao invés de assumir que ações de gerentes com maior conhecimento sempre tem prioridade).
- definição de uma base padrão de incidentes, problemas e soluções para RSSFs de forma a facilitar a integração destas redes utilizando a abordagem proposta.
- inclusão de uma linguagem padronizada para definição das políticas, como por exemplo a *Autonomic Computing Policy Language* [IBM, 2007] que está sendo desenvolvida pela IBM.

As RSSFs autonômicas promovem a independência da intervenção humana em tarefas de manutenção e gerência, reduzindo os custos de comunicação, o tempo de resposta a eventos e aumentando significativamente a disponibilidade da rede. As soluções autonômicas também promovem a escalabilidade e a qualidade dos serviços, além dos benefícios associados com o negócio.

Referências Bibliográficas

AKYILDIZ, I. et al. A survey on sensor networks. *IEEE Communications Magazine 40*, v. 8, p. 102–114., 2002.

AKYILDIZ, I. F. et al. Wireless sensor networks: a survey. *Computer Networks (Amsterdam, Netherlands: 1999)*, v. 38, n. 4, p. 393–422, 2002.

ARORA, A. et al. A line in the sand: A wireless sensor network for target detection, classification, and tracking. *Computer Networks (Elsevier)*, 2004. Disponível em: <http://www.cse.msu.edu/~sandeep/publications/lites04>. Acesso em: 11/08/2007.

ASSUNÇÃO, H. P.; RUIZ, L. B. Uma abordagem de auto-gerenciamento de serviços em redes de sensores sem fio. In: *Anais do XII Workshop de Gerência e Operação de Redes e Serviços (WGRS 2007)*. Belém, PA, Brasil: [s.n.], 2007.

ASSUNÇÃO, H. P.; RUIZ, L. B.; LOUREIRO, A. A. F. A service management approach for self-healing wireless sensor networks. In: GAITI, D. et al. (Ed.). *Autonomic Networking*. [S.l.]: Springer, 2006. (Lecture Notes in Computer Science, v. 4195), p. 215–228. ISBN 3-540-45891-3.

BAIRD, S. et al. Communicating data from wireless sensor networks using the hl7v3 standard. In: *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*. [S.l.: s.n.], 2006. p. 4pp.

BAZAN, O.; JASEEMUDDIN, M. Routing in autonomic communications. In: *Consumer Communications and Networking Conference, 2006. CCNC 2006. 2006 3rd IEEE*. [S.l.: s.n.], 2006. v. 1, p. 91–95.

BERKELEY WIRELESS RESEARCH CENTER. *PicoRadio - Wireless Sensor Network research: Berkeley Wireless Research Center*. 2006. Disponível em: <http://bwrc.eecs-.berkeley.edu/research/Pico_Radio/>. Acesso em: 18/02/2006.

BRAGA, T. R. M. *Um Elemento Autonômico para Redes de Sensores Sem Fio*. Dissertação (Mestrado) — Universidade Federal de Minas Gerais, 2006.

BRAGA, T. R. M. et al. Redes autonômicas. In: ____. [S.l.: s.n.], 2006. v. 1, in Minicursos do XXIV Simpósio Brasileiro de Redes de Computadores - SBRC'2006, p. 159–208.

BTnodes. *A Distributed Environment for Prototyping Ad Hoc Networks*. 2006. Available at http://www.btnode.ethz.ch/. Acesso em: 11/08/2007.

BULUSU, N. et al. Scalable coordination in sensor networks: Self-configuring localization systems. In: *Proceedings of the Sixth International Symposium on Communication Theory and Applications (ISCTA'01)*. [S.l.: s.n.], 2001.

CHESS, D. et al. Unity: experiences with a prototype autonomic computing system. In: *Proceeding of the International Conference on Autonomic Computing*. [S.l.: s.n.], 2004. p. 140– 147.

CROSSBOW. *MICA Wireless Measurement System*. [S.l.], 2003. Document Part Number: 6020-0041-01. Disponível em: <http://www.xbow.com>. Acesso em: 11/08/2007.

DELIN, K. A. et al. The jpl sensor webs project: Fielded technology. In: *Space Mission Challenges for IT Proceedings, Annual Conference Series*. [S.l.: s.n.], 2003. Jet Propulsion Laboratory, California Institute of Technology.

HEMPSTEAD, M. et al. An ultra low power system architecture for sensor network applications. In: *Computer Architecture, 2005. ISCA '05. Proceedings. 32nd International Symposium on*. [S.l.: s.n.], 2005. p. 208–219.

HEWLETT-PACKARD. *The HP vision for the Adaptive Enterprise: achieving business agility*. July 2003. Disponível em: <http://h71028.www7.hp.com/enterprise/downloads-/ae_business_white_paper_final0703.pdf>. Acesso em: 25/09/2006.

HOCHSTEIN, A.; ZARNEKOW, R.; BRENNER, W. Itil as common practice reference model for it service management: formal assessment and implications for practice. In: *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service, 2005. EEE '05.* [S.l.: s.n.], 2005. p. 704–710.

HOLLAR, S. *COTS Dust*. Dissertação (Mestrado) — Computer Science Department of the UC Berkeley, December 2000.

IBM. *Autonomic Computing (creating self-managing computing systems).* December 2005. Disponível em: <http://www-3.ibm.com/autonomic/index.shtml.> Acesso em: 12/04/2006.

IBM. *Autonomic Computing White Paper - An Architectural blueprint for autonomic computing*. June 2005. Disponível em: <http://www-03.ibm.com/autonomic/pdfs-/AC_Blueprint_White_Paper_4th.pdf>. Acesso em: 25/09/2006.

IBM. *Autonomic Computing Toolkit*. 2006. Disponível em: <http://www-128.ibm.com-/developerworks/autonomic/overview.html>. Acesso em: 14/06/2006.

IBM. *Policy Management for Autonomic Computing*. 2007. Available at http://www.alphaworks.ibm.com/tech/pmac. Disponível em: <http://www.alphaworks-.ibm.com/tech/pmac>. Acesso em: 25/07/2007.

INTERNATIONAL TELECOMMUNICATION UNION (ITU). *ITU-T M.3010 - Principles for a Telecommunications management network.* May 1996.

KEPHART, J. O.; CHESS, D. M. The vision of autonomic computing. *IEEE Computer*, IEEE, Los Alamitos, CA, USA, v. 36, n. 1, p. 41–50, 2003.

LITTMAN, M. et al. Reinforcement learning for autonomic network repair. In: *Autonomic Computing, 2004. Proceedings. International Conference on*. [S.l.: s.n.], 2004. p. 284–285.

LIU, T.; MARTONOSI, M. Impala: A middleware system for managing autonomic parallel sensor systems. In: *Proceedings of the 9th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP'03)*. San Diego, CA, USA: [s.n.], 2003. p. 107–118.

LOPES, C. E. R. et al. Mannasim: Simulando redes de sensores sem fio. In: *Proceedings of the 24th Brazilian Computer Networks Symposium (SBRC'06)*. [S.l.: s.n.], 2006.

LORINCZ, K. et al. Sensor networks for emergency response: challenges and opportunities. *Pervasive Computing, IEEE*, v. 3, n. 4, p. 16–23, Oct-Dec 2004.

MAINWARING, A. et al. Wireless sensor networks for habitat monitoring. In: *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. New York, NY, USA: ACM Press, 2002. p. 88–97. ISBN 1-58113-589-0.

MARSH, D. et al. Autonomic wireless sensor networks. *Engineering Applications of Artificial Intelligence*, p. 741–748, August 2004. Elsevier Publishers.

MELCHER, B.; MITCHELL, B. Towards an autonomic framework: Self-configuring network services and developing autonomic applications. *Intel Technology Journal*, v. 8, n. ISSN 1535-864X, p. 279 –290, November 2004.

MENDES, P. et al. Novel very small dual-band chip-size antenna for wireless sensor networks. In: *Radio and Wireless Conference, 2004 IEEE*. [S.l.: s.n.], 2004. p. 419–422.

Microsoft Corporation. *Dynamic Systems Initiative Overview White Paper*. February 2005.

MILLER, B. *The autonomic computing edge: The role of knowledge in autonomic systems*. September 2005. IBM Autonomic Computing DeveloperWorks. Disponível em: <http://www-128.ibm.com/developerworks/autonomic/library/ac-edge6/>. Acesso em: 08/02/2006.

MORAIS, M. V. *Relatório de Gestão - Ministério da Agricultura, Pecuária e Abastecimento*. 2007. Disponível em: <http://www.agricultura.gov.br/pls/portal/docs-/PAGE/MAPA/INSTITUCIONAL/RELATORIO_DE_GESTAO/REGIAO_SUDESTE-/RELATORIO_GESTAO_RIO_DE_JANEIRO/DFA_RJ_2002_US.PDF>. Acesso em: 20/07/2007.

NASA JET PROPULSION LAB. *JPL Sensor Webs*. 2001. Available at. http://sensorwebs.jpl.nasa.gov/.

NS-2. *UCB/LBNL/VINT Network Simulator (ns-2)*. November 2005. Available at http://www.isi.edu/nsnam/ns.

OFFICE OF GOVERNMENT COMMERCE. *ITIL Service Support*. [S.l.]: The Stationery Office, 2000. ISBN 0113300158.

OFFICE OF GOVERNMENT COMMERCE. *ITIL Service Delivery*. [S.l.]: The Stationery Office, 2001. ISBN 0113300174.

OFFICE OF GOVERNMENT COMMERCE. *Planning to Implement Service Management Manual*. [S.l.]: The Stationery Office, 2002. ISBN 0113308779.

PATTERSON, D. A. et al. *Recovery-Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies*. [S.l.], March 2002.

PERAZOLO, M. *Symptoms Deep Dive, Part 1: Know thy symptoms, heal thyself*. October 2005. IBM Autonomic Computing DeveloperWorks. Disponível em: <http://www-128.ibm.com/developerworks/autonomic/library/ac-symptom1/index.html>. Acesso em: 25/09/2006.

PERAZOLO, M. *Symptoms Deep Dive, Part 2: Cool things you can do with symptom*. December 2005. IBM Autonomic Computing DeveloperWorks. Disponível em: <http:/-/www-128.ibm.com/developerworks/autonomic/library/ac-symptom2/index.html>. Acesso em: 25/09/2006.

PRAS, A.; BEIJNUM, B.-J. van; SPRENKELS, R. *Introduction to TMN*. Enschede, The Netherlands, abr. 1999. Disponível em: <http://www.simpleweb.org/tutorials/tmn/tmn.pdf>. Acesso em: 11/07/2007.

RABAEY, J. et al. Picoradios for wireless sensor networks: The next challenge in ultra-low-power design. In: *Proceedings of the International Solid-State Circuits Conference, San Francisco, CA*. [S.l.: s.n.], 2002.

RAJARAMAN, R. Topology control and routing in ad hoc networks: a survey. *SIGACT News*, ACM Press, New York, NY, US, v. 33, n. 2, p. 60–73, 2002. ISSN 0163-5700.

RUIZ, L. B. *MANNA: A Management Architecture for Wireless Sensor Networks*. Tese (Doutorado) — Computer Science Department of the Federal University of Minas Gerais, Belo Horizonte, MG, Brazil, December 2003.

RUIZ, L. B. et al. Conception of a self-management application for wireless sensor networks (in frech). In: *Proceedings of the 6th Networks and Services Management (GRES'05)*. Luchon, France: [s.n.], 2005. p. 129–141. ISBN 2-9520326-5-3.

RUIZ, L. B. et al. On the design of a self-managed wireless sensor network. *IEEE Communications Magazine*, v. 43, n. 8, p. 95–102, August 2005.

RUIZ, L. B. et al. Self-managed wireless sensor networks: A study case. In: *Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Network Management (IM'05)*. Nice, France: [s.n.], 2005. Short paper.

RUIZ, L. B. et al. Uma abordagem de auto-gerenciamento para redes de sensores sem fio. In: *Anais do XXIII Simpósio Brasileiro de Redes de Computadores (XXIII SBRC 2005)*. Fortaleza, Brasil: [s.n.], 2005. Short paper.

SCHMID, S.; SIFALAKIS, M.; HUTCHISON, D. Towards autonomic networks. In: *In proceedings of 3rd Annual Conference on Autonomic Networking, Autonomic Communication Workshop (IFIP AN/WAC)*. [S.l.: s.n.], 2006.

SENSICAST. *SensiNet: The Wireless Sensor Network*. 2006. Available at http://www.sensicast.com/.

SENSORNET. *SensorNet Project - Architecture, Protocols, Management, and Applications in Wireless Sensor Networks*. 2004. Disponível em: <http://www.sensornet.dcc.ufmg.br>. Acesso em: 25/03/2006.

SINHA, A.; CHANDRAKASAN, A. Dynamic power management in wireless sensor networks. *IEEE Design & Test of Computers*, IEEE Computer Society Press, v. 18, n. 2, p. 62–74, 2001. ISSN: 0740-7475.

SLOMAN, M. Policy Driven Management For Distributed Systems. *Journal of Network and Systems Management*, v. 2, n. 4, p. 333–360, December 1994.

STALLINGS, W. *SNMP, SNMPv2, and CMIP: the practical guide to network management*. [S.l.]: Addison-Wesley Longman Publishing Co, 1993.

STERRITT, R. Autonomic networks: engineering the self-healing property. In: *Journal of Advanced Engineering Informatics, Engineering Applications of Artificial Intelligence*. [S.l.]: Elsevier Publishers, 2004. v. 17, p. 727–739 187.

SUN MICROSYSTEMS. *Sun N1: Products for automating servers and applications life-cycle management*. September 2006. Disponível em: <http://www.sun.com/software/gridware/>. Acesso em: 10/12/2006.

TENNENHOUSE, D. et al. A survey of active network research. *Communications Magazine, IEEE*, v. 35, n. 1, p. 80–86, Jan. 1997.

TENNENHOUSE, D. L.; WETHERALL, D. J. Towards an active network architecture. In: *In Proceedings of Multimedia Computing and Networking*. [S.l.: s.n.], 1996.

WARNEKE, B. et al. Smart dust: Communicating with a cubic-millimeter computer. *IEEE Computer*, IEEE Computer Society Press, v. 34, n. 1, p. 44–51, 2001. ISSN 0018-9162.

WINS. *Wireless Integrated Network Sensors (WINS)*. 2002. Available at http://www.janet.ucla.edu/WINS/. Disponível em: <http://www.janet.ucla.edu/WINS-/>. Acesso em: 30/04/2005.

ZORKOT, A. C.; ASSUNÇÃO, H. P.; RUIZ, L. B. Uma ferramenta para detecção de movimentos utilizando redes de sensores sem fio. In: *Anais do Congresso Da Sociedade Brasileira de Computação - XXXIII Seminário Integrado de Software e Hardware*. [S.l.: s.n.], 2006.